ABSTRACT

Title of thesis: A STUDY ON DISTRIBUTED RECEDING HORIZON CONTROL

Wann-Jiun Ma, Master of Science, 2011

Thesis directed by: Professor Nuno Martins
Department of Electrical and Computer Engineering

We consider a distributed control problem comprising of multiple sub-systems with one-controller at each sub-system. We apply a recent result about suboptimal receding horizon control that analytically relates a receding horizon control suboptimal solution and system performance loss to quantify the necessary number of iterations for the dual and primal decomposition algorithm to achieve a solution that guarantees stability. We also use this result to explore the idea of ”incremental robustness”, meaning that the overall system is robustly stable and its performance varies gracefully with the inclusion of sub-systems and sub-controllers. We demonstrate these ideas in a consensus seeking and a formation control problem and provide simulation results. To our best knowledge, this is the first time the result is applied to a distributed receding horizon control framework based on dual and primal decomposition.

A STUDY ON DISTRIBUTED RECEDING HORIZON CONTROL

by

Wann-Jiun Ma

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2011

Advisory Committee:
Professor Nuno Martins, Chair/Advisor
Professor Steve Marcus
Professor Andre Tits

Acknowledgments

I would like to thank my advisor Professor Nuno Martins. During the past two years, I had a great time with him thinking about and solving the problems. My deep appreciation to him for his company all this time. I would also like to thank Professor Andre Tits and Professor Steve Marcus for many helpful and most valuable discussions. In addition, I would like to thank my fellow student Eduardo Arvelo for working out together the conference paper to be presented at the IFAC World Congress. Finally, my great appreciation to Professor Martins, Professor Tits and Professor Marcus for serving as members of my thesis committee and reviewing the thesis.

Table of Contents

List of Figures

Chapter 1

Introduction

Receding horizon control also known as model predictive control [2] is a form of control, in which at each time instant, basically we solve an open loop finite horizon optimization problem; then as time horizon rolls, we apply the first control value in the optimal solution to the system.[1] After this first element of the optimal control sequence is applied to the system, the corresponding controlled state can be measured if we assume an observation case of perfect state. It might be corrupted with state noise. Then solve the optimization again taking this new observed state as the initial state. Since the implementation results from solving open loop optimization problems, we can expect that the constraints on the states and controls can be easily added into the system, but it becomes difficult if we adopt other approaches such as dynamical programming.

1.1 Stability

An important issue of receding horizon control is stability. During the 90's, there have been numerous research works on the stability problem [14]. Depending on the system's dynamic structure such as linear or nonlinear; continuous or discontinuous; with constraints or without constraints; discrete-time or continuous-time, there are many different approaches to offer the sufficient conditions for the stability of the system. Basically, there are three essential ingredients in the stability analysis of receding horizon control, which are the final state constraints, the penalty of final state, and the local feedback

[1] As [14] points out, in time, the difference between model predictive control and receding horizon control becomes irrelevant; in the sequel, we use receding horizon control as a generic term.

control of the final state to ensure that it lies in a specific invariant set. Based on the corresponding stability sufficient conditions, one needs to pre-compute the above three ingredients to satisfy the conditions and stabilize the system. One of the most widely adopted principles relies on using the value function of a finite time horizon optimal control problem as the Lyapunov function to establish the stability conditions.

1.2 Sub-optimality and Stability

Most of the discussions about the stability of receding horizon control during the 90's are based on the fact that the optimization problem at each time instant can be computed exactly, i.e. the optimal value is attained. The practical difficulty behind the stability conditions is that solvers are able usually only to provide sub-optimal solution for each optimization subproblem, otherwise there would be real-time concerns. Therefore, the stability results for sub-optimal solution become important if we want to implement it in a practical way. There are two approaches about this kind of research. First, in [19], sufficient conditions are established to stabilize the system for the sub-optimal solutions. If the conditions are satisfied, the feasibility implies stability. One of the main conditions is that the cost of the receding horizon control should decrease along the horizon. The other different approach [11] does not put additional constraints on the cost function, but gives the explicit relation between the deviation of the optimal solution and the deterioration of the stability of the closed-loop receding horizon control system. They also show that under slightly stronger conditions, the stability can be achieved. In [11], a powerful notion, input-to-state stability (ISS), is used to deal with the problem with bounded state disturbances. The explicit relation between the deviation of the optimal control and the deterioration of ISS property is also established.

1.3 Real-time Receding Horizon Control

In order to compute the control sequence quickly, in [20], a tailored solver is used. They consider the structure of the problem, use the sparsity of the structure, and consider related sub-optimal optimization techniques to compute the control sequence quickly on-line. In [21], a performance guarantee and the stability analysis of real-time receding horizon control are presented.

1.4 Stochastic and Robust Receding Horizon Control

Standard receding horizon control does not include state disturbances into the analysis. We could argue that since in the standard receding horizon control, the first element of the computed control sequence is applied, and then the resulting controlled state is used as the initial state for the optimization subproblem at the next time instant. Since it is already in a closed-loop control form, it can overcome the disturbances to some extent. However, adding the disturbances into the system, and considering the performance of the disturbed receding horizon control are still necessary. There are many works about stochastic and robust receding horizon control over the past ten years. Among numerous proposals, there are two categories. For the first category, people use input-to-state stability notion, and consider tightened state constraints to establish the stability sufficient conditions for a disturbed receding horizon control system with the bounded disturbances (see [13]). The basic idea of using tightened state constraints is simple. Because of triangle inequality, if one uses carefully chosen tightened state constraints to solve a nominal receding horizon control problem, the disturbed system might stabilize in ISS sense. The other approach is relatively new, in which the disturbances are modeled as random variables, and stochastic programming is applied to solve the associated uncertain receding horizon control problem [17]. For this kind of approach, since we directly solve the problem with uncertainty using stochastic programming instead of solving a nominal problem, the structure of the problem is destroyed that explains the reason why there do not exist many stability or performance analyses for such an approach.

1.5 Decentralized and Distributed Receding Horizon Control

In accordance with a survey paper [18], here we clearly define decentralized and distributed receding horizon control based on the kind of information exchanged among agents and interaction between the agents. Decentralized receding horizon control is for a system that all agents have some kind of interactions either coming from the coupled states or inputs. And each agent has its own local regulator and computes its own controller. One novel way to ensure the stability of this kind of system treats the agents interactions as disturbances and again uses ISS notion analysis.

Distributed receding horizon control, on the other hand, allows agents to exchange the information. Depending on different communication structure assumption and necessary information needed to be transmitted, there are different approaches to follow. In particular, there is a special type of distributed receding horizon control problem, in which all agents are independent, but still have some kind of information exchange mechanism. This type of problem is called coordinated control problem. Consensus control problem in this group further assumes that all agents do not know the equilibrium information in advance.

In [5], dual decomposition is adopted to solve a consensus receding horizon control problem. And in [7] and [12], primal decomposition is used to solve a similar problem. Dual and primal decompositions are specific distributed optimization techniques. In dual decomposition, the public variables are coupled among the agents, and each agent uses its own dual prices to maintain the consistence constraints with other agents in the neighborhood. The equilibrium information is our public variables, i.e. all agents converge to a consensus point. Using consistence constraints of the public variables, one can be sure that all agents converge to the consensus point, even though each agent solves the optimal control problem independently. On the other hand, in primal decomposition, one assigns the same amount of resources to each agent, each agent minimizes its own cost function based on this fixed resource, then we minimize the cost function with respect to this resource (public variable).

As mentioned in Section 1.2, classical sub-optimal receding horizon control analysis (see [19]) shows that under necessary assumptions, instead of optimality, feasibility is sufficient for stability. This result reduces the computational requirements to implement a stabilizing receding horizon control. On the other hand, in [11], an explicit relationship is established between the optimality loss of the control and the deterioration of the input-to-state stability of a receding horizon control system. Unlike the classical sub-optimal analysis, this explicit relationship quantifies the required computational requirements for establishing a receding horizon control system that is stable within a given error range. [12] applies this relationship to a consensus seeking problem and establishes a practical implementation. They use the result from [11] to calculate a bound on the number of iteration steps necessary for stabilizing the system within a given error range. The closest work related to [12] is [7], in which a consensus problem is implemented using distributed receding horizon control. Primal decomposition based on incremental subgradient method is adopted there. Classical sub-optimal receding horizon control analysis is used to reduce the computational requirements. However, in [12], they use a fundamentally different approach to study stability and suboptimality.

In this study, first, we try to extend the results in [12] to dual decomposition method. The sensitivity analysis and first order approximation are used to quantify the deviation of the primal variables. We also solve a formation control problem using dual and primal decompositions to discuss the relation between the optimality loss of the control and the deterioration of the system stability.

The contents following the introduction section are organized as follows. Chapter 2 describes the main problem formulation. Then, in Chapter 3, we recall the results in [11] and discuss the necessary modifications. We establish the explicit relationship between the optimality loss of the control and the deterioration of the system performance. A consensus seeking problem based on dual decomposition and receding horizon control is discussed in Chapter 4. Furthermore, the results in [12] for incremental primal decomposition based approach and the idea of incremental robustness are also discussed in this chapter for completeness. Chapter 5 solves a formation control problem based on dual and primal decompositions with receding horizon control. The simulation is given in Chapter 6. The conclusion and future work can be found in Chapter 7.

Chapter 2

Problem Formulation

Consider an N-agents system where each agent has the following time invariant discrete-time dynamics

$$x_{k+1}^i = \phi^i(x_k^i, u_k^i), \tag{2.1}$$

for $i = 1, ..., N$, where the state $x_k^i \in \Re^n$, the control input $u_k^i \in \Re^m$, and the dynamics $\phi^i : \Re^n \times \Re^m \to \Re^n$, which is continuous. In this and Chapter 3, we only assume that the dynamics is continuous. The dynamics can be linear or nonlinear. Each agent has its own cost function:

$$J^i := \sum_{k=0}^{\infty} h^i(x_k^i - x_s^i, u_k^i - u_s^i), \tag{2.2}$$

where h^i is convex and nonnegative, $x_s^i \in \Theta^i$ is an equilibrium state, and $\Theta^i \subseteq \Re^n$ is the feasible set for x_s^i. $u_s^i \in \Omega^i$ is the control input associated with the steady state x_s^i and $\Omega^i \subseteq \Re^m$ is the feasible set for u_s^i. The cost function penalizes the deviation from the equilibrium state. The objective of this thesis is to minimize the sum of the individual cost, while satisfying state and control constraints. We formulate the problem as follows:

$$\begin{aligned} \text{minimize} \quad & \sum_{k=0}^{\infty} h(x_k - x_s, u_k - u_s) \\ \text{subject to} \quad & x_{k+1} = \phi(x_k, u_k), \\ & x_k \in \mathcal{X}, \\ & u_k \in \mathcal{U}, \\ & x_s \in \Theta, \\ & u_s \in \Omega, \end{aligned} \tag{2.3}$$

where $h = \sum_{i=1}^{N} h^i$, $\phi = (\phi^1, ..., \phi^N)$, $x_k = (x_k^1, ..., x_k^N)$, $u_k = (u_k^1, ..., u_k^N)$, $x_s = (x_s^1, ..., x_s^N)$, and $u_s = (u_s^1, ..., u_s^N)$. The sets $\mathcal{X}$ and $\mathcal{U}$ are compact and are called the state and control constraint sets, respectively. Note that $\mathcal{X} = \mathcal{X}^1 \times ... \times \mathcal{X}^N$ and $\mathcal{U} = \mathcal{U}^1 \times ... \times \mathcal{U}^N$, where $\times$ denotes the Cartesian product of sets and $\mathcal{X}^i \subseteq \Re^n$, $\mathcal{U}^i \subseteq \Re^m$ for $i = 1, ..., N$. Similarly, $\Theta = \Theta^1 \times ... \times \Theta^N$ and $\Omega = \Omega^1 \times ... \times \Omega^N$. Where x_s and u_s may be fixed desired targets or decision variables of the associated optimization problem depending on different applications. This is an infinite horizon optimal control problem with state and control constraints. In the following sections, we adopt receding horizon control and dual and primal decomposition scheme to solve this problem in a practical and distributed way.

There are two scenarios associated with this problem set-up. First, we formulate a consensus seeking problem. In this scenario, after the algorithm converges, all x_s^i's converge to the same value, which is set to be the desired target that each agent need to track. In this set-up, the agents are coupled via cost functions. In the usual consensus or rendezvous seeking problem, the agents do not know the consensus point in advance.

For the second scenario, we formulate a formation control problem in a similar way. In formation control scenario, the desired target is fixed in advance, and each agent's desired target might be different. The consistency constraints about the states and controls of agents are embedded in the state and control constraints in (2.3). Hence, the agents might track different fixed targets and should follow some predetermined trajectories at the same time. In this set-up, the agents are coupled via state and control constraints.

Chapter 3

Suboptimal Receding Horizon Control Scheme

Receding horizon control also known as model predictive control, is one of the practical ways to tackle infinite horizon control problems such as the one in previous chapter. However, many factors such as real-time constraints and online computational limitations may render the resulting closed-loop system unstable. To solve this problem, classical suboptimal receding horizon control [19] uses an explicit endpoint and cost constraint to ensure the stability of the system even if each optimization subproblem is solved partially. Recently, in [11] an analytical formula of the relationship between the optimality loss and the performance degradation of a model predictive control hybrid system was established. Based on this formula, given a tolerable degradation range in the performance, the difference between the cost function values of optimal and suboptimal solutions for each optimization subproblem can be obtained analytically. In the following discussion, we modify this result to fit our framework.

Consider an N-agents system as in Chapter 2. the cost function $h = \sum_{i=1}^{N} h^i$, the dynamics $\phi = (\phi^1, ..., \phi^N)$, the state $x_k = (x_k^1, ..., x_k^N)$, and the control input $u_k = (u_k^1, ..., u_k^N)$. The sets $\mathcal{X} = \mathcal{X}^1 \times ... \times \mathcal{X}^N$ and $\mathcal{U} = \mathcal{U}^1 \times ... \times \mathcal{U}^N$. $\mathcal{X}^i \subseteq \Re^n$ and $\mathcal{U}^i \subseteq \Re^m$ for $i = 1, ..., N$.

First, we introduce two basic definitions: A real value function $\alpha : \Re_+ \to \Re_+$ is a $\mathcal{K}$ function if it is continuous, strictly increasing and $\alpha(0) = 0$. A function $\beta : \Re_+ \times \Re_+ \to \Re_+$ is a $\mathcal{KL}$ function if for each fixed $k \in \Re_+$, $\beta(\cdot, k)$ is a $\mathcal{K}$ function and for each fixed $t \in \Re_+$, $\beta(t, \cdot)$ is decreasing and $\lim_{k\to\infty} \beta(t, k) \to 0$.

We define an $\varepsilon -$ *asymptotically stable* system, where ε is an index of performance loss.

Definition 3.1: Let $\mathcal{X}$ be a subset of $\Re^{N \cdot n}$. Given an $\varepsilon \in \Re_+$, if there exists a $\mathcal{KL}$-function β such that for each $x_0 \in \mathcal{X}$, $||x_k|| \leq \beta(||x_0||, k) + \varepsilon$ for all $k = 1, 2, ...$, where x_k is the state trajectories of the discrete-time system $x_{k+1} = \phi(x_k, u_k)$ with initial state x_0, then the system is $\varepsilon -$ *asymptotically stable* $(\varepsilon - AS)$ *in* $\mathcal{X}$.

Next, consider Theorem 4 in [11] but without disturbance, which is crucial to obtain the analytical formula of the relationship about the performance loss and suboptimality.

Theorem 3.1: Let $\alpha_1(t) := at^p$, $\alpha_2(t) := bt^p$, $\alpha_3(t) := ct^p$, where a, b, c, p are positive real numbers, and $c \leq b$. Moreover, let $d, e \in \Re_+$, and $\mathcal{X}_f \subseteq \Re^{N \cdot n}$ be a positively invariant set for the system $x_{k+1} = \phi(x_k, u_k)$. If there exists a function $V(\cdot)$ such that

$$\begin{aligned} &\alpha_1(||x_k||) \leq V(x_k) \leq \alpha_2(||x_k||) + d, \\ &V(x_{k+1}) - V(x_k) \leq -\alpha_3(||x_k||) + e \end{aligned} \tag{3.1}$$

for all $x_k \in \mathcal{X}_f$, and $x_{k+1} = \phi(x_k, u_k)$, then the system $x_{k+1} = \phi(x_k, u_k)$ is $\varepsilon - AS$ *in* $\mathcal{X}_f$ with $\beta(t, k) = \alpha_1^{-1}(2\rho^k \alpha_2(t))$, $\varepsilon = \alpha_1^{-1}(2(d + \frac{e}{1-\rho}))$, $\rho = 1 - \frac{c}{b}$.

Here we introduce standard notations used to describe receding horizon control strategies. Given the initial condition x_k, the decision variable for k-th optimization subproblem is given by $U_k := (u_{0,k}, ..., u_{T-1,k})$, where T is the prediction horizon. $X_k := (x_{1,k}, ..., x_{T,k})$ is the corresponding predicted state trajectory. $\mathcal{X}_D \subseteq \mathcal{X}$ is the invariant target set, where $\mathcal{X}$ is a subset of $\Re^{N \cdot n}$. Each subproblem cost function is given by

$$J(x_k, U_k) := \sum_{j=0}^{T-1} h(x_{j,k}, u_{j,k}) + h_T(x_{T,k}), \tag{3.2}$$

where h_T is the endpoint penalty. $J^*(x_k)$ denotes the optimal cost with the initial state x_k. The optimal control sequence is given by $(u^*_{0,k}, ..., u^*_{T-1,k})$. Based on the receding horizon control scheme, only the first element of the optimal control sequence is applied to the system, i.e. $u_k^{RH} := u_{0,k}^*$. The admissible control set is denoted as $\mathcal{U}_T(x_k) := \{U_k \in \mathcal{U}^T | X_k \in \mathcal{X}^T, x_{T,k} \in \mathcal{X}_D\}$, and given a positive constant δ, the suboptimal admissible control set is $\Gamma_\delta(x_k) := \{U_k \in \mathcal{U}_T(x_k) | J(x_k, U_k) \leq J^*(x_k) + \delta\}$, where $\mathcal{X}^T$ is the Cartesian product of $\mathcal{X}$'s (there are T of them), and $\mathcal{U}^T$ is the Cartesian product of $\mathcal{U}$'s. $\gamma_\delta(x_k) = \{u_{0,k} \in \Re^{N\times m} | U_k \in \Gamma_\delta(x_k)\}$ is the first element of it. Pick any element in $\gamma_\delta(x_k)$, for example $\bar{\gamma}_\delta(x_k)$ to be the suboptimal admissible control input. We also use the notation $u_k^{RH\_\delta}$ for $\bar{\gamma}_\delta(x_k)$. After applying this suboptimal admissible control input to the system, the receding horizon closed-loop control system is $\theta_\delta(x_k) = \phi(x_k, \bar{\gamma}_\delta(x_k))$. $\mathcal{X}_U := \{x \in \mathcal{X} | \sigma(x) \in \mathcal{U}\}$ is the safe set with respect to the state and input constraints for σ (see [11]). σ is the local feedback control corresponding to the invariant target set $\mathcal{X}_D$. $\sigma(\cdot) : \Re^{N\times n} \to \Re^{N\times m}$, and $\sigma(0) = 0$. Each optimization subproblem is formulated as:

Problem 3.1: Given $\mathcal{X}_D \subseteq \mathcal{X}$, the prediction horizon T, and the initial state $x_k \in \mathcal{X}$

$$\begin{aligned} &\text{minimize} && J(x_k, U_k) \\ &\text{subject to} && U_k \in \mathcal{U}_T(x_k) \end{aligned} \tag{3.3}$$

We use $\mathcal{X}_f(T) \subseteq \mathcal{X}$ to denote the feasible set. If the initial condition $x_k \in \mathcal{X}_f(T)$, the admissible control set $\mathcal{U}_T(x_k)$ is not empty.

The following theorem is the main result in this section. The proof is similar to Theorem 9 in [11] with some modifications and is presented in Appendix A.

Theorem 3.2: Assume there exists a positively invariant set $\mathcal{X}_D \subseteq \mathcal{X}_U$ with $0 \in int(\mathcal{X}_D)$ for the closed-loop system with the local feedback control $u_k = \sigma(x_k)$. Assume the following conditions are satisfied

$$\begin{aligned} &h(x,u) \geq \alpha_1(||x||),\ \forall x \in \mathcal{X}_f(T),\ u \in \mathcal{U}, \\ &h_T(x) \leq \alpha_2(||x||),\ \forall x \in \mathcal{X}_f(T), \\ &h_T(\phi(x,\sigma(x))) - h_T(x) + h(x,\sigma(x)) \leq 0,\ \forall x \in \mathcal{X}_D, \end{aligned} \tag{3.4}$$

where $\alpha_1(t) := at^p$, $\alpha_2(t) := bt^p$, a, b, p are positive real numbers. Then given a $\delta \in \Re_+$,

(1) If *Problem 3.1* is feasible at time k for state $x_k \in \mathcal{X}$, then it is feasible at time $k+1$ for any state $x_{k+1} = \theta_\delta(x_k)$, and $\mathcal{X}_D \subseteq \mathcal{X}_f(T)$.

(2) The closed-loop system $x_{k+1} = \theta_\delta(x_k)$ is $\varepsilon - AS$ *in* $\mathcal{X}_f(T)$ with $\varepsilon(\delta) := (\frac{2b}{a^2}\delta)^{1/p}$.

In order to apply *Theorem 3.2*, we need to compute those stage cost $h(\cdot,\cdot)$, final cost $h_T(\cdot)$, final state invariant set $\mathcal{X}_D$, and final state local feedback control $\sigma(\cdot)$, which satisfy the hypotheses in *Theorem 3.2*. In [8], [9], [10], [11], there are examples and procedures telling us how to formulate them. Here we use some simple examples to summarize the procedure to compute those essential elements. First, consider quadratic cost functions $h(x,u)$ and $h_T(x)$, i.e. $h(x,u) = x'Qx + u'Ru$, and $h_T(x) = x'Q_Tx$, where Q, R, and Q_T are weighting matrices in the cost functions with appropriate dimensions. In this simple case, the first two conditions are satisfied if we assume that Q, R, and Q_T are positive definite matrices. It is because $h(x,u) \geq x'Qx \geq \lambda_{min}(Q)||x||_2^2$, and $h_T(x) \leq \lambda_{max}(Q_T)||x||_2^2$. Hence, if define $\alpha_1(||x||) = \lambda_{min}(Q)||x||_2^2$, and $\alpha_2(||x||) = \lambda_{max}(Q_T)||x||_2^2$, then the first two conditions in (3.4) are satisfied. In choosing the weighting Q_T, and local feedback control law $\sigma(\cdot)$ such that the third condition is satisfied, we can formulate the following matrix inequality to compute σ and Q_T. As a simple example, assume now that the dynamics and the local feedback control law are linear, i.e. $x_{k+1} = Ax_k + Bu_k$, and $\sigma(x) = Kx$. After we substitute this dynamics and local feedback control into the third sufficient condition in (3.4), we can obtain the matrix inequality $Q_T - (A+BK)'Q_T(A+BK) - Q - K'RK > 0$, i.e. $Q_T - (A+BK)'Q_T(A+BK) - Q - K'RK$ is positive definite, to satisfy the corresponding sufficient condition. Finally, for computing the positively invariant set $\mathcal{X}_D$, consider the following sequence of sets. Define a safe set $\tilde{\mathcal{X}}_U$, which is the largest compact polyhedron set inside $\mathcal{X}_U$. For an arbitrary target set $\mathcal{X}$, define

$\mathcal{Q}(\mathcal{X}) = \{x \in \Re^{N \times n} | Ax + Ku \in \mathcal{X}\}$. $\mathcal{X}_0 := \tilde{\mathcal{X}}_U$, and $\mathcal{X}_j := \mathcal{Q}(\mathcal{X}_{j-1}) \bigcap \mathcal{X}_{j-1}$. Then the maximal positively invariant set contained in the set $\tilde{\mathcal{X}}_U$ is given by $\bigcap_{j=1}^{\infty} \mathcal{X}_j$. It contains the origin and is convex. For more details about the description and the proof, one can refer to [8], [9], [10], [11].

Here come three remarks. First, for the system whose equilibrium is different from the origin, for example the problem in Chapter 2, whose equilibrium state is given by $x_s \in \Theta$, where $\Theta \subseteq \Re^{N \cdot n}$. And the steady state control $u_s \in \Omega$, where $\Omega \subseteq \Re^{N \cdot m}$. We can define $\tilde{x}_k = x_k - x_s$ and $\tilde{u}_k = u_k - u_s$ to translate the equilibrium (x_s, u_s) to the origin. Then one can apply the theorem to the problem.

Secondly, if x_s is a decision variable like the case in [7], then it is sufficient to add an assumption $\mathcal{X}_D(k) \subseteq \mathcal{X}_D(k+1)$, for $k = 0, 1, 2, ...$, where $\mathcal{X}_D(k)$ is the invariant set at k-th time instant to make sure that the theorem is applicable. The following corollary concludes this statement, and the proof is given in Appendix B.

Corollary 3.1: If the cost function in (3.2) is replaced by

$$J(x_k, U_k, x_s) := \sum_{j=0}^{T-1} h(x_{j,k} - x_s, u_{j,k} - u_s) + h_T(x_{T,k} - x_s), \tag{3.5}$$

where $x_s \in \Theta$ is also a decision variable in the optimization subproblem and $u_s \in \Omega$ is the corresponding steady state control with respect to x_s. Where $\Theta \subseteq \Re^{N \cdot n}$ is the feasible set for x_s, and $\Omega \subseteq \Re^{N \cdot m}$ is the feasible set for u_s. If $\mathcal{X}_D(k) \subseteq \mathcal{X}_D(k+1)$, where $\mathcal{X}_D(k)$ is the invariant set at k-th time instant. At each time instant k, assume there exists a positively invariant set $\mathcal{X}_D(k) \subseteq \mathcal{X}_U$, for $k = 0, 1, 2, ...$, with $x_s^* \in int(\mathcal{X}_D(k))$ for the closed-loop system with $u_k = \sigma(x_k)$, where x_s^* is the optimal solution for x_s in the corresponding optimization subproblem. Assume the following conditions are satisfied

$$
\begin{aligned}
& h(x - x_s, u - u_s) \geq \alpha_1(||x - x_s||),\ \forall x \in \mathcal{X}_f(T),\ u \in \mathcal{U},\ x_s \in \Theta,\ u_s \in \Omega \\
& h_T(x - x_s) \leq \alpha_2(||x - x_s||),\ \forall x \in \mathcal{X}_f(T),\ x_s \in \Theta \\
& h_T(\phi(x, \sigma(x)) - x_s) - h_T(x - x_s) + h(x - x_s, \sigma(x) - u_s) \leq 0,\ \forall x \in \mathcal{X}_D,\ x_s \in \Theta,\ u_s \in \Omega \\
& c \leq \tfrac{||x - x_s||^2}{||x - x_s^*||^2} a,\ \forall x \in \mathcal{X}_f(T),\ x_s \in \Theta,\ x_s^* \in \Theta
\end{aligned}
\tag{3.6}
$$

where $\alpha_1(t) := at^p$, $\alpha_2(t) := bt^p$, a, b, c, p are positive real numbers. Then given a $\delta \in \Re_+$,

(1) If *Problem 3.1* is feasible at time k for state $x_k \in \mathcal{X}$, then it is feasible at time $k+1$ for any state $x_{k+1} = \theta_\delta(x_k)$, and $\mathcal{X}_D(k) \subseteq \mathcal{X}_f(T)$.

(2) The closed-loop system $x_{k+1} = \theta_\delta(x_k)$ is $\varepsilon - AS\ in\ \mathcal{X}_f(T)$ with $\varepsilon(\delta) := (\frac{2b}{ac}\delta)^{1/p}$, and with the origin replaced by x_s^*.

The third remark is crucial that motivates the analysis in Chapter 4, where we use the dual decomposition method to solve *Problem 3.1* in a distributed way. If the desired target set is characterized by a single point, it becomes an equality constraint of the type $x_{T,k} = x_s$. In this case, any suboptimal solution (i.e., any iterate of the dual decomposition algorithm other than the optimal one) is unfeasible. For this reason we consider a desired target set $\mathcal{X}_D$ constraint instead of final state equality constraint in *Theorem 3.2*.

On the other hand, if we consider a primal decomposition based approach instead of dual decomposition, the feasibility of the suboptimal solution is not an issue. Primal decomposition method always generates feasible suboptimal solutions at each iteration. Therefore the equality endpoint constraints can be used to ensure the stability. In Chapter 4, a primal decomposition based approach is also adopted to implement a consensus problem in a distributed way (also see [12]). In order to have the similar analysis for that case, in the following discussion, we consider the relation of suboptimality and stability of a consensus problem using equality endpoint constraints. The proof for the following corollary is given in Appendix C.

Corollary 3.2: If the cost function in (3.2) is replaced by

$$J(x_k, U_k, x_s) := \sum_{j=0}^{T-1} h(x_{j,k} - x_s, u_{j,k} - u_s), \tag{3.7}$$

where $x_s \in \Theta$ ($u_s \in \Omega$) is also a decision variable in the optimization subproblem. Where $\Theta \subseteq \Re^{N \cdot n}$ is the feasible set for x_s, and $\Omega \subseteq \Re^{N \cdot m}$ is the feasible set for u_s. x_s^* is the corresponding optimal consensus point. The endpoint constraint is given by $x_{T,k} = x_s$. Assume the following conditions are satisfied

$$\begin{aligned}
&h(x - x_s, u - u_s) \geq \alpha_1(||x - x_s||), \; \forall x \in \mathcal{X}_f(T), \; u \in \mathcal{U}, \; x_s \in \Theta, \; u_s \in \Omega \\
&h(x_{j,k}^* - x_s^*, u_{j,k}^* - u_s^*) \leq \tilde{\alpha}_2(||x - x_s^*||), \; \forall x \in \mathcal{X}_f(T), \; x_s^* \in \Theta, \; u_s^* \in \Omega \\
&c \leq \tfrac{||x - x_s||^2}{||x - x_s^*||^2} a, \; \forall x \in \mathcal{X}_f(T), \; x_s \in \Theta, \; x_s^* \in \Theta
\end{aligned} \tag{3.8}$$

where $\alpha_1(t) := at^p$, $\tilde{\alpha}_2(t) := bt^p$, a, b, c, p are positive real numbers. Then given a $\delta \in \Re_+$,

(1) If *Problem 3.1* is feasible at time k for state $x_k \in \mathcal{X}$, then it is also feasible at time $k+1$ for any state $x_{k+1} = \theta_\delta(x_k)$.

(2) The closed-loop system $x_{k+1} = \theta_\delta(x_k)$ is $\varepsilon - AS \; in \; \mathcal{X}_f(T)$ with $\varepsilon(\delta) := (\frac{2b}{ac}\delta)^{1/p}$, and with the origin replaced by x_s^*.

Chapter 4

A Consensus Seeking Strategy Based on Dual and Primal Decomposition

4.1 Dual Decomposition

In this section, we consider a consensus seeking problem ([12]) to illustrate the idea of the study. We use the dual decomposition method to solve it in a distributed way that follows the receding horizon control framework. The main drawback of the dual decomposition method is the large number of iterations needed for the algorithm to converge in a system with a large number of agents. We apply *Theorem 3.2* in Chapter 3 with the objective of estimating the required number of iterations necessary to guarantee performance within a fixed error tolerance in the $\varepsilon - AS$ sense.

We formulate the problem using linear dynamics and quadratic cost with endpoint penalty. Because in this chapter we focus on each subproblem of the receding horizon control and use dual and primal decomposition to solve it, in order to simplify the notations, we use $(u_0^i, u_1^i, ..., u_{T-1}^i)$ to denote the optimization variables for each subproblem of the receding horizon control for agent i, $i = 1, 2, ..., N$. N is the number of agents in the system. Similarly, we use $(x_0^i, x_1^i, ..., x_T^i)$ to denote the corresponding predicted state trajectories for agent i with the initial condition x_0^i. Moreover, we add a decision variable r into each agent's cost function to denote the consensus point of agents. The following constrained minimization problem is given by:

$$
\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{N} \sum_{k=0}^{T-1} (x_k^i - r)' Q^i (x_k^i - r) + (u_k^i - u_r)' R^i (u_k^i - u_r) + \\
& (x_T^i - r)' Q_T^i (x_T^i - r) \\
\text{subject to} \quad & x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& x_k^i \in \mathcal{X}^i, \\
& u_k^i \in \mathcal{U}^i, \\
& x_T^i \in \mathcal{X}_D^i, \; i = 1, 2, ..., N, \\
& r \in \Theta,
\end{aligned} \tag{4.1}
$$

where $A^i \in \Re^{n \times n}$, $B^i \in \Re^{n \times m}$, Q^i, R^i, and Q_T^i are symmetric positive definite matrices of appropriate dimensions. And $\mathcal{X}^i \subseteq \Re^n$, $\mathcal{U}^i \subseteq \Re^m$ for $i = 1, ..., N$, and Θ[1] $\subseteq \Re^n$ is the feasible set for r. As the remarks in Chapter 3 point out, the final state constraint is that all agent's final states lie in their own desired target sets given by $\mathcal{X}_D^i = \{x \in \mathcal{X}^i | ||x_T^i - r^*||_S \leq \xi, \; \xi > 0\}$, where $||\cdot||_S$ is an appropriate norm, and r^* is the optimal consensus point among agents, instead of having an endpoint constraint of the type $x_T^i = r^*$ for $i = 1, ..., N$. This relaxation enables the algorithm iterates to become feasible before convergence as long as they lie in the desired target set.

We apply the dual decomposition method by adding additional decision variables r^i's and necessary consistency constraints:

[1] Note that here, the feasible set Θ for the consensus point r is a subset of $\Re^n$. However, in Chapter 3, we also use Θ to denote the feasible set for x_s, and in that case, Θ is a subset of $\Re^{N \cdot n}$.

$$
\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{N}\sum_{k=0}^{T-1}(x_k^i - r^i)'Q^i(x_k^i - r^i) + u_k^i - u_r^i)'R^i(u_k^i - u_r^i) + \\
& (x_T^i - r^i)'Q_T^i(x_T^i - r^i) \\
\text{subject to} \quad & x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& x_k^i \in \mathcal{X}^i, \\
& u_k^i \in \mathcal{U}^i, \\
& x_T^i \in \bar{\mathcal{X}}_D^i \\
& r^i = r^j, \ j \in \mathcal{N}_i, \\
& r^i \in \Theta, \ i = 1, 2, ..., N,
\end{aligned}
\tag{4.2}
$$

where $\mathcal{N}_i$ is the the set of indexes of neighbors of agent i, $\bar{\mathcal{X}}_D^i := \{x \in \mathcal{X}_D^i | ||x_T^i - r^i||_S \leq \xi - \zeta, \ \zeta > 0\}$. Note that the desired sets are smaller than the original one, which means that if $||r^i - r^*||_S \leq \triangle$, where $\triangle$ is a positive constant, then from the triangle inequality, $||x_T^i - r^*||_S \leq \xi - \zeta + \triangle$. If $-\zeta + \triangle \leq 0$, then the final state x_T^i lies in $\mathcal{X}_D^i$ and the solution is feasible.

Given r^i, each agent can compute the following optimization problem individually

$$
\begin{aligned}
f^i(r^i) := \quad & \min \textstyle\sum_{k=0}^{T-1}(x_k^i - r^i)'Q^i(x_k^i - r^i) + u_k^i - u_r^i)'R^i(u_k^i - u_r^i) + \\
& (x_T^i - r^i)'Q_T^i(x_T^i - r^i) \\
& \text{subject to } x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& \qquad x_k^i \in \mathcal{X}^i, \\
& \qquad u_k^i \in \mathcal{U}^i, \\
& \qquad x_T^i \in \bar{\mathcal{X}}_D^i, \\
& \qquad r^i \in \Theta.
\end{aligned}
\tag{4.3}
$$

For simplicity, we rewrite (4.2) as:

$$\begin{aligned} &\text{minimize} && f^i(r^i) \\ &\text{subject to} && g^i(r^i) = 0, \\ &&& r^i \in \Theta, \end{aligned} \tag{4.4}$$

where $g^i(r^i)$ is the corresponding consistency constraint in (4.2).

In the following discussion, we use sensitivity analysis, as in [1], to quantify the variation of the primal variable.

The difference between the dual price at the $(l+1)$-th iteration μ_{l+1} and any given $\mu \geq 0$ is given by the following basic iterate ([4])

$$||\mu_{l+1} - \mu||^2 \ \leq ||\mu_l - \mu||^2 - 2\alpha(q(\mu) - q(\mu_l)) + \alpha^2 ||g_l||^2, \ \forall l \geq 0, \tag{4.5}$$

where α is the step size used to update the dual variables, $q(\mu)$ is the dual function value at μ and g_l is the computed subgradient at the l-th iteration. Then it follows that:

$$||\mu_{l+1} - \mu||^2 \ \leq \ ||\mu_0 - \mu||^2 - \sum_{j=0}^{l} [2\alpha(q(\mu) - q(\mu_j)) - \alpha^2 ||g_j||^2], \ \forall l \geq 0. \tag{4.6}$$

Now we use sensitivity analysis to quantify the variation of the primal variable as the dual prices change. First, we remove the constraint $r^i \in \Theta$. If the resulting solution satisfies the constraint, then it is a feasible solution. In particular, we formulate the problem as:

$$\text{minimize} \quad f^i(r^i) + \mu^{i\prime} g^i(r^i), \tag{4.7}$$

where μ^i's are the dual prices.

The corresponding Lagrange function is given by

$$L^i(r^i, \mu^i) := f^i(r^i) + \mu^{i\prime} g^i(r^i). \tag{4.8}$$

Let $r^i(\mu^i)$ denote the minimizer of $L^i(r^i, \mu^i)$ corresponding to μ^i. Because of the first order condition,

$$\nabla_r L^i(r^i, \mu^i)|_{r^i = r^i(\mu^i)} = 0, \tag{4.9}$$

and by differentiating (4.9) with respect to μ^i we have

$$\nabla r^i(\mu^i) \nabla^2_{rr} L^i(r^i(\mu^i), \mu^i) + \nabla^2_{r\mu} L^i(r^i(\mu^i), \mu^i) = 0. \tag{4.10}$$

Moreover, if $\nabla^2_{rr} L^i(r^i(\mu^i), \mu^i)$ invertible we have

$$\nabla r^i(\mu^i) = -\nabla^2_{r\mu} L^i(r^i(\mu^i), \mu^i)(\nabla^2_{rr} L^i(r^i(\mu^i), \mu^i))^{-1}. \tag{4.11}$$

Now, we express the difference between the primal variable and the optimal primal solution as

$$||r^i - r^{i*}|| = ||r^i - r^i(n \cdot |\mathcal{N}_i|) + r^i(n \cdot |\mathcal{N}_i|) - r^i(n \cdot |\mathcal{N}_i| - 1) + r^i(n \cdot |\mathcal{N}_i| - 1) \cdots - r^{i*}||, \tag{4.12}$$

where the notation $r^i(j)$ means that for the primal variable, which is a function of the corresponding dual prices i.e. $r^i = r^i([\mu^i]_1, ..., [\mu^i]_{n \cdot |\mathcal{N}_i|})$, the l-th arguments, for $l = j, j+1, ..., n \cdot |\mathcal{N}_i|$ are replaced by the optimal dual solutions, where $[\mu]_j$ denotes the j-th element of μ. Then, the following inequality follows:

$$||r^i - r^{i*}|| \leq ||r^i - r^i(n \cdot |\mathcal{N}_i|)|| + ||r^i(n \cdot |\mathcal{N}_i|) - r^i(n \cdot |\mathcal{N}_i| - 1)|| + \cdots ||r^i(2) - r^{i*}||. \tag{4.13}$$

Using first order approximation to approximate each term on the right hand side of (4.13), we get:

$$||r^i(j+1) - r^i(j)||^2 \simeq [([\mu^i]_j - [\mu^{i*}]_j)\tfrac{\partial [r^i]_1}{\partial [\mu^i]_j}]^2 + ... + [([\mu^i]_j - [\mu^{i*}]_j)\tfrac{\partial [r^i]_n}{\partial [\mu^i]_j}]^2. \tag{4.14}$$

Now, we can relate the dual price variation to the primal variation via the following formula:

$$
\begin{aligned}
||r^i - r^{i*}|| &\leq \sum_{j=1}^{n\cdot|\mathcal{N}_i|}([\mu^i]_j - [\mu^{i*}]_j)(\sum_{k=1}^{n}(\frac{\partial[r^i]_k}{\partial[\mu^i]_j})^2)^{1/2} \\
&\leq ||\mu^i - \mu^{i*}|| || \begin{bmatrix} (\sum_{k=1}^{n}(\frac{\partial[r^i]_k}{\partial[\mu^i]_1})^2)^{1/2} \\ \vdots \\ (\sum_{k=1}^{n}(\frac{\partial[r^i]_k}{\partial[\mu^i]_{n\cdot|\mathcal{N}_i|}})^2)^{1/2} \end{bmatrix} ||.
\end{aligned} \tag{4.15}
$$

From (4.5) and (4.6), the difference between the dual price at $(l+1)$-th iteration and the optimal dual solution is

$$||\mu^i_{l+1} - \mu^{i*}||_2^2 \leq ||\mu^i_l - \mu^{i*}||_2^2 + 2\alpha(q^i(\mu^i_l) - q^{i*}) + \alpha^2||g^i_l||_2^2, \tag{4.16}$$

$$||\mu^i_{l+1} - \mu^{i*}||_2^2 \leq ||\mu^i_0 - \mu^{i*}||_2^2 + 2\sum_{j=0}^{l}\alpha(q^i(\mu^i_j) - q^{i*}) + \sum_{j=0}^{l}\alpha^2||g^i_j||_2^2. \tag{4.17}$$

Combining (4.15) and (4.17), the variation of the primal variables can be found for a specific iteration. Note that in (4.15) and (4.17), the upper bound of the variation of the primal variables is related to the optimal dual function value q^{i*} and optimal dual price μ^{i*}. For all agents, they do not know the exact values of q^{i*} and μ^{i*}. In order to let each agent be able to compute this upper bound individually, we assume there exists a center, which should transmit the quantized values of q^{i*} and μ^{i*} to each agents in the system.

Using the above analysis, one could measure the necessary number of iterations for updating the dual prices to ensure that the primal variables lie in the desired target set.

Next, we quantify the optimality loss. There are different methods to compute the optimal loss for suboptimal solutions. Here we directly compute the difference of the suboptimal cost function and the optimal cost function value. For any given iterate of r, each agent can compute the corresponding optimal control input u^i_k and state x^i_k using (4.3). Substitute these control input and state into the cost function and which gives us a higher suboptimal cost function value comparing to the optimal cost function value. The upper bound of the optimality loss is given by

$$
\begin{aligned}
&\sum_{k=0}^{T-1}(x_k^i - r^*)'Q^i(x_k^i - r^*) + \sum_{k=0}^{T-1}(u_k^i - u_r^*)'R^i(u_k^i - u_r^*) + (x_T^i - r^*)'Q^i(x_T^i - r^*) \\
&- \sum_{k=0}^{T-1}(x_k^{i*} - r^*)'Q^i(x_k^{i*} - r^*) + \sum_{k=0}^{T-1}(u_k^{i*} - u_r^*)'R^i(u_k^{i*} - u_r^*) + (x_T^{i*} - r^*)'Q^i(x_T^{i*} - r^*) \\
&\leq \sum_{k=0}^{T-1}[\lambda_{\max}\cdot(||r^i - r^*||^2 + 2||x_k^i - r^i||||r^i - r^*|| + ||x_k^i - r^i||^2) - \triangle_k] + \\
&\sum_{k=0}^{T-1}[\tilde{\lambda}_{\max}\cdot(||u_r^i - u_r^*||^2 + 2||u_k^i - u_r^i||||u_r^i - u_r^*|| + ||u_k^i - u_r^i||^2) - \tilde{\triangle}_k] + \\
&\bar{\lambda}_{\max}\cdot(||r^i - r^*||^2 + 2||x_T^i - r^i||||r^i - r^*|| + ||x_T^i - r^i||^2) - \bar{\triangle}_T,
\end{aligned}
\tag{4.18}
$$

where

$$
\begin{aligned}
&\triangle_k := \lambda_{\min}\cdot(||x_k^{i*} - r^*||^2),\ \bar{\triangle}_T := \bar{\lambda}_{\min}\cdot(||x_T^{i*} - r^*||^2),\ \tilde{\triangle}_k := \tilde{\lambda}_{\min}\cdot(||u_k^{i*} - u_r^*||^2), \\
&\lambda_{\max} := \lambda_{\max}(Q^i),\ \lambda_{\min} := \lambda_{\min}(Q^i),\ \tilde{\lambda}_{\max} := \lambda_{\max}(R^i),\ \tilde{\lambda}_{\min} := \lambda_{\min}(R^i), \\
&\bar{\lambda}_{\max} := \lambda_{\max}(Q_T^i),\ \bar{\lambda}_{\min} := \lambda_{\min}(Q_T^i).
\end{aligned}
\tag{4.19}
$$

If we have information or bounds about these quantities in (4.19), then we can use the deviation of the primal variable, i.e. $||r^i - r^{i*}||$ to infer the optimality loss for certain iteration steps. We should note that since in (4.18) we substitute the optimal consensus point r^* into the cost function to compute the suboptimal cost function value, the upper bound is conservative. The reason is that for any given feasible iterate of r, each agent can compute the corresponding optimal control and state to minimize the cost. Therefore the actual suboptimal cost is less than the one we compute in (4.18).

4.2 Incremental Primal Decomposition

In this section, a consensus seeking problem based on incremental primal decomposition is discussed ([15]). Incremental primal decomposition is a variant of the primal decomposition method, for which any intermediate iterate of the algorithm is feasible.

Because any suboptimal solution is feasible, we can adopt endpoint constraints to ensure the stability of the receding horizon control system. The incremental primal decomposition approach assumes that at each iteration of the public variable update, each agent updates its iterate incrementally, through a sequence of N steps, where N is the total number of agents in the system. Here we assume that the communication network has a ring topology. When each agent updates the public variable, only its local objective function is used. After all agents contribute to the update of the public variable, a cycle is complete. The total number of iterations of the incremental primal method is the product of the cycle number K and agent number N.

For a consensus seeking problem set-up, the agents do not know the consensus point in advance. The consensus seeking among an N-agent system is formulated as follows:

$$\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{N} \sum_{k=0}^{T-1} (x_k^i - r)' Q^i (x_k^i - r) + (u_k^i - u_r)' R^i (u_k^i - u_r) + \\
& (x_T^i - r)' Q_T^i (x_T^i - r) \\
\text{subject to} \quad & x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& x_k^i \in \mathcal{X}^i, \\
& u_k^i \in \mathcal{U}^i, \\
& x_T^i = r, \ \forall i = 1, 2, ... N, \\
& r \in \Theta.
\end{aligned} \tag{4.20}$$

where $A^i \in \Re^{n \times n}$, $B^i \in \Re^{n \times m}$, Q^i, R^i, and Q_T^i are symmetric positive definite matrices of appropriate dimensions. And $\mathcal{X}^i \subseteq \Re^n$, $\mathcal{U}^i \subseteq \Re^m$ for $i = 1, ..., N$, and $\Theta \subseteq \Re^n$. Note that the consensus point r is a decision variable in each subproblem, and it is the target that all agents would like to converge to. Now we use primal decomposition to solve the problem in a distributed way.

Given $y \in \Theta$, the public resource, each agent can compute the following optimization problem individually

$$
\begin{aligned}
f^i(y) := \quad & \min \textstyle\sum_{k=0}^{T-1} (x_k^i - y)' Q^i (x_k^i - y) + (u_k^i - u_y)' R^i (u_k^i - u_y) + \\
& (x_T^i - y)' Q_T^i (x_T^i - y) \\
& \text{subject to } x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& \qquad\qquad x_k^i \in \mathcal{X}^i, \\
& \qquad\qquad u_k^i \in \mathcal{U}^i, \\
& \qquad\qquad x_T^i = y,
\end{aligned}
\tag{4.21}
$$

where u_y is the corresponding steady state control input with the public resource y, i.e. $y = A^i y + B^i u_y$.

After every agent solves the problem in (4.21), the public variable y is updated as:

$$
y_{l+1} = \mathcal{P}_\Theta[y_l - \alpha \textstyle\sum_{i=1}^N g_l^i], \tag{4.22}
$$

where g_l^i is a subgradient (in this setup, the gradient) of f^i at y_l. $\mathcal{P}_\Theta$ is the projection operator on set Θ, and the set Θ is the set of feasible consensus points.

If y_l is the iterate after l cycles, then y_{l+1} is found by the following incremental algorithm:

$$
\begin{aligned}
& \psi_l^i = \mathcal{P}_\Theta[\psi_l^{i-1} - \alpha g_{i-1,l}], \;\; i = 1, 2, ..., N, \\
& \psi_l^0 = y_l, \\
& \psi_l^N = y_{l+1},
\end{aligned}
\tag{4.23}
$$

where $g_{i-1,l}$ is the subgradient, in our case gradient of f^i at ψ_l^{i-1}.

In order to quantify the optimality loss, now assume that the gradients are bounded by a constant C, i.e. $||g|| \leq C$. Based on Proposition 2.3 in [15], the optimality loss for suboptomal solutions is given by:

$$\min_{0\geq l\leq K} \sum_{i=1}^{N} f^i(y_l) \leq f^* + \frac{\alpha N^2 C^2+\varepsilon}{2}, \tag{4.24}$$

where, $K = \lfloor \frac{(dist(y_0,\Theta^*))^2}{\alpha\varepsilon} \rfloor$, and $dist(y_0, \Theta^*)$ is the Euclidean distance from the point y_0 to the set of optimal solutions Θ^*. Therefore, one can use (4.24) to measure the necessary cycle number for a given optimality loss, and use the analysis in Chapter 3 to achieve a stable receding horizon control system within a tolerable error range.

Note that to solve (4.23), each agent can relay on its own objective function and the associated subgradient or gradient. This means that agents do not need to exchange information to compute the subgradient (or gradient) as in the dual decomposition method. Each agent only needs to pass the computed iterate to the next adjacent neighbor. In order to measure the necessary number of cycles, the information about the Euclidean distance from the initial iterate to the optimal solution set should be known by the agents. In a practical implementation, only one agent needs to compute the upper bound of the optimality loss, and thus only this agent needs to know the information about the distance from the initial iterate to the optimal set. When the optimality loss is guaranteed to be within the desired margin, the update cycles are terminated, and one more cycle is needed to pass the most recent estimate of the public variable to each agent to ensure that all agents use the same value when computing their control input. The total number of subiterations of the incremental primal decomposition algorithm is $N \cdot K$.

4.3 Incremental Robustness

This section focuses on a receding horizon control - primal decomposition framework in which performance can be guaranteed with the addition of agents to the system. This is what we mean by "incremental robustness." The incremental primal decomposition

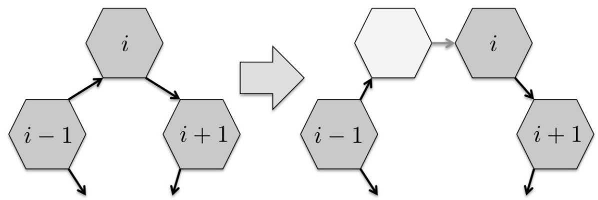

Figure 4.1: A system with the addition of agent

method is crucial, not only for the purpose of designing decentralized algorithms, but also to ensure that there is no need for redesign when agents are added to the system even in the presence of physical coupling (see Fig. 4.1). A system topology that allows for this type of "plug-and-play" featuring the ring topology that has been illustrated in the previous subsection.

Moreover, a similar approach to the previous discussions can be used to quantify the performance degradation of the primal decomposition algorithm when new nodes are added to the system. This approach enables us to find limitations on the number of nodes the system can accommodate while the performance degrades gracefully within a tolerable range.

Specifically, when there are additional agents entering into the system, in order to keep the same stability guarantees, the original agents need to recompute the optimality loss of the previous subproblem. They need to increase the number of cycles to achieve a smaller loss, which is a function of the cost incurred by the new agents. The computation analysis of the new number of cycles which is necessary to ensure the same stability guarantees of the new system is as follows.

Assume there are M extra agents entering into the system at the $(k + 1)$-th subproblem, the difference of the optimality loss $D(M, r)$ is given by

$$D(M, r) = \min \sum_{i=N+1}^{N+M} \sum_{j=0}^{T-1} h^i(x^i_{j,k} - r, u^i_{j,k} - u_r), \tag{4.25}$$

where r is the suboptimal consensus point at k-th subproblem, and the minimization is with respect to the control inputs. Then the new number of cycles necessary to keep the same stability guarantee of the system is given by

$$K_{\text{new}} = \lfloor \frac{(dist(y_0, \Theta^*))^2}{\alpha \varepsilon_{\text{new}}} \rfloor, \tag{4.26}$$

where $\varepsilon_{\text{new}} = \varepsilon - D(M, x_s^{(N)})$. Note that since ε_{new} has to be positive, this equation can be used to quantify the limitation of the number of agents and their initial states that can be accommodated while keeping the same stability guarantees. Moreover, if the designer is willing to accept degradation of the stability guarantees, a similar analysis can be used to compute the number of cycles of the incremental primal decomposition algorithm.

Chapter 5

A Formation Control Strategy Based on Dual and Primal Decomposition

In this chapter, we consider a formation control problem. Assume a linear dynamics and quadratic cost with endpoint penalty. The following constrainted minimization problem is given by:

$$
\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{N}\sum_{k=0}^{T-1}(x_k^i - r^i)' Q^i (x_k^i - r^i) + (u_k^i - u_r^i)' R^i (u_k^i - u_r^i) + \\
& (x_T^i - r^i)' Q_T^i (x_T^i - r^i) \\
\text{subject to} \quad & x_{k+1}^i = A^i x_k^i + B^i u_k^i \\
& x_k^i \in \mathcal{X}^i, \\
& u_k^i \in \mathcal{U}^i, \\
& x_T^i \in \mathcal{X}_D^i \\
& x_k^i = x_k^j + d^i, \quad \forall j \in \mathcal{N}_i, \quad i = 1, 2, ..., N.
\end{aligned}
\tag{5.1}
$$

where $A^i \in \Re^{n\times n}$, $B^i \in \Re^{n\times m}$, Q^i, R^i, and Q_T^i are symmetric positive definite matrices of appropriate dimensions. And $\mathcal{X}^i \subseteq \Re^n$, $\mathcal{U}^i \subseteq \Re^m$ for $i = 1, ..., N$, and $r^i \in \Theta^i \subseteq \Re^n$ for $i = 1, 2, ..., N$. $d^i \in \Re^n$ for $i = 1, 2, ..., N$. As an example, these state consistency constraints, $x_k^i = x_k^j + d^i, \quad \forall j \in \mathcal{N}_i, \quad i = 1, 2, ..., N$, represent the formation of vehicles' locations.

First, we use dual decomposition method to solve this problem in a distributed way. For each consistency constraint, we assign a dual price to control its evolution. Then simply use subgradient ascent method to update these dual prices. Agents need to exchange their primal variables to compute the associated subgradients. Since before the algorithm converges, the primal variables are not feasible, it means that the consistency constraints are not satisfied until the solution is optimal. However, if one would like to apply the result of *Theorem 3.2*, the solution must be feasible. It does not need to be optimal, though. Therefore, it is necessary to construct a feasible primal solution at each iteration of the dual decomposition algorithm. One straightforward way to find a feasible solution is using average estimate, i.e. for each k, $x_k^i = 1/N \sum_{j=1}^{N} x_k^j$ for $i = 1, 2, ..., N$ at each iteration for $d^i = 0$ case. If there are no state and control constraints, the average estimate is a feasible solution. If there are state $\mathcal{X}^i$, and control constraints $\mathcal{U}^i$, then it is possible that the average estimate is not feasible, i.e. there does not exist a control sequence such that the average states can be attained for all agents. One way to construct a feasible solution which satisfies all constraints is that for each neighborhood, the members in the neighborhood exchange the primal solutions to test if there exist feasible solutions, then propagate it to the next neighborhood. Repeat this procedure until all agents get the feasible primal solutions. At each iteration, the dual function value is the lower bound of the optimal value function. Furthermore, after each agent finds the feasible solution, the value function of this feasible solution becomes the upper bound of the optimal value function. Use these upper and lower bounds of the optimal value function, one can compute the optimality loss at each iteration of the dual decomposition algorithm. Finally, based on *Theorem 3.2* the associated performance loss of the receding horizon control system can be computed.

Secondly, we can use primal decomposition to solve the same problem. In dual decomposition, we use dual prices to adjust the necessary resources in order to minimize the overall cost function. However, in primal decomposition, one assigns the same amount of resources to each agent, each agent minimizes its own cost function based on this fixed resource, then we minimize the cost function with respect to this resource. The advantage of using primal decomposition is that at each iteration, the primal solution is feasible. Similarly, the optimality loss can be obtained as in the dual decomposition method, and again we can use *Theorem 3.2* to relate the optimality loss to the stability of the receding horizon control system.

The primal decomposition is given by the following formulation. In the first step, each agent solves the following minimization problem individually based on a given public resource y.

$$\begin{aligned} f^i(y) := \quad & \min \textstyle\sum_{k=0}^{T-1} (x_k^i - r^i)' Q^i (x_k^i - r^i) + (u_k^i - u_r^i)' R^i (u_k^i - u_r^i) + \\ & (x_T^i - r^i)' Q_T^i (x_T^i - r^i) \\ & \text{subject to } x_{k+1}^i = A^i x_k^i + B^i u_k^i \\ & \qquad\qquad x_k^i \in \mathcal{X}^i, \\ & \qquad\qquad u_k^i \in \mathcal{U}^i, \\ & \qquad\qquad x_T^i \in \mathcal{X}_D^i \\ & \qquad\qquad x_k^i = y + d^i. \end{aligned} \tag{5.2}$$

In the second step of the primal decomposition, we minimize $\sum_{i=1}^{N} f^i(y)$ with respect to y. This can be done by subgradient descent method, $y := y - \alpha \sum_{i=1}^{N} g^i$, where α is a constant step size, and g^i is the associated subgradient of $f^i(y)$.

Chapter 6

Simulation

In this chapter, we use simulation examples to illustrate the results of the study. It is important to note that the results in the thesis should be viewed as analytical guarantees of receding horizon control stability under suboptimality. The simulation is used to illustrate some of the ideas used in the thesis, but no claims on tightness are made.

First, consider a case that there are three agents in the system ($N = 3$) with prediction horizon $T = 10$. The state and control dimensions are $n = 2$, $m = 2$, respectively. $A^1 = [0.3, 0.5; 0.2, 0.6]$ and $B^1 = [0.1, 0.3; 0.6, 0.2]$, $A^2 = A^3 = [0.5, 0.2; 0.6, 0.1]$ and $B^2 = B^3 = [0.5, 0; 0, 0.5]$. Q^i and R^i, $i = 1, 2, 3$ are identity matrices with proper dimensions. The reference r is set to be $[5, 5]'$. Each state element is constrained to lie in a closed interval from -10 to 10, while each control element is constrained to lie in a closed interval from -20 to 20. The initial states for each agent are given by $x_0^1 = [8, 8]'$, $x_0^2 = [10, 10]'$, and $x_0^3 = [7, 7]'$, respectively. The states represent the 2-D locations of the agents. We allow agent 1 to be able to communicate with agent 2 and agent 3, but agent 2 and agent 3 cannot exchange the information. Dual decomposition is used to solve a leader-follower tracking problem. In a leader-follower tracking scenario, after the algorithm converges, all x_s^i's converge to the same value, and which is set to be the desired target that each agent need to track. In this scenario, only the leader knows this reference target and all other followers do not know it exactly. After putting extra consistency constraints on each agent's own reference target, all followers tend to track the same reference target. In this leader-follower tracking scenario, the leader knows the desired target to track, and followers do not know this reference target exactly. It is different from the usual consensus or rendezvous seeking problem, in which all agents do not know the consensus point in advance. The step size to update dual variables is 0.5. We assume that agent 1 is the leader, and agent 2 and agent 3 are followers. Fig. 6.1 represents the result of the first subproblem using 250 iterations to update dual prices, and the evolution of the dual function value is in Fig. 6.2. In Fig. 6.3, Fig. 6.4, and Fig. 6.5, we present the suboptimal solution results. In Fig. 6.3, the algorithm converges. In Fig. 6.4, the optimality loss is 20. In Fig. 6.5, the optimality loss is 40. For the real realizations, the norms of the deviation from the target are 0.5685, 0.8645, and 1.5400 in the cases that optimality loss is 0, 20, and 40, respectively. When the optimality loss is zero, the total number of iterations is 250 for each subproblem. When the optimality loss is 20, the numbers of iterations are 105, 85, 70, 55, 37 for the first, second, third, forth, and fifth subproblem. When the optimality loss is set to be 40, the numbers of iterations are 85, 59, 39, 23, and 11 for the corresponding subproblems. One can compare the results of iterations, optimality loss, and the norms of the deviations to find a appropriate trade off between the suboptimal solutions and the stability of the receding horizon control system. In this leader-follower tracking example, the stage cost $h(\cdot,\cdot)$, final cost $h_T(\cdot)$, final state invariant set $\mathcal{X}_D$, and final state local feedback control $\sigma(\cdot)$ are chosen arbitrarily. We have not computed the necessary iterations to let the final state lie in the invariant sets. Since those elements are chosen arbitrarily and not necessarily satisfy the sufficient conditions in *Theorem 3.2*, we cannot apply the theorem directly. One could use the methods summarized in Chapter 3 to compute these elements, then apply the procedure mentioned in Chapter 3 to find the necessary number of iterations to let the final states of all agents lie in the invariant sets. And also apply *Theorem 3.2* to compute the upper bound of the norms of the deviations from the target. We use this example to illustrate the idea of using less iteration steps to achieve a stable receding horizon control system within certain error range in a leader-follower tracking set-up.

For the second example, we consider the scalar and unconstrained case for simplicity. In this example the system is comprised of three agents that negotiate to find a consensus point. The dynamics of the agents are given by: $A^1 = 0.8,\ B^1 = 0.75,\ A^2 = 0.5,\ B^2 = 0.35$, and $A^3 = 0.7,\ B^3 = 0.55$. The initial states were randomly generated in Matlab. The prediction horizon is $T = 10$. The step size for the incremental primal decomposition is 0.01. The results of the first receding horizon subproblem can be seen in Fig. 6.6. Here we demonstrate how the primal decomposition works within one receding horizon subproblem. The total number of iterations is 50, and the optimal consensus point is 4.54.

In Fig. 6.7 and Fig. 6.8, we present the state evolutions of the receding horizon control both for the converging case and suboptimal case. The set-up is the same as in Fig. 6.6, except that the initial states are $x_0^1 = 10,\ x_0^2 = -20,\ x_0^3 = -5$. In Fig. 6.7 we use a large number of iterations (50) to simulate the case where convergence is achieved. The consensus point is -3.94. On the other hand, for the suboptimal case in Fig. 6.8, we artificially impose a suboptimality of 30 to demonstrate the idea of suboptimal receding horizon control. When the agents are close, we relax the suboptimality imposition. Here the consensus point is approaching zero.

In Fig. 6.9, we maintain the same suboptimality as before and add an extra agent to the system at time $k = 4$, however we let the previous subproblem converge to compensate for the extra cost incurred by the new agent. The simulation suggests that the system with the added agent converges within the same error range as the system without the addition.

In the last example, consider a scalar three agent formation control problem. $A^1 = 0.8$, $B^1 = 0.75$, $A^2 = 1$, $B^2 = 0.85$, $A^3 = 0.8$, and $B^3 = 0.75$ with prediction horizon $T = 5$. $Q^i = 1$ and $R^i = 1$, for $i = 1, 2, 3$, respectively. The final cost $Q^i_T = 2$, for $i = 1, 2, 3$. The reference r is set to be 5. Each state element is constrained to lie in a closed interval from -10 to 10. Agent 1 and agent 3's control element is constrained in the closed interval from -10 to 10. Agent 2's control element is constrained in the closed interval from -0.5 to 0.5. The initial states for each agent are given by $x^1_0 = 10$, $x^2_0 = 8$, and $x^3_0 = 1$, respectively. We add formation constraints such that the locations of all agents should be the same at time 1, 2, and 3. The targets of all agents are also the same for these three agents. For the first subproblem, the predicted state trajectory is in Fig. 6.11, and the predicted control sequence is in Fig. 6.12. The evolution of the corresponding dual function value with respect to the iterations is in Fig. 6.13. The state trajectory of the receding horizon control system is in Fig. 6.14. Now we apply *Theorem 3.2* to compute the associated upper bound of the norm of the deviation from the target. Given that the optimality loss is 4, the upper bounds of the stability results are 14, 11.07, and 9 at time 1, 2, and 3 respectively. The stability results of actual realizations at time 1, 2, 3 are 4.48, 3.76, and 3.04, respectively.

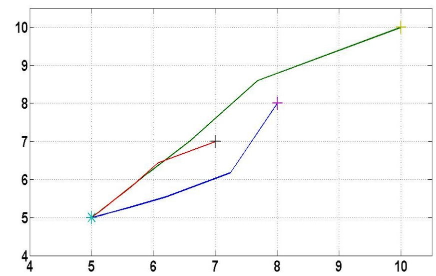

Figure 6.1: Optimal predicted state trajectory of the first subproblem. Horizontal axis: horizontal location; vertical axis: vertical location; cross: initial location; star: the target; green line: trajectory of agent 2; blue line: trajectory of agent 1; red line: trajectory of agent 3.

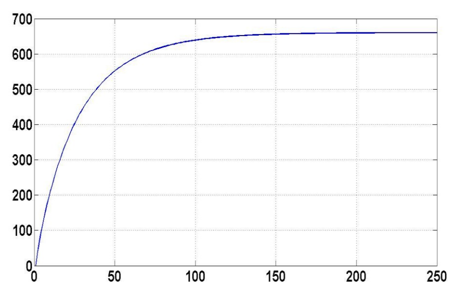

Figure 6.2: Dual function value evolution of the first subproblem. Horizontal axis: iteration.

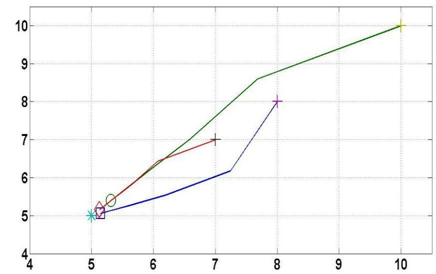

Figure 6.3: State trajectory (optimality loss: 0). Horizontal axis: horizontal location; vertical axis: vertical location; cross: initial location; star: the target; green line: trajectory of agent 2; blue line: trajectory of agent 1; red line: trajectory of agent 3.

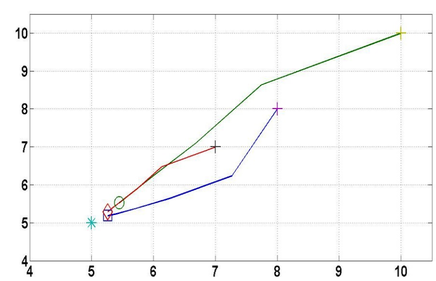

Figure 6.4: State trajectory (optimality loss: 20). Horizontal axis: horizontal location; vertical axis: vertical location; cross: initial location; star: the target; green line: trajectory of agent 2; blue line: trajectory of agent 1; red line: trajectory of agent 3.

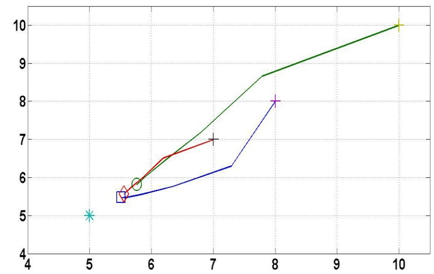

Figure 6.5: State trajectory (optimality loss: 40). Horizontal axis: horizontal location; vertical axis: vertical location; cross: initial location; star: the target; green line: trajectory of agent 2; blue line: trajectory of agent 1; red line: trajectory of agent 3.

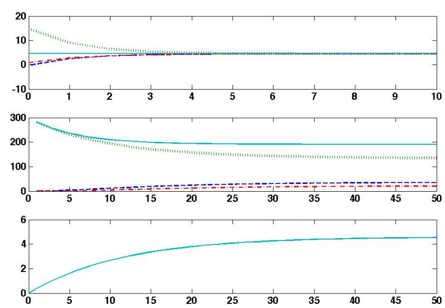

Figure 6.6: The results of the first receding horizon subproblem. On the top: y-axis: the state trajectories of agents; x-axis: prediction horizon; solid line is the optimal consensus. In the middle: y-axis: the cost function values of agents; x-axis: iterations; solid line is the sum of the individual cost function values. On the bottom: The consensus evolution versus iterations.

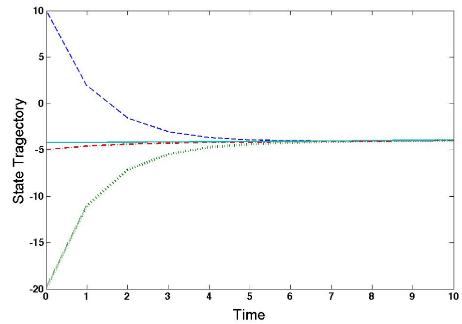

Figure 6.7: State trajectories versus time for converging case.

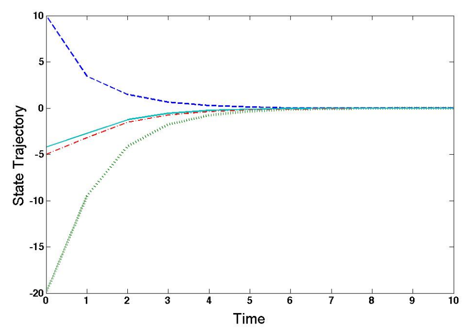

Figure 6.8: State trajectories versus time for suboptimal case.

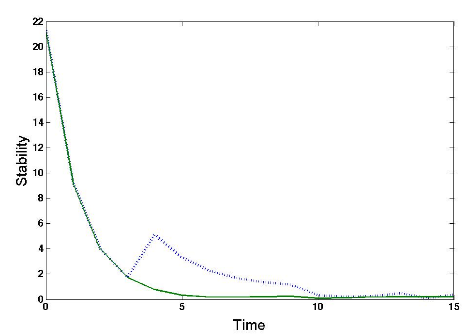

Figure 6.9: Stability versus time. Solid line: the system without addition; dot line: the system with addition.

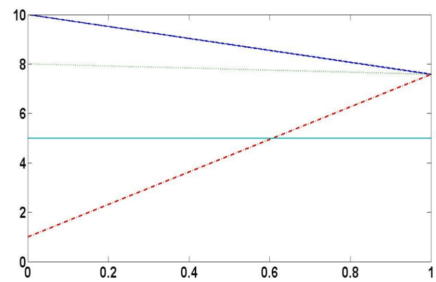

Figure 6.10: The state trajectory of the receding horizon control at time 1. Horizontal axis: time index; green dot line: trajectory of agent 2; blue dot line: trajectory of agent 1; red dot line: trajectory of agent 3.

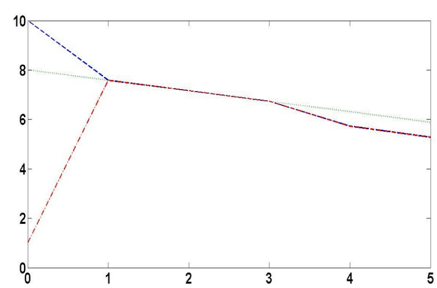

Figure 6.11: The predicted state trajectory of the first subproblem. Horizontal axis: time index; green dot line: trajectory of agent 2; blue dot line: trajectory of agent 1; red dot line: trajectory of agent 3.

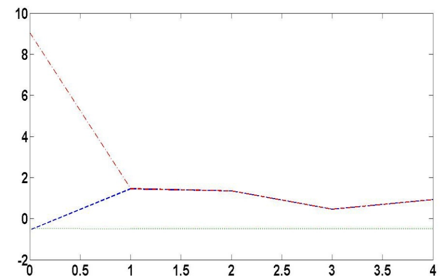

Figure 6.12: The predicted control input of the first subproblem. Horizontal axis: time index; green dot line: the control sequence of agent 2; blue dot line: the control sequence of agent 1; red dot line: the control sequence of agent 3.

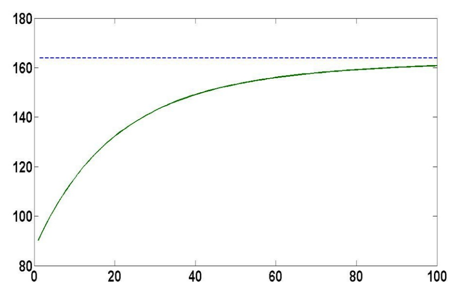

Figure 6.13: The evolution of the dual function value of the first subproblem. Horizontal axis: iteration; blue dot line: the value function of the feasible solution; green line: dual function value.

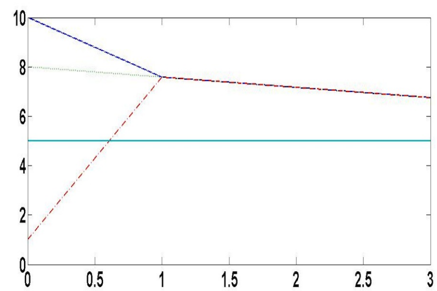

Figure 6.14: The state trajectory of the receding horizon control. Horizontal axis: time index; green dot line: trajectory of agent 2; blue dot line: trajectory of agent 1; red dot line: trajectory of agent 3.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, an incremental primal decomposition approach is used to solve a consensus problem, and the stability for suboptimal solutions is also discussed ([12]). In addition to our previous work ([12]), its results are extended in this thesis to the scenario with dual decomposition. We use sensitivity analysis and first order approximation to quantify the primal variables as dual prices change. Finally, The formation control problem based on dual and primal decompositions is also considered in the thesis.

7.2 Future Work

As [6] points out, approaches to decentralized and distributed receding horizon control design differ from each other in the assumptions such as the structure of interaction between different systems, the structure of interaction between constraints, objective functions, dynamics, the model of system (such as linear, non-linear, hybrid, constrained, no constrained, discrete-time, continuous-time), and the model of information exchange between the systems. Different scenarios usually require different assumptions and approaches to deal with. In particular, if we consider the existence of random noises or deterministic disturbances, then the situation becomes more challenging and interesting. Therefore, a unified approach for robust distributed and decentralized receding horizon control to ensure desired properties would be necessary. It is thus an open problem and future research direction in the study of receding horizon control.

Secondly, we try to adopt the approximate saddle points analysis in [16] to quantify the deviation of the primal variables in the dual decomposition. Basically, dual decomposition method is an iterative algorithm to construct the saddle points of the associated Lagrange function. In [16], the approximate saddle points are constructed based on subgradient descent method for primal variable and subgradient ascent method for dual variable. Then the average estimates are used to construct the approximate saddle points. The reason for using subgradient descent method to update primal variable is because in some cases, the optimal solution of the primal problem or the subgradient of the dual problem cannot be found exactly. In our case, since now we use subgradient descent to update the primal variable which is an iterative method, the variation of the primal variable can be found directly.

Moreover, we have in mind of introducing the methodology of robust optimization [3] to deal with the uncertainties in the system. In contrast to stochastic optimization assuming uncertainties with a probabilistic description, a more recent optimization-theoretical concept assuming uncertainty model is deterministic and set-based. Feasible solutions could be constructed for any realization of the uncertainty in a given set by decision makers. As a function of the type of uncertainty set and the structure of the nominal problem, it would lead to different results of tractability [3]. Since in principle, solving a receding horizon control problem is equivalent to solving an optimization problem, the single-shot robust optimization can be extended to sequential decision-making by using receding horizon control. However, if we only try to implement robust optimization in a receding horizon control form, we will not get any adaptability. And the results might lead to a solution far from optimal. How to design an adaptable robust optimization based receding horizon control would therefore be desirable.

Appendix A

Proof of Theorem 3.2

Proof: Consider k-th and $(k+1)$-th suproblems, and define the corresponding predicted control inputs as $U_k := (u_{0,k}, ..., u_{T-1,k})$ and $U_{k+1} := (u_{1,k}, ..., u_{T-1,k}, \sigma(x_{T,k}))$.

(1) If *Problem 3.1* is feasible at time k for the initial state $x_k \in \mathcal{X}$, then there exists a control sequence $U_k \in \mathcal{U}_T(x_k)$ such that $x_{T,k} \in \mathcal{X}_D$. Since we choose U_{k+1} to be $(u_{1,k}, ..., u_{T-1,k}, \sigma(x_{T,k}))$, $x_{T,k+1} \in \mathcal{X}_D$ also, and this means $U_{k+1} \in \mathcal{U}_T(x_{k+1})$. For any initial condition $x_k \in \mathcal{X}_D$, apply $\sigma(x_k)$ into the dynamics, then the updated state will still lie in $\mathcal{X}_D$. And it shows that $\mathcal{X}_D \subseteq \mathcal{X}_f(T)$.

(2) First, because of the first sufficient condition,

$$J^*(x) \geq h(x, u^{RH}(x)) \geq \alpha_1(||x||), \forall x \in \mathcal{X}_f(T), \tag{A.1}$$

where $u^{RH}(x)$ is the first control value in the optimal solution for the given initial condition $x \in \mathcal{X}_f(T)$.

Secondly,

$$\begin{aligned} &J^*(x_{k+1}) - J^*(x_k) \leq J(x_{k+1}, U_{k+1}) - J(x_k, U_k) + \delta \\ &= -h(x_k, u_{0,k}) - h_T(x_{T,k}) + h_T(x_{T,k+1}) + h(x_{T,k}, \sigma(x_{T,k})) + \delta, \end{aligned} \tag{A.2}$$

and because of the first and third sufficient condition, which implies that

$$J^*(\phi(x_k, u_k^{RH\_\delta})) - J^*(x_k) \leq -h(x_k, u_{0,k}) + \delta \leq -\alpha_1(||x_k||) + \delta, \ \forall x_k \in \mathcal{X}_f(T). \tag{A.3}$$

Thirdly, directly apply Theorem 3.3.3 in [8],

$$J^*(x) \leq h_T(x) \leq \alpha_2(||x||), \forall x \in \mathcal{X}_f(T). \tag{A.4}$$

Finally, applying *Theorem 3.1*, we conclude that $x_{k+1} = \phi(x_k, u_k^{RH\_\delta})$ is $\varepsilon - AS$ *in* $\mathcal{X}_f(T)$. Note that in this case, $d = 0$, $e = \delta$, and $a = c$, therefore $\varepsilon(\delta) := (\frac{2b}{a^2}\delta)^{1/p}$.

□

Appendix B

Proof of Corollary 3.1

Proof: Consider k-th and (k + 1)-th suproblems, and define the corresponding predicted control inputs as $U_k := (u_{0,k}, ..., u_{T-1,k})$, $U_{k+1} := (u_{1,k}, ..., u_{T-1,k}, \sigma(x_{T,k}))$.

(1) If *Problem 3.1* is feasible at time k for the initial state $x_k \in \mathcal{X}$, then there exists a control sequence $U_k \in \mathcal{U}_T(x_k)$ such that $x_{T,k} \in \mathcal{X}_D(k)$. Since we choose U_{k+1} to be $(u_{1,k}, ..., u_{T-1,k}, \sigma(x_{T,k}))$, $x_{T,k+1} \in \mathcal{X}_D(k) \subseteq \mathcal{X}_D(k+1)$ also, and this means $U_{k+1} \in \mathcal{U}_T(x_{k+1})$. For any initial condition $x_k \in \mathcal{X}_D(k)$, apply $\sigma(x_k)$ into the dynamics, then the updated state will still lie in $\mathcal{X}_D(k)$. And it shows that $\mathcal{X}_D(k) \in \mathcal{X}_f(T)$.

(2) First, because of the first sufficient condition,

$$J^*(x, x_s^*) \geq h(x - x_s^*, u^{RH}(x) - u_s^*) \geq \alpha_1(||x - x_s^*||),\ \forall x \in \mathcal{X}_f(T),\ x_s^* \in \Theta, \tag{B.1}$$

where $u^{RH}(x)$ is the first control value in the optimal solution for the given initial condition $x \in \mathcal{X}_f(T)$.

Secondly, given that $\tilde{x}_s^*$ and x_s^* are the optimal consensus points for $(k+1)$ and k-th subproblem respectively, we have

$$\begin{aligned} &J^*(x_{k+1}, \tilde{x}_s^*) - J^*(x_k, x_s^*) \leq J(x_{k+1}, U_{k+1}, x_s) - J(x_k, U_k, x_s) + \delta \\ &= -h(x_k - x_s, u_{0,k} - u_s) - h_T(x_{T,k} - x_s) \\ &\quad + h_T(x_{T,k+1} - x_s) + h(x_{T,k} - x_s, \sigma(x_{T,k}) - u_s) + \delta \end{aligned} \tag{B.2}$$

because of the first, third, and forth sufficient conditions, which implies that

$$\begin{aligned} &J^*(\phi(x_k, u_k^{RH\_\delta}) - \tilde{x}_s^*) - J^*(x_k - x_s^*) \leq -h(x_k - x_s, u_{0,k} - u_s) + \delta \\ &\leq -\alpha_1(||x_k - x_s||) + \delta \leq -\alpha_3(||x_k - x_s^*||) + \delta,\ \forall x_k \in \mathcal{X}_f(T),\ x_s^* \in \Theta. \end{aligned} \tag{B.3}$$

Thirdly, directly apply Theorem 3.3.3 in [8]

$$J^*(x - x_s^*) \leq h_T(x - x_s^*) \leq \alpha_2(||x - x_s^*||), \ \forall x \in \mathcal{X}_f(T), \ x_s^* \in \Theta. \tag{B.4}$$

Finally, applying *Theorem 3.1*, we conclude that $x_{k+1} = \phi(x_k, u_k^{RH\_\delta})$ is $\varepsilon - AS \ in \ \mathcal{X}_f(T)$ with the origin replaced by x_s^*. Note that in this case, $d = 0$ and $e = \delta$, therefore $\varepsilon(\delta) := (\frac{2b}{ac}\delta)^{1/p}$ □

Appendix C

Proof of Corollary 3.2

Proof: Consider k-th and (k + 1)-th suproblems, and define the corresponding predicted control inputs as $U_k := (u_{0,k}, ..., u_{T-1,k})$, $U_{k+1} := (u_{1,k}, ..., u_{T-1,k}, u_s)$, where u_s is the corresponding steady state control with steady state x_s. And $x_{T,k} = x_s$.

(1) If *Problem 3.1* is feasible at time k for the initial state $x_k \in \mathcal{X}$, then there exists a control sequence $U_k \in \mathcal{U}_T(x_k)$ such that $x_{T,k} = x_s$. Since we choose U_{k+1} to be $(u_{1,k}, ..., u_{T-1,k}, u_s)$, $x_{T,k+1} = x_s$ also, and this means $U_{k+1} \in \mathcal{U}_T(x_{k+1})$.

(2) First, because of the first sufficient condition,

$$J^*(x, x_s^*) \geq h(x - x_s^*, u^{RH}(x) - u_s^*) \geq \alpha_1(||x - x_s^*||),\ \forall x \in \mathcal{X}_f(T),\ x_s^* \in \Theta, \tag{C.1}$$

where $u^{RH}(x)$ is the first control value in the optimal solution for the given initial condition $x \in \mathcal{X}_f(T)$.

Secondly, given that $\tilde{x}_s^*$ and x_s^* are the optimal consensus points for $(k+1)$ and k-th subproblem respectively, we have

$$\begin{aligned} &J^*(x_{k+1}, \tilde{x}_s^*) - J^*(x_k, x_s^*) \leq J(x_{k+1}, U_{k+1}, x_s) - J(x_k, U_k, x_s) + \delta \\ &= -h(x_k - x_s, u_{0,k} - u_s) + h(x_{T,k} - x_s, u_s - u_s) + \delta \end{aligned} \tag{C.2}$$

because of the first and third sufficient conditions, which implies that

$$\begin{aligned} &J^*(\phi(x_k, u_k^{RH\_\delta}) - \tilde{x}_s^*) - J^*(x_k - x_s^*) \leq -h(x_k - x_s, u_{0,k} - u_s) + \delta \\ &\leq -\alpha_1(||x_k - x_s||) + \delta \leq -\alpha_3(||x_k - x_s^*||) + \delta,\ \forall x \in \mathcal{X}_f(T),\ x_s^* \in \Theta. \end{aligned} \tag{C.3}$$

Thirdly, because of the second sufficient condition,

$$\begin{aligned} &J^*(x - x_s^*) = \textstyle\sum_{j=0}^{T-1} h(x_{j,k}^* - x_s^*, u_{j,k}^* - u_s^*) \leq T\tilde{\alpha}_2(||x - x_s^*||),\ \forall x \in \mathcal{X}_f(T),\ x_s^* \in \Theta. \\ &J^*(x - x_s^*) \leq \alpha_2(||x - x_s^*||),\ \forall x \in \mathcal{X}_f(T),\ x_s^* \in \Theta. \end{aligned} \tag{C.4}$$

where $\alpha_2 = T\tilde{\alpha_2}$.

Finally, applying *Theorem 3.1*, we conclude that $x_{k+1} = \phi(x_k, u_k^{RH\_\delta})$ is $\varepsilon - AS$ *in* $\mathcal{X}_f(T)$ with the origin replaced by x_s^*. Note that in this case, $d = 0$ and $e = \delta$, therefore $\varepsilon(\delta) := (\frac{2b}{ac}\delta)^{1/p}$ □

Bibliography

[1] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, Mass, 1999.

[2] D. P. Bertsekas, *Dynamic Programming and Suboptimal Control: A Survey from ADP to MPC*, European Journal of Control: Fundamental Issues in Control, vol 11, no 4-5, 2005, pp 310-334.

[3] D. Bertsimas and M. Sim, *The Price of Robustness*, Operations Research, vol 52, no 1, 2004, pp 35-53.

[4] S. Boyd and A. Mutapcic, *Subgradient Methods*, Lecture Notes, 2007.

[5] B. Johansson, A. Speranzon, M. Johansson, K. H. Johansson, *On Decentralized Negotiation of Optimal Consensus*, Automatica, vol 44, 2008, pp 1175-1179.

[6] T. Keviczky, F. Borrelli, and G. J. Balas, *Decentralized Receding Horizon Control for Large Scale Dynamically Decoupled Systems*, Automatica, vol 42, 2006, pp 2105-2115.

[7] T. Keviczky and K. H. Johansson, *A Study on Distributed Model Predictive Consensus*, in Proceedings of IFAC 17th World Congress, Seoul, Korea, 2008, pp 1516-1521.

[8] M. Lazar, *Model Predictive Control of Hybrid Systems: Stability and Robustness*, Ph.D. Thesis, 2006.

[9] M. Lazar, W. P. M. H. Heemels, S. Weiland, and A. Bemporad, *Stabilizing Model Predictive Control of Hybrid Systems*, IEEE Trans. on Automatic Control, vol 51, no 11, 2006, pp 1813-1818.

[10] M. Lazar, W. P. M. H. Heemels, A. Bemporad, and S. Weiland, *Discrete-Time Non-smooth Nonlinear MPC: Stability and Robustness*, in Lectures Notes in Control And Information Sciences, vol 358, 2007, pp 93-103.

[11] M. Lazar and W. P. M. H. Heemels, *Predictive Control of Hybrid Systems: Input-to-State Stability Results for Sub-Optimal Solutions*, Automatica, vol 45, 2009, pp 180–185.

[12] W.-J. Ma, E. Arvelo, and N. C. Martins, *Designing Networked Control Architectures for Incremental Robustness*, Accepted for Publication in Proceedings of IFAC 18th World Congress, Milano, Italy, 2011.

[13] D. L. Marruedo, T. Alamo, and E. F. Camacho, *Input-to-State Stable MPC for Constrained Discrete-Time Nonlinear Systems with Bounded Additive Uncertainties*, in Proceedings of 41st IEEE Conference on Decision and Control, Las Vegas, Nevada, 2002, pp 4619-4624.

[14] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, *Constrained Model Predictive Control: Stability and Optimality*, Automatica, vol 36, 2000, pp 789-814.

[15] A. Nedic and D. Bertsekas, *Convergence Rate of Incremental Subgradient Algorithms*, Stochastic Optimization: Algorithms and Applications, S. Uryasev and P. M. Pardalos, (Eds.), 2000, pp 263-304.

[16] A. Nedic and A. Ozdaglar, *Subgradient Methods for Saddle-Point Problems*, J. on Optimization Theory Application, vol 142, 2009, pp 205-228.

[17] D. M. D. L. Pena, A. Bemporad, and T. Alamo, *Stochastic Programming Applied to Model Predictive Control*, in Proceedings of 44th IEEE Conference on Decision and Control, Seville, Spain, 2005, pp 1361-1366.

[18] R. Scattolini, *Architectures for Distributed and Hierarchical Model Predictive Control - A Review*, J. of Process Control, vol 19, 2009, pp 723-731.

[19] P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings, *Suboptimal Model Predictive Control (Feasibility Implies Stability)*. IEEE Trans. on Automatic Control, vol 44, no 3, 1999, pp 648–654.

[20] Y. Wang and S. Boyd, *Fast Model Predictive Control Using Online Optimization*, IEEE Trans. on Control Systems Technology, vol 18, no 2, 2010, pp 267-278.

[21] M. N. Zeilinger, C. N. Jones, D. M. Raimondo, and M. Morari, *Real-Time MPC - Stability Through Robust MPC Design*, in Proceedings of 48th IEEE Conference on Decision and Control, Shanghai, China, 2009, pp 3980-3986.