Performance and Information Pattern Trade-Offs in a Consensus Based Distributed Optimization Method

Alireza Farhadi, Michael Cantoni, Peter M. Dower

***Abstract*— In this paper, aspects of a consensus based distributed optimization method are studied. The method is applied to approximate the solution of a multi-variate Linear-Quadratic (LQ) optimal control problem via distributed decision makers. The decision makers are constrained in terms of the pattern of local computation and information exchange, as a mechanism for managing the corresponding overheads. Feasibility (i.e., constraints satisfaction by the approximated solutions), convergence, and optimality of the method are proved. Convergence to the solution of finite horizon LQ optimal control problem is illustrated for a system with six interacting linear time invariant subsystems. For this system, trade-offs between approximation error (i.e. performance loss relative to the optimal solution) and the patterns used to constrain information exchange and computation, are also illustrated.**

I. Introduction

A. Motivation and Background

In many emerging large scale systems, dynamical behavior can be modeled by n distributed interacting linear time invariant subsystems of the form [1]

$$S_i: \; x_i[k+1] \;=\; A_i x_i[k] + B_i u_i[k] + v_i[k], \quad i = 1, 2, ..., n, k \in \{0, 1, 2, ..., N-1\}, \tag{1}$$

where x_i is the state variable of the ith subsystem, u_i is the decision variable of the ith subsystem, and

$$v_i[k] = \sum_{j=1, j\neq i}^{n} M_{ij} x_j[k] + N_{ij} u_j[k] \tag{2}$$

is the interacting variable that summarizes the effect of other subsystems on S_i. For the system (1) we are interested in solving the following Linear-Quadratic (LQ) constrained optimization problem subject to the dynamics of subsystems and operational constraints $x_i[k] \in \mathcal{X}_i$ and $u_i[k] \in \mathcal{G}_i$, where $\mathcal{X}_i$ is a closed convex subset of the real Euclidean space with dimension $n_i > 0$ (i.e., $\mathcal{X}_i \subset \mathbf{R}^{n_i}$) modeling the constraint set on the ith state variable, and $\mathcal{G}_i$ is a closed convex subset of $\mathbf{R}^{m_i}$ modeling the time invariant constraint set on the ith decision variable. That is,

$$\min_{u} \Big\{ J(\mathbf{x}[0], u_1, ..., u_n), x_i[k] \in \mathcal{X}_i, u_i[k] \in \mathcal{G}_i, \forall i, k \Big\},$$
$$J(\mathbf{x}[0], u_1, ..., u_n) \doteq \sum_{i=1}^{n} \sum_{k=0}^{N-1} ||x_i[k] - x_i^d||_Q^2 + ||u_i[k]||_R^2, \tag{3}$$

The authors are with the Department of Electrical and Electronic Engineering, University of Melbourne, Melbourne, Victoria, Australia. E-mail: {alireza.farhadi,cantoni,pdower}@unimelb.edu.au

where $\mathbf{x}[\mathbf{0}] \doteq (\, x_1'[0] \;\; . \;\; . \;\; . \;\; x_n'[0] \,)'$ is the vector of initial states, x_i^d are the desired values for the state variables (desired set points) and $Q \geq 0$, $R > 0$ are weighting matrices.

Problem (1)-(3) involves $(\sum_{i=1}^{n} m_i) \times N$ decision variables. Techniques involving one decision maker responsible for computing all u_i (referred to henceforth as centralized optimization techniques) have been developed in the literature to solve such above optimization problems [2]. However, in large scale systems, $\sum_{i=1}^{n} m_i$ is very large. In general, as the number of decision variables n increases, the complexity of the centralized optimization techniques increases significantly in terms of the computational complexity (e.g., computational complexity $\sim\mathcal{O}(n^6)$). Consequently, centralized optimization techniques may not be able to provide a feasible solution for large scale systems within a desired time period.

A simple way to deal with the computational scalability problem just described is to use one decision maker for each subsystem decision variable. Each decision maker uses fixed values for the decision variables of all other subsystems, thereby yielding a collection of smaller optimization problems that can be solved in parallel. But this means decision makers will have to exchange information in order to approximate the optimal solution. This can happen in an iterative fashion, leading to the question of convergence. Improvements in computational scalability can be achieved in this way when the complexity associated with information exchange scales more slowly than that of the optimization problems with problem dimension. To manage the overheads associated with exchange of information between decision makers, one approach is to group decision makers into neighborhoods (see Fig. 1). Exchange of information between decision makers within a neighborhood can occur after each time their decision variables are updated, whereas exchange of information between neighborhoods is limited to be less frequent. This gives rise to two specific questions: (i) How should decision makers be grouped into neighborhoods? (ii) What are the trade-offs between size of neighborhoods, convergence rate, frequency of exchange of information between neighborhoods, and approximation error (i.e., performance loss relative to the optimal solution)?

The work in this paper is based on the consensus based optimization technique of [3], which approximates the solution of a centralized finite horizon optimal control problem by distributed decision makers. In particular, the consensus based optimization technique works as follows: Within a neighborhood, each decision maker

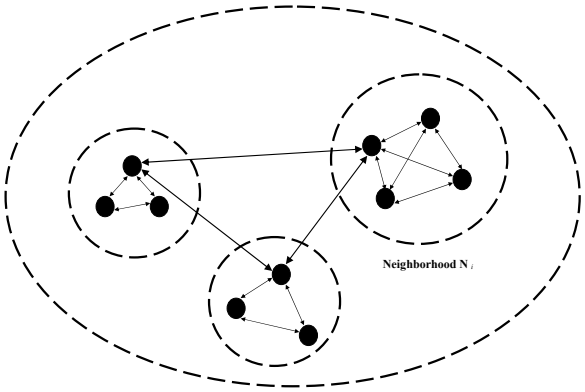

Fig. 1. Two-level architecture for exchanging information between distributed decision makers.

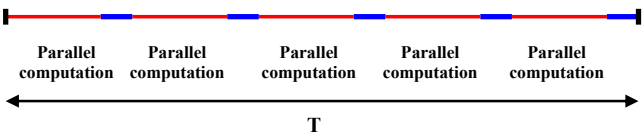

Fig. 2. The time-frame for inner and outer iterates updates and communication. Red lines represent inner iterate communication and updates and blue lines represent outer iterate updates and communication.

frequently updates its decision variable and shares the updated variable with all other neighboring decision makers. This intra-neighborhood update and communication is referred as an inner iterate. In addition to inner iterates, updates of decision variables from other neighborhoods are received periodically. These are referred to as outer iterates. Between outer iterates, distributed decision makers continue to compute and refine the local approximation of the optimal solution, with fixed values for decision variables from outside the neighborhood. These inner-outer iterates continue until consensus is reached between distributed decision makers on the value of the optimal cost. As each inner iterate involves parallel computation and a smaller optimization problem at each decision maker compared to that of centralized single decision maker optimization techniques, the above approach may yield a computationally tractable solution within a desired time period T (see Fig. 2).

In addition to [3], other examples of work that use consensus include [4] - [8]. Average consensus problems are considered in [4],[5] primarily from the perspective of convergence to the arithmetic mean of a quantity. On the other hand, similar to [3] the references [6]-[8] present distributed optimization methods, which approximate the solution of an optimal control problem; and focus on optimal performance. However, [3] suffers from the following drawbacks: There are no proofs for feasibility, convergence, and optimality of the technique presented in [3]; the example provided in [3] does not illustrate convergence to the optimal solution; and [3] does not present a method for decomposing the system into neighborhoods (a decomposition is assumed). Also, it does not study trade-offs between convergence rate, frequency of exchange of information between neighborhoods, and approximation error. Meanwhile, the techniques presented in [6], [7] may not be suitable for large scale systems, as these involve excessive communication overheads associated with exchange of information between decision makers. Moreover, the technique presented in [8] may not be suitable for some applications because (in general) it involves non-convex optimization problems at distributed decision makers (even when the original problem is convex), resulting in high computation overheads for the decision makers.

B. Paper Contributions

This paper aims to further develop the results of [3] by presenting a consensus based distributed optimization method suitable for large scale systems, which approximates the solution of a finite horizon optimal control problem via distributed decision makers. To achieve this goal, a simple method for decomposing the system into interacting neighborhoods with specified size is proposed, and mathematical proofs for feasibility, convergence, and optimality of the method are provided. Suitable measures for analyzing the performance of the consensus based optimization method are presented. As an example the convergence of the method to the solution of optimal control problem for the system (1) with six interacting subsystems subject to the LQ finite horizon cost functional (3) is illustrated. Trade-offs between approximation error (i.e. performance loss) and the aforementioned information pattern providing a handle for the management of communication and computation overheads are also studied.

C. Paper Organization

The paper is organized as follows: Section II presents the problem formulation. A consensus based distributed optimization method is presented in Section III. In Section IV, feasibility, convergence, and optimality of the method are demonstrated. In Section V, an application of the method to the LQ optimal control problem is presented, followed by a summary of the main contributions of the paper and directions for future research in Section VI.

II. Problem Formulation

Throughout the paper the following conventions are used: $\sigma_{max}(\cdot)$ denotes the largest singular value, $\sigma_{min}(\cdot)$ the smallest singular value, while $'$ denotes the transpose of a vector/matrix. I_n denotes the identity matrix with dimension $n \times n$ and "$\doteq$" means "by definition equals". $\mathbf{R}^n$ is the n dimensional real Euclidean space and "$\equiv$" means "equivalent to".

It is assumed throughout that the decision maker associated with subsystem S_i, as described by (1), has knowledge of all initial states, $\{x_i[0]\}_{i=1}^n$, and the cost functional (3) rewritten in terms of the initial states and decision variables.

Remark 2.1: i) From conditions $Q \geq 0$, $R > 0$, it easily follows that the quadratic cost functional (3) is a strictly convex function of the decision variables over the finite horizon N.

ii) As $\mathcal{X}_i \subset \mathbf{R}^{n_i}$, $i \in \{1,2,...,n\}$, are closed convex sets, their Cartesian product $\mathcal{X}_1 \times \mathcal{X}_2 \times ... \times \mathcal{X}_n \subset \prod_{i=1}^{n} \mathbf{R}^{n_i}$ is a closed convex set [9]. Hence, as affine functions preserve closedness and convexity of sets [9], the closed convex state constraints set $\mathcal{X}_i$ on the linear time invariant dynamics (1) impose additional constraints on the decision variables. In particular, for each $i \in \{1,...,n\}$, $u_i \in \mathcal{H}_i(\mathbf{x}(\mathbf{0}))$, where $\mathcal{H}_i(\mathbf{x}(\mathbf{0}))$ is a closed convex subset of $\mathbf{R}^{m_i}$. Therefore, for each i the set of control constraints for the LQ optimization problem (3) is $(u_i \in)\mathcal{U}_i$, where $\mathcal{U}_i \doteq \mathcal{G}_i \cap \mathcal{H}_i(\mathbf{x}(\mathbf{0}))$, which is a closed convex subset of $\mathbf{R}^{m_i}$.

III. Design Methodology of the Consensus Based Distributed Optimization Method

Consider the following constrained optimization problem:

$$\min_{u} J(u_1,...,u_n),\ u_i \in \mathcal{U}_i,$$

where the cost functional $J > 0$ is strictly convex over the finite horizon N, $\mathcal{U}_i$ is a closed convex subset of $\mathbf{R}^{m_i}$, and $\text{argmin}_{u_i} J(u_1,...,u_n) \in \mathbf{R}^{m_i N}$.

In order to solve the above optimization problem, a consensus based distributed optimization method is presented which uses the two-level architecture of Fig. 1 for exchanging information. The design methodology for this method involves two steps, namely, 1) decomposition, and 2) optimization, as will be described in this section.

A. Decomposition

In control and/or optimization of distributed systems with a large number of interacting subsystems, it has long been recognized [10] that a significant conceptual insight and numerical simplification can be gained by grouping strongly coupled subsystems into neighborhoods, solving the problem for neighborhoods independently, and resolving subsequently the effects of interaction between neighborhoods to get the overall solution. There are different ways for constructing these neighborhoods, such as the nested epsilon decomposition method ([10], Chapter 7). The idea behind this decomposition is to associate a graph with a given system, disconnect the edges corresponding to interconnections with strength smaller than a predefined threshold ϵ, and identify the disconnected subgraphs, as neighborhoods. Therefore, such decomposition methods do not provide or allow the specification of bounds on the size of neighborhoods. However, in the optimization method considered in this paper it is important to be able to prescribe the size of neighborhoods during the decomposition process, as the size of neighborhoods is limited by the available communication bandwidth and induced transmission delay. Such a freedom can not be provided by the available decomposition methods. Therefore, in this section a simple decomposition method is presented for the system (1), which clusters subsystems into disjoint interacting neighborhoods with specified size.

Interaction Strength Decomposition Method: Consider the system (1), where the ith subsystem is affected by other subsystems via the interacting variable of (2). The z-transform of the state of subsystem S_i is given by

$$\begin{aligned} X_i(z) &= (zI_{n_i} - A_i)^{-1}B_iU_i(z) \\ &\quad + \sum_{j\neq i}^{n}(zI_{n_i} - A_i)^{-1}M_{ij}X_j(z) \\ &\quad + \sum_{j\neq i}^{n}(zI_{n_i} - A_i)^{-1}N_{ij}U_j(z), \end{aligned}$$

so that the transfer function from input $U(z) = (\,U_1'(z)\ \ .\ \ .\ \ .\ \ U_n'(z)\,)'$ to state $X(z) = (\,X_1'(z)\ \ .\ \ .\ \ .\ \ X_n'(z)\,)'$ for the large scale system is given by

$$G(z) = V^{-1}(z)W(z),$$

where $V(z)\doteq[V_{ij}(z)]$ with

$$V_{ij}(z)\doteq \begin{cases} I_{n_i}, & \text{when } i = j \\ -(zI_{n_i} - A_i)^{-1}M_{ij}, & \text{otherwise} \end{cases}$$

and $W(z)\doteq[W_{ij}(z)]$ with

$$W_{ij}(z)\doteq \begin{cases} (zI_{n_i} - A_i)^{-1}B_i, & \text{when } i = j \\ (zI_{n_i} - A_i)^{-1}N_{ij}, & \text{otherwise.} \end{cases}$$

Lots of control systems have low bandwidth. Therefore, the DC parts of decision signals of other subsystems have the most effect on a given subsystem. Therefore, in the decomposition method presented in this section, the DC gain of the transfer function from input to state is considered, which is given by

$$G(z)|_{z=1} = \begin{pmatrix} E_1 & E_{12} & . & . & . & E_{1n} \\ E_{21} & E_2 & . & . & . & E_{2n} \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ E_{n1} & E_{n2} & . & . & . & E_n \end{pmatrix}, E_{ij} \in \mathbf{R}^{n_i \times m_j}.$$

Then, the *Interaction Strength (IS)* is defined as follows:

$$IS_{ij} \doteq \begin{cases} 0, & \text{if } i = j \\ \frac{\sigma_{max}(E_{ij})}{\sigma_{min}(E_i)}, & \text{if } \sigma_{min}(E_i) \neq 0 \text{ and } i \neq j \\ \frac{\sigma_{max}(E_{ij})}{\gamma}, & \text{if } \sigma_{min}(E_i) = 0 \text{ and } i \neq j \end{cases}$$

where γ is a fixed very small positive scalar. Subsequently, the normalized interaction strength ISN_{ij} is defined as

$$ISN_{ij} \doteq \text{round}\Big(\frac{IS_{ij}}{IS_{min}}\Big),\ IS_{min} \doteq \min_{\{i,j;IS_{ij}>0\}} IS_{ij}.$$

The interaction between subsystems may be summarized by a matrix, called interaction strength matrix, with elements/entries ISN_{ij}. Using this matrix, subsystems are clustered into neighborhoods with specified size such that subsystems within a neighborhood have the largest total interaction strength. The steps involved in this clustering are illustrated in the following examples.

Example 3.1: Consider a system with six scalar subsystems of the form (1). The aggregated system is described as follows:

$$x[k+1] = Ax[k] + Bu[k], \tag{4}$$

Subsystems	S_1	S_2	S_3	S_4	S_5	S_6
S_1	0	36	226	3	245	82
S_2	37	0	21	29	49	27
S_3	20	12	0	22	182	70
S_4	93	55	63	0	148	39
S_5	53	31	151	13	0	67
S_6	106	62	73	1	185	0

TABLE I

THE INTERACTION MATRIX ASSOCIATED WITH THE SYSTEM (4).

$(1,2) = 73$	$(1,3) = 246$	$(1,4) = 96$	$(1,5) = 298$
$(1,6) = 188$	$(2,3) = 33$	$(2,4) = 84$	$(2,5) = 80$
$(2,6) = 89$	$(3,4) = 85$	$(3,5) = 333$	$(3,6) = 143$
$(4,5) = 161$	$(4,6) = 40$	$(5,6) = 252$	$(5,6) = 252$

TABLE II

THE SW PAIRS CORRESPONDING TO THE INTERACTION MATRIX OF TABLE I.

$$x[k] = (\, x_1[k] \quad x_2[k] \quad x_3[k] \quad x_4[k] \quad x_5[k] \quad x_6[k] \,)',$$
$$u[k] = (\, u_1[k] \quad u_2[k] \quad u_3[k] \quad u_4[k] \quad u_5[k] \quad u_6[k] \,)',$$
$$A = (a_1|a_2|a_3|a_4|a_5|a_6),$$
$$a_1 = \begin{pmatrix} 1.7049 \\ 0.2328 \\ 0.1213 \\ -0.3836 \\ -0.1148 \\ -0.5148 \end{pmatrix}, \; a_2 = \begin{pmatrix} -0.0049 \\ 1.4672 \\ -0.1213 \\ 0.3836 \\ 0.1148 \\ 0.5148 \end{pmatrix},$$
$$a_3 = \begin{pmatrix} -0.9082 \\ -0.0213 \\ 0.7311 \\ 0.1393 \\ -0.6754 \\ 0.0246 \end{pmatrix}, \quad a_4 = \begin{pmatrix} -0.2732 \\ -0.4127 \\ 0.0955 \\ 1.2061 \\ 0.007 \\ -0.143 \end{pmatrix},$$
$$a_5 = \begin{pmatrix} 0.5496 \\ -0.4861 \\ 0.5566 \\ 0.132 \\ 2.3762 \\ 0.4762 \end{pmatrix}, \quad a_6 = \begin{pmatrix} -0.2756 \\ 0.5709 \\ -0.4652 \\ 0.198 \\ -0.4357 \\ 1.5143 \end{pmatrix},$$
$$B = diag(1.7, -1, 1.5, -1.2, 1.9, 0.86).$$

The interaction matrix associated with this system is shown in Table I. Suppose that we are interested in decomposing the system (4) into three neighborhoods of size two. To do so, we compute the *Strength Weights* (SWs) for each pair of different subsystems, which are defined as follows:

$$SW(ij) \doteq ISN_{ij} + ISN_{ji}, \quad i \neq j.$$

These are computed in Table II. By inspection, it follows that the pair $(3,5)$ has the largest total interaction strength followed by the disjoint pair $(1,6)$. Therefore, we decompose the system (4), as follows: $\mathcal{N}_1 = \{S_3, S_5\}$, $\mathcal{N}_2 = \{S_1, S_6\}$, and $\mathcal{N}_3 = \{S_2, S_4\}$. Alternatively, suppose that we are interested in decomposing the system (4) into two neighborhoods of size three. To do so, we compute the SWs for each triple combination of different subsystems, defined as

$$SW(ijk) \doteq ISN_{ij} + ISN_{ik} + ISN_{ji} + ISN_{jk} + ISN_{ki} + ISN_{kj}, \; i \neq j \neq k.$$

$(1,2,3) = 352$	$(1,2,4) = 253$	$(1,2,5) = 451$
$(1,2,6) = 350$	$(1,3,4) = 427$	$(1,3,5) = 877$
$(1,3,6) = 577$	$(1,4,5) = 555$	$(1,4,6) = 324$
$(1,5,6) = 738$	$(2,3,4) = 202$	$(2,3,5) = 446$
$(2,3,6) = 265$	$(2,4,5) = 325$	$(2,4,6) = 213$
$(2,5,6) = 421$	$(3,4,5) = 579$	$(3,4,6) = 268$
$(3,5,6) = 728$	$(4,5,6) = 453$	$(4,5,6) = 453$

TABLE III

THE SWS FOR TRIPLE COMBINATIONS OF DIFFERENT SUBSYSTEMS OF THE SYSTEM (4).

These are computed In Table III. By inspection, it follows that the triple combination $(1,3,5)$ has the largest total interaction strength. Therefore, we decompose the system (4), as follows: $\mathcal{N}_1 = \{S_1, S_3, S_5\}$, and $\mathcal{N}_2 = \{S_2, S_4, S_6\}$.

Remark 3.2: The above decomposition clusters subsystems into neighborhoods on the basis of interaction strength. Consequently, subsystems in a particular neighborhood are less affected by decision variables of subsystems in other neighborhoods.

B. Optimization

Now, consider the two-level architecture of Fig. 1. Suppose that the interaction strength decomposition method decomposes the system (1) with n interacting subsystems S_1, S_2, ... , S_n, into q disjoint interacting neighborhoods $\mathcal{N}_1$,..., $\mathcal{N}_q$, with specified size. Without loss of generality, for notational simplicity, let the neighborhoods have the same size and suppose the indexing is such that $\mathcal{N}_1 = \{S_{1,1}, ..., S_{1,g}\} \equiv \{S_1, ..., S_g\}$, $\mathcal{N}_2 = \{S_{2,1}, ..., S_{2,g}\} \equiv \{S_{g+1}, ..., S_{2g}\}$, ... , $\mathcal{N}_m = \{S_{m,1}, ..., S_{m,g}\} \equiv \{S_{(m-1)g+1}, ..., S_{mg}\}$, ... , and $\mathcal{N}_q = \{S_{q,1}, ..., S_{q,g}\} \equiv \{S_{(q-1)g+1}, ..., S_n\}$, where $S_{m,l}$ is the lth subsystem in the mth neighborhood. Subsequently, the decision variable of the subsystem $S_{m,l}$ is denoted by $u_{m,l}$, its corresponding control constraint set by $\mathcal{U}_{m,l}$ and the cost functional by $J(u_{1,1}, ..., u_{m-1,g}, u_{m,1}, ..., u_{q,g})$.

After the above decomposition, decision variables $u_{m,l}$, $m \in \{1, ..., q\}$, $l \in \{1, ..., g\}$, are produced using the following consensus based optimization technique. This technique was first presented in [3] for the two-neighborhoods case each with two subsystems.

The consensus based optimization technique involves outer iterate updates and communication between neighborhoods; and between every two successive outer iterates, a sequence of inner iterate communication between subsystems within a neighborhood and inner iterate updates, according to the following steps.

- *Initialization:* The information exchange between neighborhoods at outer iterate t makes it possible for subsystem $\mathcal{S}_{m,l}$ to initialize its local decision variables as $h^0_{j,r} \doteq u^t_{j,r}$, $\forall j \in \{1, ..., q\}$, $\forall r \in \{1, ..., g\}$, where $u^0_{j,r} \in \mathcal{U}_{j,r}$ are chosen arbitrarily at $t = 0$.
- *Inner Iterate:* Then, subsystem $\mathcal{S}_{m,l}$ performs $\bar{p}$ inner iterates, as follows:
  For each inner iterate $p \in \{0, 1, ..., \bar{p}-1\}$, it first updates

its decision variable via

$$h_{m,l}^{p+1} = \pi_{m,l} h_{m,l}^{*} + (1 - \pi_{m,l}) h_{m,l}^{p}, \tag{5}$$

where $\pi_{m,l}$ are chosen subject to

$$\pi_{m,l} > 0, \ \sum_{l=1}^{g} \pi_{m,l} = 1,$$

and

$$\begin{aligned} h_{m,l}^{*} \ \doteq \ & \text{argmin}_{h_{m,l} \in \mathcal{U}_{m,l}} J(h_{1,1}^{0}, ..., h_{m-1,g}^{0}, h_{m,1}^{p}, \\ & ..., h_{m,l}, ..., h_{m,g}^{p}, h_{m+1,1}^{0}, ..., h_{q,g}^{0}). \end{aligned} \tag{6}$$

Then, it trades its updated decision variable, $h_{m,l}^{p+1}$, with all other subsystems in its neighborhood.

- *Outer Iterate:* After $\bar{p}$ inner iterates, there is an outer iterate update at subsystem $\mathcal{S}_{m,l}$, as follows:

$$u_{m,l}^{t+1} = \lambda_m h_{m,l}^{\bar{p}} + (1 - \lambda_m) u_{m,l}^{t}, \tag{7}$$

where λ_m are chosen subject to

$$\lambda_m > 0, \ \ \sum_{m=1}^{q} \lambda_m = 1.$$

Then, there is an outer iterate communication, in which the updated decision variables, $u_{j,r}^{t+1}$, $j \in \{1, ..., q\}$, $r \in \{1, ..., g\}$, are shared between all neighborhoods; and subsequently, between all subsystems.

IV. Feasibility, Convergence, and Optimality Results

In this section, it is shown that given a feasible initialization (i.e., $u_{m,l}^{0} \in \mathcal{U}_{m,l}$, $m \in \{1, ..., q\}$, $l \in \{1, ..., g\}$), the iterates (7) are feasible (i.e., $u_{m,l}^{t} \in \mathcal{U}_{m,l}$, $t \in \{0, 1, 2, 3, ...\}$), the cost functional is strictly non-increasing for each outer iterate (and so converges as $t \to \infty$), and the iterates $(u_{1,1}^{t}, ..., u_{q,g}^{t})$ converges to the optimal solution $(u_{1,1}^{*}, ..., u_{q,g}^{*})$. Note that as J and the control constraint sets are convex, there exists a unique optimal solution $(u_{1,1}^{*}, ..., u_{q,g}^{*})$. Feasibility and convergence properties are shown for the general strictly convex finite horizon cost functional $J(u_{1,1}, ..., u_{q,g})$; however, for the optimality it is also assumed that the cost functional is quadratic, i.e.,

$$J(u_{1,1}, ..., u_{q,g}) = \sum_{k=0}^{N-1} \begin{pmatrix} u_{1,1}[k] \\ . \\ . \\ . \\ u_{q,g}[k] \end{pmatrix}' H \begin{pmatrix} u_{1,1}[k] \\ . \\ . \\ . \\ u_{q,g}[k] \end{pmatrix}, H > 0.$$

We start this section with a feasibility result.

Lemma 4.1: (Feasibility) Given above strictly convex finite horizon cost functional, J, convex control constraint sets, $\mathcal{U}_{m,l}$ ($m \in \{1, ..., q\}$, $l \in \{1, ..., g\}$), and a feasible initialization, the inner and outer iterates (5) and (7) are feasible.

Proof: By assumption, the initialization, $h_{m,l}^{0} = u_{m,l}^{0}$ ($m \in \{1, ..., q\}, l \in \{1, ..., g\}$) is feasible. Since $\mathcal{U}_{1,1}, ..., \mathcal{U}_{q,g}$ are convex, the convex combination (5) with $p = 0$ implies that $(h_{1,1}^{1}, ..., h_{q,g}^{1})$ is feasible. Feasibility for $p \in \{1, ..., \bar{p}\}$ follows similarly by induction. Now as $u_{m,l}^{0}$ and $h_{m,l}^{\bar{p}}$ are feasible, the convex combination (7) with $t = 0$ implies that $(u_{m,l}^{1}, ..., u_{m,l}^{1})$ is feasible. Subsequently, the feasibility for $t > 1$ and each $p \in \{1, ..., \bar{p}\}$ between every two successive outer iterates follows similarly.

Next we show the convergence of the cost functional, J, under the solution (7).

Lemma 4.2: (Convergence) Given a feasible initialization, the strictly convex finite horizon cost functional $J(u_{1,1}^{t}, ..., u_{q,g}^{t})$ is non-increasing at each outer iterate $t \in \{0, 1, 2, 3, ...\}$ and converges as $t \to \infty$.

Proof: For each $t \in \{0, 1, 2, 3, ...\}$, the cost functional satisfies the following:

$$\begin{aligned} & J(u_{1,1}^{t+1}, ..., u_{q,g}^{t+1}) \\ & = J\Big(\lambda_1 (h_{1,1}^{\bar{p}}, ..., h_{1,g}^{\bar{p}}, u_{2,1}^{t}, ..., u_{q,g}^{t}) + ... \\ & + \lambda_m (u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}}, ..., h_{m,g}^{\bar{p}}, u_{m+1,1}^{t}, ..., u_{q,g}^{t}) \\ & + ... + \lambda_q (u_{1}^{t}, ..., u_{q-1,g}^{t}, h_{q,1}^{\bar{p}}, ..., h_{q,g}^{\bar{p}})\Big) \\ & < \lambda_1 J(h_{1,1}^{\bar{p}}, ..., h_{1,g}^{\bar{p}}, u_{2,1}^{t}, ..., u_{q,g}^{t}) + ... \\ & + \lambda_m J(u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}}, ..., h_{m,g}^{\bar{p}}, u_{m+1,1}^{t}, ..., u_{q,g}^{t}) \\ & + ... + \lambda_q J(u_{1}^{t}, ..., u_{q-1,g}^{t}, h_{q,1}^{\bar{p}}, ..., h_{q,g}^{\bar{p}}), \end{aligned} \tag{8}$$

where the equality follows from (7), and the inequality follows from the strict convexity of the cost functional. Now, define

$$\begin{aligned} & J_m \doteq J(u_{1,1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}}, ..., h_{m,g}^{\bar{p}}, u_{m+1,1}^{t}, ..., \\ & u_{q,g}^{t}), \ \ m \in \{1, 2, ..., q\}. \end{aligned}$$

Then, J_m satisfies the following bound

$$\begin{aligned} & J_m \\ & = J\Big(u_{1}^{t}, ..., u_{m-1,g}^{t}, \pi_{m,1}(h_{m,1}^{*}, h_{m,2}^{\bar{p}-1}, ..., h_{m,g}^{\bar{p}-1}) + ... \\ & + \pi_{m,g}(h_{m,1}^{\bar{p}-1}, ..., h_{m,g-1}^{\bar{p}-1}, h_{m,g}^{*}), u_{m+1,1}^{t}, ..., u_{q,g}^{t}\Big) \\ & < \pi_{m,1} J(u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{*}, h_{m,2}^{\bar{p}-1}, ..., h_{m,g}^{\bar{p}-1}, \\ & u_{m+1,1}^{t}, ..., u_{q,g}^{t}) + ... \\ & + \pi_{m,g} J(u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}-1}, ..., h_{m,g-1}^{\bar{p}-1}, h_{m,g}^{*}, \\ & u_{m+1,1}^{t}, ..., u_{q,g}^{t}) \\ & \leq \Big(\sum_{l=1}^{g} \pi_{m,l}\Big) J(u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}-1}, ..., h_{m,g}^{\bar{p}-1}, \\ & u_{m+1,1}^{t}, ..., u_{q,g}^{t}) \\ & = J(u_{1}^{t}, ..., u_{m-1,g}^{t}, h_{m,1}^{\bar{p}-1}, ..., h_{m,g}^{\bar{p}-1}, u_{m+1,1}^{t}, ..., u_{q,g}^{t}), \end{aligned}$$

where $h_{m,1}^{*}$, $h_{m,2}^{*}$, $...h_{m,g}^{*}$ have been generated at inner iterate $\bar{p}$, the first equality follows from (5) for $p = \bar{p}$, the first inequality follows from the strict convexity of the cost functional, the second inequality follows from the fact that the cost functional J for $h_{m,l}^{*}$ is not greater than J for $h_{m,l}^{\bar{p}-1}$, and the second equality follows from the fact that

$\sum_{l=1}^{g} \pi_{m,l} = 1$. By following a similar argument, it can be shown for $m \in \{1, 2, ..., q\}$ that

$$\begin{aligned} J_m &< J(u_1^t, ..., u_{m-1,g}^t, h_{m,1}^{\bar{p}-1}, ..., h_{m,g}^{\bar{p}-1}, u_{m+1,1}^t, ..., u_{q,g}^t) \\ &< J(u_1^t, ..., u_{m-1,g}^t, h_{m,1}^{\bar{p}-2}, ..., h_{m,g}^{\bar{p}-2}, u_{m+1,1}^t, ..., u_{q,g}^t) \\ &< ... < J(u_{1,1}^t, ..., u_{q,g}^t). \end{aligned} \quad (9)$$

Consequently, from (8), (9) it follows that

$$\begin{aligned} J(u_{1,1}^{t+1}, ..., u_{q,g}^{t+1}) &< \Big(\sum_{m=1}^{q} \lambda_m\Big) J(u_{1,1}^t, ..., u_{q,g}^t) \\ &= J(u_{1,1}^t, ..., u_{q,g}^t). \end{aligned}$$

That is, the cost $J(u_{1,1}^t, ..., u_{q,g}^t)$ is non-increasing at each outer iterate t. Hence, the non-negative cost functional J converges as $t \to \infty$ by the monotone convergence theorem.

The optimality of the consensus based distributed optimization method is now demonstrated.

Lemma 4.3: (Optimality) Given a feasible initialization, strictly convex and quadratic cost J, and closed convex control constraint sets $\mathcal{U}_{m,l}$ ($m \in \{1, ..., q\}$, $l \in \{1, ..., g\}$), the cost $J(u_{1,1}^t, ..., u_{q,g}^t)$ converges to the optimal cost $J(u_{1,1}^*, ..., u_{q,g}^*)$, and the iterates $(u_{1,1}^t, ..., u_{q,g}^t)$ converge to the unique optimal solution $(u_{1,1}^*, ..., u_{q,g}^*)$, as $t \to \infty$.

Proof: From Lemma 4.2, it follows that the cost converges to some $\bar{J} \geq 0$. Because J is quadratic and strictly convex, its sub level sets $Lev_{\leq b}(J)$ are compact and bounded for all $b > 0$. Therefore, all iterates belong to the compact and bounded set $Lev_{\leq J(u_{1,1}^0, ..., u_{q,g}^0)}(J) \cap \mathcal{U}_{1,1} \times ... \times \mathcal{U}_{q,g}$. Hence, there is at least one accumulation point $(\bar{u}_{1,1}, ..., \bar{u}_{q,g})$ and a subsequence $\mathcal{T} \subset \{1, 2, 3, ...\}$ such that $(u_{1,1}^t, ..., u_{q,g}^t)_{t \in \mathcal{T}}$ converges to $(\bar{u}_{1,1}, ..., \bar{u}_{q,g})$ and $J(\bar{u}_{1,1}, ..., \bar{u}_{q,g}) = \bar{J}$.

Suppose for the purpose of contradiction that $\bar{J} \neq J(u_{1,1}^*, ..., u_{q,g}^*)$, and therefore $(\bar{u}_{1,1}, ..., \bar{u}_{q,g}) \neq (u_{1,1}^*, ..., u_{q,g}^*)$. Then, following a similar argument as the proof of ([7], Lemma 7), it follows from the assumption $\bar{J} \neq J(u_{1,1}^*, ..., u_{q,g}^*)$ that $\bar{J} < \bar{J}$ giving a contradiction. Consequently, it is concluded that $\bar{J} = J(u_{1,1}^*, ..., u_{q,g}^*)$ and therefore $(\bar{u}_{1,1}, ..., \bar{u}_{q,g}) = (u_{1,1}^*, ..., u_{q,g}^*)$. Moreover, as J is continuous and it converges to $\bar{J}$, i.e.,

$$\lim_{t \to \infty} J(u_{1,1}^t, ..., u_{q,g}^t) = \bar{J} = J(u_{1,1}^*, ..., u_{q,g}^*),$$

it is concluded that the entire sequence $(u_{1,1}^t, ..., u_{q,g}^t)$ confined in a compact set converges to the unique optimal solution $(u_{1,1}^*, ..., u_{q,g}^*)$.

V. Example

In this section, the proposed consensus based distributed optimization method is applied on the LQ optimal control problem and its satisfactory performance on this problem is illustrated by computer simulations. To analyze the performance of the consensus based distributed optimization method, in this section the following measures are defined:

Definition 5.1: (Performance Loss) For a given number of outer iterate updates t and $\bar{p}$, the Performance Loss $PL_t(\bar{p})$ (measured in percent) is defined as

$$PL_t(\bar{p}) \doteq 100\Big(\frac{J(u_1^t, ..., u_n^t) - \bar{J}}{\bar{J}}\Big),$$

$\bar{p}$	T_{PL}	$PL_t(\bar{p})$ at $t = T_{PL}/\bar{p}$	Computation time (sec.)
1	453	0.99	77.63
10	820	0.95	142.34
20	1400	0.93	244.93
50	3250	0.98	564.91

TABLE IV

Trade-offs between $\bar{p}$, T_{PL}, and computation time for $PL = 1$ percent and the two-neighborhoods case.

$\bar{p}$	T_{PL}	$PL_t(\bar{p})$ at $t = T_{PL}/\bar{p}$	Computation time (sec.)
1	424	0.99	74.23
10	2200	0.99	390.14
20	4320	0.98	755.36
50	10750	0.99	1885.2

TABLE V

Trade-offs between $\bar{p}$, T_{PL}, and computation time for $PL = 1$ percent and the three-neighborhoods case.

where $\bar{J}$ is the optimal cost.

Definition 5.2: (Total Number of Iterations) For a given $\bar{p}$, $T_t \doteq \bar{p} \times t$ is referred as the total number of iterations up to outer iterate t.

Definition 5.3: (Total Number of Iterations for Convergence) For a given performance loss PL, let $\bar{t}_{PL}$ be the smallest integer such that $PL_t(\bar{p}) \leq PL$ for all $t \geq \bar{t}_{PL}$. Then, $T_{PL} \doteq \bar{p} \times \bar{t}_{PL}$ is referred as the total number of iterations for convergence.

Remark 5.4: A smaller total number of iterations T_{PL} means that a smaller computation time is required for convergence. Therefore, it is desired to keep T_{PL} as small as possible for a given performance loss, PL.

Now, consider the LQ optimal control problem described earlier. From Remark 2.1 it follows that the proposed consensus based distributed optimization method is directly applicable to the LQ problem.

For the purpose of illustration, we apply the consensus based distributed optimization method to the system (4) subject to the finite horizon quadratic cost functional (3) with the following specifications: $N = 5$, $Q = 100.I_6$, $R = I_6$, $(x_1^d = 1, x_2^d = 2, x_3^d = 3, x_4^d = 4, x_5^d = 5, x_6^d = 6)$, $x_i[0] = 0$, $i \in \{1, 2, 3, 4, 5, 6\}$, $\mathcal{X}_i = [-12, 12]$, $\mathcal{G}_i = [-6, 6]$, and $u_{m,l}^0 = 0$. The optimal control cost for this constraint optimization problem is $\bar{J} = 9370.89$. The simulation results for the above optimization problem are compared for different decompositions of the system (4), as follows: Two-neighborhoods decomposition: $\mathcal{N}_1 = \{S_{1,1}, S_{1,2}, S_{1,3}\} \equiv \{S_1, S_3, S_5\}$, $\mathcal{N}_2 = \{S_{2,1}, S_{2,2}, S_{2,3}\} \equiv \{S_2, S_4, S_6\}$, Three-neighborhoods decomposition: $\mathcal{N}_1 = \{S_{1,1}, S_{1,2}\} \equiv \{S_3, S_5\}$, $\mathcal{N}_2 = \{S_{2,1}, S_{2,2}\} \equiv \{S_1, S_6\}$, $\mathcal{N}_3 = \{S_{3,1}, S_{3,2}\} \equiv \{S_2, S_4\}$, and Six-neighborhoods decomposition: $\mathcal{N}_1 = \{S_{1,1}\} \equiv \{S_1\}$, $\mathcal{N}_2 = \{S_{2,1}\} \equiv \{S_2\}$, $\mathcal{N}_3 = \{S_{3,1}\} \equiv \{S_3\}$, $\mathcal{N}_4 = \{S_{4,1}\} \equiv \{S_4\}$, $\mathcal{N}_5 = \{S_{5,1}\} \equiv \{S_5\}$, $\mathcal{N}_6 = \{S_{6,1}\} \equiv \{S_6\}$. For the two-neighborhoods case, we choose in (7), (5), $\lambda_1 = \lambda_2 = \frac{1}{2}$, $\pi_{1,1} = \pi_{1,2} = \pi_{1,3} = \frac{1}{3}$, $\pi_{2,1} = \pi_{2,2} = \pi_{2,3} = \frac{1}{3}$, for the three-neighborhoods

$\bar{p}$	T_{PL}	$PL_t(\bar{p})$ at $t = T_{PL}/\bar{p}$	Computation time (sec.)
1	1020	0.99	179.21
10	10200	0.99	1834.3
20	20400	0.99	3569.9
50	51000	0.99	9027.9

TABLE VI

TRADE-OFFS BETWEEN $\bar{p}$, T_{PL}, AND COMPUTATION TIME FOR $PL = 1$ PERCENT AND THE SIX-NEIGHBORHOODS CASE.

case, we choose $\lambda_1 = \lambda_2 = \lambda_3 = \frac{1}{3}$, $\pi_{1,1} = \pi_{1,2} = \frac{1}{2}$, $\pi_{2,1} = \pi_{2,2} = \frac{1}{2}$, $\pi_{3,1} = \pi_{3,2} = \frac{1}{2}$, and for the six-neighborhoods case, we choose $\lambda_1 = \lambda_2 = ... = \lambda_6 = \frac{1}{6}$, $\pi_{1,1} = \pi_{2,1} = ... = \pi_{6,1} = 1$.

For simulation purposes, MATLAB *quadprog.m* solver is used, which is interfaced via YALMIP [11] to compute the optimal controls numerically.

It is observed that for different decompositions and different numbers of inner iterates before each outer iterate $\bar{p}$, the convergence to the optimal solution is achieved. A number of trade-offs are observed. In particular, Table IV summarizes trade-offs between $\bar{p}$, T_{PL} and computation time (associated with the distributed decision makers processors) for convergence of the two-neighborhoods case. Similarly, Table V summarizes trade-offs between $\bar{p}$, T_{PL} and computation time for convergence of the three-neighborhoods case. Finally, Table VI summarizes trade-offs between $\bar{p}$, T_{PL} and computation time for convergence of the six-neighborhoods case. As clear from these tables for different $\bar{p}$ and different decompositions, convergence to the optimal solution is achieved. Also, as illustrated in Fig. 3, the required computation time to have 1 percent performance loss ($PL = 1$ percent) is proportional to the total number of iterations for convergence. Specifically, for all three cases the required computation time equals γT_{PL}, where $\gamma = 0.175$ is the computation time of the decision maker with the most computational complexity. That is, T_{PL} is a measure of required computation time.

Fig. 4 illustrates trade-offs between T_{PL}, $\bar{p}$ and different decompositions for $PL = 1$ percent and $PL = 10$ percent. It is clear from this figure that (for a given decomposition) the total number of iterations for convergence T_{PL} increases linearly with $\bar{p}$, and for a given $\bar{p}$ the corresponding total number of iterations for convergence increases by decomposing the system into more neighborhoods with smaller size. This result is expected because decomposing the system into more neighborhoods with smaller size means less communication for consensus between local decision makers. Fig. 4 indicates that there is no point in decomposing the system into neighborhoods and doing inner iterates if there are no constraints in communication between distributed decision makers. However, in many systems; particularly, in large scale systems, communication is subject to limited bandwidth and long delay due to multi-hopping. These constraints induce excessive communication overheads and result in very long overall computation time for producing the optimal inputs. Therefore, to manage the overheads associated with exchange of information between decision makers, the system must be decomposed into neighborhoods and inner iterates must be used, in which, as shown in Fig. 4, this decomposition results in more iterations for convergence. For illustration consider the following example.

Example 5.5: Suppose the inner iterate communication overhead for each neighborhood is 1 second, the outer iterate communication overhead is 10 seconds, and subsystems in different neighborhoods can broadcast simultaneously without collision (e.g., using Orthogonal Frequency Division Multiple Access - OFDMA). Then, if the system is decomposed into three neighborhoods and $\bar{p} = 10$, from Table V it follows that the total communication overhead equals $(220 \times 10 + 2200 \times 1 =)$ 4400 seconds and therefore the total computation time for producing the optimal inputs equals $(390.14 + 4400 =)$ 4790.14 seconds. On the other hand, if the system is not decomposed and the method of [7] is used, it takes 950 (outer) iterates to have 1 percent performance loss and the associated computation time is 174.126 seconds. Therefore, for this case the total communication overhead equals 9500 seconds and hence the total computation time for producing the optimal inputs equals 9674.126 seconds.

As clear from this example there is a gain in terms of the total computation time in decomposing the system into neighborhoods and using inner iterates when there is an excessive communication overhead in exchange of information between all decision makers.

Fig. 5 illustrates trade-offs between the performance loss, $PL_t(\bar{p})$, the total number of iterations T_t, and different decompositions for $\bar{p} = 10$ and $\bar{p} = 20$. As clear from this figure for a given decomposition, the performance loss decreases as the number of iterations increases. Also, for a given performance loss, the number of iterations increases by decomposing the system into more neighborhoods with smaller size.

As shown above, there are trade-offs between the performance loss, number of neighborhoods, number of inner and outer iterates, and the total number of iterations for the system (4) subject to the LQ cost (3). These trade-offs present the accuracy of approximation of the LQ optimal control solution by distributed decision makers for the system (4) subject to state and control constraints as well as information pattern.

VI. CONCLUSION

In this paper, aspects of a consensus based distributed optimization method were presented and studied. The method was applied to approximate the solution of a multi-variate LQ optimal control problem via distributed decision makers. The decision makers are constrained in terms of the pattern of local computation and information exchange, as a mechanism for managing the corresponding overheads. Feasibility, convergence, and optimality of the method were proved. Convergence to the solution of finite horizon LQ optimal control problem was illustrated for a system with six interacting linear time invariant subsystems. For this system, trade-offs between approximation error (i.e. performance loss

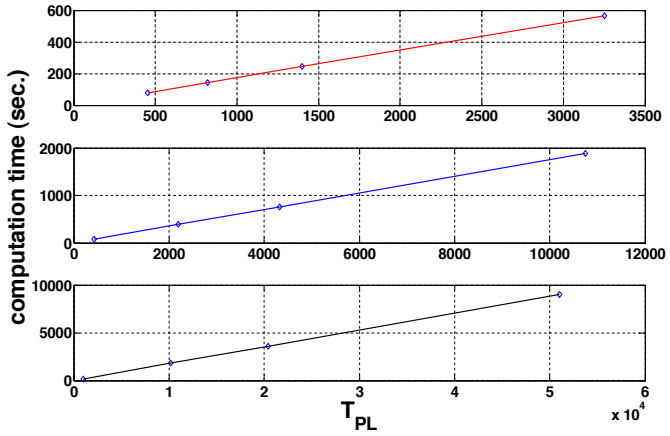

Fig. 3. Computation time versus the total number of iterations for convergence T_{PL} for different decompositions and $PL = 1$ percent. Red: The two-neighborhoods case. Blue: The three-neighborhoods case. **Black**: The six-neighborhoods case.

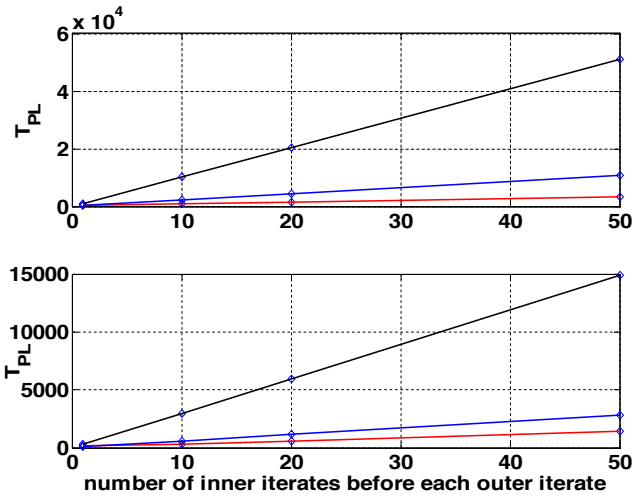

Fig. 4. Trade-offs between the total number of iterations for convergence T_{PL} and $\bar{p}$ for different decompositions and $PL = 1$ percent (top figure) and $PL = 10$ percent (bottom figure).Red: The two-neighborhoods case. Blue: The three-neighborhoods case. **Black**: The six-neighborhoods case.

relative to the optimal solution) and the patterns used to constrain information exchange and computation were illustrated. Future work will include computational complexity analysis of the proposed method and its comparison with the centralized optimization techniques.

ACKNOWLEDGMENT: This work is supported by an Australian Research Council (ARC) Linkage Project LP0989497.

REFERENCES

[1] M. Cantoni, E. Weyer, Y. Li, S. K. Ooi, I. Mareels, and M. Ryan, Control of large-scale irrigation networks, *Proceedings of the IEEE*, vol. 95, No. 1, pp. 75-91, January 2007.

[2] K. G. Murty, *Linear Complementarity, Linear and Nonlinear Programming*, Sigma Series in Applied Mathematics, 1988.

[3] B. T. Stewart, J. B. Rawlings, and S. J. Wright, Hierarchical cooperative distributed model predictive control, *2010 American Control Conference*, Baltimore, MD, USA, June 30-July 02, 2010, pp. 3963-3968.

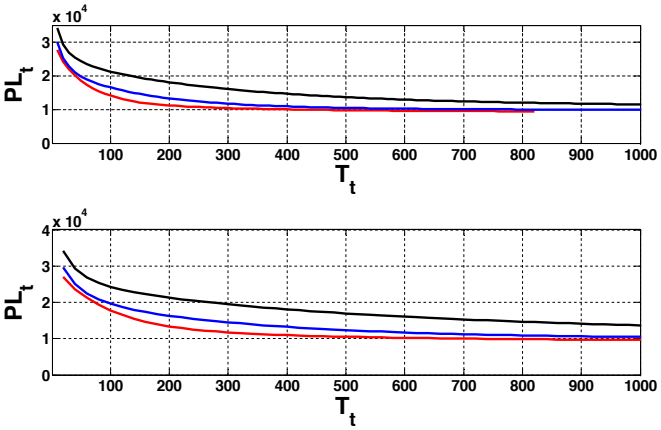

Fig. 5. Trade-offs between $PL_t(\bar{p})$ and T_t for different decompositions and $\bar{p} = 10$ (top figure) and $\bar{p} = 20$ (bottom figure). Red: The two-neighborhoods case. Blue: The three-neighborhoods case. **Black**: The six-neighborhoods case.

[4] R. Olfati-Saber and R. M. Murray, Consensus problems in networks of agents with switching topology and time delays, *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520-1533, Sep. 2004.

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

[6] J. M. Maestre, D. Munoz de la Pena, E. F. Camacho, and T. Alamo, Distributed model predictive control based on agent negotiation, *Journal of Process Control*, vol. 21, pp. 685-697, 2011.

[7] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, Cooperative distributed model predictive control, *Systems and Control Letters*, vol. 59, pp. 460-469, 2010.

[8] R. R. Negenborn, Multi-agent model predictive control with applications to power networks, *Ph.D. Thesis*, Delft University of Technology.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[10] D. D. Silijak, *Decentralized Control of Complex Systems*, Mathematics in Science and Engineering, vol. 184, 1991.

[11] J. Lofberg, Yalmip: a toolbox for modeling and optimization in MATLAB, in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: http://users.isy.liu.se/johanl/yalmip