Distributed Optimal Control Using an Efficient Method for Decomposition

Kiarash Aryankia[1], Alireza Farhadi[2]
[1]M.Sc. graduated from the Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran
Email: aryankia@alum.sharif.edu
[2]Assistant Professor in the Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran
Email: afarhadi@sharif.edu

***Abstract*—This paper is concerned with a new method for decomposing large scale systems into disjoint neighborhoods of subsystems with specific size determined a priori for managing communication load of a Jacobi based distributed optimal control method. The proposed decomposition method clusters subsystems into disjoint neighborhoods based on the strength of control inputs interaction between subsystems. The proposed method is a new method and uses the time response of system for decomposition. Using extensive computer simulation it is illustrated that this method finds the best decomposition.**

***Keywords*—Decomposition, Distributed Optimal Control , Interaction of Subsystems, Large Scale Systems**

I. Introduction

A. Motivation and Background

Energy efficiency and quality considerations restrict the control of large scale systems. For controlling large scale systems, in general, there are three approaches: centralized, decentralized and distributed methods. In centralized method [1] there is one computing unit that produces control commands for the whole system using the whole knowledge from the system. In decentralized control [2], [3], the interaction between subsystems is due to mutual states and controls; and regulators do not exchange information and the interaction between subsystems is represented as disturbance. However, in distributed method, information is also transmitted from any local regulator to all or a given subset of the others [4]. One such a method is the distributed control method of [5], [6] that is based on the Jacobi iterative optimization technique with two-level architecture for communication between subsystems. This method has superiority over the centralized methods in terms of computational complexity and reliability [7], [8]. This iterative optimization method has the following features: it handles hard control input and state constraints, it is a convex optimization problem and in the limit, its solution converges to the optimal solution. Hence, it is a suitable method for the optimal control of large scale systems, such as automated irrigation networks and smart building, in which their control problem can be written in the form of a convex optimization problem [7], [9].

In [10] it is shown that the decomposition of a large scale system into disjoint neighborhoods of subsystems in which subsystems with strong interaction are clustered into a neighborhood; and subsequently, solving an optimal control problem for each neighborhood results in a simple distributed optimal/suboptimal control. Therefore, in distributed optimal control methods an important issue is how to pair inputs and outputs. To answer this question, a few decomposition methods have been developed in the last three decades. One developed method is Relative Gain Array (RGA) [11] that expresses interaction based on the d.c. gain. Although RGA is a successful method for many systems, incapability to afford non-minimum phase systems, insensitivity to delays and focusing on just one frequency is the weakness of this method. Similar methods have been also developed: Nidelinski Index(NI) [12] in 1971 and Relative Interaction Array(RIA) [13] in 1976 were developed which are also based on d.c gain. Later in 1977, Relative Dynamic Gain Array(RDGA) [14] was presented that expresses how interaction varies with frequency and demonstrates bandwidth as an alternative for pairing. This line of research was continued through Generalized Relative Dynamic Gains(GRDG) [15]. The main restriction of GRDG is to cope mainly with 2×2 systems. In 1986, using graph theory, Nested ϵ Decompositions [16] with neglecting vectors smaller than ϵ leads to decomposition of systems. In 1998, Scaling algorithm and Reordering algorithm [17] were presented to transform system into diagonal dominance or block diagonal dominance form. In 2000, using controllability and observability gramians, Hankel Interaction Index array(HIIA) [18], for choosing control structure was presented. This work continued by Salgado and in 2004 Participation Matrix(PM) was presented [19]. In 2003, a method for decomposition of large scale systems to block diagonal dominance, using a permutation matrix was presented [20].

All of these methods are incapable of decomposing large scale systems into disjoint neighborhoods of subsystems with prespecified size for each neighborhood, which is the subject of investigation in this paper.

B. Paper Contributions

In this paper, the distributed optimal control method of [5], [6], [9], which is based on Jacobi iteration, is used to solve the constrained linear quadratic finite horizon optimal control problem, in which the size of each neighborhood is assumed to be specific a priori for managing communication load. Hence, choosing subsystems inside of each neighborhood is an issue. New method for decomposition of systems into disjoint neighborhoods of subsystems with specific size is presented. To develop this method, two new notions of Interaction Strength(IS) and Strength Weight(SW) are introduced and used to obtain the best decomposition in terms of minimum number of iterations for convergence to the optimal solution.

978-1-5090-5963-8/17/ $31.00 2017 IEEE

The satisfactory performance of the proposed decomposition method is illustrated via extensive computer simulation.

C. Paper Organization

The paper is organized as follows: In Section II the distributed optimal control method with two-level architecture for communication is presented and the problem formulation is given. Section III is devoted to the new method for decomposition. In this section, the new notions of Interaction Strength and Strength Weight for decomposition are presented. Simulation results are given in Section IV; and the paper is concluded in Section V by summarizing the main contributions of the paper.

II. DISTRIBUTED OPTIMAL CONTROL METHOD WITH HIERARCHICAL ARCHITECTURE FOR COMMUNICATION

In many large scale systems, dynamical behavior can be modeled by n distributed interacting linear time invariant subsystems of the following form [8]

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

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 the other subsystems on S_i. For the system (1) we are interested in solving the following Linear-Quadratic (LQ) constrained optimal control problem (problem (3)-(4)) subject to the dynamics of subsystems and operational constraints: $x_i[k] \in X_i$ and $u_i[k] \in G_i$, where X_i is a closed convex subset of real Euclidean space with dimension $n_i > 0$ (i.e., $X_i \subset \mathbf{R}^{n_i}$) modeling the constraint set on the ith state variable, and G_i is closed convex subset of $\mathbf{R}^{m_i}$ modeling the time invariant constraint set on the ith decision variable.

$$\min_{u} \{J(x[0], u_1, ..., u_n), x_i[k] \in X_i, u_i[k] \in G_i, \forall i, k\} \tag{3}$$

$$J(x[0], u_1, ..., u_n) = \sum_{i=1}^{n} \sum_{k=0}^{N-1} \parallel x_i[k] - x_i^d \parallel_Q^2 + \parallel u_i[k] \parallel_R^2 \tag{4}$$

where $x[0]$ 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.

It is shown in [6] that the above LQR optimal control problem can be written as follows:

$$\min_{(u_1, ..., u_n)} \{J(u_1, ..., u_n), u_i \in \mathcal{H}_i, \forall i\}. \tag{5}$$

where $J(u_1, ., u_n)$ is a quadratic functional of decision variables $u_i \in \mathbf{R}^{Nm_i}$ and $\mathcal{H}_i$ is a closed convex subset of $\mathbf{R}^{Nm_i}$. In the above problem it is assumed that each of n interacting subsystems: $S_1, ..., S_n$ is equipped with a decision maker with limited computational power for solving the optimization problem (5).

To solve this optimization problem, the distributed optimization technique of [5], [6], [9] is used. To manage the communication load, this optimization technique uses a two-level architecture for exchanging information among distributed decision makers. This communication architecture involves a collection of disjoint neighborhoods of subsystems. In each neighborhood at least one decision maker is selected as the neighborhood cluster head such that all the subsystems of the neighborhood and also all the subsystems of the nearest neighboring neighborhoods are within the effective communication range of the neighborhood cluster head so that the communication graph between cluster heads is connected. That is, there is a communication path between a cluster head to any other cluster heads. Within a neighborhood, subsystems are frequently communicate with each other, whereas communication between neighborhoods via cluster heads is less frequent. Obviously, the larger the size of neighborhood is, the larger is the communication load in exchanging information between subsystems inside the neighborhood. That is, communication load is affected by the size of neighborhoods; and hence, communication load can be managed by the size of neighborhoods. Without loss of generality, suppose subsystems $S_1, ..., S_n$ are distributed into q disjoint neighborhoods, as follows: $N_1 = \{S_1, ..., S_{l_1}\}$,$N_2 = \{S_{l_1+1}, ..., S_{l_2}\}$, ... ,$N_q = \{S_{l_{q-1}+1}, ..., S_n\}$. Then, the distributed optimization technique of [5], [6] approximates the solution of the optimization problem (5) by taking the following three steps:

- *Initialization*: The information exchange between neighborhoods at outer iterate $t \in \{0, 1, 2, ..., T_\epsilon - 1\}$ makes it possible for subsystem S_i to initialize its local decision variables as $h_i^0 = u_i^t \in \mathcal{H}_i, \forall i \in \{1, ..., n\}$, where $u_i \in \mathcal{H}_i$ is chosen arbitrary at $t = 0$.
- *Inner iterate*: Between every two successive outer iterates there are $\bar{p}$ inner iterates. Subsystem $S_i \in N_r (r \in \{1, 2, ..., q\})$ performs $\bar{p}$ inner iterates, as follows:
  For each inner iterate $p \in \{1, 2, ..\bar{p} - 1\}$, subsystem S_i first updates its decision variable via
  $$h_i^{p+1} = \pi_j h_i^* + (1 - \pi_j) h_i^p, \tag{6}$$
  where π_j are chosen subject to $\pi_j > 0, (j = 1, ..., n), \sum_{j=1}^{l_1} \pi_j = 1, ..., \sum_{j=l_{q-1}+1}^{n} \pi_j = 1$ and $h_i^* = \arg\min_{h_i \in \mathcal{H}_i} J(h_1^0, ..., h_{l_r}^0, h_{l_{r-1}}^0, h_{l_{r-1}+1}^p, ..., h_i, ..., h_{l_r}^p, h_{l_r+1}^0, ..., h_n^0)$ (note that $l_0 = 0, l_q = n$). Then, it trades its updated decision variable h_i^{p+1} with all other subsystems in its neighborhood N_r.
- *Outer iterate*: After $\bar{p}$ inner iterates, there is an outer iterate update as follows:
  $$u_i^{t+1} = \lambda_i h_i^{\bar{p}} + (1 - \lambda_i) u_i^t, \tag{7}$$
  where $t \in \{0, 1, ..., T_\epsilon - 1\}$ and $\lambda_i, i = \{1, 2, ..., n\}$, are chosen subject to $\lambda_i > 0$, $\lambda_1 = ... = \lambda_{l_1}$, $\lambda_{l_1+1} = ... = \lambda_{l_2}$, ... , $\lambda_{l_{q-1}+1} = ... = \lambda_{l_q}$ ($\lambda_{l_q} = \lambda_n$), $\lambda_{l_1} + \lambda_{l_2} + ... + \lambda_{l_q} = 1$. Then, three is an outer iterate communication, in which the updated decision variable u_i^{t+1} are shared between all neighborhoods; and subsequently, between all subsystems.

The above Three-step algorithm is repeated T_ϵ times. When $T_\epsilon \to \infty$, then $u_i^{T_\epsilon}$ converges to the optimal solution of the optimization problem [9]. Hence, $u_i^{T_\epsilon} =$

$(u_i^{T_\epsilon}{}'[0]\ u_i^{T_\epsilon}{}'[1]\ ...\ u_i^{T_\epsilon}{}'[N-1])' \in \mathbf{R}^{Nm_i}$ represents the approximated solution of the optimization problem (5).

The communication load of the above iterative optimization technique is managed by the size of neighborhoods [8]. Hence, the size of neighborhoods must be determined so that the communication load is acceptable. Now, for given values for parameters π_j, λ_i and specific size for each neighborhood, the question is how to find the best clustering of subsystems inside of each neighborhood that yields to the fastest convergence rate to the optimal solution. In other words, the objective in this paper is to find the best clustering of subsystems inside each neighborhood that results in the minimum number of total iteration for convergence as defined below.

Definition 2.1: (Total Number of Iteration for Convergence): For a given $\epsilon > 0$, the above three-step algorithm is terminated as soon as the following inequality holds

$$|J(u_1^t, ..., u_n^t) - J(u_1^{t-1}, ..., u_n^{t-1})| \leq \epsilon \tag{8}$$

Note that for small values for ϵ, there will be very small improvement in the approximation of the optimal solution by the above distributed optimization algorithm; and as the algorithm converges, the algorithm can therefore be terminated as soon as the above inequality holds. Now, let T_ϵ be the smallest integer such that (8) holds. Then, T_ϵ is referred as the total number of iterations for convergence.

We refer to $J(u_1^{T_\epsilon}, ..., u_n^{T_\epsilon})$ as an approximation of the optimal cost and the sequence $(u_1^{T_\epsilon}, ..., u_n^{T_\epsilon})$ as an approximation of the optimal solution. For a given size of neighborhoods, π_j and λ_i, the objective in this paper is to find the best decomposition for system that results in the minimum value for T_ϵ. A method for such a decomposition is given next.

III. Decomposition of Large Scale System Based on the Response of System

In the distributed optimization technique of previous section, decision variables are exchanged between subsystems and/or neighborhoods. Consequently, the interaction between subsystems must be expressed in terms of decision variables u_is. To achieve this goal, the distributed system (1) is represented in the following augmented form:

$$x[k+1] = Ax[k] + Bu[k], \tag{9}$$

where $x[k] = \left(x_1'[k]\ \ x_2'[k]\ \ ...\ \ x_n'[k]\right)'$ and $u[k] = \left(u_1'[k]\ \ u_2'[k]\ \ ...\ \ u_n'[k]\right)'$. Subsequently, the response of the augmented system is as follows:

$$x[k] = A^k x[0] + \sum_{i=0}^{k-1} A^{k-i-1} B u[i]. \tag{10}$$

Throughout, it is assumed that the system matrix A of the augmented system is stable. Many large scale systems, such as automated irrigation network and smart building have stable dynamic. As system (9) is stable, when $k \to \infty$, then $\mu_\infty = \lim_{k\to\infty} \sum_{i=1}^{k-1} A^{k-1-i}B$ exists, where μ_∞ is an indicator for interactions between subsystems in terms of decision variables. Based on this indicator and in order to decompose system into disjoint neighborhoods with specific size, the Interaction Strength (IS) is defined as follows.

Definition 3.1: (Interaction Strength): Let μ_a be a matrix that contains the absolute value of each element of the matrix μ_∞. The IS_{ij} that represents the interaction between the decision variable of the jth subsystem to the ith subsystem is defined as follows:

$$IS_{ij} = \frac{\mu_a(i,j)}{\max_j(\mu_a(i,j))},$$

where $\mu_a(i,j)$ is the element of the matrix μ_a located at the ith row and the jth column.

After finding the Interaction Strength matrix, $IS = [IS_{ij}]$, the Strength Weight (SW) is defined as follows.

Definition 3.2: (Strength Weight): For each pair of subsystems, the SW is defined as follows:

$$SW(ij) = IS_{ij} + IS_{ji} \quad , i \neq j. \tag{11}$$

Similarly, for each collection of n subsystems, the SW is defined as follows:

$$SW(i_1, ..., i_n) = IS_{i_1 i_2} + IS_{i_2 i_1} + ... + IS_{i_{n-1} i_n} + IS_{i_n i_{n-1}}. \quad (i_1 \neq i_2 \neq ... \neq i_n) \tag{12}$$

Hence, SW represents the total amount of interactions among a collection of subsystems.

Now, to see how using these two notions of IS and SW, a decomposition with specific size for neighborhoods is obtained, consider the following example.

Example 3.1: Consider a distributed system with stable augmented form as given below:

$$x[k+1] = Ax[k] + Bu[k], \tag{13}$$

where the A matrix is given by (14) (the next page) and $B = diag(0.4322, 0.5649, 0.7460, 0.3170, 0.3459, 0.4390, 0.5426, 0.3456, 0.6427, 0.3356)$

Suppose that we are interested in decomposing the system (13) into five neighborhoods of size two. To achieve this goal, the IS matrix and SWs for each pair of subsystems are computed. In (15) the IS matrix for system (13) is given.

Using this matrix, SWs are calculated (using *D*efinition3.2) from the above IS matrix. Now, by inspection of SWs, it follows that the pair $\{9, 10\}$ has the largest SW followed by pairs $\{5, 6\}$, $\{3, 4\}$, $\{1, 2\}$, $\{7, 8\}$. Therefore, we decompose the system (13) as follows:

$$N_1 = \{s_9, s_{10}\}, N_2 = \{s_5, s_6\}, N_3 = \{s_3, s_4\},$$
$$N_4 = \{s_1, s_2\}, N_5 = \{s_7, s_8\}$$

Now, suppose we are interested in decomposing the system (13) into two neighborhoods of size five. SWs for this case is calculated by the following formula from the IS matrix.

$$SW(i_1, ..., i_5) = IS_{i_1 i_2} + IS_{i_2 i_1} + ... + IS_{i_4 i_5} + IS_{i_5 i_4}. \quad (i_1 \neq i_2 \neq ... \neq i_5). \tag{16}$$

$$A = \begin{bmatrix} 0.6930 & -0.6300 & -0.0839 & 0.0832 & -0.0088 & -0.0188 & -0.0216 & 0.0393 & -0.0541 & -0.0169 \\ 0.5931 & 0.6371 & -0.0645 & -0.0262 & 0.0562 & 0.0643 & 0.0633 & -0.0133 & -0.0652 & -0.0274 \\ 0.0123 & 0.0514 & 0.5951 & -0.5940 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0269 & 0.0555 & 0.6550 & 0.7356 & 0 & 0 & 0 & 0 & 0 & 0.0186 \\ 0.0244 & 0.0481 & 0 & 0 & 0.5100 & -0.5200 & 0 & 0 & 0 & 0.0958 \\ 0.0248 & 0.0505 & 0 & 0 & 00.5122 & 0.6260 & 0 & 0 & 0 & 0 \\ 0.0472 & 0.0236 & 0 & 0 & 0 & 0 & 0.6835 & 0.7151 & 0 & \\ 0.0135 & 0.0494 & 0 & 0 & 0 & 0 & -0.6000 & 0.612 & 0 & 0 \\ 0.0593 & 0.0205 & 0 & 0 & 0 & 0 & 0 & 0 & 0.6100 & -0.6335 \\ 0.0420 & 0.0274 & 0 & 0.0926 & 0.0854 & 0 & 0 & 0 & 0.7000 & 0.7180 \end{bmatrix} \quad (14)$$

$$IS = \begin{bmatrix} 0.4331 & 1.0000 & 0.1340 & 0.0206 & 0.0850 & 0.0057 & 0.0881 & 0.0474 & 0.0276 & 0.0213 \\ 1.0000 & 0.6661 & 0.0694 & 0.1490 & 0.0307 & 0.0162 & 0.0350 & 0.0260 & 0.1102 & 0.0648 \\ 0.0781 & 0.0153 & 1.0000 & 0.9788 & 0.0017 & 0.0029 & 0.0065 & 0.0010 & 0.0418 & 0.0160 \\ 0.0747 & 0.0121 & 1.0000 & 0.2704 & 0.0003 & 0.0016 & 0.0047 & 0.0002 & 0.0065 & 0.0070 \\ 0.0150 & 0.0103 & 0.0091 & 0.0004 & 0.5664 & 1.0000 & 0.0005 & 0.0004 & 0.1294 & 0.0362 \\ 0.1763 & 0.0147 & 0.0378 & 0.0257 & 0.8311 & 1.0000 & 0.0136 & 0.0014 & 0.1766 & 0.0612 \\ 0.1333 & 0.0177 & 0.0178 & 0.0162 & 0.0040 & 0.0021 & 0.8406 & 1.0000 & 0.0085 & 0.0046 \\ 0.0210 & 0.0558 & 0.0071 & 0.0008 & 0.0047 & 0.0003 & 1.0000 & 0.3402 & 0.0018 & 0.0013 \\ 0.0622 & 0.0123 & 0.2799 & 0.0787 & 0.0709 & 0.1299 & 0.0052 & 0.0008 & 0.8298 & 1.0000 \\ 0.0758 & 0.0730 & 0.0944 & 0.0279 & 0.0142 & 0.0387 & 0.0096 & 0.0037 & 1.0000 & 0.2914 \end{bmatrix} \quad (15)$$

Then, by inspection, it follows that the collection $\{1, 2, 3, 9, 10\}$ has the largest SW. Hence, the system (13) is decomposed as follows:

$$N_1 = \{s_1, s_2, s_3, s_9, s_{10}\}, N_2 = \{s_4, s_5, s_6, s_7, s_8\}$$

Now, suppose we are interested in decomposing the system (13) into two neighborhoods of size four and one neighborhood with size two. SWs for this case is calculated by the following formula from the IS matrix.

$$SW(i_1, ..., i_4) = IS_{i_1 i_2} + IS_{i_2 i_1} + ... + IS_{i_3 i_4} + IS_{i_4 i_3}. \quad (i_1 \neq i_2 \neq ... \neq i_4). \quad (17)$$

Then, by inspection, it follows that the collection $\{1, 2, 3, 4\}$ has the largest SW after that $\{5, 6, 7, 8\}$ has the large value. Therefore, these two set cluster as two neighborhoods and the rest $\{9, 10\}$ puts in one neighborhood. Hence, the system (13) is decomposed as follows:

$$N_1 = \{s_1, s_2, s_3, s_4\}, N_2 = \{s_5, s_6, s_7, s_8\}, N_3 = \{s_9, s_{10}\}$$

IV. SIMULATION STUDY

TABLE I. COMPARISON OF T_ϵ FOR $\epsilon = 8$ FOR SEVERAL DECOMPOSITION WITH TWO SUBSYSTEMS IN EACH NEIGHBORHOOD

Subsystems placement	Numbers of iteration (T_ϵ)
$\{s_1, s_2\}, \{s_3, s_5\}, \{s_4, s_6\}, \{s_7, s_9\}, \{s_8, s_{10}\}$	65
$\{s_1, s_2\}, \{s_3, s_4\}, \{s_5, s_6\}, \{s_7, s_8\}, \{s_9, s_{10}\}$	$\mathbf{T}^*_\epsilon = \mathbf{24}$
$\{s_1, s_9\}, \{s_2, s_5\}, \{s_3, s_{10}\}, \{s_4, s_7\}, \{s_6, s_8\}$	25
$\{s_1, s_2\}, \{s_3, s_4\}, \{s_5, s_8\}, \{s_6, s_7\}, \{s_9, s_{10}\}$	43
$\{s_1, s_{10}\}, \{s_2, s_9\}, \{s_3, s_8\}, \{s_4, s_7\}, \{s_5, s_6\}$	32

In the distributed optimization technique of Section II, the decision variables are exchanged between subsystems and neighborhoods. Now, as the Interaction Strength (IS) in the proposed decomposition method of Section III is defined in terms of decision variables, this decomposition method results in the best clustering of subsystems into disjoint neighborhoods with specific size. For the purpose of illustration, we apply the Jacobi based distributed optimization method to the system (13) subject to the finite horizon quadratic cost functional (3)-(4) with the following specifications: $N = 6, Q = 100.I, R = I$, $x_i[k] \in [-12, 12]$, $u_i[k] \in [-8, 8]$, $n = 10, \bar{p} = 10, x_i[0] = 0 (i = \{1, 2, ..., 10\})$, $x_d = (3.5377, 2.8339, -2.2588, -4.8622, 3.3188, -1.3077, -3.4336, 0.3426, 3.5784, 2.7694)'$.

TABLE II. COMPARISON OF T_ϵ FOR $\epsilon = 8$ FOR SEVERAL DECOMPOSITION WITH FIVE SUBSYSTEMS IN EACH NEIGHBORHOOD

Subsystems placement	Numbers of iteration (T_ϵ)
$\{s_1, s_3, s_5, s_7, s_9\}, \{s_2, s_4, s_6, s_8, s_{10}\}$	38
$\{s_1, s_4, s_7, s_9, s_{10}\}, \{s_2, s_3, s_5, s_6, s_8\}$	32
$\{s_1, s_3, s_4, s_5, s_6\}, \{s_2, s_7, s_8, s_9, s_{10}\}$	27
$\{s_1, s_5, s_6, s_9, s_{10}\}, \{s_2, s_3, s_4, s_7, s_8\}$	27
$\{s_1, s_2, s_5, s_6, s_9\}, \{s_3, s_4, s_7, s_8, s_{10}\}$	33
$\{s_1, s_2, s_3, s_5, s_9\}, \{s_4, s_6, s_7, s_8, s_{10}\}$	33
$\{s_1, s_2, s_3, s_9, s_{10}\}, \{s_4, s_5, s_6, s_7, s_8\}$	$\mathbf{T}^*_\epsilon = \mathbf{25}$

TABLE III. COMPARISON OF T_ϵ FOR $\epsilon = 8$ FOR SEVERAL DECOMPOSITION WITH TWO NEIGHBORHOODS WITH FOUR SUBSYSTEMS IN ADDITION ONE NEIGHBORHOOD WITH TWO SUBSYSTEMS

Subsystems placement	Numbers of iteration (T_ϵ)
$\{s_1, s_2, s_3, s_8\}, \{s_4, s_5, s_6, s_7\}, \{s_9, s_{10}\}$	37
$\{s_1, s_4, s_7, s_9\}, \{s_2, s_5, s_6, s_8\}, \{s_3, s_{10}\}$	49
$\{s_1, s_2, s_3, s_4\}, \{s_5, s_6, s_8, s_9\}, \{s_7, s_{10}\}$	44
$\{s_1, s_3, s_9, s_{10}\}, \{s_2, s_5, s_6, s_8\}, \{s_4, s_7\}$	44
$\{s_1, s_2, s_3, s_4\}, \{s_5, s_6, s_7, s_8\}, \{s_9, s_{10}\}$	$\mathbf{T}^*_\epsilon = \mathbf{15}$
$\{s_1, s_2, s_5, s_6\}, \{s_3, s_4, s_7, s_8\}, \{s_9, s_{10}\}$	16

The simulation results for the above optimization problem are compared for different decompositions of the system (13), as follow:

In five-neighborhood decomposition with size two, we set $\lambda_i = \frac{1}{2}$; and for each subsystem inside a neighborhood, we set the parameter π_j as $\pi_j = \frac{1}{2}$. We also set $\epsilon = 8$ and use the stopping criterion of Definition 2.1. Via extensive search

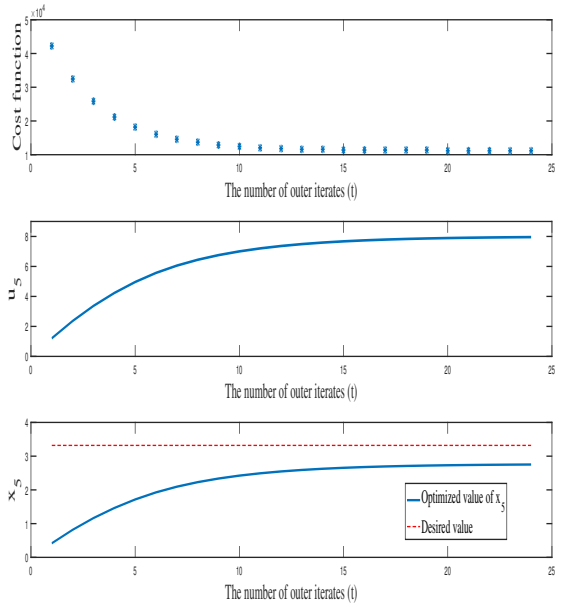

Fig. 1. Simulation Results for 5 neighborhoods with 2 subsystems

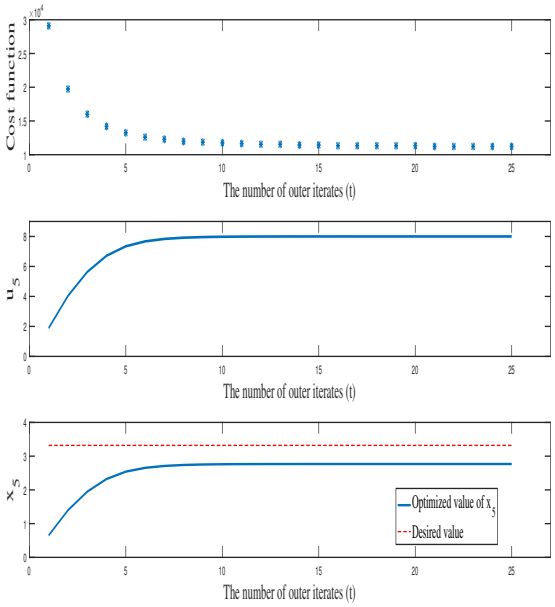

Fig. 2. Simulation Results for 2 neighborhoods with 5 subsystems

and by calculation of T_ϵ of all pairs, it is determined that $T_\epsilon^* = 24$, for $\epsilon = 8$. For $\epsilon = 8$, T_ϵs for several of other decompositions have been given in Table I. From this table, it is clear that the proposed decomposition of Section III finds the best decomposition without requiring exhausting search.

In two-neighborhood decomposition with size five, we set $\lambda_i = \frac{1}{2}$; and for each subsystem inside a neighborhood, we set the parameter π_j as $\pi_j = \frac{1}{5}$. Via extensive search and by calculation T_ϵ of all collections of five subsystems, it is determined that $T_\epsilon^* = 25$, $\epsilon = 8$. For $\epsilon = 8$, T_ϵ for several decomposition is given in Table II. From this table it is clear that the proposed decomposition of Section III finds the best decomposition without requiring any frustrating effort.

Fig. 1 illustrates the value of $J(u_1^t, ..., u_{10}^t)$, u_5 and x_5 for the condition simulated and five-neighborhood case. As it is clear from this figure after $T_\epsilon^* = 24$ iterations, there is no significant improvement in the value of the cost functional; while the value of x_5 is close enough to the desired set point. Therefore, the iterative algorithm can be terminated at $T_\epsilon^* = 24$.

Fig. 2 illustrates the value of $J(u_1^t, ..., u_{10}^t)$, u_5 and x_5 for the condition simulated and two-neighborhood case. As it is clear from this figure after $T_\epsilon^* = 25$ iterations, there is no significant improvement in the value of the cost functional; while the value of x_5 is close enough to the desired set point. Therefore, the iterative algorithm can be terminated at $T_\epsilon^* = 25$.

Fig. 3 illustrates the value of $J(u_1^t, ..., u_{10}^t)$, u_5 and x_5 for the condition simulated and 3-neighborhood case which are heterogeneously. As it is manifest from the figure, after $T_\epsilon^* = 15$ iterations the cost function would not change and the value of x_5 reach to desired value.

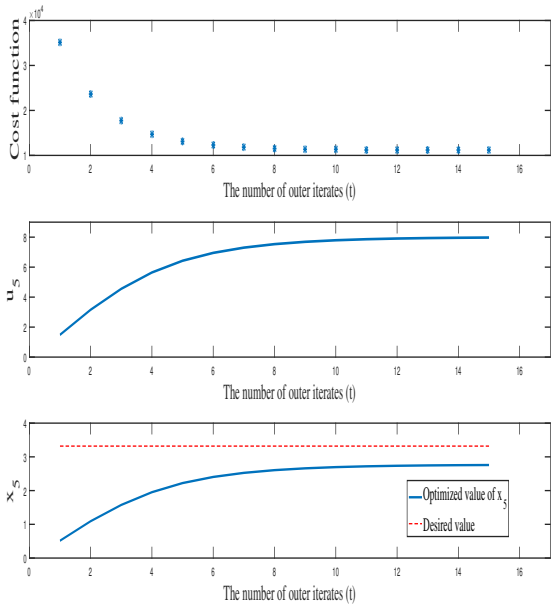

Fig. 3. Simulation Results for 2 neighborhoods with 4 subsystems and 1 neighborhood with 2 subsystems

V. Conclusion

This paper presented a new method for decomposing large scale stable systems into disjoint neighborhoods of subsystems with specific size determined a priori for managing communication load of the Jacobi based distributed optimal control method of [5], [6], [9]. The proposed decomposition method clusters subsystems into disjoint neighborhoods based on the strength of control inputs interaction between subsystems. The

proposed method is a new method and uses the time response of system for decomposition. Using extensive computer simulation it was illustrated that this method finds the best decomposition. For future, it is interesting to extend the results of this paper to unstable systems.

ACKNOWLEDGMENT

This work was supported by the research office of Sharif University of Technology.

REFERENCES

[1] D. E. Kirk, *Optimal Control Theory: A Introduction*. Prentice Hall, 1970.

[2] N. R. Sandell Jr, P. Varaiya, M. Athans, and M. G. Safonov, "Survey of decentralized control methods for large scale systems," *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 108–128, 1978.

[3] M. Ikeda, "Decentralized control of large scale systems," in *Three decades of mathematical system theory*, pp. 219–242, Springer, 1989.

[4] R. Scattolini, "Architectures for distributed and hierarchical model predictive control–a review," *Journal of Process Control*, vol. 19, no. 5, pp. 723–731, 2009.

[5] B. T. Stewart, J. B. Rawlings, and S. J. Wright, "Hierarchical cooperative distributed model predictive control," in *American Control Conference (ACC), 2010*, pp. 3963–3968, IEEE, 2010.

[6] A. Farhadi, M. Cantoni, and P. M. Dower, "Performance and information pattern trade-offs in a consensus based distributed optimization method," in *The 2nd AustralianControl Conference (AUCC)*, pp. 373–380, 2012.

[7] C. M. M. Farhadi, Alireza and P. M. Dower, "Computation time analysis of a distributed optimization algorithm applied to automated irrigation networks," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pp. 2193–2199, IEEE, 2013.

[8] C. Farhadi, A and Dower, "Computation time analysis of a centralized and distributed optimization algorithms applied to automated irrigation networks," *in Proceedings of the 2013 Australian Control Conference*, pp. 263–269, 2015.

[9] A. Farhadi and A. Khodabandehlou, "Distributed model predictive control with hierarchical architecture for communication: application in automated irrigation channels," *International Journal of Control*, vol. 89, no. 8, pp. 1725–1741, 2016.

[10] D. Siljak, *Decentralized Control of Complex Systems*. Dover Books on Electrical Engineering, Dover Publications, 2013.

[11] E. Bristol, "On a new measure of interaction for multivariable process control," *IEEE transactions on automatic control*, pp. 133–134, 1966.

[12] A. Niederlinski, "A heuristic approach to the design of linear multivariable interacting control systems," *Automatica*, vol. 7, no. 6, pp. 691–701, 1971.

[13] Z.-X. Zhu, "Variable pairing selection based on individual and overall interaction measures," *Industrial & engineering chemistry research*, vol. 35, no. 11, pp. 4091–4099, 1996.

[14] M. Witcher and T. McAvoy, "Interacting control-systems-steady-state and dynamic measurement of interaction," *ISA transactions*, vol. 16, no. 3, pp. 35–41, 1977.

[15] E. Gagnon, A. Desbiens, and A. Pomericau, "Selection of pairing and constrained robust decentralized pi controllers," in *American Control Conference, 1999. Proceedings of the 1999*, vol. 6, pp. 4343–4347, IEEE, 1999.

[16] M. Sezer and D. Šiljak, "Nested ε-decompositions and clustering of complex systems," *Automatica*, vol. 22, no. 3, pp. 321–331, 1986.

[17] J. Edmunds, "Input and output scaling and reordering for diagonal dominance and block diagonal dominance," *IEE Proceedings-Control Theory and Applications*, vol. 145, no. 6, pp. 523–530, 1998.

[18] B. Wittenmark and M. E. Salgado, "Hankel-norm based interaction measure for input-output pairing," in *Proc. of the 2002 IFAC World Congress*, 2002.

[19] M. E. Salgado and A. Conley, "Mimo interaction measure and controller structure selection," *International Journal of Control*, vol. 77, no. 4, pp. 367–383, 2004.

[20] X.-B. Chen and W. Wang, "Block diagonally dominant decomposition for large scale systems," in *Control and Automation, 2003. ICCA'03. Proceedings. 4th International Conference on*, pp. 555–559, IEEE, 2003.