This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

Analysis of Model and Iteration Dependencies in Distributed Feasible-Point Algorithms for Optimal Control Computation*

Mark A. Fabbro, Iman Shames, Michael Cantoni

***Abstract*—The problem of computing optimal control inputs is studied for networks of dynamical linear systems, with respect to separable input constraints and a separable quadratic cost over a finite time-horizon. The main results concern network structures for which the iterates of three feasible-point algorithms can be computed exactly on a subsystem-by-subsystem basis with access restricted to local model data and algorithm-state information. In particular, hop-based network proximity bounds are investigated for algorithms based on projected gradient, random co-ordinate descent and Jacobi iterations, via graph-based characterisations of various aspects of an equivalent static formulation of the optimal control problem.**

I. Introduction

Within the context of controlling large-scale systems, a problem of interest is the computation of constrained system inputs that optimise a control objective over a finite time-horizon. Iterative algorithms that distribute across a network of computing agents exist for solving such optimisation problems. The ability to respect limitations on the model-data and algorithm-state information available to each agent may be important. For example, when privacy is a concern, a restriction on the level of model-data and algorithm-state information exchange can be viewed as a limit on the extent to which privacy is sacrificed; see [1]–[3]. Similarly, in situations with a deadline for the result (e.g. receding-horizon/model-predictive control), a reduction of communication overheads, achieved by respecting limits on information exchange, can afford time for computing additional iterates. Algorithms based on dual-decomposition techniques achieve distributed computation in a way that respects one-hop limits on the information available at each step, assuming a computing network architecture that coincides with that of the network of dynamical systems under control. However, this comes at the expense of no guarantees on the feasibility of all iterations, precluding early termination of such algorithms without special care [4].

This paper is focused on the computation of optimal control inputs over a finite time-horizon for networks of linear dynamical systems subject to separable input constraints and a separable quadratic cost. The optimal control problem is formulated as an equivalent static quadratic programme by stacking the variable in a particular way. The implementation of three iterative feasible-point algorithms (variants of *projected gradient*, *random co-ordinate descent* and *Jacobi iterations*) for solving this quadratic programme are investigated from the perspective of the model-data and algorithm-state information required at each step and at each computing agent. The convergence properties of these algorithms, which naturally distribute across a computing network architecture that matches the network of dynamical systems, is well understood to be linear (i.e., exponential) in the case of positive definite cost [5], [6]. Moreover, the feasible-point property of each iteration ensures that pre-convergence termination is acceptable, at least in terms of input constraint satisfaction and consistency with the model dynamics.

The main contribution of this paper is an analysis of network-hop based limits on the model-data and algorithm-state information available to each computing agent such that *exact* implementation of the aforementioned iterative algorithms is possible. Exact implementation of the algorithms ensures inheritance of corresponding convergence and feasibility properties. Graph models of the network interconnection structure and the information available for computation underpins the analysis. This leads to conditions on the network interconnection graph that are sufficient for iterates to be computable with information limited to a specified ℓ-hop neighbourhood of each agent.

The results are strongest for networks in which the dynamical subsystems are coupled exclusively through the subsystem control inputs, without delay, such as urban traffic networks [7] and gravity-powered water distribution networks [8]. Structure of this kind is not captured by the aforementioned network interconnection graph; i.e., it is finer level structure. In networks with input-coupling only, the iterates associated with each computing agent can be implemented with access restricted to model-data and algorithm-state information from agents that lie within a *two*-hop neighbourhood; this is independent of the time-horizon of the optimal control problem. By contrast, in the absence of any structural properties finer than that captured by the network interconnection graph, the analysis yields a network-hop bound that grows doubly with the length of the time horizon.

This paper extends [9], [10] by generalising results to include conditions for networks with interconnection graphs involving subsystem coupling via control *and* state signals. An outline of the paper is as follows. The problem investigated, including details of the three iterative algorithms studied and graph-based characterisations of network structure and model/algorithm-state information limits, is formulated in

*This work is supported in part by a McKenzie Fellowship and the Australian Research Council (DP130104510).

The authors are with the Department of Electrical and Electronic Engineering, University of Melbourne {iman.shames, cantoni}@unimelb.edu.au, m.a.fabbro@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

Section II. Network proximity analysis of model and iteration dependencies is pursued in Section III, where the main results are all stated. Concluding remarks and future directions are given in Section IV.

II. Problem Formulation

First, some basic notation is established. A_{ij} denotes the ij-th block of block matrix A, and a_{ij} is the j-th entry of vector a_i. For given integers a and b, $i = a : b$ means the same as $i = a, a+1, \ldots, b$. The expression $A \succ 0$ is used if A is symmetric positive definite, $A^\top$ for the transpose of A and, I_m and 0_m for the $m \times m$ identity-matrix and zero-matrix respectively. Moreover, $A \otimes B$ denotes the Kronecker product of A by B. The notation $x(t)$ and $x(t_1 : t_2)$ means the value of x at time t and the concatenation of values of x at times $t = t_1 : t_2$, respectively. Finally, $\wedge$ is logical conjunction and $\vee$ logical disjunction.

The following graph theoretic terminology is used in subsequent analysis to characterise important aspects of the problem of computing optimal control inputs for the network over a finite time-horizon. A graph is an ordered pair $\mathcal{R} = (\mathcal{P}, \mathcal{Q})$, where $\mathcal{P}$ is a set of elements called *vertices* or *nodes* and $\mathcal{Q}$ is a set of ordered pairs of vertices, called *edges* or *arcs*. A graph is called *undirected* if $(i,j) \in \mathcal{Q} \iff (j,i) \in \mathcal{Q}$; otherwise it is *directed*. A *walk* is a sequence $w_{v_1 v_k} = (v_1, e_1, v_2, e_2, \ldots, e_{k-1}, v_k)$ of vertices $v_i \in \mathcal{P}$ and edges $e_i \in \mathcal{Q}$ such that $e_i = (v_i, v_{i+1})$ for all $i < k$. The vertex v_1 is called the *initial* vertex of $w_{v_1 v_k}$ and v_k is the *terminal* vertex of $w_{v_1 v_k}$. The *length* of the walk is the number of edges in $w_{v_1 v_k}$. A *path* in $\mathcal{R}$ is a walk in which all vertices are distinct and a *cycle* in $\mathcal{R}$ is a walk of non-zero length with the same initial and terminal vertex. The *distance* $d_{\mathcal{R}}(i,j)$ in $\mathcal{R}$ between two vertices $i, j \in \mathcal{P}$ is the length of a *shortest* path with initial vertex i and terminal vertex j, or ∞ if no such path exists, or 0 if $i = j$. The p-th *power* of a graph $\mathcal{R}$ is a graph with the same set of vertices as $\mathcal{R}$ and an edge between two vertices if and only if there is a path of length at most p between them in $\mathcal{R}$. A *rooted tree* is a directed graph such that the underlying undirected graph has no cycles and all edges point away from a *root* vertex which has zero in-degree. A *cycle* graph consists of a single cycle, and the *disjoint union* of two graphs with disjoint vertex sets is constructed from the union of the vertex sets and the union of the edge sets. For a given integer $\ell \geq 0$, the *ℓ-hop in-neighbours* and *ℓ-hop out-neighbours* of node i in the graph $\mathcal{R}$ are denoted by $\mathcal{N}_i^-(\mathcal{R}, \ell) = \{v : d_{\mathcal{R}}(v,i) \leq \ell\}$ and $\mathcal{N}_i^+(\mathcal{R}, \ell) = \{v : d_{\mathcal{R}}(i,v) \leq \ell\}$, respectively. Moreover, the set of *mutual neighbours* that are within a distance of α from a node i and within a distance of θ from a node j, simultaneously, is denoted by $\mathcal{W}_{ij}(\mathcal{R}, \alpha, \theta) = \{l : i \in \mathcal{N}_l^-(\mathcal{R}, \alpha) \wedge j \in \mathcal{N}_l^-(\mathcal{R}, \theta)\}$.

Now, consider a network of $n > 1$ interconnected discrete-time linear systems in set $\mathcal{V} = \{1 : n\}$. Each subsystem $i \in \mathcal{V}$ is described by the state-space model

$$\begin{aligned} x_i(t+1) &= A_{ii}x_i(t) + B_{ii}u_i(t) + v_i(t), \quad t \geq t_0 \\ x_i(t_0) &= \xi_i, \end{aligned} \tag{1}$$

where $x_i(t)$, $\xi_i \in \mathbb{R}^{p_i}$, $u_i(t) \in \mathbb{R}^{m_i}$, $A_{ii} \in \mathbb{R}^{p_i \times p_i}$, $B_{ii} \in \mathbb{R}^{p_i \times m_i}$ for some positive integers p_i, m_i, and

$$v_i(t) = \sum_{l \in \mathcal{V}\setminus\{i\}} B_{il}u_l(t) + \sum_{l \in \mathcal{V}\setminus\{i\}} A_{il}x_l(t). \tag{2}$$

The control inputs are restricted to convex polytopes $\mathcal{U}_i$, i.e. $u_i(t) \in \mathcal{U}_i$ for all $i \in \mathcal{V}$ and all $t \in t_0 : t_0 + \tau - 1$.

Definition 1. For the network of systems $\mathcal{V}$ with subsystem dynamics (1), the *interconnection graph* is defined to be $\mathcal{I} = (\mathcal{V}, \mathcal{E}_{\mathcal{I}})$, where $\mathcal{E}_{\mathcal{I}} = \{(j,i) \in \mathcal{V} \times \mathcal{V} : i \neq j \wedge (A_{ij} \neq 0 \vee B_{ij} \neq 0)\}$. Similarly, the *state-coupling interconnection graph* is defined to be the subgraph $\mathcal{A} = (\mathcal{V}, \mathcal{E}_{\mathcal{A}}) \subset \mathcal{I}$, where $\mathcal{E}_{\mathcal{A}} = \{(j,i) \in \mathcal{V} \times \mathcal{V} : i \neq j \wedge A_{ij} \neq 0\}$, and the *control-coupling interconnection graph* is defined to be the subgraph $\mathcal{B} = (\mathcal{V}, \mathcal{E}_{\mathcal{B}}) \subset \mathcal{I}$, where $\mathcal{E}_{\mathcal{B}} = \{(j,i) \in \mathcal{V} \times \mathcal{V} : i \neq j \wedge B_{ij} \neq 0\}$. Note $\mathcal{I} = (\mathcal{V}, \mathcal{E}_{\mathcal{A}} \cup \mathcal{E}_{\mathcal{B}})$.

Remark 1. Coupling delay between subsystems can be accommodated within the model structure (1-2) by augmenting the state of each node with a sufficiently large number of delay states as required for out-neighbours to access an appropriately delayed version of the input or state according to an equation of the form (2). This would lead to structure in the matrices A_{il} and A_{ii}, which is finer than that captured by the interconnection graph $\mathcal{I} = (\mathcal{V}, \mathcal{E})$. Such structure is not exploited in subsequent analysis, which is therefore conservative.

Definition 2. The *underlying network graph* associated with the interconnection graph $\mathcal{I}$ is defined to be the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E} = \mathcal{E}_{\mathcal{I}} \bigcup \{(j,i) : (i,j) \in \mathcal{E}_{\mathcal{I}}\}$. Moreover, the *$\ell$-hop neighbour set* of i in graph $\mathcal{G}$ is defined to be $\mathcal{N}_i(\mathcal{G}, \ell) = \{v \in \mathcal{V} : d_{\mathcal{G}}(i,v) \leq \ell\}$.

Given $\tau \geq 1$, the object is to calculate a τ-horizon optimal control input for the network of subsystem with dynamics (1) by minimising the following cost-function:

$$f(\mathbf{u}_1, \ldots, \mathbf{u}_n) = \tfrac{1}{2} \sum_{i \in \mathcal{V}} \sum_{t=t_0}^{t_0+\tau-1} f_i(x_i(t+1), u_i(t)) \tag{3}$$

where $f_i(x,u) = x^\top Q_i x + u^\top R_i u$, $Q_i \succ 0$, $R_i \succ 0$, and

$$\mathbf{u}_i = u_i(t_0 : t_0 + \tau - 1) \in \mathcal{U}_i^\tau = \mathcal{U}_i \times \cdots \times \mathcal{U}_i. \tag{4}$$

Defining

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1^\top & \ldots & \mathbf{u}_n^\top \end{bmatrix}^\top, \tag{5}$$

and $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_n$, the optimal control can be computed by solving the following quadratic programme:

$$\mathbf{u}^\star = \arg\min_{\mathbf{u} \in \mathcal{U}} f(\mathbf{u}). \tag{6}$$

The information needed to implement (exactly) the following three feasible-point algorithms for solving (6) is investigated below. These algorithms are all amenable to distributed implementation by virtue of the separable structure of the input constraints and cost function under consideration, assuming a computing network architecture that coincides with the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

network of dynamical systems under control. The algorithms can be viewed as special cases of the following:

$$\mathbf{u}^{k+1} = \left[\mathbf{u}^k - D(k)\nabla f(\mathbf{u}^k)\right]_{\mathcal{U}} \tag{7}$$

where

$$[u]_{\mathcal{U}} = \arg\min_{\mu\in\mathcal{U}} ||u-\mu||. \tag{8}$$

Algorithm 1 (Projected-gradient)**.** Let ρ_0, ρ, $\sigma > 0$ be such that $\nabla^2 f(\mathbf{u}) \leq \rho_0^{-1} I_{n(\tau-1)}$ for all $\mathbf{u} \in \mathcal{U}$ and $0 < \sigma \leq \rho \leq 2\rho_0 - \sigma$. In (7) set $D(k) = \rho$ for all k. See [11] for further details.

Remark 2. Note that choosing ρ requires some *a priori* global knowledge about the cost function, namely, an upper bound on the largest eigenvalue of the Hessian. This is simple to compute for a quadratic programme.

Algorithm 2 (Random co-ordinate descent)**.** At each iteration k randomly select a subsystem $i \in \mathcal{V}$ from some distribution for which each subsystem has non-zero probability. In (7) set $D(k) = \text{diag}\{0_{m_0}, \ldots, \rho_i I_{m_i}, \ldots, 0_{m_n}\}$ where ρ_i is a positive constant. See [6], [12] for further details.

Algorithm 3 (Jacobi iterations)**.** These involve the iterative solution of a collection of smaller sub-problems as indicated below:

$$\mathbf{u}^{k+1} = \begin{bmatrix} \arg\min\limits_{\mathbf{u}_1\in\mathcal{U}_1} f(\mathbf{u}_1, \mathbf{u}_2^k, \ldots, \mathbf{u}_n^k) \\ \vdots \\ \arg\min\limits_{\mathbf{u}_n\in\mathcal{U}_n} f(\mathbf{u}_1^k, \ldots, \mathbf{u}_{n-1}^k, \mathbf{u}_n) \end{bmatrix}. \tag{9}$$

See [5] for further details.

Remark 3. The update in (9) is the standard form of Jacobi iterations, however, the following iterations found in [13], for example, also converge to the same value:

$$\mathbf{u}_i^{k+1} = (1-w_i)\mathbf{u}_i^k + w_i\left(\arg\min_{\mathbf{u}_i\in\mathcal{U}_i} f(\mathbf{u}_1^k, \ldots, \mathbf{u}_i, \ldots, \mathbf{u}_n^k)\right)$$

where $w_i > 0$ and $\sum_{i\in\mathcal{V}} w_i = 1$. As both versions of the Jacobi iterations are the same in terms of the information required to implement an iterate, only (9) is considered subsequently (i.e. $w_i = 1$ for all i).

Conditions are sought such that at each node i the update $\mathbf{u}_i^{k+1}$ can be implemented exactly, given access to local model-data and algorithm-state information.

Definition 3. Given $\ell \geq 0$ and $i \in \mathcal{V}$, a so-called ℓ-hop limit restricts the information available to node i at time k to the following collection of objects

$$\mathcal{K}_i^k(\ell) = \left\{A_{jl}, B_{jl}, Q_j, \xi_j, \mathbf{u}_j^k\right\}_{\substack{j\in\mathcal{N}_i(\mathcal{G},\ell) \\ l\in\mathcal{N}_j^-(\mathcal{I},1)}} \bigcup \left\{R_i, \mathcal{U}_i\right\}.$$

Problem 1. For given $\ell > 0$, find conditions on the interconnection graph $\mathcal{I}$ such that, for all $i \in \mathcal{V}$, the iterates $\mathbf{u}_i^{k+1}$ in (7) or (9) can be implemented from the information in $\mathcal{K}_i^k(\ell)$ alone.

Remark 4. By definition, $\mathcal{K}_i^k(\ell)$ contains model-data and algorithm-state information collected from only those nodes within ℓ-hops of node i in the underlying interconnection graph $\mathcal{G}$. Hence, if $\mathcal{K}_i^k(\ell)$ contains enough information to implement the update of $\mathbf{u}_i^{k+1}$, then agent i need only communicate with agent $j \in \mathcal{N}_i(\mathcal{G}, \ell)$.

III. Graph-based Analysis of Iterate Dependencies

Subsequently, it is shown that for each $i \in \mathcal{V}$ the cost function in (3) can be decomposed as

$$f(\mathbf{u}) = g_i(\mathbf{u}) + h_i(\ldots, \mathbf{u}_{i-1}, \mathbf{u}_{i+1}, \ldots),$$

with $g_i(\cdot)$ dependent on $\mathbf{u}_i$ and $h_i(\cdot)$ not dependent on $\mathbf{u}_i$. Correspondingly, by exploiting the separability of the input constraint set $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_n$, the iterations in (7) can be decomposed as

$$\mathbf{u}_i^{k+1} = \left[\mathbf{u}_i^k - \varrho(k)\nabla g_{ii}(\mathbf{u}_i^k)\right]_{\mathcal{U}_i} \tag{10}$$

where $g_{ii} = \mathbf{u}_i \mapsto g_i(\ldots, \mathbf{u}_{i-1}^k, \mathbf{u}_i, \mathbf{u}_{i+1}^k, \ldots)$ and $\varrho(k)$ is either ρ, ρ_i or 0, depending on $D(k)$. Similarly, the Jacobi-iterations can be written as

$$\mathbf{u}_i^{k+1} = \arg\min_{\mathbf{u}_i\in\mathcal{U}_i} g_{ii}(\mathbf{u}_i). \tag{11}$$

The dependence of the mapping

$$g_{ii} = \mathbf{u}_i \mapsto g_i(\ldots, \mathbf{u}_{i-1}^k, \mathbf{u}_i, \mathbf{u}_{i+1}^k, \ldots) \tag{12}$$

on the model data and the algorithm state $\mathbf{u}^k$ is explored below. A graph-based characterisation of the dependencies is sought in a form that facilitates investigation of Problem 1. This leads to several results about interconnection graph structures for which the functional dependence of g_{ii} on $\mathbf{u}_i$ can be completely determined from information in $\mathcal{K}_i^k(\ell)$ at iteration $k > 0$, for given instances of the network-proximity based information limit $\ell > 0$.

A. Coupling Through Controls and States

The subsystem coupling (2) can be expressed as

$$v_i(t) = \sum_{l\in\mathcal{N}_i^-(\mathcal{A},1)\setminus\{i\}} A_{il}x_l(t) + \sum_{l\in\mathcal{N}_i^-(\mathcal{B},1)\setminus\{i\}} B_{il}u_l(t), \tag{13}$$

for $i \in \mathcal{V}$, where the state-coupling graph $\mathcal{A}$ and control-coupling graph $\mathcal{B}$ are as in Definition 1. Generally, $\mathcal{E}_\mathcal{A} \neq \emptyset$ and $\mathcal{E}_\mathcal{B} \neq \emptyset$. This case is considered first. The cases $\mathcal{E}_\mathcal{B} = \emptyset$ and $\mathcal{E}_\mathcal{A} = \emptyset$ are considered in Sections III-B and III-C, respectively.

Let

$$\mathbf{A} = \begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nn} \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} B_{11} & \cdots & B_{1n} \\ \vdots & \ddots & \vdots \\ B_{n1} & \cdots & B_{nn} \end{bmatrix}.$$

A state-space model for the overall network is then given by

$$x(t+1) = \mathbf{A}x(t) + \mathbf{B}u(t),$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

where $x(t) = [x_1(t)^\top, \dots, x_n(t)^\top]^\top$ and $u(t)$ defined similarly. Let $\Phi(k) = \mathbf{A}^k$ and

$$\mathbf{M}_{ij}(k) = \begin{bmatrix} \Phi_{ij}(0) & 0 & \cdots & 0 \\ \Phi_{ij}(1) & \Phi_{ij}(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \Phi_{ij}(k) & \cdots & \Phi_{ij}(1) & \Phi_{ij}(0) \end{bmatrix},$$

for $k = 0, 1, \dots$. Then, for $t = t_0 + 1 : t_0 + \tau$, each subsystem $i = 1 : n$ evolves according to

$$x_i(t) = \sum_{j \in \mathcal{V}} \Phi_{ij}(t - t_0)\xi_j + \sum_{k=t_0}^{t-1} \sum_{l,j \in \mathcal{V}} \Phi_{il}(t - k - 1)B_{lj}u_j(k).$$

Stacking $x_i(t)$ over the horizon, it follows that

$$\begin{bmatrix} x_i(t_0+1) \\ \vdots \\ x_i(t_0+\tau) \end{bmatrix} = \sum_{l,j \in \mathcal{V}} \Phi_{ij}(1:\tau)\xi_j + \mathbf{M}_{il}(\tau - 1)\mathbf{B}_{lj}\mathbf{u}_j \quad (14)$$

where $\mathbf{B}_{ij} = I_\tau \otimes B_{ij}$ and $\mathbf{u}_j$ is as in (4). As such, with (5), the cost function (3) can be re-written compactly as

$$\begin{aligned} f(\mathbf{u}) &= \tfrac{1}{2}\mathbf{u}^\top \Pi \mathbf{u} + \mathbf{u}^\top \Psi \xi + \xi^\top \Lambda \xi \\ &= g_i(\mathbf{u}) + h_i(\dots, \mathbf{u}_{i-1}, \mathbf{u}_{i+1}, \dots), \end{aligned}$$

where

$$g_i(\mathbf{u}) = \tfrac{1}{2}\mathbf{u}^\top \Pi_i^+ \mathbf{u} + \mathbf{u}^\top \Psi_i^- \xi.$$

The non-zero entries of Π_i^+ coincide in value and location with those of Π in the i-th row and column. All other entries are zero. Similarly, the non-zero entries of Ψ_i^- coincide in value and location with those of Ψ in the i-th row; remaining entries are zero valued. The matrices Π, Ψ, Λ are given by

$$\begin{aligned} \Pi_{ii} &= \mathbf{R}_i + \sum_{r,s,l \in \mathcal{V}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \mathbf{M}_{ls}(\tau - 1)\mathbf{B}_{si}, \quad i \in \mathcal{V}, \\ \Pi_{ij} &= \Pi_{ji}^\top = \sum_{r,s,l \in \mathcal{V}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \mathbf{M}_{ls}(\tau - 1)\mathbf{B}_{sj}, \quad i \neq j, \\ \Psi_{ij} &= \sum_{r,l \in \mathcal{V}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \Phi_{lj}(1:\tau), \quad i,j \in \mathcal{V}, \\ \Lambda_{ij} &= \tfrac{1}{2} \sum_{l \in \mathcal{V}} \Phi_{li}^\top(1:\tau)\mathbf{Q}_l \Phi_{lj}(1:\tau), \qquad i,j \in \mathcal{V}, \end{aligned} \quad (15)$$

with $\mathbf{Q}_l = I_\tau \otimes Q_l$ and $\mathbf{R}_i = I_\tau \otimes R_i$.

The following lemma is useful in establishing a relationship between Π, Ψ, and the coupling graphs $\mathcal{A}$ and $\mathcal{B}$. The proof is deferred to the appendix.

Lemma 1. *Let $\mathbf{A}$ be a square matrix, partitioned into blocks A_{ij} where $i, j \in \mathcal{V} = \{1 : n\}$. For any positive integer k, let $\Phi_{ij}(k)$ denote the ij-block of $\Phi = \mathbf{A}^k$. Associate with $\mathbf{A}$ the directed graph $\mathcal{D} = (\mathcal{V}, \mathcal{E}_\mathcal{D})$, where $\mathcal{E}_\mathcal{D} = \{(j,i) \in \mathcal{V} \times \mathcal{V} : A_{ij} \neq 0\}$ and define $\mathcal{P}(k,j,i)$ to be the set of all k-length walks in $\mathcal{D}$ from initial vertex j to terminal vertex i. Then,*

$$\mathcal{P}(k,j,i) = \emptyset \implies \Phi_{ij}(k) = 0.$$

Through the use of Lemma 1, the relationship between Π, Ψ, $\mathcal{A}$ and $\mathcal{B}$ can be understood by excluding the terms in (15) that are known to be zero by virtue of the network structure.

Consider, for example, the following term in the summation for Π_{ij} in (15), given fixed l, i, j, r, s and τ:

$$\Xi = (\mathbf{M}_{lr}(\tau - 1)\mathbf{B}_{ri})^\top \mathbf{Q}_l \mathbf{M}_{ls}(\tau - 1)\mathbf{B}_{sj}. \quad (16)$$

Clearly, $\Xi = 0$ if any of the multiplicands in (16) are zero. By Lemma 1, it follows that $\mathbf{M}_{lr}(\tau - 1) = 0$ if $\mathcal{D}$ contains no walks of length $(\tau - 1)$-or-fewer from vertex r to vertex l. Hence, $\mathbf{M}_{lr}(\tau - 1)$ and $\mathbf{M}_{ls}(\tau - 1)$ are both potentially non-zero if $\mathcal{D}$ contains a $(\tau - 1)$-or-fewer length walk from both r to l and s to l. The set $\mathcal{W}_{rs}(\mathcal{D}, \tau - 1, \tau - 1)$ (Definition 1) captures all vertices that satisfy this condition. Moreover, as $\mathcal{A}$ is equivalent to the graph constructed by removing all self-loops from $\mathcal{D}$, it follows that $\mathcal{W}_{rs}(\mathcal{D}, \tau - 1, \tau - 1) = \mathcal{W}_{rs}(\mathcal{A}, \tau - 1, \tau - 1)$. Similarly, if $s \notin \mathcal{N}_j^+(\mathcal{B}, 1)$, then $\mathbf{B}_{sj} = 0$. Hence, $\mathbf{B}_{sj}$ is potentially non-zero when $s \in \mathcal{N}_j^+(\mathcal{B}, 1)$.

Using the same analysis on the other summands in (15), the matrices Π, Ψ and Λ can be restated in a form which illuminates the relationship to the coupling graphs $\mathcal{A}$ and $\mathcal{B}$:

$$\begin{aligned} \Pi_{ii} &= \mathbf{R}_i + \sum_{\substack{r,s \in \mathcal{N}_i^+(\mathcal{B},1) \\ l \in \mathcal{W}_{rs}(\mathcal{A}, \tau-1, \tau-1)}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \mathbf{M}_{ls}(\tau - 1)\mathbf{B}_{si}, \quad i \in \mathcal{V} \\ \Pi_{ij} &= \Pi_{ji}^\top = \sum_{\substack{r \in \mathcal{N}_i^+(\mathcal{B},1), s \in \mathcal{N}_j^+(\mathcal{B},1) \\ l \in \mathcal{W}_{rs}(\mathcal{A}, \tau-1, \tau-1)}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \mathbf{M}_{ls}(\tau - 1)\mathbf{B}_{sj}, \quad i \neq j \\ \Psi_{ij} &= \sum_{\substack{r \in \mathcal{N}_i^+(\mathcal{B},1) \\ l \in \mathcal{W}_{rj}(\mathcal{A}, \tau-1, \tau)}} \mathbf{B}_{ri}^\top \mathbf{M}_{lr}^\top(\tau - 1)\mathbf{Q}_l \Phi_{lj}(1:\tau), \qquad i,j \in \mathcal{V} \\ \Lambda_{ij} &= \tfrac{1}{2} \sum_{l \in \mathcal{W}_{ij}(\mathcal{A}, \tau, \tau)} \Phi_{li}^\top(1:\tau)\mathbf{Q}_l \Phi_{lj}(1:\tau). \qquad i,j \in \mathcal{V} \end{aligned} \quad (17)$$

Definition 4. The *undirected optimisation graph* $\mathcal{O}(\Pi, \Psi) = \{\mathcal{V}, \mathcal{E}_\mathcal{O}\}$ is such that $(i,j), (j,i) \in \mathcal{E}_\mathcal{O} \iff (\Pi_{ij} \neq 0) \vee (\Psi_{ij} \neq 0)$, where Π and Ψ are defined in (17). Moreover,

$$\mathcal{J}_i^k = \{A_{jl}, B_{jl}, Q_j, \xi_j, \mathbf{u}_j^k\}_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{N}_j^-(\mathcal{I},1)}} \bigcup \{R_i, \mathcal{U}_i\}$$

where $\mathcal{M}_i = \{j : (i,j) \in \mathcal{E}_\mathcal{O}\}$.

By construction, at iteration $k > 0$, access to the model data in $\mathcal{J}_i^k$ is sufficient to completely determine $g_i(\mathbf{u}) = (1/2)\mathbf{u}^\top \Pi_i^+ \mathbf{u} + \mathbf{u}^\top \Psi_i^- \xi$ for arbitrary $\mathbf{u} \in \mathcal{U}$. Moreover, in view of the algorithm-state information also contained in $\mathcal{J}_i^k$, the mapping g_{ii} in (12) can be constructed, as required to implement the iteration (10) or (11) for node $i \in \mathcal{V}$. The next theorem establishes conditions on the network and interconnection graphs $\mathcal{G}$ and $\mathcal{I}$ that imply $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$, yielding an answer to Problem 1 in the generic case of both state and input coupling, for specific instances of the information limit ℓ. The proof is deferred to the appendix, with the proofs of all subsequent results in this section.

Theorem 1. *In the general case of state and control coupling, the following hold for all time-horizons $\tau > 0$, iterations $k > 0$ and nodes $i \in \mathcal{V}$:*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

1) *For* $\ell = 0$, $\mathcal{J}_i^k = \mathcal{K}_i^k(\ell)$ *if and only if* $\mathcal{E}_\mathcal{I} = \emptyset$;
2) *For* $\ell = \tau$, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$ *if* $d_\mathcal{G}(i,j) \leq \ell$ *for all* $i,j \in \mathcal{V}$ *such that* $\mathcal{W}_{ij}(\mathcal{I},\ell,\ell) \neq \emptyset$;
3) *For* $\ell \geq 2\tau$, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$.

Remark 5. In essence, a subsystem's contribution to the global cost can be computed locally, and hence locally minimised, provided it can calculate the states of all the subsystems influenced by its control $\mathbf{u}_i$ over the τ-length horizon. Under worst-case conditions $\mathbf{u}_i$ may affect the states of all subsystems $j \in \mathcal{N}_i^+(\mathcal{I},\tau)$. Hence, subsystem i must be able to reconstruct the states of all such subsystems. Similarly, under worst-case conditions, the state of a subsystem $j \in \mathcal{N}_i^+(\mathcal{I},\tau)$ may be influenced by subsystems $l \in \mathcal{N}_j^-(\mathcal{I},\tau)$. Thus, as $d_\mathcal{G}(i,l) \leq 2\tau$ for all l, the gathering of model data and algorithm state from subsystems $\mathcal{N}_i(\mathcal{G},2\tau)$ is sufficient for subsystem i to compute $\mathbf{u}_i$.

Remark 6. The second statement of Theorem 1 illustrates that iterates can be computed locally using τ-hop information when the interconnection graph is structured in a particular way. When there exists a pair (i,j) such that $\mathcal{W}_{ij}(\mathcal{I},\tau,\tau) \neq \emptyset$, the model data and algorithm-state of subsystem j must be collected by i. This is due to subsystems i and j both affecting the state of some common subsystem over the τ-length horizon. Hence, if all such pairs of (i,j) are also within τ-hops in $\mathcal{G}$ then i can compute $\mathbf{u}_i$ using information collected from just $\mathcal{N}_i(\mathcal{G},\tau)$.

B. Coupling Through States

Consider the case of state coupling only, i.e.

$$v_i(t) = \sum_{l \in \mathcal{N}_i^-(\mathcal{A},1)\setminus\{i\}} A_{il}x_l(t). \tag{18}$$

Theorem 2. *In the case of state coupling only, in that* $\mathcal{E}_\mathcal{B} = \emptyset$, *the following hold for all time-horizons* $\tau > 0$, *iterations* $k > 0$ *and nodes* $i \in \mathcal{V}$:

1) *For* $\ell = \tau$, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$ *if* $d_\mathcal{G}(i,j) \leq \ell$ *for all* $i,j \in \mathcal{V}$ *such that* $\mathcal{W}_{ij}(\mathcal{I},\ell-1,\ell) \neq \emptyset$.
2) *For* $\ell \geq 2\tau - 1$, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$.

Remark 7. When no control coupling exists, a subsystem's control $\mathbf{u}_i$ can affect the states of subsystems $j \in \mathcal{N}_i^+(\mathcal{I},\tau-1)$. Thus, similar to Remark 5, model data and algorithm-state collected from $\mathcal{N}_i^+(\mathcal{G},2\tau-1)$ is sufficient for i to compute $\mathbf{u}_i$.

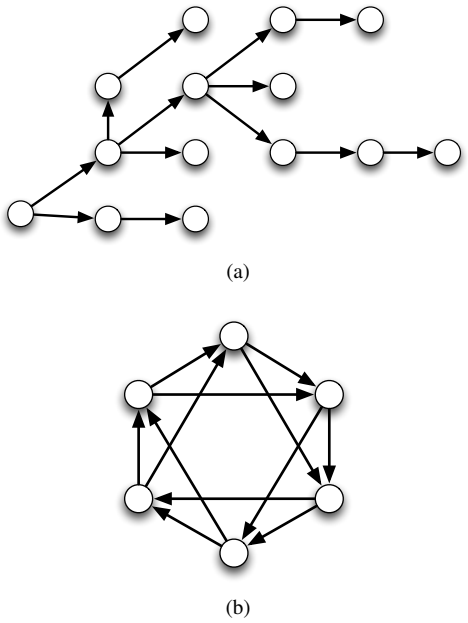
C. Coupling Through Controls

Consider the case of control input coupling only, i.e.

$$v_i(t) = \sum_{l \in \mathcal{N}_i^-(\mathcal{B},1)\setminus\{i\}} B_{il}u_l(t). \tag{19}$$

Theorem 3. *In the case of control input coupling only, in that* $\mathcal{E}_\mathcal{A} = \emptyset$, *the following hold for all time-horizons* $\tau > 0$, *iterations* $k > 0$ *and nodes* $i \in \mathcal{V}$:

1) *For* $\ell = 1$, $\mathcal{J}_i^k = \mathcal{K}_i^k(\ell)$ *if and only if* $d_\mathcal{G}(i,j) \leq \ell$ *for all* $i,j \in \mathcal{V}$ *such that* $\mathcal{W}_{ij}(\mathcal{I},\ell,\ell) \neq \emptyset$;
2) *For* $\ell \geq 2$, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$.



(a)

(b)

Figure 1: Examples of different interconnection graphs that satisfy the conditions of Theorem 3 for $\ell = 1$. Figure 1a is a rooted tree and Figure 1b is the square of a five node cycle graph.

Remark 8. When coupling is through control inputs only, a subsystem i can only affect the states of its immediate neighbours in the interconnection graph. Hence, in order to calculate the states of subsystems $j \in \mathcal{N}_i^+(\mathcal{I},1)$, it is sufficient to collect model-data and algorithm-state from the set $\mathcal{N}_i(\mathcal{G},2)$.

In the special case of a one-hop information limit (i.e. $\ell = 1$), the condition $\mathcal{J}_i^k = \mathcal{K}_i^k(1)$ is equivalent to $\mathcal{O}(\Pi,\Psi) = \mathcal{G}$. Graph structures for which this holds include:

1) $(\mathcal{V}, \mathcal{E}_\mathcal{I})$ is a power of a disjoint union of rooted trees.
2) $(\mathcal{V}, \mathcal{E}_\mathcal{I})$ is a power of a disjoint union of cycle graphs.
3) $(\mathcal{V}, \mathcal{E}_\mathcal{G})$ is a complete graph.

See Figure 1 for some examples.

The remainder of this section provides examples of how Theorem 3 can be applied. Specifically, Theorem 3 can be used to construct procedures that verify if a control coupling can be added to or removed from the network without compromising the ability for iterations to be implemented locally with an $\ell = 1$ network-hop limit on model and algorithm-state information. Such coupling additions and removals are characterised in Procedures 1 and 2. In particular, given a graph that already satisfies the $\ell = 1$ conditions, Procedure 1 checks if the addition of a new edge (i,j) maintains these conditions by verifying that all l such that $j \in \mathcal{W}_{il}(\mathcal{I},1,1) \neq \emptyset$ also satisfies $d_\mathcal{G}(i,l) \leq 1$. An example of the couplings that Procedure 1 would allow is depicted in Figure 2. Similarly, by repeatedly applying Procedures 1 and 2, it is also possible to devise a method for verifying if a new subsystem, i.e. vertex in the interconnection graph, can be connected to or removed from the network without compromising the $\ell = 1$ conditions.

Remark 9. Although Procedures 1 and 2 only verify the $\ell = 1$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

Procedure 1 Adding a new edge (i,j) to $\mathcal{I}$ that maintains Theorem 3 ($\ell=1$) conditions.

Require: $\mathcal{G}=\mathcal{O}(\Pi,\Psi)$, $\iota=$ **true** {subsystem iterates computable using one-hop information only.}
  for $l\in\mathcal{N}_j^-(\mathcal{I},1)\setminus\{j\}$ **do**
    if $(l\notin\mathcal{N}_i^-(\mathcal{I},1)\setminus\{i\}$ **and** $l\notin\mathcal{N}_i^+(\mathcal{I},1)\setminus\{i\})$ **then**
      $\iota\leftarrow$ **false**
      break
    end if
  end for
  if $\iota=$ **true then**
    $\mathcal{E}_\mathcal{I}\leftarrow\mathcal{E}_\mathcal{I}\bigcup\{(i,j)\}$ {edge (i,j) is added.}
  end if

Procedure 2 Removing an edge (i,j) from $\mathcal{I}$ that maintains Theorem 3 ($\ell=1$) conditions.

Require: $\mathcal{G}=\mathcal{O}(\Pi,\Psi)$, $\iota=$ **true** {subsystem iterates computable using one-hop information only.}
  for $l\in\mathcal{V}$ **do**
    if $(i\notin\mathcal{N}_l^-(\mathcal{I},1)\setminus\{l\}$ **and** $j\notin\mathcal{N}_l^-(\mathcal{I},1)\setminus\{l\})$ **then**
      $\iota\leftarrow$ **false**
      break
    end if
  end for
  if $\iota=$ **true then**
    $\mathcal{E}_\mathcal{I}\leftarrow\mathcal{E}_\mathcal{I}\setminus\{(i,j)\}$ {edge (i,j) is removed.}
  end if

network conditions in Theorem 3, alternative procedures for checking the conditions established in Theorem 1, 2 and 3 for any fixed ℓ-hop limit can be constructed in a similar manner.

IV. Conclusion

Model and iteration dependencies of three feasible-point algorithms for input constrained optimal control computation are investigated in this paper using graph-based characterisations of the network structure of the system and the structure of the optimisation problem. Results are established on the structure of network interconnection graphs such that computation of the iterates for each subsystem is possible while respecting specific ℓ-hop neighbourhood limits on the model and algorithm-state information available. In the special case of subsystem coupling via control inputs only, the conditions hold independently of the time-horizon of the optimal control problem, while more generally the proximity of information needed for computation grows with the time-horizon. More graphs structures for which one-hop limits are possible are identified in this special case.

Future research directions include consideration of the following: 1) state constraints; 2) non-linear dynamics; and 3) reduction of conservativeness by exploiting finer level structures than those captured by the network interconnection graph, such as structure arising from coupling delays.

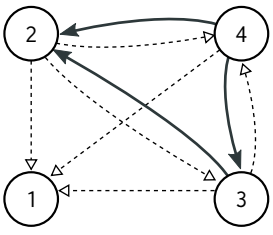


Figure 2: The interconnection graph $\mathcal{I}$ associated with the solid edges satisfies the $\ell=1$ conditions in Theorem 3. The dashed edges are new control couplings that could be added to $\mathcal{I}$ without compromising the $\ell=1$ conditions, as identified by Procedure 1 given $\mathcal{I}$. Once an edge has been added the procedure must be run again on the new interconnection graph.

Appendix

Proof of Lemma 1*:* To prove the contrapositive, suppose $\Phi_{ij}(k)\neq 0$. Since

$$\Phi_{ij}(k)=\sum_{l_{k-1}\in\mathcal{V}}\cdots\sum_{l_1\in\mathcal{V}}A_{il_{k-1}}A_{l_{k-1}l_{k-2}}\cdots A_{l_2l_1}A_{l_1j},\quad(20)$$

there exists at least one set of indices $S=(l_1,\ldots,l_{k-1})$ such that $A_{il_{k-1}}A_{l_{k-1}l_{k-2}}\cdots A_{l_2l_1}A_{l_1j}$ is a non-zero matrix. For this to be the case, each sub-matrix in the expression must be non-zero. From the definition of $\mathcal{A}$, this implies that $(j,l_1),(l_1,l_2),\ldots,(l_{k-1},i)\in\mathcal{E}_\mathcal{A}$. Therefore, $\omega=(j,(j,l_1),l_1,(l_1,l_2),l_2,\ldots,l_{k-1},(l_{k-1},i),i)$ is a walk in $\mathcal{A}$ that begins at vertex j, traverses k edges and terminates at vertex i. Hence, $\omega\in\mathcal{P}(k,j,i)$. ■

Proof of Theorem 1*:* Note, $\mathcal{J}_i^k=\mathcal{K}_i^k(\ell)\iff\mathcal{M}_i=\mathcal{N}_i(\mathcal{G},\ell)$ with $\mathcal{M}_i=\{j:(\Pi_{ij}\neq 0)\vee(\Psi_{ij}\neq 0)\}$. We now proceed by considering each case of ℓ.

($\Rightarrow|\ell=0$) Fix $\ell=0$ and suppose $\mathcal{I}=(\mathcal{V},\emptyset)$. This implies that $\mathcal{N}_i(\mathcal{G},\ell)=\{i\}$ for $i=1:n$, and since $\mathcal{I}=(\mathcal{V},\mathcal{E}_\mathcal{A}\cup\mathcal{E}_\mathcal{B})$, $\mathcal{A}$ and $\mathcal{B}$ are edgeless graphs. Hence, $\mathcal{N}_i^+(\mathcal{B},1)=\{i\}$, $\mathcal{N}_j^+(\mathcal{B},1)=\{j\}$, $\mathcal{W}_{ij}(\mathcal{A},\tau-1,\tau-1)=\emptyset$ and $\mathcal{W}_{ij}(\mathcal{A},\tau-1,\tau)=\emptyset$ when $i\neq j$, which implies $\Pi_{ij}=\Psi_{ij}=0$, therefore $j\notin\mathcal{M}_i$ if $j\neq i$. As $\mathbf{R}_i$ is assumed to be non-zero, then Π_{ii} is always non-zero hence $i\in\mathcal{M}_i$, $i=1:n$. Therefore, $\mathcal{M}_i=\mathcal{N}_i(\mathcal{G},\ell)$ and equivalently $\mathcal{J}_i^k=\mathcal{K}_i^k(\ell)$.

($\Leftarrow|\ell=0$) Fix $\ell=0$ and suppose $\mathcal{M}_i=\mathcal{N}_i(\mathcal{G},\ell)$. $\ell=0$ implies $\mathcal{N}_i(\mathcal{G},0)=\{i\}=\mathcal{M}_i$. We wish to show that both $\mathcal{A}$ and $\mathcal{B}$ are edgeless. Now $\mathcal{M}_i=\{i\}$ implies that $\Psi_{ij}=\Pi_{ij}=0$ for all $i\neq j$. $\Psi_{ij}=0$ in conjunction with $i\in\mathcal{N}_i^+(\mathcal{B},1)$ implies that $\mathcal{W}_{ij}(\mathcal{A},\tau-1,\tau)=\emptyset$ for all $i\neq j$, and since it is always true that $i\in\mathcal{N}_i^-(\mathcal{A},\tau-1)$, the definition of $\mathcal{W}_{ij}(\mathcal{A},\tau-1,\tau)$ shows that $(\mathcal{W}_{ij}(\mathcal{A},\tau-1,\tau)=\emptyset)\wedge(i\in\mathcal{N}_i^-(\mathcal{A},\tau-1))\implies j\notin\mathcal{N}_i^-(\mathcal{A},\tau)$. If j is not a τ-hop in-neighbour of i, then $j\notin\mathcal{N}_i^-(\mathcal{A},1)$ hence, no two distinct nodes in $\mathcal{A}$ are neighbours, therefore $\mathcal{A}$ is edgeless. We now show that $\mathcal{B}$ is edgeless by contradiction. Suppose a $j\neq i$ exists such that $j\in\mathcal{N}_i^+(\mathcal{B},1)$, since it's always true that $j\in\mathcal{N}_j^+(\mathcal{B},1)$ but by assumption $\Psi_{ij}=0$ we need $\mathcal{W}_{jj}(\mathcal{A},\tau,\tau-1)=\emptyset$, however, it is always true that $j\in\mathcal{W}_{jj}(\mathcal{A},\tau,\tau-1)$, implying that Ψ_{ij} is generically non-zero, i.e. a contradiction. Therefore, for all $j\neq i$, $j\notin\mathcal{N}_i^+(\mathcal{B},1)$, hence, no two distinct nodes are neighbours in graph $\mathcal{B}$, therefore $\mathcal{B}$ is edgeless. Ergo, $\mathcal{I}$ too is edgeless.

($\ell=\tau$) Suppose $\mathcal{I}$ satisfies the structure described by Theorem 1:2. Select an arbitrary $j\in\mathcal{M}_i$, from this we know

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

that either $\Pi_{ij} \neq 0$ or $\Psi_{ij} \neq 0$. Suppose $\Pi_{ij} \neq 0$, this implies that $\exists r, s, l$ such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (s \in \mathcal{N}_j^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{sr}(\mathcal{A}, \tau - 1, \tau - 1))$. Since $\mathcal{I}$ is the graph union of $\mathcal{A}$ and $\mathcal{B}$, the existence of r, s and l ensures that $l \in \mathcal{W}_{ij}(\mathcal{I}, \tau, \tau)$. As the structure imposed on $\mathcal{I}$ requires $d_{\mathcal{G}}(i, j) \leq \ell$ when $\mathcal{W}_{ij}(\mathcal{I}, \tau, \tau) \neq \emptyset$, it follows that $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Now suppose $\Psi_{ij} \neq 0$. Similar to $\Pi_{ij} \neq 0$ case, the existence of r and l such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{rj}(\mathcal{A}, \tau - 1, \tau))$ implies $\mathcal{W}_{ij}(\mathcal{I}, \tau, \tau) \neq \emptyset$, hence by the structure imposed on $\mathcal{I}$, $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Combining the two cases above we have shown that $\mathcal{M}_i \subseteq \mathcal{N}_i(\mathcal{G}, \ell)$.

($\ell \geq 2\tau$) Let $j \in \mathcal{M}_i$, again, this is equivalent to $\Pi_{ij} \neq 0$ or $\Psi_{ij} \neq 0$. From the $(\ell = \tau)$ proof above, we saw that $\Pi_{ij} \neq 0$ or $\Psi_{ij} \neq 0$ implies $\exists l$ such that $l \in \mathcal{W}_{ij}(\mathcal{I}, \tau, \tau)$. From the definition of $\mathcal{W}_{ij}(\mathcal{I}, \tau, \tau)$ we know that $d_{\mathcal{G}}(i, j) \leq d_{\mathcal{G}}(i, l) + d_{\mathcal{G}}(j, l) \leq \tau + \tau \leq \ell$, therefore $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. It immediately follows from $\mathcal{M}_i \subseteq \mathcal{N}_i(\mathcal{G}, \ell)$ that $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$. ■

Proof of Theorem 2*:* ($\ell = \tau$) Suppose $\mathcal{I}$ satisfies the structure described by Theorem 2:2. Select an arbitrary $j \in \mathcal{M}_i$, from this we know that either $\Pi_{ij} \neq 0$ or $\Psi_{ij} \neq 0$. Suppose $\Pi_{ij} \neq 0$, this implies that $\exists r, s, l$ such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (s \in \mathcal{N}_j^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{sr}(\mathcal{A}, \tau - 1, \tau - 1))$. Since $\mathcal{B}$ is edgeless, the existence of r, s and l ensures that $l \in \mathcal{W}_{ij}(\mathcal{I}, \tau-1, \tau-1)$. As the structure imposed on $\mathcal{I}$ requires $d_{\mathcal{G}}(i, j) \leq \ell$ when $\mathcal{W}_{ij}(\mathcal{I}, \tau - 1, \tau) \neq \emptyset$, it follows that $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Now suppose $\Psi_{ij} \neq 0$. Similar to the $\Pi_{ij} \neq 0$ case, the existence of r and l such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{rj}(\mathcal{A}, \tau - 1, \tau))$ implies $\mathcal{W}_{ij}(\mathcal{I}, \tau - 1, \tau) \neq \emptyset$, hence by the structure imposed on $\mathcal{I}$, $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Combining the two cases above we have shown that $\mathcal{M}_i \subseteq \mathcal{N}_i(\mathcal{G}, \ell)$.

($\ell \geq 2\tau - 1$), Let $j \in \mathcal{M}_i$, this is equivalent to $\Pi_{ij} \neq 0$ or $\Psi \neq 0$. Suppose $\Pi_{ij} \neq 0$, this implies that $\exists r, s, l$ such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (s \in \mathcal{N}_j^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{sr}(\mathcal{A}, \tau - 1, \tau - 1))$. Since $\mathcal{B}$ is edgeless, $\mathcal{N}_i^+(\mathcal{B}, 1) = \{i\}$ and $\mathcal{N}_j^+(\mathcal{B}, 1) = \{j\}$, hence, $l \in \mathcal{W}_{ij}(\mathcal{I}, \tau - 1, \tau - 1)$. This implies that $d_{\mathcal{G}}(i, j) \leq 2\tau - 2 < \ell$, i.e. $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Now suppose $\Psi_{ij} \neq 0$, this implies the existence of r and l such that $(r \in \mathcal{N}_i^+(\mathcal{B}, 1)) \wedge (l \in \mathcal{W}_{jr}(\mathcal{A}, \tau, \tau - 1))$. Again, since $\mathcal{N}_i^+(\mathcal{B}, 1) = \{i\}$, this implies $l \in \mathcal{W}_{ij}(\mathcal{I}, \tau - 1, \tau)$, hence $d_{\mathcal{G}}(i, j) \leq 2\tau - 1 \leq \ell$ and $j \in \mathcal{N}_i(\mathcal{G}, \ell)$. Therefore, $\mathcal{J}_i^k \subseteq \mathcal{K}_i^k(\ell)$. ■

Proof of Theorem 3*:* First note that $\mathcal{I} = \mathcal{B}$ enforces that $\mathcal{W}_{ji}(\mathcal{A}, \tau - 1, \tau - 1) \subseteq \mathcal{W}_{ij}(\mathcal{A}, \tau - 1, \tau) = \emptyset$ if $i \neq j$, and $\mathcal{W}_{ji}(\mathcal{A}, \tau - 1, \tau - 1) \subseteq \mathcal{W}_{ij}(\mathcal{A}, \tau - 1, \tau) = \{i\}$ if $i = j$. Moreover, $\Pi_{ij} = 0 \iff \Psi_{ij} = 0$ since $\mathcal{W}_{ji}(\mathcal{I}, 1, 1) = \emptyset \iff j \notin \mathcal{N}_i^+(\mathcal{I}, 1)$, hence $\mathcal{M}_i = \{j : \Pi_{ij} \neq 0\}$.

($\Rightarrow$: $\ell = 1$) Fix $\ell = 1$, and suppose the interconnection graph $\mathcal{I}$ satisfies the structure described by Theorem 3:2. We now wish to show that this structure implies $\mathcal{M}_i = \mathcal{N}_i(\mathcal{G}, 1)$. Let, $j \in \mathcal{M}_i$. This implies $\Pi_{ij} \neq 0$ and from (17), since $\mathcal{A}$ is edgeless, this can only be true if and only if $\exists l \in \mathcal{W}_{ij}(\mathcal{I}, 1, 1)$. As the structure imposed on $\mathcal{I}$ requires $d_{\mathcal{G}}(i, j) \leq 1$, then $j \in \mathcal{N}_i(\mathcal{G}, 1)$; hence $\mathcal{M}_i \subseteq \mathcal{N}_i(\mathcal{G}, 1)$. Now, suppose $j \in \mathcal{N}_i(\mathcal{G}, 1)$, from the definition of $\mathcal{W}_{ij}(\mathcal{I}, 1, 1)$, this implies either $j \in \mathcal{W}_{ij}(\mathcal{I}, 1, 1)$ or $i \in \mathcal{W}_{ij}(\mathcal{I}, 1, 1)$, thus $\Pi_{ij} \neq 0$ and $j \in \mathcal{M}_i$. Therefore $\mathcal{M}_i = \mathcal{N}_i(\mathcal{G}, 1)$ for all $i \in \mathcal{V}$. This is equivalent to $\mathcal{G} = \mathcal{O}(\Pi, \Psi)$.

($\Leftarrow$: $\ell = 1$) Fix $\ell = 1$, and suppose the interconnection graph $\mathcal{I}$ has the following structure: there exists a pair (i, j) such that $\mathcal{W}_{ij}(\mathcal{I}, 1, 1) \neq \emptyset$ and $d_{\mathcal{G}}(i, j) > 1$. We wish to show that this implies $\mathcal{M}_i \neq \mathcal{N}_i(\mathcal{G}, 1)$. Fix i and j to be this pair. This implies $\Pi_{ij} \neq 0$ as $\exists l \in \mathcal{W}_{ij}(\mathcal{I}, 1, 1)$. However, $j \notin \mathcal{N}_i(\mathcal{G}, 1)$ as $d_{\mathcal{G}}(i, j) > 1$, therefore $\mathcal{M}_i \neq \mathcal{N}_i(\mathcal{G}, 1)$.

($\ell \geq 2$) Let $j \in \mathcal{M}_i$, this is equivalent to $\Pi_{ij} \neq 0$. From the $(\ell = 1)$ case, we saw that $\Pi_{ij} \neq 0$ implies $\exists l$ such that $l \in \mathcal{W}_{ij}(\mathcal{I}, 1, 1)$. From the definition of $\mathcal{W}_{ij}(\mathcal{I}, 1, 1)$ we know that $d_{\mathcal{G}}(i, j) \leq d_{\mathcal{G}}(i, l) + d_{\mathcal{G}}(j, l) \leq 1 + 1 \leq 2$, therefore $j \in \mathcal{N}_i(\mathcal{G}, 1)$; hence, $\mathcal{M}_i \subseteq \mathcal{N}_i(\mathcal{G}, 1)$. ■

REFERENCES

[1] F. Deroo, M. Ulbrich, S. Hirche, and B. D. O. Anderson, "Distributed controller design for a class of sparse singular systems with privacy constraints," in *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems (Necsys)*, 2013.

[2] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *Security & Privacy, IEEE*, vol. 7, no. 3, pp. 75–77, 2009.

[3] F. Farokhi, C. Langbort, and K. H. Johansson, "Optimal structured static state-feedback control design with limited model information for fully-actuated systems," *Automatica*, vol. 49, no. 2, pp. 326–337, 2013.

[4] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer, "Accelerated gradient methods and dual decomposition in distributed model predictive control," *Automatica*, vol. 49, no. 3, pp. 829–833, 2013.

[5] D. Bertsekas and J. Tsitsiklis, *Parallel and distributed computation:Numerical methods*. New York: Athena Scientific, 1997.

[6] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[7] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, "Fast model predictive control for urban road networks via milp," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 3, pp. 846–856, 2011.

[8] E. Weyer, "Control of irrigation channels," *Control Systems Technology, IEEE Transactions on*, vol. 16, no. 4, pp. 664–675, 2008.

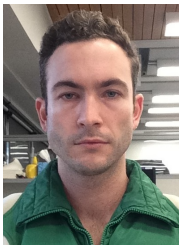
[9] I. Shames and M. Cantoni, "Structured computation of constrained quadratic optimal controls for heterogeneous networks," in *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems (Necsys)*, 2013.

[10] ——, "Constrained Linear Quadratic Control in Networks with Limited Model-Information Sharing," in *21st Int. Symp. on Mathematical Theory of Networks and Systems (MTNS)*, Groningen, The Netherlands, 2014.

[11] A. A. Goldstein, "Convex programming in hilbert space," *Bulletin of the American Mathematical Society*, vol. 70, no. 5, pp. 709–710, 1964.

[12] J. Liu, S. J. Wright, C. Ré, and V. Bittorf, "An asynchronous parallel stochastic coordinate descent algorithm," *arXiv preprint arXiv:1311.1873*, 2013.

[13] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, Aug. 2010. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0167691110000691



Mark A. Fabbro was born in Ayr, Australia. He received the B.E. degree in electrical and electronic engineering and the D.MathSc. diploma in applied mathematics from the University of Melbourne, Australia, in 2013, and is currently pursuing the Ph.D. degree in electrical engineering from the same institution. His research interests include mathematical systems theory, clock synchronisation in distributed systems, and sampled-data control systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2016.2590138, IEEE Transactions on Control of Network Systems

Iman Shames is currently a Senior Lecturer (Assistant Professor) at the Department of Electrical and Electronic Engineering, the University of Melbourne. Previously, he had been a McKenzie fellow at the same department from 2012 to 2014. Previously, he was an ACCESS Postdoctoral Researcher at the ACCESS Linnaeus Centre, the KTH Royal Institute of Technology, Stockholm, Sweden. He received his B.Sc. degree in Electrical Engineering from Shiraz University, Iran in 2006, and the Ph.D. degree in engineering and computer science from the Australian National University, Canberra, Australia in 2011. His current research interests include, but are not limited to, optimisation theory and numerical optimisation, mathematical systems theory, and security and privacy in cyber-physical systems.

Michael Cantoni was born in Australia. He received the BSc in Applied Mathematics and BE(Hons I) in Electrical Engineering from The University of Western Australia, Crawley, in 1995. He subsequently completed a PhD in the Department of Engineering at The University of Cambridge, UK, where postdoctoral positions were also held from 1998-2000. Since then he has been with The University of Melbourne, Australia, where he is now a full Professor in the Department of Electrical and Electronic Engineering. He is an Associate Editor for Automatica (2015-present), and he has served on the editorial boards of IET Control Theory and Applications (2007âĂŞ2010) and Systems & Control Letters (2007âĂŞ2014). He was co-recipient of the Control Systems Technology Award from the IEEE Control Systems Society in 2014. His research interests include mathematical systems theory, robust and optimal control, distributed control and optimization, and applications such as the automation of large-scale water distribution networks.