Distributed Nonlinear Control Design using Separable Control Contraction Metrics

Humberto Stein Shiromoto, *Member, IEEE,* Ian R. Manchester, *Member, IEEE,*

***Abstract*—This paper gives convex conditions for synthesis of a distributed control system control for a class of large-scale networked nonlinear dynamic systems. It is shown that the technique of control contraction metrics (CCMs) can be extended to this problem by utilizing *separable* metric structures, resulting in controllers that only depend on information from local sensors and communications from immediate neighbours. The conditions given are pointwise linear matrix inequalities, and are necessary and sufficient for linear positive systems. Distributed synthesis methods based on chordal graphs are also proposed. A simple example demonstrates feasibility for network of nonlinear dynamic systems with up to 512 nodes.**

***Index Terms*—Nonlinear Systems, Feedback Design, Contraction Theory, Distributed Control, Network Systems**

I. Introduction

In recent years, rapid advances in communication and computation technology have enabled the development of large-scale engineered systems such as smart grids [1], sensor networks [2], smart manufacturing plants [3], traffic networks [4] and pipe networks [5]. Despite these advances, the design of feedback controllers for such large systems remains challenging.

When it is assumed that a system has linear dynamics and that all sensor information can be collected in a single location for control computation, well-developed synthesis methods such as LQG and H^∞ can be applied [6], [7]. However, emerging applications motivate going beyond these assumptions:

i) For geographically distributed systems with thousands or millions of nodes, such as transportation and power networks, it is not practical to collect all sensor information in one location for control. In this case there is a need for *distributed* methods that rely only on local information or information communicated from nearby nodes.

ii) Most real systems exhibit *nonlinear* dynamics. The use of linear dynamic models is usually based on the premise that the system always operates near a prescribed setpoint. When large excursions in operating conditions are expected, e.g. due to changing production demands in a flexible manufacturing system, or recovery from a fault in a smart electrical grid, one must take into account the system nonlinearity.

Decentralised and distributed control are long-standing problems in control theory, with important early work surveyed in [8]. Terminology is not completely uniform in the literature, but in this paper we take “decentralised” to mean that at each node there is a controller that uses *only* local information, and “distributed” to mean that *some* communication is allowed between nearby nodes. In either case, there is a desire to impose a *structure* on information flow, and this turns out to be the main source of difficulty.

Even for linear systems, it has long been known that apparently simple problems with decentralized information flow can be surprisingly challenging [9]. A recent breakthrough was the discovery that a property called *quadratic invariance* characterises a convex subclass of problems in which, roughly speaking, communication is faster than the propagation of dynamical effects [10].

More directly related to this paper is a large body of work on linear matrix inequality (LMI) methods. For linear state feedback, information flow can be encoded by a sparsity structure on the feedback gain matrix, however in general the problem of finding such a gain which is stabilizing is NP-Hard [11]. It has been recognized by many authors that if the search is restricted to *diagonal* (or *block diagonal*) Lyapunov matrices, then the problem is convex (see, e.g., [12], [13], [14] and references therein). The main benefit is that sparsity structure in the gain matrix is preserved under the standard change of variables for LMI-based design.

In general, restricting the set of Lyapunov functions is conservative: it produces sufficient conditions for stabilizability, but not necessary. However, for the important sub-class of systems for which internal states are always non-negative, known as *positive systems*, existence of a diagonal Lyapunov function is actually necessary and sufficient (see, e.g., [15] and references therein). This result has been extended to H^∞ design [13], robust stability [16], and scalable algorithms for control design [14] and identification [17] of networked positive systems.

Design of controllers for nonlinear systems has also been a major topic of research for many years, see e.g. [18], [19], [20], [21] for established approaches. Most methods require (at least implicitly) the construction of a *control Lyapunov function*. While for certain structured systems, constructive methods such as backstepping and energy-based control can be used [18], [19], no general methodology exists. Indeed, the set of control Lyapunov functions can be non-convex and disconnected [22].

Another drawback of Lyapunov functions is the fact that they are defined with respect to a particular set point or trajectory, which must be known *a priori*. In many cases, e.g. robotics or flexible manufacturing, it is more appropriate to define a function depending on the distance *between* pairs of points. Tools such as contraction metrics [23] and incremental

Both authors are with the Australian Centre for Field Robotics, Department of Aerospace, Mechanical, and Mechatronic Engineering, Rose Street Building J04, The University of Sydney, NSW 2006, Australia. Corresponding author: h.stein.shiromoto@gmail.com, orcid.org/0000-0002-0883-0231

Lyapunov functions [24] provide such a capability for stability analysis, but the problem of going from analysis to synthesis remains.

An alternative approach is to construct a control contraction metric (CCM), introduced in [25], [26]. The main advantages this method offers over the Lyapunov approach are that the synthesis conditions are convex, and that provides a stabilizing controller for *all* solutions, not just a single set point. It was shown in [26] that the CCM conditions are necessary and sufficient for feedback-linearizable nonlinear systems.

At this point, the three contributions of this work can be highlighted. First, using the methodology for synthesis of CCMs introduced in [26] and imposing a (block-)diagonal structural constraint, a distributed feedback law is obtained. Second, because the CCM has a (block-)diagonal structure, the computation of the online integration phase, which corresponds to an optimization problem, is also distributed. Third, the structure imposed on the CCM allows to reduce the number of equations needed to verify stability. Note that the proposed method provides an explicit algorithm to design the controller. A particular case of this paper has also been published in [27].

a) Overview: The motivation and the problem formulation are provided in Section II-A, with technical background in Section II-B. Sections III and IV present the proposed method and main theoretical results. Illustrations of the proposed approach are provided in Section V. Section VI collects final remarks.

II. Preliminaries and Problem Formulation

a) Notation: Let $\mathbb{S}$ be a totally ordered set and $s_1, s_2, s_3 \in \mathbb{S}$. The notation $\mathbb{S}_{[s_1,s_2)}$ (resp. $\mathbb{S}_{\diamond s_3}$) stands for the set $\{s \in \mathbb{S} : s_1 \leq s < s_2\}$ (resp. $\{s \in \mathbb{S} : s \diamond s_3\}$, where $\diamond$ is a comparison operator, i.e., $\diamond \in \{<, \geq, =, \text{ etc}\}$). Let $n > 0$ be any integer, the vector e_i denotes the vector with zeros in all rows except the i-th where it is 1. A matrix $M \in \mathbb{R}^{n\times n}$ with zero elements except (possibly) those $m_{ii}, \dots, m_{nn}$ on the diagonal is denoted as $\mathtt{diag}(m_{ii}, \dots, m_{nn})$. The notation $M \succ 0$ (resp. $M \succeq 0$) stands for M being positive (resp. semi)definite such a class of matrices is denoted as $\mathbf{S}^n_{\star 0} = \{M \in \mathbb{R}^{n\times n} : M \star 0, M = M^\top\}$, where $\star \in \{\succ, \succeq, \prec, \preceq\}$.

The notation $\mathcal{L}^\infty_{\text{loc}}(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ stands for the class of functions $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ that are locally essentially bounded. Given differentiable functions $M : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ and $f : \mathbb{R}^n \to \mathbb{R}^n$ the notation $\partial_f M$ stands for matrix with dimension $n \times n$ and with (i,j) element given by $\frac{\partial m_{ij}}{\partial x}(x) f(x)$. The notation $\dot{f}$ stands for the total derivative of f.

Let $N > 0$ be an integer, a *graph* consists of a set of *nodes* $\mathscr{V} \subset \mathbb{N}_{[1,N]}$ and a set of *edges* $\mathscr{E} \subset \mathscr{V} \times \mathscr{V}$ and it is denoted by the pair $(\mathscr{V}, \mathscr{E}) = \mathscr{G}$. A node $i \in \mathscr{V}$ is said to be *adjacent* to a node $j \in \mathscr{V}$ if $(i,j) \in \mathscr{E}$, the set of nodes that are adjacent to j is defined as $\mathscr{N}(j) = \{i \in \mathscr{V} : i \neq j, (i,j) \in \mathscr{E}\}$. A graph is said to be *undirected* if, for every edge $(i,j) \in \mathscr{E}$, there exists $(j,i) \in \mathscr{E}$. It is said to be *directed* if otherwise. Given two nodes $i, j \in \mathscr{V}$, an ordered sequence of edges $\{(k, k+1)\}_{k=i}^{j-1}$ is said to be a *path from node i to the node j*. A path is said to be *cycle* if node i equals node j, no edges are repeated, and the nodes i and $j-1$ are distinct. A graph is said to be *strongly connected* if, for every two nodes $i, j \in \mathscr{V}$, there exists a path connecting them. It is also said to be *complete* is every node is adjacent to any other node, and it is *incomplete* otherwise. A graph is said to be a *tree* if it is connected and does not contain cycles. A *leaf* is a node that is adjacent to only one node. The following concepts are recalled from [28], [29], [30]. A *clique* $\mathscr{C} \subset \mathscr{V}$ of the graph $\mathscr{G}$ is a maximal set of nodes that induces a complete subgraph on $\mathscr{G}$. A *chord* of a path is any edge joining two nonconsecutive nodes. A graph is said to be *chordal* if every cycle of length greater than three has a chord. The importance of a graph being chordal is that it has a tree-decomposition into cliques [31, Proposition 12.3.11] such a tree is said to be a *clique tree* and it is denoted as $\mathscr{T}(\mathscr{G})$.

A. Problem Formulation and Motivation

Consider the class of systems described by the differential equation

$$\dot{x}(t) = f(x(t)) + B(x(t))u(t) \;, \tag{1}$$

where, for every positive times t, the *system state* $x(t)$ and the *system input variable* $u(t)$ evolve in the Euclidean spaces $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. The functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $B : \mathbb{R}^n \to \mathbb{R}^m$ are assumed to be smooth, i.e., infinitely differentiable. From now on the dependence on the time t will be omitted.

A function $u^* \in \mathcal{L}^\infty_{\text{loc}}(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ is said to be an *input signal or control for* (1). For such a control for (1), and for every *initial condition* x^*, there exists a unique solution to (1) ([32]) that is denoted by $X(t, x^*, u^*)$, when computed at time t. This solution is defined over an open interval $(\underline{t}, \bar{t})$, and it is said to be *forward complete* if $\bar{t} = +\infty$.

A forward complete solution $X^*(\cdot, x^*, u^*)$ to (1) is said to be *globally exponentially uniformly stabilizable* with *rate* $\lambda > 0$ if there exist a constant value $C > 0$ and a feedback law $k^* : \mathbb{R}^n \times \{X^*\} \times \{u^*\} \to \mathbb{R}^m$, denoted as k^*, such that the inequality

$$|X^*(t, x^*, u^*) - X(t, x, k^*)| \leq C e^{-\lambda t} |x^* - x| \tag{2}$$

holds, for every $t \geq 0$, and for every initial condition $x \in \mathbb{R}^n$ ([25]). Equivalently, if inequality (2) holds, then the solution $X^*(\cdot, x^*, u^*)$ to (1) is said to be *globally exponentially uniformly stable* for system (1) in closed loop with k^*

A stronger condition than the global exponential stabilizability of a particular solution is the requirement for every forward complete solution to be globally exponentially stabilizable. This concept is formalized in the following definition recalled from [25].

Definition 1. The system (1) is said to be *universally stabilizable* with rate λ if, for every forward complete solution $X^*(\cdot, x^*, u^*)$ to (1), there exists a static feedback law k^* for system (2) rendering $X^*(\cdot, x^*, u^*)$ globally exponentially uniformly stable with rate λ.

Let $N > 0$ be an integer and consider the graph $\mathscr{G}_p$ defined by the set of vertices $\mathscr{V}_p = \mathbb{N}_{[1,N]}$ and a set of edges $\mathscr{E}_p \subset \mathbb{N}_{[1,N]} \times \mathbb{N}_{[1,N]}$. For each index $i \in \mathbb{N}_{[1,N]}$, system (1) can be decomposed into smaller components as follows.

$$\dot{x}_i = f_i(x_i, \breve{x}_i) + b_i(x_i)u_i \;, \tag{3}$$

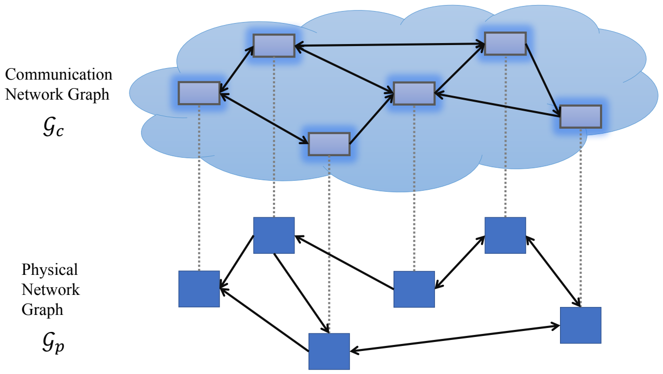

Figure 1. Illustration of the directed graphs representing the physical interaction between nodes $\mathscr{G}_p$, and the communication network $\mathscr{G}_c$. These may or may not be identical.

where $\breve{x}_i \in \mathbb{R}^{\breve{n}_i}$ stands for the vector composed of states x_j of systems that are adjacent to system (3). Given a graph $\mathscr{G}_c = (\mathbb{N}_{[1,N]}, \mathscr{E}_c)$ that specifies a communication network. One of the objectives of this work is to design a feedback law $k_i^* = k_i(t, x_i, \vec{x}_i, X_i^*, \vec{X}_i^*, u_i^*)$ for (3) under the constraints defined by $\mathscr{G}_c$, where $\vec{x}_i \in \mathbb{R}^{\vec{n}_i}$ stands for the vector composed of states $x_j \in \mathbb{R}^{n_j}$, that are adjacent to system i, according to the set of edges $\mathscr{E}_c$. The difference between both graphs is illustrated in Figure 1.

At this point, the problem under consideration in this paper can be formulated as follows.

Problem 1. For any forward complete solution $X^*(\cdot, x^*, u^*)$, find a feedback law k^* such that, for every initial condition $x \in \mathbb{R}^n$, the issuing solution to (1) in closed loop satisfies the inequality (2). Moreover, for each index $i \in \mathbb{N}_{[1,N]}$, the i-th component k^* depends on specific components of x described by the graph $\mathscr{G}_c$.

In other words, Problem 1 states that each component of the controller depends on the states prescribed by the graph $\mathscr{G}_c$.

The constraint on the components of k^* described in Problem 1 is particularly relevant for the design of distributed feedback laws with a prescribed topology (described possibly by $\mathscr{G}_c$). Consider the network composed of systems described by the following equation.

$$\left\{ \begin{array}{rcl} \dot{x}_i & = & -x_i - x_i^3 + y_i^2 + 0.01\left(x_{i-1}^3 - 2x_i^3 + x_{i+1}^3\right) \\ \dot{y}_i & = & u_i \; . \end{array} \right. \tag{4}$$

As explained in Section V, when more than eight of these systems are connected, the CCM approach was unable to provide unconstrained controller on a standard desktop machine, due to memory constraints. However, by utilizing a separable structure (see definition below), up to 512 of theses systems can be connected and the distributed controller can be computed.

B. Control Contraction Metrics

A *Riemannian metric on* $\mathbb{R}^n$ is a symmetric positive-definite bilinear form that depends smoothly on $x \in \mathbb{R}^n$. In a particular coordinate system, for any pair of vectors δ_0, δ_1 of $\mathbb{R}^n$ the *metric* is defined as the inner product $\langle \delta_0, \delta_1 \rangle_x = \delta_0^\top M(x) \delta_1$, where $M : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ is a smooth function. Consequently, local notions of norm $|\delta_0|_x^2 = \langle \delta_0, \delta_0 \rangle_x =: V(x, \delta_0)$ and orthogonality $\langle \delta_0, \delta_1 \rangle_x = 0$ can be defined. The metric is said to be *bounded* if there exists constant values $\underline{m} > 0$ and $\overline{m} > 0$ such that, for every $x \in \mathbb{R}^n$, $\underline{m} I_n \leq M(x) \leq \overline{m} I_n$, where $I_n \in \mathbb{R}^{n \times n}$ is the identity matrix. From now, metric stands for a bounded Riemannian metric.

Let $\Gamma(x_0, x_1)$ be the set of piecewise-smooth curves $c : [0, 1] \to \mathbb{R}^n$ connecting $x_0 = c(0)$ to $x_1 = c(1)$. The *length* and *energy* of c are, respectively, defined by the values

$$\ell(\gamma) = \int_0^1 |c_s(s)|_{c(s)} \, ds$$
$$e(\gamma) = \int_0^1 V(c(s), c_s(s)) \, ds \; ,$$

where the notation c_s stands for the derivative $\frac{\partial c}{\partial s}$. The Riemannian distance between x_0 and x_1, denoted as $\mathtt{dist}(x_0, x_1)$, is defined as the curve with the smallest length connecting them. This curve is said to be a *geodesic* and it is the solution to the optimization problem

$$\mathtt{dist}(x_0, x_1) = \inf_{c \in \Gamma(x_0, x_1)} \ell(c) \; . \tag{5}$$

A suitable notion to deal with exponential convergence of pair of solutions to (1) is provided by the *differential* (also known as variational or prolonged) dynamical system

$$\dot{\delta}_x = A(x, u)\delta_x + B(x)\delta_u \; , \tag{6}$$

where δ_x (resp. δ_u) is a vector of the Euclidean space $\mathbb{R}^n$ (resp. $\mathbb{R}^m$). More precisely, it is the vector tangent to a piecewise smooth curve connecting a pair of points in $\mathbb{R}^n$ (resp. $\mathbb{R}^m$). The matrix $A \in \mathbb{R}^{n \times n}$ has components given by

$$A_{jk}(x, u) = \frac{\partial}{\partial x_k} \left[f_j + \sum_{i=1}^{m_j} B_{ji} u_i \right]$$

for indexes $j, k \in \mathbb{N}_{[1,n]}$.

The resulting system composed of Equations (1) and (6) is analyzed on the state space spanned by the vector $(x, \delta_x) \in \mathbb{R}^n \times \mathbb{R}^n$.

Similarly to (1), given a control δ_u for system (6), the solution to (6) computed at time $t \geq 0$, along signals $(x(t), u(t)) \in \mathbb{R}^n \times \mathbb{R}^n$ and issuing from the initial condition $\delta_x \in \mathbb{R}^n$ is denoted by $\Delta_x(t, x, \delta_x, u, \delta_u)$.

Lyapunov stability notions of solutions to (6) are similar to those of linear time-varying systems (LTVS) (see [33] for more information on LTVS).

The importance of the stability of (6) for system (1) can be understood as follows. Given a particular control signal $u(\cdot)$ for system (1), $\delta_u \equiv 0$. If the Euclidean norm of every solution $\Delta_x(t, x, \delta_x, u, 0)$ converges to zero exponentially, as the time tends to infinity, then every pair of solutions to (1) under the input u converge to each other exponentially. The interested reader may address [23], [34] and references therein for further details.

A sufficient condition for the stability of (6) is provided by analyzing the derivative of a particular function along the solutions of systems (1) and (6). This function is recalled from [35] and [25].

Definition 2. A metric $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ receives the adjective *contraction* if, for a fixed control u for system (1), there exists a value $\lambda > 0$ such that the inequality

$$\tfrac{dV}{dt}(X(t), \Delta(t)) \leq -\lambda V(X(t), \Delta(t)) \tag{7}$$

holds, where $X(t) := X(t, x, u)$ and $\Delta(t) := \Delta(t, x, \delta_x, u, 0)$, for every pair $(x, \delta_x) \in \mathbb{R}^n \times \mathbb{R}^n$.

The existence a contraction metric for system (1) implies that every two solutions to this system converge to each other exponentially. The proof of this claim, for the autonomous case, can be found in [36, Theorem 1], and [37, Theorems 5.7 and 5.33], and [38, Lemma 3.3].

For the class of systems considered in this paper, the following kind of metric is of interest, since it also allows the design a feedback law for system (6).

Definition 3 ([25])**.** A metric for system (1) is said to be a *control contraction metric for system* (1) if there exists a constant value $\lambda > 0$ such that the condition

$$\delta_x \neq 0, \quad \delta_x^\top M(x) B(x) = 0 \tag{8a}$$

implies that the inequality

$$\delta_x^\top(\dot{M} + A^\top M + MA)\delta_x < \ -2\lambda \delta_x^\top M \delta_x \tag{8b}$$

holds, where $\dot{M} := \partial_{f+Bu} M$.

The set of equations (8) is an adaptation of Artstein-Sontag's condition to the differential system (6). Given a control contraction metric for system (1), Finsler's lemma (cf. [39, Lemma 11.1]) provides a stabilizing feedback law of the form $\delta_u = K(x)\delta_x$ for system (6), defined for every $(x, \delta_x, u) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m$, where $K : \mathbb{R}^n \to \mathbb{R}^{n \times m}$.

The following result is recalled from [25] and provides a feedback law for system (1) given a feedback law for system (6).

Theorem 1. *If there exists a control contraction metric for system* (1)*, then there exists a differential feedback law for system* (6) *rendering the equilibrium of the origin globally exponentially stable for system* (6) *in closed loop. Moreover, a stabilizing feedback law for system* (1) *is obtained by integrating the differential feedback law for system* (6) *along geodesics.*

As remarked in [25], one of the main advantages to look for control contraction metric over control-Lyapunov functions is that the former case can be formulate in terms of a convex optimization problem while the latter is non-convex [22]. The steps to obtain a control to system (1) are recalled below.

Step 1 (Offline MI computation). Consider the change of variables $\eta = M\delta$ and define the matrix $W = M^{-1}$. The set of equations (8) is equivalent (cf. [39, Lemma 11.1]) to the existence of a bounded differentiable function $W : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ such that $W(\cdot) = W(\cdot)^\top \succ 0$ and a function $Y : \mathbb{R}^n \to \mathbb{R}^{m \times n}$ satisfying the following matrix inequality (MI)

$$-\dot{W} + AW + WA^\top + BY + (BY)^\top + 2\lambda W \prec 0 \tag{9}$$

uniformly over $(x, u) \in \mathbb{R}^n \times \mathbb{R}^m$. Note that (9) is linear on the matrix variables W and Y, for every $(x, u) \in \mathbb{R}^n \times \mathbb{R}^m$. Consequently, the MI (9) can be solved with positive semidefinite programming techniques.

Once a solution to MI (9) has been computed, the function defined, for every $(x, \delta_x, u) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m$, by

$$\delta_k = Y(x) W^{-1}(x) \delta_x := K(x)\delta_x \tag{10}$$

is a differential feedback law that renders the origin globally exponentially stable for system (6) in closed loop with δ_k.

Step 2 (Online controller integration). The feedback law for system (1) can be obtained by integration as follows. Let $u^* : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ be a control for system (1), from Hopf-Rinow theorem (cf. [40, Theorem 7.7]), for every pair of solutions $X(\cdot, x, u)$ and $X^*(\cdot, x^*, u^*) \in \mathbb{R}^n$, and at each time instant $t \geq 0$, there exist a smooth geodesic curve $\gamma : \{t\} \times [0, 1] \to \mathbb{R}^n$ connecting them. Then the differential feedback law $K(x)$ can be integrated along γ:

$$k(t) = u^*(t) + \int_0^1 K(\gamma(t, s))\gamma_s(t, s)\, ds\ , \tag{11}$$

where $s \in [0, 1]$, is a feedback law rendering any forward complete solution X^* to system (1) globally exponentially stable.

Note about existence a.e.

Note about constant metric/constant Y case.

From the above steps, even if the problem of imposing a particular structure on K to correspond to the constraints of Problem 1 was tractable, the integration of the controller would not necessarily satisfy these constraints. This is due to the fact that the solutions to the optimization problem (5) cannot generally be computed in a distributed manner. In this paper, these limitations are addressed by imposing a separable structure over W.

Based on [25], the energy function also provides a control-Lyapunov function for system (1) that can be employed to design a distributed controller for system (1). More precisely, from the first variation of the energy function, the time derivative of e yields the equation

$$\frac{1}{2} De^+(X^*, X) = \langle \gamma_s(t, 0), \dot{X}^* \rangle_{X^*} - \langle \gamma_s(t, 1), f(x) \rangle_X \\ - \langle \gamma_s(t, 1), B(x)u \rangle_X \tag{12}$$

where De^+ stands for the Dini's upper right-hand time derivative of the energy e along the solutions X and X^*. A distributed controller can be obtained by finding the appropriate feedback law for system (1) rendering (12) negative definite. This can be achieved by noticing that (12) is the sum of N terms of the form

$$\tfrac{\partial \gamma_i}{\partial s_i}(t, 0) M_i(X_i^*) X_i^* - \tfrac{\partial \gamma_i}{\partial s_i}(t, 1) M_i(X_i) f_i(X_i, \breve{X}_i) \\ - \tfrac{\partial \gamma_i}{\partial s_i}(t, 1) M_i(X_i) B_i(X_i) u_i\ .$$

This was used in [41] to guarantee stability in distributed economic model predictive control.

III. Design of Distributed Controllers

Inspired by the notion of sum-separable Lyapunov functions (see e.g. [42]), we introduce the following class of control contraction metrics:

Definition 4. A control contraction metric V for system (1) is called *sum-separable* if it can be decomposed like so:

$$V(x,\delta_x) = \sum_{i=1}^{N} \delta_{x_i}^\top M_i(x_i)\delta_{x_i}\ ,$$

where, for each index $i \in \mathbb{N}_{[1,N]}$, and for every $(x_i,\delta_{x_i}) \in \mathbb{R}^{n_i} \times \mathbb{R}^{n_i}$, the function $V_i(x_i,\delta_{x_i}) := \delta_{x_i}^\top M_i(x_i)\delta_{x_i}$ is a metric on $\mathbb{R}^{n_i}$.

In other words, Definition 4 states that the metric V on $\mathbb{R}^n$ can be decomposed into a sum of smaller components, each of which depends only on the *local* information x_i, δ_{x_i}.

To address the constraint on the components of k^* described in Problem 1, the structure of the feedback defined by Equation (11) is obtained by imposing a suitable constraint on the function Y to be satisfied together with the MI (9).

Define Ξ as the set of functions $Y : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^{m\times n}$ with components defined by

$$Y_{ij} \begin{cases} = Y_{ij}(x_i,\vec{x}_i) \in \mathbb{R}^{m_i \times n_i}, & \text{if } (i,j) \in \mathscr{E}_c, \\ \equiv 0_{m_i \times n_i}, & \text{otherwise,} \end{cases}$$

for every $i,j \in \mathscr{V}_c$, and for every $(x_i,\vec{x}_i) \in \mathbb{R}^{n_i} \times \mathbb{R}^{\vec{n}_i}$.

The set Ξ defines the topology of the differential feedback law to be designed for system (6) and the dependence of each element of the matrix Y on the state-space variables. Note that, whenever system (1) is linear, the map Y_{pq} is constant. For each index $i \in \mathbb{N}_{[1,N]}$, define Π_i as the set of functions $W_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i \times n_i}$ such that, for each $j \in \mathbb{N}_{[1,N]} \setminus \{i\}$, $\frac{\partial W_i}{\partial x_j} \equiv 0$.

Theorem 2. *If there exists a solution to the matrix inequalities*

$$-\dot{W} + AW + WA^\top + BY + (BY)^\top + 2\lambda W \prec 0 \qquad \text{(13a)}$$

with $W = \mathtt{diag}(W_1,\ldots,W_N)$,

$$W_i \succ 0, W_i \in \Pi_i, \forall i \in \mathbb{N}_{[1,N]} \qquad \text{(13b)}$$

and

$$Y \in \Xi. \qquad \text{(13c)}$$

Then, W is a separable control contraction metric for system (1) and there exists a solution to Problem 1. Moreover, for each index $i \in \mathbb{N}_{[1,N]}$, each function W_i depends only on the state of system i. In particular, the graph $\mathscr{G}_c$ describing the communication network among different controllers specifies the set Ξ.

Proof of Theorem 2. By assumption, the functions $W : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ and $Y : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^{m\times n}$ satisfy the set of equations (13). Apply the coordinate change $\eta = M\delta$ and define the matrices $W = M^{-1}$ and $K = YW^{-1}$. Since W is separable, the sparsity pattern and local dependence of Y is preserved and, consequently, $K \in \Xi$.

The MI (13a) implies that the inequality

$$\delta_x^\top \left(\dot{M} + (A+BK)M + M(A+BK)^\top - 2\lambda M\right)\delta_x \le 0$$

holds, for every $(x,\delta_x,u) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m$. Consequently, the condition defined by the set of equations (8) holds. Thus, M is a sum-separable control contraction metric for system (1) and, according to Theorem 1, (10) is a differential feedback rendering the equilibrium of the origin globally exponentially stable for system (6) in closed loop.

It remains to integrate δ_k to obtain a feedback law for system (1) satisfying the constraints of Problem 1. Because M is a block-diagonal matrix, the energy of any curve $c : [0,1] \to \mathbb{R}^n$ satisfies the following equation

$$e(c) = \int_0^1 \sum_{i=1}^{N} V_i\left(c_i(s_i), \frac{\partial c_i}{\partial s_i}(s_i)\right) ds\ . \qquad (14)$$

Since M has a block-diagonal structure, it can be split into N separate optimization problems depending on different decision variables. Moreover, the positive definiteness of M implies also that the minimum of Equation (14) corresponds to a minimum of each component $i \in \mathbb{N}_{[1,N]}$. Moreover, for a geodesic curve γ, $e(\gamma) = \ell(\gamma)$. Consequently,

$$\gamma_i = \arg\min e(c_i), \forall i \in \mathbb{N}_{[1,N]} \ \Leftrightarrow\ \gamma = \arg\min \ell(c).$$

Thus, the minimization of Equation (14) can be computed in parallel.

From Hopf-Rinow theorem (cf. [40, Theorem 7.7]), for every index $i \in \mathbb{N}_{[1,N]}$, and for every x_i and $x_i^* \in \mathbb{R}^{n_i}$, there exists a solution to the optimization problem

$$\gamma_i = \arg \min_{c_i \in \Gamma(x_i^*, x_i)} e(c_i)\ .$$

This solution is a geodesic smooth curve $\gamma_i : [0,1] \to \mathbb{R}^{n_i}$ connecting x_i to $x_i^* \in \mathbb{R}^{n_i}$, because $e(\gamma) = \ell(\gamma)^2$.

From the definition of the tangent vectors δ_x and δ_u, given an input u^* for system (1), a feedback for system (1) is given by Equation (11).

Note that, due to the constraint $Y \in \Xi$, the independence of Y with respect to the input variable u and the fact that W is diagonal, each component $i \in \mathbb{N}_{[1,N]}$ of the feedback law k^*, defined by Equation (11), is given by

$$k_i^*(t,s) = u_i^*(t) + \int_0^s K_i(\widetilde{\gamma}_i(t,\sigma))\tfrac{\partial \widetilde{\gamma}_i}{\partial \sigma}(t,\sigma)\, d\sigma\ ,$$

where $\widetilde{\gamma}_i(\cdot) \in \mathbb{R}^{n_i} \times \mathbb{R}^{\vec{n}_i}$. Thus, k^* satisfies the constraint imposed in Problem 1.

To see how the graph $\mathscr{G}_c$ specifies the set Ξ, not that the set of edges $\mathscr{E}_c$ specifies the dependence of K_i on the variables x_i and $\vec{x}_i$. This concludes the proof of Theorem 2. □

Corollary 1 ([27])**.** *Assume that the matrix B satisfies the identity*

$$\partial_B W - \tfrac{\partial B}{\partial x} W - W \tfrac{\partial B}{\partial x}^\top \equiv 0$$

and there exist N functions $\rho_i : \mathbb{R}^{n_i + \vec{n}_i} \to \mathbb{R}$ such that the matrix inequality

$$-\dot{W} + \tfrac{\partial f}{\partial x} W + W \tfrac{\partial f}{\partial x}^\top - BRB^\top - 2\lambda W \prec 0 \qquad (15)$$

holds uniformly with respect to $(x, u) \in \mathbb{R}^n \times \mathbb{R}^m$*, where* $R = \mathtt{diag}(\rho_1, \ldots, \rho_N)$*. Then,* W *is a sum-separable control contraction metric for system* (1) *and there exists a solution to Problem 1. Also, the graph* $\mathscr{G}_c$ *describing the communication network among different controllers specifies the functions* ρ_i*.*

To see that Corollary 1 is a particular case of Theorem 2, note that imposing Y to be $Y = -RB^\top/2$ on the set Ξ (cf. (13c)), the matrix inequality (13a) yields (15).

Although Theorem 2 provides a methodology to design distributed controllers, *a priori*, nothing has been imposed to compute the set of equations (13) in a distributed manner. The next section provides sufficient conditions to parallelize the search for a solution to the set of equations (13).

IV. Scalable Design of Distributed Controllers

For large scale systems, the dimension of the matrices W and Y can be a problem to satisfy the set of equations (13). This is due to the fact that semidefinite programming become less efficient when those matrices have high rank [43]. Thus, the computation of a solution to MI (13a) need to be scalable, in the sense that adding more agents to the network will make the time taken to compute the controller linearly. To that end chordal graphs are employed to decompose the network topology into smaller complete components.

Proposition 1. *Let* $\mathscr{G} := \mathscr{G}_p \cup \mathscr{G}_c$ *and* $l \in \mathbb{N}_{>0}$ *be the number of nodes of the clique tree* $\mathscr{T}(\mathscr{G})$*. Then, the MI* (13a) *can be decomposed into* l *matrix inequalities with smaller dimension. Furthermore, each of the matrix inequality correspond to a clique, and for each node* $i \in \mathscr{V}_p \cup \mathscr{V}_c$ *contained in it, the matrix inequality depends on* $x_i, \breve{x}_i$ *and* $\vec{x}_i$*. Finally, whenever each matrix inequality holds uniformly over its argument, the set of Equations* (13a) *is satisfied.*

Proof. Using the Algorithm 3.1 from [29], it is possible to decompose the graph $\mathscr{G}$ into the clique tree $\mathscr{T}(\mathscr{G})$. Let the integer $l > 0$ be the number of nodes of $\mathscr{T}(\mathscr{G})$. The remainder of this proof is based on Section II of [28].

Let the sets $\mathscr{C}_1, \ldots, \mathscr{C}_l$ be the nodes of $\mathscr{T}(\mathscr{G})$, and $\mathtt{card}_k$ be cardinality (number of elements) of the set $\mathscr{C}_k$, $k \in \mathbb{N}_{[1,l]}$. For each index $k \in \mathbb{N}_{[1,l]}$, define the matrix $E_k \in \mathbb{R}^{\mathtt{card}_k \times n}$ obtained from the $n \times n$ identity matrix with blocks of rows indexed by $\mathbb{N}_{[1,N]} \setminus \mathscr{C}_k$ removed.

Denote the left-hand side of the MI (13a) by T. The existence of l cliques implies that there exist matrices $F_k : \mathbb{R}^{\mathtt{card}_k} \to \mathbb{R}^{\mathtt{card}_k \times \mathtt{card}_k}$, where $k \in \mathbb{N}_{[1,l]}$, satisfying the equation

$$T = \sum_{k=1}^{p} E_k^\top F_k E_k \ .$$

Without loss of generality, assume that the node $i \in \mathscr{V}_c$ is contained in the clique $\mathscr{C}_k$. Whenever the following matrix inequality

$$F_k \prec 0, \quad \forall k \in \mathbb{N}_{[1,l]}, \tag{16}$$

holds uniformly, the matrix T is uniformly negative definite. Thus, the MI (13a) holds.

For each node $i \in \mathscr{V}_p \cup \mathscr{V}_c$ contained in the clique $\mathscr{C}_k$. The corresponding matrix F_k has arguments $x_i, \breve{x}_i$ and $\vec{x}_i$. In other words, F_k depends on how strongly the nodes of the of the system (defined by $\mathscr{G}_c$) and communication network (defined by $\mathscr{G}_p$) are connected among each other. This concludes the proof. □

Note that if a graph is not chordal, it is possible to make it chordal by adding edges to the graph. The obtained graph is said to be a *chordal embedded* and the above procedure can be applied for this graph.

A. Discussion

Although the requirement of a sum-separable structure seems to be restrictive, this structure is equivalent to the (possibly local) stability of the origin, for some classes of systems.

For monotone systems (i.e., systems for which the some order of initial conditions is preserved along the flow of the vector field) described by a vector field that is continuously differentiable and whose gradient matrix is Hurwitz at the origin, there exists a sum-separable Lyapunov function in a neighborhood of the origin [42, Theorem 3.4].

For linear time-invariant systems with the property that solutions starting in the positive orthant remain in the positive orthant (also known as positive systems), the stability of the origin is equivalent to the existence of a quadratic Lyapunov function described by a matrix $P = P^\top \succ 0$ with diagonal structure [15]. Thus, locally (around the origin) the assumption of a diagonal structure is not restrictive.

Based on these paragraphs, the question of how restrictive is the requirement of M to have a block-diagonal structure remains open. As mentioned before, the main advantage to look for control contraction metric with a diagonal structure is that it allows to exploit sparsity to design controllers for large scale systems. This is illustrated in the next section.

V. Numerical Experiments

A case where structured controller design is relevant is on distributed design for network systems. For a given integer $N > 0$, consider the system that describes the dynamics of each agent

$$\left\{ \begin{array}{rcl} \dot{x}_i & = & -x_i - x_i^3 + y_i^2 + 0.01\left(x_{i-1}^3 - 2x_i^3 + x_{i+1}^3\right) \\ \dot{y}_i & = & u_i \ , \end{array} \right. \tag{4}$$

where $i \in \mathbb{N}_{[1,N]}$ and for convenience $x_0 = x_1$ and $x_N = x_{N+1}$.

For each index $i \in \mathbb{N}_{[1,N]}$, define the vectors $q_i = (x_i, y_i)$, $\breve{q}_i = (x_{i-1}, x_{i+1})$ and let $q = (q_1, \ldots, q_N)$. Denote also

$$f_i(q_i, \breve{q}_i) = \begin{bmatrix} -x_i - x_i^3 + y_i^2 + 0.01\left(x_{i-1}^3 - 2x_i^3 + x_{i+1}^3\right) \\ 0 \end{bmatrix}$$
$$B_i = \begin{bmatrix} 0, 1 \end{bmatrix}^\top .$$

Note that system (4) is not feedback linearizable in the sense of [44], because the vector fields

$$B = \mathtt{diag}\,(B_1, \ldots, B_N),$$
$$\frac{\partial f}{\partial q} B - \frac{\partial B}{\partial q} f = \mathtt{diag}\left(\begin{bmatrix} 2y_1 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 2y_N \\ 0 \end{bmatrix} \right)$$

are not linearly independent when $y_1 = \cdots = y_N = 0$. Furthermore, due to the quadratic term on y, the only possible action by the controller on the x-subsystem is to move the x-component of solution to (4) towards the positive semi-axis. In other words, the controller cannot reduce the value of the x-component.

In the remainder of this section, two network topologies are considered to illustrate the results presented in this paper. The numerical results were obtained employing the optimization parser Yalmip [45], [46] and the solver Mosek running on an Intel Core i7, 32GB RAM, Ubuntu and on Matlab 2015b

A. Chain topology

The graph describing the network structure is provided in Figure 2. Consider the case with $N = 4$ to compute the matrix W, chordal graphs are employed.

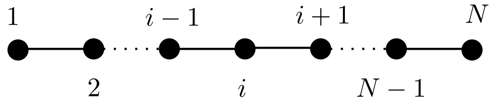

Figure 2. Graph describing the network composed of systems of the form (4).

When $N > 8$ and Y is a full-rank matrix with elements described by polynomials, the set of matrix inequalities (13) could not be solved on a standard desktop computer. A reason for this issue is the dependence of each element of the matrix Y on the variable q. Another reason is the need to verify if the matrix inequality (13a) is negative definite. For instance, when polynomials with degree d are employed, the number of monomials to be optimized in the (i, j) position of the matrix T is

$$\binom{n+d}{d}$$

[47]. Consequently, when no structural constraints are imposed on Y, this verification can lead to

$$n\frac{n+1}{2}\binom{n+d}{d}$$

monomials to be computed. This motivates the approach proposed in this paper.

To show the advantages of the method proposed in this paper, a benchmark composed of three scenarios, according to the constraints imposed on the matrix Y, has been created. Namely, the unconstrained case, the “neighbor” case, and the fully decentralized case. For the two latter cases, it was possible to consider up to $N = 512$ systems. In these scenarios the techniques provided by chordal graphs have been employed.

In the first one, the graph $\mathscr{G}_c$ describing the communication network is fully connected, i.e., $\mathscr{E}_c = \mathbb{N}_{[1,4]} \times \mathbb{N}_{[1,4]}$ and no constraints were imposed on the matrix Y. Consequently, for each $i \in \mathbb{N}_{[1,4]}$, for every $j \in \mathbb{N}_{[1,4]}$, $Y_{ij} = Y_{ij}(q_1, \ldots, q_4) \in \mathbb{R}^{1\times 2}$.

The solver took 31 seconds to compute a solution to the set of matrix inequalities (13). The matrix W is given by

$$W = \texttt{diag}(1, 1, 2, 1, 2, 1, 1, 1)$$

and the matrix Y has elements defined, for every $q \in \mathbb{R}^8$, by

$$\begin{aligned}
\overline{Y}_i(q) &= -\,2\begin{bmatrix} y_i & 1 + \sum_{k=1}^{4} y_k^2 + x_k^2 \end{bmatrix},\\
\hat{Y}_i(q) &= -\begin{bmatrix} y_i, & 0 \end{bmatrix},\\
0_{1\times 2} &= \begin{bmatrix} 0, & 0 \end{bmatrix},
\end{aligned}$$

where $i \in \mathbb{N}_{[1,4]}$, and structure given by the matrix

$$\begin{bmatrix}
\overline{Y}_1(q), & 0_{1\times 2}, & -0.03\hat{Y}_1(q), & 0_{1\times 2}\\
0_{1\times 2}, & \overline{Y}_2(q), & 0_{1\times 2}, & -0.04\hat{Y}_2(q)\\
-0.04\hat{Y}_3(q), & 0_{1\times 2}, & \overline{Y}_3(q), & 0_{1\times 2}\\
0_{1\times 2}, & -0.03\hat{Y}_4(q), & 0_{1\times 2}, & \overline{Y}_4(q)
\end{bmatrix}. \tag{17}$$

For the remaining cases, the structure of the network composed of systems of the form (4) has been exploited. More specifically, Proposition 1 can be employed, as the graph describing the interconnection of the network is a subclass of chordal graphs. Consequently, the MI (13a) can be decomposed into $N - 1$ cliques described by the matrix

$$F_k = \begin{bmatrix} F_{11} & F_{12}\\ F_{12}^{\top} & F_{22} \end{bmatrix}.$$

In the “neighbor” case, the communication network topology $\mathscr{G}_c$ has the same structure as the network system. Consequently, the set of edges $\mathscr{E}_c$ is defined as $\mathscr{E}_c = \mathscr{E}_p$.

The time taken by solver to find a solution to the set of matrix inequalities (13), in this case, was 3 seconds. The matrix W is given by

$$W = \texttt{diag}(2, 1, 1, 1, 1, 1, 2, 1)$$

and the matrix Y has the structure defined, over $q \in \mathbb{R}^8$ as

$$Y(q) = \begin{bmatrix}
\overline{Y}_1(\cdot) & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2}\\
0_{1\times 2} & \overline{Y}_2(\cdot) & 0_{1\times 2} & 0_{1\times 2}\\
0_{1\times 2} & 0_{1\times 2} & \overline{Y}_3(\cdot) & 0_{1\times 2}\\
0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & \overline{Y}_4(\cdot)
\end{bmatrix}, \tag{18}$$

with elements given, for every $(q_i, \breve{q}_i) \in \mathbb{R}^2 \times \mathbb{R}^{\breve{n}_i}$, by

$$\overline{Y}_i(q_i, \breve{q}_i) = -2\begin{bmatrix} y_i, & 1 + \sum_{j\in\mathscr{N}(i)} x_j^2 + y_j^2) \end{bmatrix}, i \in \mathbb{N}_{[1,4]}.$$

In the fully decentralized case, the set of edges $\mathscr{E}_c$ is defined as $\mathscr{E}_c = \{(1, 1), (2, 2), (3, 3), (4, 4)\}$. The time taken by the solver to find a solution to the set of matrix inequalities (13) was 2.5 seconds, the matrix W is given by

$$W = \texttt{diag}\,(2, 1, 1, 1, 1, 1, 1, 1),$$

and the matrix Y with structure given by (18) with elements defined, for every $q_i \in \mathbb{R}^2$, by

$$\overline{Y}_i(q_i) = -\begin{bmatrix} 2y_i, 3 + 2(x_i^2 + y_i^2) \end{bmatrix}, \quad i \in \mathbb{N}_{[1,4]}.$$

Figure 3 shows simulations of the network (4) composed of four systems with the network structure as represented by the graph in Figure 2, according to each constraint imposed on the controller. Figure 4 show the Riemannian metric V computed along solutions to network (4). Note that, when random initial random initial conditions are considered and the input is zero to be identically zero, the system may diverge from the neighborhood of the origin.

Figure 5 shows a plot of the time taken by parser to the set of matrix inequalities (13) for the three cases considered in this

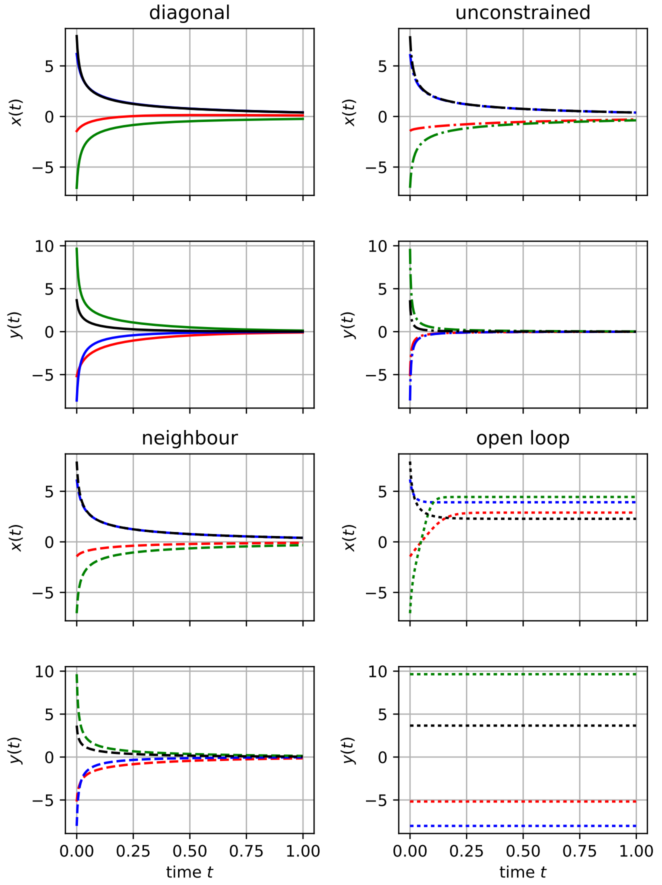

Figure 3. Simulation of network (4) with the target trajectory being the origin and under the controller obtained according to different constrains for the computation of Y: diagonal, neighbour and unconstrained. In contrast to these, the open loop simulation (performed with $u \equiv 0$) does not converge to the origin.

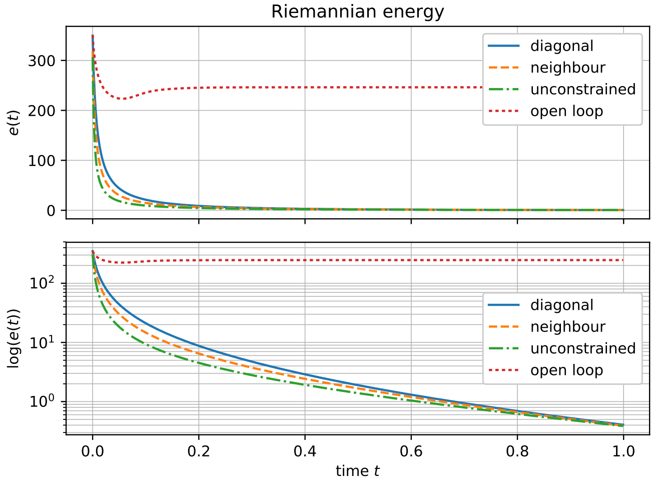

Figure 4. Computation of the Riemannian energy along solutions to (4), according to each of each case. The bottom graph is the logarithm of the Riemannian energy.

Case	Time taken by the solver (4 agents)
Unconstrained	31s
Neighbor	3s
Fully decentralized	2.5s

Table I
A SUMMARY OF THE TIME TAKEN TO SOLVE THE SET OF MATRIX INEQUALITIES (13) USING CHORDAL GRAPH DECOMPOSITION, IN THE TOPOLOGY ILLUSTRATED BY FIG. 2.

topology: unconstrained, "neighbor" and fully decentralized. According to this graph, for $N = 1, 2$, the time taken by the parser to solve the set of matrix inequalities (13) has the same order of magnitude for the three cases. However, as the number of systems increases, the unconstrained case takes more time to be solved than the "neighbor" case which, in turn, takes more time than the fully decentralized case. The time difference between the two latter cases can be explained by fact that, for the "neighbor" case, the matrix Y contains more non-identically zero elements than the fully decentralized case. In addition to this, in the former case, each element is defined as a polynomial of second degree on the variables q_i and $\breve{q}_i$, while for the latter, they are defined on the variable q_i only.

Table I shows a summary of the time taken by the solver to compute the matrices W and Y satisfying the set of matrix inequalities (13), according to the unrestricted, "neighbor" and fully decentralized cases.

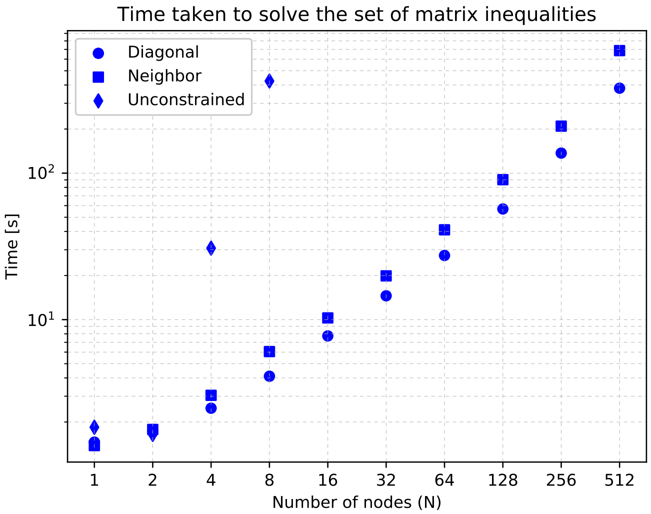

Figure 5. Time taken to solve the set of matrix inequalities (13) for the three cases under consideration. The number of agents started with one system and doubled until the maximum that the computer could process (8 for the unconstrained case, 512 for the other cases). Axes are on log scale.

Regarding the on-line integration phase, since the matrices W computed in each case are constant, for each index $i \in \mathbb{N}_{[1,4]}$, the geodesic curve connecting the origin to a point $q_i \in \mathbb{R}^2$ is a straight line: $\gamma_i(s) = sq_i$, where $s \in [0, 1]$. For each index $i \in \mathbb{N}_{[1,4]}$ and assuming that $u^* \equiv 0$, the feedback law is given by the formula

$$k = \int_0^1 Y(sq)W^{-1}q\,ds.$$

Given the structure of the matrices Y and W, each line $i \in \mathbb{N}_{[1,4]}$ is given by

$$k_i = \begin{cases} \int_0^1 Y_i(sq)W_i^{-1}q_i\, ds, & \text{if unconstrained} \\ \int_0^1 Y_i(sq_i, s\breve{q}_i)W_i^{-1}q_i\, ds, & \text{if “neighbor”} \\ \int_0^1 Y_i(sq_i)W_i^{-1}q_i\, ds, & \text{if fully decentralized.} \end{cases}$$

Moreover, from Theorem 2, the feedback law k solves Problem 1 for the network resulting from the interconnection of systems (4).

B. A topology with a cycle

The graph describing the network structure is provided in Figure 6. As in the Section V-A, here same three scenarios are considered. Namely, the unconstrained case, the “neighbor” case and the fully decentralized case.

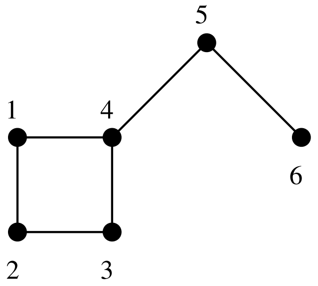

Figure 6. Graph describing the network composed of systems of the form (4).

In the first scenario, the unconstrained case, the graph $\mathscr{G}_c$ describing the communication network is fully connected, i.e., $\mathscr{E}_c = \mathbb{N}_{[1,6]} \times \mathbb{N}_{[1,6]}$ and no constrained were imposed on the matrix Y. Consequently, the matrix Y have elements given by $Y_{ij} = Y_{ij}(q_1, q_2, q_3, q_4)$, for each $i, j \in \mathbb{N}_{[1,4]}$, for each $(q_1, q_2, q_3, q_4) \in \mathbb{R}^8$.

The solver took 32s to solve the set of matrix inequalities (13). The matrix W obtained is given by

$$W = \texttt{diag}(0.1, 1, 0.1, 1, 0.1, 1, 0.1, 1, 0.1, 1, 0.3, 1).$$

The matrix Y is given by

$$\begin{bmatrix} \overline{Y}_{11}(q) & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} \\ 0_{1\times 2} & \underline{Y}_{22}(q) & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} \\ 0_{1\times 2} & 0_{1\times 2} & \overline{Y}_{33}(q) & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} \\ 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & \overline{Y}_{44}(q) & 0_{1\times 2} & 0_{1\times 2} \\ 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & \overline{Y}_{55}(q) & 0_{1\times 2} \\ 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & 0_{1\times 2} & \underline{Y}_{66}(q) \end{bmatrix},$$

for every $q \in \mathbb{R}^{12}$, where

$$\begin{aligned} \overline{Y}_{ij}(q) &= -\left[3y_i, \quad 4 + 4\textstyle\sum_{i=1}^{6}(x_i^2 + y_i^2)\right], \\ \underline{Y}_{ij}(q) &= -\left[3y_i, \quad 2 + 2\textstyle\sum_{i=1}^{6}(x_i^2 + y_i^2)\right], \end{aligned}$$

for each index $i, j \in \mathbb{N}_{[1,6]}$.

In the “neighbor” case, the topology of the controller is the same as of the network. More precisely, $\mathscr{G}_c = \mathscr{G}_p$. The solver took 17s to solve the set of matrix inequalities (13).

Case	Time taken by the solver (4 agents)
Unconstrained	32s
Neighbor	17s
Fully decentralized	12s

Table II
A SUMMARY OF THE TIME TAKEN TO SOLVE THE SET OF MATRIX INEQUALITIES (13) USING CHORDAL GRAPH DECOMPOSITION, IN THE TOPOLOGY ILLUSTRATED BY FIG. 6.

The matrix W obtained is given by

$$W = \texttt{diag}(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1).$$

The matrix Y is given by Equation (19).

In the last scenario, the fully decentralized case, the set of edges $\mathscr{E}_c$ of the graph $\mathscr{G}_c$ is specified as $\mathscr{E}_c = \{(1,1), (2,2), (3,3), (4,4)\}$ and no constrained were imposed on the matrix Y. The solver took 12s to solve the set of matrix inequalities (13).

The matrix W obtained is given by

$$W = \texttt{diag}(0.6, 1, 0.8, 1, 0.6, 1, 0.4, 1, 0.7, 1, 1, 1)$$

The matrix Y have elements given by $Y_{ij} = Y_{ij}(q_i)\delta_{ij} \in \mathbb{R}^{2\times 1}$, for each $i, j \in \mathbb{N}_{[1,6]}$, and for each $q_i \in \mathbb{R}^2$, where δ_{ij} is the Kronecker’s delta. More specifically, the matrix Y is presented in Equation (20).

VI. Conclusion

In terms of semidefinite programming, employing control contraction metrics allows one to formulate the search for the controller as a convex optimization problem. Although this is one of its main advantages over the search of control-Lyapunov functions, it cannot handle systems with a large number of states, in general.

For large-scale systems, the approach presented in this paper exploits sparsity to design feedback laws with a prescribed structure. More precisely, during the off-line phase, by imposing a block-diagonal structure on the matrix describing the metric, a controller with a prescribed structure can be synthesized. Moreover, the on-line integration phase computation can be in parallel, because each block defines a metric to the corresponding component of the state space.

References

[1] D. J. Hill, T. Liu, and G. Verbic, “Smart grids as distributed learning control,” in *2012 IEEE Power and Energy Society General Meeting*, pp. 1–8, July 2012.

[2] M. Pajic, S. Sundaram, G. J. Pappas, and R. Mangharam, “The wireless control network: A new approach for control over networks,” *IEEE Trans. Autom. Control*, vol. 56, pp. 2305–2318, oct 2011.

[3] S. Wang, J. Wan, D. Li, and C. Zhang, “Implementing Smart Factory of Industrie 4.0: An Outlook,” *International Journal of Distributed Sensor Networks*, vol. 12, p. 3159805, Jan. 2016.

[4] C. Canudas de Wit, F. Morbidi, L. Leon Ojeda, A. Y. Kibangou, I. Bellicot, and P. Bellemain, “Grenoble Traffic Lab: An Experimental Platform for Advanced Traffic Monitoring and Forecasting,” *IEEE Control Systems Magazine*, vol. 35, pp. 23–39, jun 2015.

[5] C. de Persis and C. S. Kallesoe, “Pressure regulation in nonlinear hydraulic networks by positive and quantized controls,” *IEEE Transactions on Control Systems Technology*, vol. 19, pp. 1371–1383, nov 2011.

[6] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, 1990. 02941.

$$-\begin{bmatrix} 2y_1 & 4+3(|q_1|^2+|\breve{q}_1|^2) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2y_2 & 2+2(|q_2|^2+|\breve{q}_2|^2) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2y_3 & 4+3(|q_3|^2+|\breve{q}_3|^2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2y_4 & 3+3(|q_4|^2+|\breve{q}_4|^2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2y_5 & 4+3(|q_5|^2+|\breve{q}_5|^2) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2y_6 & 4+3(|q_6|^2+|\breve{q}_6|^2) \end{bmatrix} \tag{19}$$

$$Y = -\begin{bmatrix} 2y_1 & 5+3|q_1|^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2y_2 & 3+2|q_2|^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2y_3 & 5+3|q_3|^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2y_4 & 4+2|q_4|^2 \end{bmatrix} \tag{20}$$

[7] G. Dullerud and F. Paganini, “A Course in Robust Control Theory: A Convex Approach,” 2000.

[8] N. Sandell, P. Varaiya, M. Athans, and M. Safonov, “Survey of decentralized control methods for large scale systems,” *IEEE Transactions on Automatic Control*, vol. 23, pp. 108–128, Apr. 1978.

[9] H. Witsenhausen, “A Counterexample in Stochastic Optimum Control,” *SIAM Journal on Control*, vol. 6, pp. 131–147, Feb. 1968.

[10] M. Rotkowitz and S. Lall, “A Characterization of Convex Problems in Decentralized Control,” *IEEE Transactions on Automatic Control*, vol. 51, pp. 274–286, Feb. 2006.

[11] V. Blondel and J. N. Tsitsiklis, “NP-Hardness of Some Linear Control Design Problems,” *SIAM J. Control Optim.*, vol. 35, pp. 2118–2127, nov 1997.

[12] A. Zecevic and D. D. Siljak, *Control of Complex Systems: Structural Constraints and Uncertainty*. Springer Science & Business Media, Jan. 2010.

[13] T. Tanaka and C. Langbort, “The bounded real lemma for internally positive systems and h-infinity structured static state feedback,” *IEEE Trans. Autom. Control*, vol. 56, pp. 2218–2223, sep 2011.

[14] A. Rantzer, “Scalable control of positive systems,” *European Journal of Control*, vol. 24, pp. 72–80, jul 2015.

[15] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*. SIAM, 1994.

[16] M. Colombino and R. S. Smith, “A Convex Characterization of Robust Stability for Positive and Positively Dominated Linear Systems,” *IEEE Transactions on Automatic Control*, vol. 61, pp. 1965–1971, July 2016.

[17] J. Umenberger and I. R. Manchester, “Scalable Identification of Positive Linear Systems,” in *Proceedings of the 55th IEEE Conference on Decision and Control*, (Las Vegas, NV), Dec. 2016.

[18] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1991.

[19] M. Krstić, I. Kanellakopoulos, and P. V. Kokotović, *Nonlinear and Adaptive Control Design*. Wiley, 1995.

[20] E. D. Sontag, *Mathematical Control Theory: Deterministic Finite Dimensional Systems*. No. 6 in Textbooks in Applied Mathematics, New York: Springer, 2nd ed., 1998.

[21] H. K. Khalil, *Nonlinear Systems*. Prentice Hall, 3rd ed., 2001.

[22] A. Rantzer, “A dual to Lyapunov’s stability theorem,” *Syst. & Contr. Lett.*, vol. 42, pp. 161–168, 2001.

[23] W. Lohmiller and J.-J. Slotine, “On contraction analysis for nonlinear systems,” *Automatica*, vol. 34, no. 6, pp. 683–696, 1998.

[24] D. Angeli, “A Lyapunov approach to incremental stability properties,” *IEEE Trans. Autom. Control*, vol. 47, pp. 410–421, Mar. 2002.

[25] I. R. Manchester and J.-J. E. Slotine, “Control contraction metrics and universal stabilizability,” in *Proceedings of the 19th IFAC World Congress*, vol. 19, (Cape Town, South Africa), pp. 8223–8228, Aug 2014.

[26] I. R. Manchester and J. J. E. Slotine, “Control Contraction Metrics: Convex and Intrinsic Criteria for Nonlinear Feedback Design,” *IEEE Transactions on Automatic Control*, vol. 62, pp. 3046–3053, June 2017.

[27] H. Stein Shiromoto and I. R. Manchester, “Decentralized nonlinear feedback design with separable control contraction metrics,” in *Proceedings of the 55th Conference on Decision and Control (CDC)*, (Las Vegas, NV, USA), pp. 5551–5556, Dec 2016.

[28] S. K. Pakazad, A. Hansson, M. S. Andersen, and A. Rantzer, “Distributed Semidefinite Programming with Application to Large-scale System Analysis (arxiv:1504.07755),” 2015.

[29] L. Vandenberghe and M. S. Andersen, “Chordal Graphs and Semidefinite Optimization,” *Foundations and Trends in Optimization*, vol. 1, no. 4, pp. 241–433, 2015.

[30] J. R. S. Blair and B. Peyton, “An Introduction to Chordal Graphs and Clique Trees,” in *Graph Theory and Sparse Matrix Computation*, pp. 1–29, 1993.

[31] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. Springer, 2005.

[32] G. Teschl, *Ordinary differential equations and dynamical systems*, vol. 1XX. American Mathematical Society, 2012.

[33] J. P. Hespanha, *Linear Systems Theory*. Princeton University Press, 2009.

[34] E. D. Sontag, “Contractive systems with inputs,” in *Perspectives in Mathematical System Theory, Control, and Signal Processing*, pp. 217–228, Springer, 2010.

[35] F. Forni and R. Sepulchre, “A Differential Lyapunov Framework for Contraction Analysis,” *IEEE Transactions on Automatic Control*, vol. 59, pp. 614–628, mar 2014.

[36] D. C. Lewis, “Differential equations referred to a variable metric,” *American Journal of Mathematics*, vol. 73, pp. 48–58, jan 1951.

[37] S. Reich and D. Shoikhet, *Nonlinear Semigroups, Fixed Points, and Geometry of Domains in Banach Spaces*. Imperial College Press, Jul 2005.

[38] G. Isac and S. Z. Nemeth, *Scalar and Asymptotic Scalar Derivatives: Theory and Applications*, vol. 13. Springer, 2008.

[39] G. Calafiore and L. El Ghaoui, *Optimization Models*. Cambridge University Press, 2014.

[40] W. M. Boothby, *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, 1986.

[41] R. Wang, I. R. Manchester, and J. Bao, “Distributed Economic MPC With Separable Control Contraction Metrics,” *IEEE Control Systems Letters*, vol. 1, pp. 104–109, July 2017.

[42] G. Dirr, H. Ito, A. Rantzer, and B. S. Rüffer, “Separable lyapunov functions for monotone systems: Constructions and limitations,” *Discrete and Continuous Dynamical Systems Series B (DCDS-B)*, vol. 20, pp. 2497–2526, aug 2015.

[43] L. Vandenberghe and S. Boyd, “Semidefinite Programming,” *SIAM Review*, vol. 38, pp. 49–95, mar 1996.

[44] A. Isidori, *Nonlinear Control Systems*. Communications and Control Engineering, Springer, 1995.

[45] J. Löfberg, “YALMIP : a toolbox for modeling and optimization in MATLAB,” in *Proceedings of the IEEE International Symposium on Computer Aided Control Systems Design*, (Taipei, Rep. of China), pp. 284–289, Sep 2004.

[46] J. Löfberg, “Pre- and post-processing sum-of-squares programs in practice,” *IEEE Trans. Autom. Control*, vol. 54, pp. 1007–1011, may 2009.

[47] H. Waki, S. Kim, M. Kojima, and M. Muramatsu, “Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity,” *SIAM J. Optim.*, vol. 17, pp. 218–242, Jan 2006.