2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

# Decomposition of Nonlinear Dynamical Networks via Comparison Systems

Abdullah Al Maruf
Washington State University
Pullman, WA 99164 USA
Email: abdullahal.maruf@wsu.edu

Soumya Kundu, Enoch Yeung
Pacific Northwest National Laboratory
Richland, WA 99354 USA
Email: {soumya.kundu,enoch.yeung}@pnnl.gov,

Marian Anghel
Los Alamos National Laboratory
Los Alamos, NM 87545 USA
Email: manghel@lanl.gov

***Abstract*—In analysis and control of large-scale nonlinear dynamical systems, a distributed approach is often an attractive option due to its computational tractability and usually low communication requirements. Success of the distributed control design relies on the separability of the network into weakly interacting subsystems such that minimal information exchange between subsystems is sufficient to achieve satisfactory control performance. While distributed analysis and control design for dynamical network have been well studied, decomposition of nonlinear networks into weakly interacting subsystems has not received as much attention. In this article we propose a vector Lyapunov functions based approach to quantify the energy-flow in a dynamical network via a model of a comparison system. Introducing a notion of power and energy flow in a dynamical network, we use sum-of-squares programming tools to partition polynomial networks into weakly interacting subsystems. Examples are provided to illustrate the proposed method of decomposition.**

## I. Introduction

Control of large-scale dynamical systems is challenging because of the computational complexity. In such scenarios, distributed control design is an attractive option which often offers a trade off between computational tractability and controller performance [1], [2]. Distributed control design has become a critical challenge with the advent of interdependent large-scale infrastructure systems, smart cities, cloud architectures, and coordination and control problems for large fleets of independent agents. The combination of scale and dynamical complexity of these systems makes it computationally difficult to design global control solutions. Design of distributed control architectures can be roughly described as a three step process: 1) decomposition of the network into subsystems, 2) distributed control design to stabilize isolated subsystems, and 3) verification of stability of the closed-loop network under distributed control. The first step of this process is critical, and is the focus of this paper, since a poor system or model decomposition can affect all subsequent steps of distributed control design.

In some applications, the physical layout or construction of a network dictates the subsystem structure. For example, in critical infrastructure systems the subsystem structure is traditionally defined based on distance and connectivity of buses, nodes or junctions. The increasing integration and instrumentation of heterogeneous infrastructure systems, e.g. power-gas systems [3], [4] or water-power systems [5], introduces spatial and temporal overlap in time-scales across different systems, removing the insularity required previously to guarantee control performance. Moreover, there are scenarios where a suitable system decomposition may not be known *a priori*, e.g. design of large-scale or ad-hoc communication networks [3], [6] or cyber-physical systems made of agile teams of agents. These reasons motivate the development of new system decomposition algorithms that identify appropriate subsystems that facilitate and enhance distributed control design and performance.

While the idea of using a decomposed network model for distributed stability analysis and control design has been proposed in the literature [1], [2], [7], [8] , decomposition of nonlinear dynamical networks for distributed analysis and control has not received as much attention. In [9], [10], authors proposed an algorithm to decompose nonlinear networks into sub-networks to perform stability analysis using composite Lyapunov functions. However, the proposed method involved linearization of the dynamics, and hence 1) is not suitable for networks that have nonlinear interactions, and 2) provides a decomposition that is only appropriate in the close neighborhood of the operating point. In a recently concluded work [11], authors used data-driven deep learning techniques to decompose a nonlinear dynamical network using Koopman grammians.

In this paper we propose a model-based decomposition method for nonlinear dynamical networks using comparison systems representation. Vector Lyapunov functions and comparison systems have been used for distributed stability analysis and control design for nonlinear networks [12]–[16]. In [7], [8], [17] authors proposed sum-of-squares programming methods to compute the comparison systems for generic polynomial dynamical networks. In this paper, we use vector Lyapunov functions and comparison systems based approach to quantify the energy-flow in the network and perform a spectral clustering of the network such that the energy-flow between the sub-networks is minimized. The proposed method is dynamic since it allows to find a decomposition that best suits a given set of operating conditions. The paper is structured as follows. In Section II, we describe the problem. Section III presents a brief overview of the necessary background. The proposed algorithms is discussed in details in Section IV, while

978-3-9524-2699-9 ©2018 EUCA

numerical examples are presented in Section V. The paper is concluded in Section VI.

## II. Problem Description

Consider an $n$-dimensional nonlinear dynamical system of the form

$$\dot{x}(t) = f(x(t)) \;\forall t \geq 0\,,\;\; x \in \mathbb{R}^n\,, \tag{1}$$

with an equilibrium at the origin ($f(0) = 0$), where $f : \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz. For brevity, we would drop the argument $t$ from the state variables, whenever obvious. Often complex real-life systems can be modeled in the general form (1). Analysis of such systems for control and decision making, however, can be non-trivial due to the nonlinearities. Especially, computational techniques for nonlinear systems analysis generally scale poorly with the size of the system. Furthermore, in real-time operations and decision making, often the information required for a global system-wide analysis is unavailable, rendering a decomposition-based distributed analysis the preferred option. The system in (1) can be viewed as a nonlinear network of $m$ dynamical subsystems as follows:

$$\mathcal{S}_i \;:\;\; \dot{x}_i = f_i(x_i) + \sum_{j\in\mathcal{N}_i\backslash i} g_{ij}(x_i, x_j)\,, \tag{2a}$$

$$\text{where,}\;\; x_i \in \mathbb{R}^{n_i},\, x_j \in \mathbb{R}^{n_j} \;\; \forall j \in \mathcal{N}_i\backslash i\,, \tag{2b}$$

$$f_i(0) = 0 \text{ and } g_{ij}(x_i, 0) = 0 \;\; \forall x_i \;\; \forall j \in \mathcal{N}_i\backslash i\,. \tag{2c}$$

Each subsystem $\mathcal{S}_i$ has some local dynamical state variables $x_i$ associated with it, while $\mathcal{N}_i$ (with $i \in \mathcal{N}_i$) represent the neighboring nodes that interact with the subsystem-$i$. $f_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ represents the isolated subsystem dynamics, while the terms $g_{ij} : \mathbb{R}^{n_i}{\times}\mathbb{R}^{n_j} \to \mathbb{R}^{n_i}$ represent the interactions from the neighbors. Note that the interaction model (total interaction as a sum of the $g_{ij}$'s) used here is chosen for simplicity, but more generic interaction models can also be considered. Also, it is assumed that the decomposition is overlapping in a way that no two subsystems share any common state variable, and

$$\sum\nolimits_{i=1}^{m} n_i = n\,.$$

A decomposed system model of the form (2) has been used for stability analysis studies in the literature (see [8], [13]–[16], [18]). However, finding a suitable decomposition of the large system that facilitates distributed analysis is not trivial. In [9], [10] authors applied spectral graph partitioning algorithms on linearized system dynamics to obtain a system decomposition that aims to minimize worst-case energy flows between subsystems. However, because of the linearization, such methods are only applicable in the close neighborhood of the operating points. For systems under large disturbance, i.e. far away from the operating point, the decomposition structure is expected to change. Furthermore, quadratic and higher order terms in the interactions do not show up in the linearized representation and are therefore completely ignored in the decomposition.

In this article, we propose a decomposition algorithm to partition a nonlinear dynamical network into weakly-interacting subsystems, whereby the nonlinearity of the system is explicitly considered in the decomposition algorithm, via the use of a *comparison systems* representation, as explained in the following sections.

## III. Preliminaries

Let us use $|\cdot|$ to denote both the Euclidean norm of a vector and the absolute value of a scalar; and use $\mathbb{R}\,[x]$ to denote the ring of all polynomials in $x \in \mathbb{R}^n$.

### A. Stability Analysis

The equilibrium point at the origin of (1) is Lyapunov stable if, for every $\varepsilon > 0$ there is a $\delta > 0$ such that $|x(t)| < \varepsilon \;\forall t \geq 0$ whenever $|x(0)| < \delta$. Moreover, it is asymptotically stable in a domain $\mathcal{D} \subseteq \mathbb{R}^n$, $0 \in \mathcal{D}$, if it is Lyapunov stable and $\lim_{t\to\infty} |x(t)| = 0$ for every $x(0) \in \mathcal{D}$.

*Theorem 1:* (Lyapunov, [19], [20]) If there exists a domain $\mathcal{D} \subseteq \mathbb{R}^n$, $0 \in \mathcal{D}$, and a continuously differentiable positive definite function $\tilde{V} : \mathcal{D} \to \mathbb{R}_{\geq 0}$, i.e. the 'Lyapunov function' (LF), then the origin of (1) is asymptotically stable if $\nabla\tilde{V}^T f(x)$ is negative definite in $\mathcal{D}$.

An estimate of the region-of-attraction (ROA) can be given by [21]

$$\mathcal{R} := \{x \in \mathcal{D}\,|\,V(x) \leq 1\}\;,\;\text{ with } V(x) = \frac{\tilde{V}(x)}{\gamma^{max}}, \tag{3}$$

$$\text{where } \gamma^{max} := \max\left\{\gamma \;\middle|\; \left\{x \in \mathbb{R}^n \;\middle|\; \tilde{V}(x) \leq \gamma\right\} \subseteq \mathcal{D}\right\}\,,$$

i.e. the boundary of the ROA is estimated by the unit level-set of a suitably scaled LF $V(x)$.

Relatively recent studies have explored how sum-of-squares (SOS) based methods can be utilized to find LFs by restricting the search space to SOS polynomials [22]–[25]. A (multivariate) polynomial $p \in \mathbb{R}\,[x]\,,\; x \in \mathbb{R}^n$, is called a *sum-of-squares* (SOS) if there exist some polynomial functions $h_i(x), i = 1 \ldots s$ such that $p(x) = \sum_{i=1}^{s} h_i^2(x)$. We denote the ring of all SOS polynomials in $x \in \mathbb{R}^n$ by $\Sigma[x]$. Checking if $p \in \mathbb{R}[x]$ is an SOS is a semi-definite problem which can be solved with a MATLAB® toolbox SOSTOOLS [26], [27] along with a semidefinite programming solver such as SeDuMi [28]. The SOS technique can be used to search for polynomial LFs by translating the conditions in Theorem 1 to equivalent SOS conditions [22], [26], [27], [29]–[31]. An important result from algebraic geometry, called Putinar's Positivstellensatz (P-Satz) theorem [32], [33], helps in translating the SOS conditions into SOS feasibility problems.

*Theorem 2:* Let $\mathcal{K} = \{x \in \mathbb{R}^n \,|\, k_1(x) \geq 0\,, \ldots, k_m(x) \geq 0\}$ be a compact set, where $k_j \in \mathbb{R}[x]$, $\forall j \in \{1, \ldots, m\}$. Suppose there exists a $\mu \in \left\{\sigma_0 + \sum_{j=1}^{m}\sigma_j\, k_j \,|\, \sigma_0, \sigma_j \in \Sigma[x]\,, \forall j\right\}$ such that $\{\,x \in \mathbb{R}^n |\;\mu(x) \geq 0\}$ is compact. Then,

$$p(x) > 0 \;\forall x \in \mathcal{K} \implies p \in \left\{\sigma_0 + \sum\nolimits_{j}\sigma_j k_j \,|\, \sigma_0, \sigma_j \in \Sigma[x], \forall j\right\}.$$

### B. Linear Comparison Principle

In [34], [35] the authors proposed to view the LF as a dependent variable in a first-order auxiliary differential equation, often termed as the 'comparison equation' (or, 'comparison system'). It was shown in [12], [36] that, under certain conditions, the comparison equation can be effectively reduced to a set of linear differential equations. Noting that all the elements of the matrix $e^{At}$, $t \geq 0$, where $A = [a_{ij}] \in \mathbb{R}^{m\times m}$, are non-negative if and only if $a_{ij} \geq 0, i \neq j$, it was shown in [36], [37]:

*Lemma 1:* Let $A \in \mathbb{R}^{m\times m}$ have non-negative off-diagonal elements, $v : [0,\infty) \to \mathbb{R}^m$ and $r : [0,\infty) \to \mathbb{R}^m$. If $v(0) = r(0)$, $\dot{v}(t) \leq A\,v(t)$ and $\dot{r}(t) = A\,r(t)$, then $v(t) \leq r(t)\ \forall t \geq 0$.

We henceforth refer to Lemma 1 as the 'comparison principle'; the linear time-invariant system $\dot{r}(t) = A\,r(t)$ as a 'comparison system' (CS); and the matrix $A$ as the comparison matrix (CM).

### C. Similarity Graphs and Spectral Partitioning

Consider a graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ where $\mathcal{V} = \{1, 2, \ldots, m\}$ represent a set of $m$ vertices (or, nodes) and $\mathcal{E} \subseteq \mathcal{V}\times\mathcal{V}$ denote a set of $l$ edges between the vertices (i.e. $\mathbf{card}(\mathcal{E}) = l$). A notion of *similarity* (or, edge weight), denoted by a scalar $w_{ij} \geq 0$, is associated with each pair of nodes $(i, j)$ in the graph, such that $w_{ij} > 0$ if an edge exists between the nodes $i$ and $j$, i.e. $(i, j) \in \mathcal{E}$, and $w_{ij} = 0$ if there does not exist an edge between the nodes $i$ and $j$ (i.e. $(i, j) \notin \mathcal{E}$). The edge weights quantify how similar two nodes in the graph are to each other, i.e. higher the value of the edge weights more similar the corresponding vertices are.

The spectral clustering technique used in this paper for graph partitioning is applicable for graphs that are *undirected*, i.e. the edge weights (or similarity values) satisfy $w_{ij} = w_{ji}\,\forall i \neq j$ (clearly, for undirected graphs, $(i, j) \in \mathcal{E}$ implies that $(j, i) \in \mathcal{E}$). If a graph is *directed*, we can convert it into an *undirected* graph by assigning

$$w_{ij} \leftarrow \frac{1}{2}\left(w_{ij} + w_{ji}\right)\ , \text{ or } w_{ij} \leftarrow \max\{w_{ij}, w_{ji}\}\,.$$

A symmetric *weighted adjacency matrix* $W(\mathcal{G}) = [w_{ij}] \in \mathbb{R}^{m\times m}$ is then constructed, such that $w_{ii} = 0\ \forall i \in \mathcal{V}$, and $w_{ij} = w_{ji} > 0\ \forall (i, j) \in \mathcal{E}$. Finally, a normalized (symmetric positive semi-definite) *graph Laplacian matrix*, $L_{sym}(\mathcal{G}) \in \mathbb{R}^{m\times m}$, is defined as,

$$L_{sym} = I_m - D^{-1/2} W D^{-1/2}, \quad D = \mathbf{diag}\left(W\,\mathbf{1}_m\right)\,,$$

where $I_m$ is an $m\times m$ identity matrix, $\mathbf{1}_m$ is a $m$-dimensional column vector with each entry equal to $1$, $D$ is a diagonal matrix with diagonal entries equal to the sum of the corresponding row in $W$; and $(\cdot)^{-1/2}$ denotes the inverse of the square root a matrix.

The idea behind spectral graph partitioning is to partition the graph $\mathcal{G}$ into sub-graphs in such a way that the vertices within a sub-graph are *similar* to each other (i.e. have high edge weights), while vertices from two different sub-graphs are *dissimilar* (i.e. have low edge weights). In particular, the $K$-means spectral clustering algorithm uses the normalized graph Laplacian $L_{sym}$ to partition the vertices into $K$ clusters. The first $K$ eigenvectors of the graph Laplacian, in the order of increasing eigenvalues, are used as columns to construct a matrix $U \in \mathbb{R}^{m\times K}$. Then the normalized rows of the matrix $U$ are clustered using the $K$-means clustering algorithm. For further details on the algorithm, please refer to [38], [39].

## IV. Decomposition via Comparison Systems

In this section we describe our proposed approach of decomposing a nonlinear (polynomial) dynamical network using a comparison system. The proposed approach is composed of two steps - 1) computing a comparison system, and 2) apply spectral graph partitioning on the comparison system.

### A. Constructing Comparison System

Given a polynomial network of the form (2), vector Lyapunov functions of the subsystems can be used to construct a linear comparison system. In [17] authors used sum-of-squares programming to compute the vector Lyapunov functions and the comparison matrix for polynomial dynamical networks. It is assumed that, in absence of any interaction, each subsystem $\mathcal{S}_i$ is locally asymptotically stable at the origin. Specifically, we assume that for each $i \in \{1, \ldots, m\}$

$$\mathcal{S}_i \text{ (isolated) } : \dot{x}_i = f_i(x_i)$$

has an asymptotically stable equilibrium point at the origin, and admits a polynomial Lyapunov function $V_i(x_i)$. The expanding interior algorithm [22], [25] can be used to compute these LFs and a corresponding region of attraction such that $\mathcal{R}_i := \{x_i \in \mathbb{R}^{n_i}\,|V_i(x_i) \leq 1\}$ is an estimate of the region of attraction of the $i$-th isolated subsystem.

The vector LFs thus computed are used to define the states of the *comparison system*,

$$v(t) := \begin{bmatrix} V_1(x_1(t)) & V_2(x_2(t)) & \ldots & V_m(x_m(t)) \end{bmatrix}^T$$

such that the comparison system is a *positive system*, i.e. each state of the comparison system only takes non-negative values at all time $t$. Then the objective is to compute the comparison matrix (CM) $A = [a_{ij}]$ such that in some domain

$$\mathcal{D} := \{x \in \mathbb{R}^n\,|\,V_i(x_i) \leq \gamma_i\ \forall i\} \subseteq \{x \in \mathbb{R}^n\,|\,x_i \in \mathcal{R}_i\ \forall i\}$$

where $\gamma_i \leq 1$, the following comparison equations hold:

$$\dot{v} \leq A\,v\ \ \forall x \in \mathcal{D}\,,\ \text{where } A = [a_{ij}]\,,\, a_{ij} \geq 0\,\forall i \neq j\,.$$

Sum-of-squares programming can be used to compute the CM which has non-negative off-diagonal elements. In order to obtain a stable comparison system, the CM $A$ has to be Hurwitz. While solving for a generic Hurwitz CM for a large system is difficult, a sufficient condition for CM to be Hurwitz is given by the Gershgorin circle theorem [40] which says that a strictly diagonally dominant matrix[1] with negative diagonal elements is Hurwitz. This condition motivates us to solve for

[1] $A = [a_{ij}]$ is strictly diagonally dominant if $\sum_{j\neq i}|a_{ij}| < |a_{ii}|\,, \forall i$.

each row of the CM, in a parallel and scalable way, by solving the following problem:

$$\forall i:\ \min_{a_{ij}} \sum\nolimits_{j\in\mathcal{N}_i} a_{ij},\ \text{s.t. } \dot{V}_i \le \sum\nolimits_{j\in\mathcal{N}_i} a_{ij} V_j\ \forall x\in\mathcal{D}, \quad (4)$$

which can be formulated as an SOS optimization:

$$\begin{aligned}&\forall i\colon \underset{a_{ij},\sigma_{ij}}{\text{minimize}} \sum\nolimits_{j\in\mathcal{N}_i} a_{ij} && (5)\\ &\text{s.t. } a_{ii}\in\mathbb{R}\,,\ a_{ij}\ge 0\ \forall i\neq j\,,\ \sigma_{ij}\in\Sigma[x_{\mathcal{N}_i}]\,,\\ &\quad -\nabla V_i^T\big(f_i+\sum_{j\neq i} g_{ij}\big)+\sum_{j\in\mathcal{N}_i}\big(a_{ij}V_j-\sigma_{ij}(\gamma_j-V_j)\big)\in\Sigma[x_{\mathcal{N}_i}].\end{aligned}$$

Here $\nabla$ is the gradient operator, and $x_{\mathcal{N}_i}$ denote the state variables that belong to the neighborhood of the subsystem $i$ (recall that $i\in\mathcal{N}_i$ ). The CM is guaranteed to be Hurwitz, if the sum of each row of the CM $A$ turns out to be negative. Otherwise, one can do an eigenvalue analysis to determine whether or not the CM is Hurwitz.

Note that the CM is not unique, and depends on the vector LFs computed earlier, as well as the domain of definition (also referred to as the domain of interest) $\mathcal{D}$. For a given set of vector LFs, the CM could be different if the domain of interest changes. This particular feature is useful in monitoring the change in the energy-flow pattern in a nonlinear dynamical network under different operating conditions.

### *B. Energy-Flow in the Network*

The comparison system (CS)

$$\dot{r}(t) = A\,r(t)\,,\quad r(0)=v(0)\,, \quad (6)$$

where we denote the CS states as $r=[r_1\,,\,\ldots\ r_m]^T$, provides an upper bound on the values of the level-set of the subsystem vector LFs, i.e. $V_i(x_i(t))\ \le\ r_i(t)\ \forall i\,\forall t$. The comparison system can be represented as a *similarity graph* $\mathcal{G}(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=\{i\}_{i=1}^m$ represent the nodes, while $\mathcal{E}=\{E_k\}_{k=1}^l$ denote the directional edges (where $l$ is the number of positive off-diagonal elements in the CM). We consider the case when the CM $A$ is Hurwitz and the original system (2) is guaranteed to be asymptotically stable. The goal here is to partition the network such that the *worst case* energy flow between the sub-networks is minimized. However, for generic nonlinear dynamical networks of the form (2), the concept of *energy-flow* along the edges is not well defined, but are usually informed by physical knowledge of the system (e.g. in power systems, the edge flow could be represented by the active power flowing from one node to another). It may be convenient to view the vector LFs as some form of energy-levels of the subsystems of the network, while the CS provides some bound on the rate of change of the energy levels of the subsystems. We use the vector LFs and their time derivatives to quantify the energy flow across edges as follows:

*Definition 1:* Let us define the *'power-flow'*, $\phi_{ij}(t)$, from node $j$ into the node $i$ along a directional edge $(i,j)$ in the dynamical network (2) which admits vector LFs $V_i(x_i)\,\forall i$, as

$$\phi_{ij}(t) := \nabla V_i^T g_{ij}(x_i,x_j)\,,$$

and *energy-flow*, $\psi_{ij}(t_0,t_f)$, across the directional edge $(i,j)$ between time $t_0$ and $t_f$ is defined as the following integral,

$$\psi_{ij}(t_0,t_f) := \int_{t_0}^{t_f} |\,\phi_{ij}(t)\,|\;dt\,.$$

Note that the *power-flow* can have both positive and negative values, where the positive values are referred to as the *'in-flow'* of power, while negative values are referred to as *'out-flow'* of power. The *energy-flow* is the time integral of the absolute value of the power-flow. Computing the power and energy-flows requires solving the nonlinear dynamics for the given initial conditions. However, for a given set of possibly (or likely) initial conditions, we can compute upper bounds on the power and energy-flow using the vector LFs. Specifically, note that, since the dynamical equations are polynomial, and if the states are bounded, we can find finite positive scalars $\alpha_{ij}\in\mathbb{R}_{\ge 0}\,\forall(i,j)\in\mathcal{E}$, such that

$$x\in\mathcal{D}\implies |\phi_{ij}(t)|\le \alpha_{ij}\,V_j(x_j)\ \ \forall(i,j)\in\mathcal{E}\,. \quad (7)$$

Recall that $x_j=0$ implies $V_j(0)=0$ and $\phi_{ij}=0$. The scalars $\alpha_{ij}$ are bounded (since $\nabla V_i^T$ and $g_{ij}(\cdot)$ are polynomials), but not unique and depend on the domain of interest. Since the evolution of the vector LFs are governed by the CS, a closed-form expression can be found for the upper bound of the energy-flow in the network within a domain of interest. Specifically, let us define the observable (or, output) vector of the CS as

$$y=C^T r\,,\quad C=\begin{bmatrix}c_1 & c_2 & \ldots & c_l\end{bmatrix}\in\mathbb{R}_{\ge 0}^{m\times l} \quad (8)$$

where each column of $C$ corresponds to the a different edge in the network. If the $k$-th edge $(i,j)\in\mathcal{E}$ is from the node $j$ to the node $i$, then the $k$-th column is given by $c_k=\alpha_{ij}e_j$ where $e_j$ denotes an $m$-dimensional *standard basis vector* with the $j$-th entry equal to 1 and all other entries equal to $0$.

*Proposition 1:* Suppose that a nonlinear dynamical network (2) is represented by the CS (6) in some domain of interest $\mathcal{D}$ with a Hurwitz CM. Given a trajectory within $\mathcal{D}$, the net energy-flow in the network over time $t\in[0,\infty)$ is upper bounded by $-(C\,\mathbf{1}_l)^T A^{-1}v(0)$, while the energy-flow in the directional edge $(i,j)$ is upper bounded by $-\alpha_{ij}e_j^T A^{-1}v(0)$.

**Proof** The CS is a *positive system* with a CM that is Metzler (i.e. non-negative off-diagonal elements). If the CM is also Hurwitz, its inverse exists and has only non-positive entries [16]. Therefore, the integral $\int_0^\infty \exp(At)dt$ is equal to $-A^{-1}$ with all non-negative entries. Note that,

$$\begin{aligned}\psi_{ij}(0,\infty) &= \int_0^\infty |\phi_{ij}(t)|\;dt \le \int_0^\infty \alpha_{ij}V_j(x_j(t))\,dt\\ &\le \int_0^\infty \alpha_{ij}r_j(t)\,dt = \int_0^\infty \alpha_{ij}e_j^T r(t)\,dt\\ &= \alpha_{ij}e_j^T\int_0^\infty e^{At}\,dt\,r(0) = -\alpha_{ij}e_j^T A^{-1}v(0)\end{aligned}$$

The total energy-flow across all edges is given by

$$\sum_{(i,j)\in\mathcal{E}}\psi_{ij}(0,\infty)\le\sum_{k=1}^{l}-c_k^T A^{-1}v(0) = -(C\,\mathbf{1}_l)^T A^{-1}v(0)\,.\ \blacksquare$$

Note that, since $-A^{-1}$ has only non-negative entries, the upper bound on the energy flow in any edge is monotonically increasing with respect to the initial level-sets, $v(0)$, of the vector LFs. This is useful if we want to compute the *worst-case* energy flow in the network given some domain of interest. For example, if the domain of interest is defined in the form of $\mathcal{D} = \{x \in \mathbb{R}^n \,|\, V_i(x_i) \le \gamma_i \;\forall i\}$, for some $\gamma_i \le 1$, then the *worst-case* upper bound on the energy flow across the $k$-th edge is $-c_k^T A^{-1}\Gamma$, where $\Gamma = \begin{bmatrix}\gamma_1 & \ldots & \gamma_m\end{bmatrix}^T$.

### C. $K$-means Clustering on Comparison System

The formulation of energy-flow along the edges proposed here allows us to quantify the energy-flow pattern in the network given an initial condition. It is expected that, under different operating conditions, the energy-flow pattern of the network changes and henceforth the decomposition that minimizes the energy-flow between sub-networks may be different as well. While the exact operating conditions may not be known *apriori*, one can proceed with the analysis of the *worst-case* energy-flow within some domain of interest.

For a domain of interest $\mathcal{D} = \{x \in \mathbb{R}^n \,|\, v(t) \le \Gamma \;\forall t\}$, where $\Gamma \le 1$ (element-wise), we use the *worst-case* upper bound on the energy flow across the edge in either direction to construct a symmetric *weighted adjacency matrix* of the network. Specifically, we construct the symmetric weighted adjacency matrix $W(\mathcal{G}) = [w_{ij}]$ as,

$$w_{ij} = \begin{cases} \max\{-\alpha_{ij} e_j^T A^{-1}\Gamma\,,\ -\alpha_{ji} e_i^T A^{-1}\Gamma\}, & (i,j) \in \mathcal{E} \\ 0\,, & \text{otherwise} \end{cases}$$

Note, of course, that while this construction of weighted adjacency matrix is concerned with the *worst-case* energy-flow in a domain of definition, the formulation can be adapted to investigate the energy-flow under a given set of initial conditions, by replacing $\Gamma$ with $v(0)$. The final step involves choosing the desired number of sub-networks and applying the $K$-means clustering algorithm on the weighted adjacency matrix $W$.

## V. Simulation Results

In this section we present some numerical examples to illustrate how the proposed method can be used to compute the energy-flow across nonlinear dynamical networks under varying operating conditions, and decompose the network minimizing energy-flow between sub-networks. In the figures presented below, the width of the line between any two nodes represents the relative values of the energy flowing through the edge (i.e. thicker lines represent higher energy-flow).

### A. Lotka-Volterra System

Our first example is a Lotka-Volterra system that describes the evolution of population in a network of 16 communities:

$$\dot{x}_i = (b_i - x_i)\,x_i - \sum\nolimits_{j \in \mathcal{N}_i \setminus \{i\}} x_j\,(c_{ij} + d_{ij} x_i) \qquad (9)$$

where $i = 1, 2, \ldots, 16,$. We use a similar model as in [10], with slightly modified network structure and interaction coefficients ($c_{ij}$ and $d_{ij}$) that are scaled down by a factor of $0.3$, in order to facilitate the construction of comparison systems over a wider range of operating conditions. After shifting the equilibrium point of the system to origin, the Lotka-Volterra system is expressed in the form (2). Quadratic vector LFs $V_i(x_i)$ are computed for each community, such that $\mathcal{R}_i = \{x_i \,|\, V_i(x_i) \le 1\}$ is an estimation of the region of attraction. Defining the domain of interest as $\mathcal{D} = \left\{x \in \mathbb{R}^{16} \,|\, V_i(x_i) \le \gamma_i \;\forall i\right\}$, for some $\gamma_i \le 1$, a CS is constructed based on which the energy-flow pattern of the network can be computed.

Fig. 1 shows calculated (based on the entries of the weighted adjacency matrix $W = [w_{ij}]$) and simulated (by randomly selecting an initial condition and evaluating the maximum value of $\psi_{ij}(0, \infty)$ over a number of scenarios) energy-flow patterns in the network for different domains of interest, and the associated decomposition into two sub-networks. When the domain of interest is defined to be a close neighborhood of the equilibrium point (by choosing $\gamma_i = 0.01 \,\forall i$), the simulated and calculated energy patterns look similar resulting in the same decomposition. When we choose $\gamma_i = 0.6 \,\forall i$, the calculated and simulated energy-flow patterns differ, although the decomposition remains the same. Note that, even when the energy-flow patterns differ in the calculation and simulation, the energy-flow between the sub-networks turns out to be small in both cases.

Fig. 2 shows the energy-flow patterns when the initial condition of the network, determined by the initial energy level $V_i(x_i(0))$ of the nodes, is varied. Specifically, we consider three different scenarios - 1) when the initial level-sets of the vector LFs of the nodes in group-1 are set at high value of $0.5$ and the nodes in group-1 are assigned a low value of $0.1$ (Fig. 2(a)), 2) when group-2 nodes are assigned high initial level-sets and group-1 nodes low initial level-sets (Fig. 2(b)), and 3) when the level-sets of the nodes at the interconnection of the two sub-networks are assigned high initial level-sets and others low initial level-sets (Fig. 2(c)). We can see that in all three scenarios, the energy-flows between two sub-networks remain small, while the energy-flow patterns within each group varies. This shows that such a decomposition indeed favors a distributed analysis and control design, by dividing the network into weakly interacting subsystems.

### B. Network of Van der Pol Systems

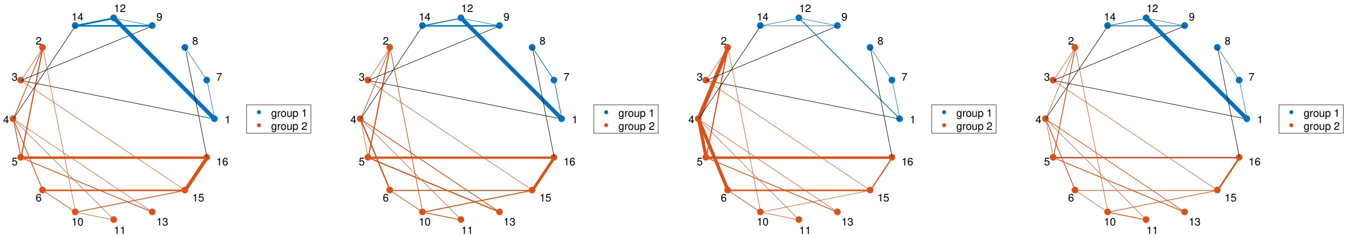
Next we consider another example of a nonlinear network, composed of 9 Van der Pol systems from [8], described by

$$\dot{x}_i = f_i(x_i) + \sum_{j \in \mathcal{N}_i \setminus \{i\}} g_{ij}(x_i, x_j) \qquad (10a)$$

$$f_i(x_i) = \begin{bmatrix} x_{i,2} \\ \mu_i\, x_{i,2}(c_i^{(1)} - c_i^{(2)} x_{i,1} - x_{i,1}^2) - c_i^{(3)} x_{i,1} \end{bmatrix} \qquad (10b)$$

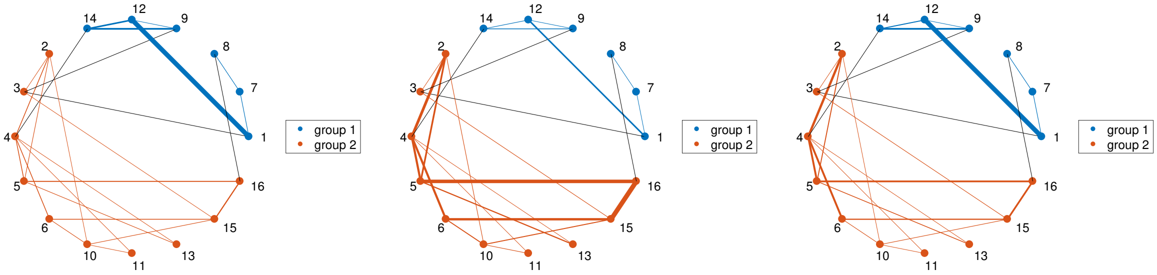
$$g_{ij}(x_i, x_j) = \begin{bmatrix} 0\,, & \beta_{ij}^{(1)}\, x_{j,2} + \beta_{ij}^{(2)}\, x_{j,2}\, x_{i,1} \end{bmatrix}^T . \qquad (10c)$$

where, $c_i^{(1)} = 1 - \left(0.5\, c_i^{(2)}\right)^2$, $c_i^{(3)} = 1 - \sum_{j \in \mathcal{N}_i\{i\}} (0.5\, \beta_{ij}^{(2)} c_i^{(2)} - \beta_{ij}^{(1)})$, $\mu_i$, $\beta_{ij}^{(1)}$ and $\beta_{ij}^{(2)}$ are chosen randomly and $c_i^{(2)}$ are related to the equilibrium point before shifting. Vector LFs for



(a) $\gamma_i = 0.01$ (calculated) (b) $\gamma_i = 0.01$ (simulated) (c) $\gamma_i = 0.6$ (calculated) (d) $\gamma_i = 0.6$ (simulated)

Fig. 1: Calculated and simulated energy-flow patterns and decomposition for varying domains of interest in (9).



(a) $V_i(x_i(0))$ high for group 1 nodes (b) $V_i(x_i(0))$ high for group 2 nodes (c) $V_i(x_i(0))$ high for $\{1, 3, 4, 8, 9, 14, 16\}$

Fig. 2: Calculated energy-flow patterns and decomposition of (9) for varying initial conditions: high values of initial level-sets ($V_i(x_i(0)) = 0.5$) at selected nodes and low values ($V_i(x_i(0)) = 0.1$) at others.

the nine nodes are computed using sum-of-squares methods. Fig. 3 shows the energy-flow patterns and decomposition for varying operating conditions. Specifically, we choose a domain of interest by selecting $\gamma_i = 0.6\,\forall i$, and plot the worst-case energy flow pattern in Fig. 3(a), along with the decomposition. Keeping the same decomposition, we then monitor the change in energy-flow patterns as we vary the initial conditions across the network. In specific, we consider three scenarios: 1) choose high initial level-sets (equal to 0.6) for the group 1 nodes and low level-sets (equal to 0.1) for others (Fig. 3(b)), 2) high initial level-sets (=0.6) for group 2 nodes and low (=0.1) for others (Fig. 3(c)), and 3) high level-sets (=0.6) only for the nodes $1, 5, 7$ and 8 that connect the two sub-networks. We observe that most of the energy-flow is contained within the sub-network, with minimal flow between the two sub-networks, implying that the decomposition yields a weakly interacting network.

## VI. Conclusion

In this article we considered the problem of decomposing a nonlinear network into weakly interacting subsystems, to facilitate distributed analysis and control design. Using a vector Lyapunov functions based approach we showed that the evolution of the energy levels of the nodes of the system can be modeled via a linear comparison system. Further, introducing a notion of power and energy flows between two nodes in a dynamical network, we proposed a method to compute the dynamic edge weights in the network. Finally, using spectral clustering techniques on the weighted adjacency matrix the nonlinear dynamical network is decomposed into weakly interacting subsystems. Sum-of-squares programming tools are used to demonstrate the working on the algorithm on two examples of nonlinear networks. Future work will investigate the applicability of this method to larger systems (interested readers are referred to the recent work in [41]), and to real-world problems such as coherency detection in power systems.

## Acknowledgment

This work was supported by the United States Department of Energy (contract DE-AC02-76RL01830) under the Control of Complex Systems Initiative at Pacific Northwest National Laboratory.

## References

[1] F. Bullo, J. Cortés, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton, New Jersey: Princeton University Press, 2009.

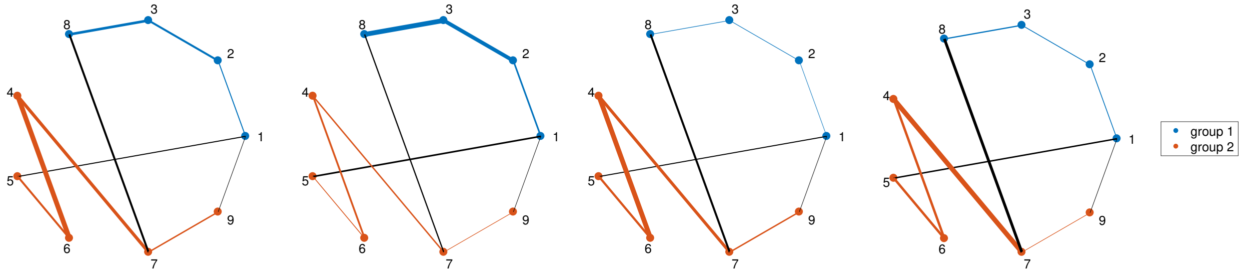
[2] A. I. Zečević and D. D. Šiljak, *Control of Complex Systems: Structural Constraints and Uncertainty*, ser. Communications and Control Engineering. New York: Springer, 2010.

[3] N.-Y. Chiang and V. M. Zavala, "Large-scale optimal control of interconnected natural gas and electrical transmission systems," *Applied Energy*, vol. 168, pp. 226–235, 2016.

[4] T. Li, M. Eremia, and M. Shahidehpour, "Interdependency of natural gas network and power system security," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1817–1824, 2008.

[5] S. J. Pereira-Cardenal, B. Mo, A. Gjelsvik, N. D. Riegels, K. Arnbjerg-Nielsen, and P. Bauer-Gottwein, "Joint optimization of regional water-power systems," *Advances in Water Resources*, vol. 92, pp. 200–207, 2016.

[6] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.



(a) $\gamma_i = 0.6$ for each node (b) $V_i(x_i(0))$ high for group 1 (c) $V_i(x_i(0))$ high for group 2 (d) $V_i(x_i(0))$ high for $\{1, 5, 7, 8\}$

Fig. 3: Calculated energy-flow patterns and decomposition for (10), under varying initial conditions (defined by $V_i(x_i(0))$).

[7] S. Kundu and M. Anghel, “Distributed coordinated control of large-scale nonlinear networks,” *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 240–245, 2015.

[8] ——, “A multiple-comparison-systems method for distributed stability analysis of large-scale nonlinear systems,” *Automatica*, vol. 78, pp. 25–33, 2017.

[9] J. Anderson, Y.-C. Chang, and A. Papachristodoulou, “Model decomposition and reduction tools for large-scale networks in systems biology,” *Automatica*, vol. 47, no. 6, pp. 1165–1174, 2011.

[10] J. Anderson and A. Papachristodoulou, “A decomposition technique for nonlinear dynamical system analysis,” *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1516–1521, 2012.

[11] Z. Liu, S. Kundu, L. Chen, and E. Yeung, “Decomposition of Nonlinear Dynamical Systems Using Koopman Gramians,” *arXiv preprint arXiv:1710.01719*, 2017.

[12] F. N. Bailey, “The application of Lyapunov’s second method to interconnected systems,” *J. SIAM Control*, vol. 3, pp. 443 – 462, 1966.

[13] M. Araki, “Stability of large-scale nonlinear systems quadratic-order theory of composite-system method using M-matrices,” *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 129 – 142, 1978.

[14] N. Michel, A. “On the status of stability of interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 28, no. 6, pp. 639–653, 1983.

[15] D. D. Šiljak, “Stability of large-scale systems under structural perturbations,” *Systems, Man and Cybernetics, IEEE Transactions on*, vol. SMC-2, no. 5, pp. 657–663, Nov 1972.

[16] S. Weissenberger, “Stability regions of large-scale systems,” *Automatica*, vol. 9, no. 6, pp. 653–663, 1973.

[17] S. Kundu and M. Anghel, “Computation of linear comparison equations for stability analysis of interconnected systems,” in *54th IEEE Conf. on Decision and Control*, Dec 2015, pp. 1762–1768.

[18] ——, “A sum-of-squares approach to the stability and control of interconnected systems using vector Lyapunov functions,” in *The 2015 American Control Conference*, July 2015, pp. 5022–5028.

[19] A. M. Lyapunov, *The General Problem of the Stability of Motion*. Kharkov, Russia: Kharkov Math. Soc., 1892.

[20] H. K. Khalil, *Nonlinear Systems*. New Jersey: Prentice Hall, 1996.

[21] R. Genesio, M. Tartaglia, and A. Vicino, “On the estimation of asymptotic stability regions: State of the art and new proposals,” *IEEE Transactions on Automatic Control*, vol. 30, no. 8, pp. 747–755, Aug. 1985.

[22] Z. W. Jarvis-Wloszek, “Lyapunov based analysis and controller synthesis for polynomial systems using sum-of-squares optimization,” Ph.D. dissertation, University of California, Berkeley, CA, 2003.

[23] P. A. Parrilo, “Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization,” Ph.D. dissertation, Caltech, Pasadena, CA, 2000.

[24] W. Tan, “Nonlinear control analysis and synthesis using sum-of-squares programming,” Ph.D. dissertation, University of California, Berkeley, CA, 2006.

[25] M. Anghel, F. Milano, and A. Papachristodoulou, “Algorithmic construction of Lyapunov functions for power system stability analysis,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 60, no. 9, pp. 2533–2546, Sep 2013.

[26] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, and P. A. Parrilo, “SOSTOOLS: Sum of squares optimization toolbox for MATLAB,” 2013, available from `http://www.eng.ox.ac.uk/control/sostools`.

[27] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo, *Positive Polynomials in Control*. Springer-Verlag, 2005, ch. SOSTOOLS and Its Control Applications, pp. 273–292.

[28] J. F. Sturm, “Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones,” *Optimization Methods and Software*, vol. 11-12, pp. 625–653, Dec. 1999, software available at http://fewcal.kub.nl/sturm/software/sedumi.html.

[29] Z. J. Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, *Positive Polynomials in Control*. Berlin, Heidelberg: Springer-Verlag, 2005, ch. Control Applications of Sum of Squares Programming, pp. 3–22.

[30] A. Papachristodoulou and S. Prajna, *Positive Polynomials in Control*. Springer-Verlag, 2005, ch. Analysis of non-polynomial systems using the sum of squares decomposition, pp. 23–43.

[31] G. Chesi, “LMI techniques for optimization over polynomials in control: a survey,” *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2500 – 2510, 2010.

[32] M. Putinar, “Positive polynomials on compact semi-algebraic sets,” *Indiana University Mathematics Journal*, vol. 42, no. 3, pp. 969–984, 1993.

[33] J.-B. Lasserre, *Moments, Positive Polynomials and Their Applications*. World Scientific, 2009, vol. 1.

[34] R. Conti, “Sulla prolungabilità delle soluzioni di un sistema di equazioni differenziali ordinarie.” *Bollettino dell’Unione Matematica Italiana*, vol. 11, no. 4, pp. 510–514, 1956.

[35] F. Brauer, “Global behavior of solutions of ordinary differential equations,” *Journal of Mathematical Analysis and Applications*, vol. 2, no. 1, pp. 145–158, 1961.

[36] R. Bellman, “Vector Lyapunov functions,” *SIAM Journal of Control*, vol. 1, no. 1, pp. 32–34, 1962.

[37] E. F. Beckenbach and R. Bellman, “Inequalities,” *Spring-Verlag, New York/Berlin*, 1961.

[38] U. Von Luxburg, “A tutorial on spectral clustering,” *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[39] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *Advances in neural information processing systems*, 2002, pp. 849–856.

[40] H. E. Bell, “Gershgorin’s theorem and the zeros of polynomials,” *American Mathematical Monthly*, pp. 292–295, 1965.

[41] A. A. Ahmadi, G. Hall, A. Papachristodoulou, J. Saunderson, and Y. Zheng, “Improving efficiency and scalability of sum of squares optimization: Recent advances and limitations,” *arXiv preprint arXiv:1710.01358*, 2017.