# Coarse-Graining Complex Networks for Control Equivalence

Daniele Toller, Mirco Tribastone, Max Tschaikowski, and Andrea Vandin

***Abstract*—The ability to control complex networks is of crucial importance across a wide range of applications in natural and engineering sciences. However, issues of both theoretical and numerical nature introduce fundamental limitations to controlling large-scale networks. In this article, we cope with this problem by introducing a coarse-graining algorithm. It leads to an aggregated network which satisfies control equivalence, i.e., such that the optimal control values for the original network can be exactly recovered from those of the aggregated one. The algorithm is based on a partition refinement method originally devised for systems of ordinary differential equations, here extended and applied to linear dynamics on complex networks. Using a number of benchmarks from the literature we show considerable reductions across a variety of networks from biology, ecology, engineering, and social sciences.**

***Index Terms*—Control systems, costs, linear systems, mathematical models, optimal control, reduced order systems.**

## I. Introduction

A large variety of complex systems of biological, ecological, social, and technological nature can be conveniently modeled as networks. Their dynamics can thus be studied as a (typically high-dimensional) system of differential equations where the evolution of each node depends on how it is connected to its neighbors and how it responds to external stimuli. Network control, i.e., the ability to steer the system toward a desired target behavior, is of crucial importance both theoretically and practically [1]. This involves two main steps [2]: 1) Determining through which nodes to input a control action; and 2) finding the actual values of the inputs.

Network science has made fundamental progress with approaches that identify a minimum subset of nodes to use as drivers in order to achieve control of the system, e.g., [3], [4], [5], and [6]. The actual computation of a control law is usually a nontrivial task [7], [8]. For example, using standard results from systems theory [9], the controls that minimize energy dissipation can be computed explicitly in networks with linear dynamics. However, this involves the inversion of the controllability matrix, which ultimately gives a cubic dependency of the computational cost with the cardinality of the network [10]. Furthermore, the realizability of explicit solutions may be hindered in practice by numerical ill-conditioning of the controllability matrix [11], by physical constraints that preclude the use of arbitrarily large input signals [10], [12], or control trajectories that evolve far away in the state space [13], [14]. Overall, these issues have been recognized as fundamental limitations to controlling large networks [1], [11]. Across many branches of science and engineering, coarse graining is a typical strategy to cope with large-scale, intractable problems. Broadly speaking, it can be defined as an approach that maps a model onto a simpler one on which the analysis can be carried out more efficiently. Here, we propose a coarse-graining algorithm for complex networks, which preserves the control trajectories and the computation of optimal control values.

Although our ultimate goal is the control of networks with nonlinear dynamics, the study of the linear case is a fundamental step in this endeavor. We follow the standard setting [1] whereby the linear dynamical system induced by a network with $N$ nodes and adjacency matrix $A \in \mathbb{R}^{N\times N}$ is given as follows:

$$\partial_t x(t) = Ax(t) + Bu(t) \tag{1}$$

where $x(t) \in \mathbb{R}^N$ is network's state at time $t$, $B \in \mathbb{R}^{N\times K}$ is the input matrix, and $u(t) \in \mathbb{R}^K$ is the control input. The columns of $B$ are given by $K$ different unit vectors and describe the fact that $K$ nodes in the network are *driver nodes*, each with its own distinct control input [4]; the remaining $N - K$ nodes, instead, are not controlled. We assume that $u(t)$ is bounded, which can thus take into account constraints for physical realizability [10], [12], [15].

In this context, the synthesis of a control law can be phrased as a problem of minimizing a cost function using $u(t)$ as the set of decision variables and (1) as the constraints. We consider a rather general cost function in the following:

$$J(x[0], u, T) = F(x(T)) + \int_0^T [S(t, x(t)) + Q(t, u(t))]\, dt \tag{2}$$

where the function $F$, called the final cost, may be used to steer the system toward a given target state at final time $T$; the integrand $R = S + Q$ is called the running cost. Note that, the above integral is computed along the trajectory $x(t)$ determined by the initial condition $x[0]$ and the control input $u$. This covers the classical setting where only the energy of the control is to be minimized [1]; the term $S$ allows for objectives where one is interested in driving the system toward a desired trajectory [1], possibly in a targeted fashion where only a subset of the state space, and not full control, is of interest, for instance [6] and [16].

Our method partitions the nodes of a network into $n \leq N$ macronodes and builds a coarse-grained adjacency matrix $\hat{A} \in \mathbb{R}^{n\times n}$ and input matrix $\hat{B} \in \mathbb{R}^{n\times k}$, where $k \leq K$ is the number of macroinputs. The state of the coarse-grained network $\hat{x}(t) \in \mathbb{R}^n$ is the solution of the following system:

$$\partial_t \hat{x}(t) = \hat{A}\hat{x}(t) + \hat{B}\hat{u}(t) \tag{3}$$

and $\hat{u}(t) \in \mathbb{R}^k$ is the control macroinput. The partition is such that each macrostate of the coarse-grained network exactly preserves the sum of the values of the corresponding states in the original network whenever

Received 30 May 2024; accepted 18 August 2024. Date of publication 22 August 2024; date of current version 30 January 2025. This work was supported in part by the Poul Due Jensen Foundation under Grant 883901, in part by the Villum Investigator Grant under Grant S4OS, in part by the project SERICS under Grant PE00000014, in part by the Fsc regional Tuscan project under Grant AUTOXAI2 J53D21003810008, in part by the project Tuscany Health Ecosystem (THE), CUP: under Grant B83C22003920001, and in part by the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU. Recommended by Associate Editor F. Pasqualetti. *(Corresponding author: Max Tschaikowski.)*

Daniele Toller and Max Tschaikowski are with the Aalborg University, 9400 Aalborg, Denmark (e-mail: tschaikowski@cs.aau.dk).

Mirco Tribastone is with IMT, 55100 Lucca, Italy.

Andrea Vandin is with Sant'Anna, 56127 Pisa, Italy, and also with DTU Compute, 2800 Kongens Lyngby, Denmark.

Digital Object Identifier 10.1109/TAC.2024.3448240

each macroinput is equal to the sum of the corresponding original input values.

Here we introduce *control equivalence*, a property that guarantees the preservation of optimal costs. For this, we assume that the cost functional in (2) can be equivalently rewritten in terms of a cost functional $\hat{J}(\hat{x}, \hat{u}, T)$ that depends on the coarse-grained model. Under these conditions, the solution of the control problem for minimizing $\hat{J}(\hat{x}, \hat{u}, T)$ with the constraints given from (3) is such that one can recover a mapping from the optimal macroinputs $\hat{u}^*(t)$ to optimal original inputs $u^*(t)$ that give rise to the optimal trajectory in (1). With this, the analysis of the original control problem can be entirely circumvented. Control equivalence can be computed using the partition-refinement algorithm defined in [17]. The algorithm provides the reduction as the coarsest partition that refines an initial candidate partition, iteratively splitting its blocks until certain criteria are met [17], [18]. As a straightforward consequence of this fact, from the computational complexity analysis in [17] we obtain that control equivalence can be computed efficiently in $O(E \log N)$ time, where $E$ is the number of nonzero entries in $A$. The initial partition may be arbitrarily chosen. Similarly to the other applications of partition refinement [19], [20], [21], [22], this freedom permits certain state variables to be kept observable in the coarse-grained model. For example, let us assume that the original cost function $J(x, u, T)$ only concerns a given state component $x_i$; then, one can ensure by construction that the coarse-grained cost function $\hat{J}(\hat{x}, \hat{u}, T)$ can be written in terms of $x_i$ by placing it in an initial singleton block for the partition-refinement algorithm.

*Results:* Our main technical contribution is a characterization result. Here, we prove that the reduction provided by our reduction algorithm is maximal in the sense that there exists no coarser refinement of a given initial partition, which yields an optimality-preserving reduction. In addition, to the technical result, we conduct a large-scale evaluation of our framework on networks from public repositories. Here, the main finding is that networks that are controlled by a minimal set of driver nodes, as computed through [4], often allow for further substantial optimality-preserving reductions.

*Notation:* We denote by $N$ and $K$, respectively, the number of nodes and driver nodes in the original network; similarly, $n \leq N$ and $k \leq K$ have the respective meaning in the reduced network. The respective dynamics of the original and the reduced networks are described by (1) and (3), where $x, \hat{x}$ are state vectors, while $u, \hat{u}$ are control vectors. Letters $i, j$ denote node indices, $l, l'$ refer to control inputs, $\mathcal{H}, \mathcal{H}'$ refer to partitions of $\{1, \ldots, N\}$, while $H, H'$ to partition blocks. A partition $\mathcal{H}$ is said to refine a partition $\mathcal{H}'$ if every block $H' \in \mathcal{H}'$ can be expressed in terms of a disjoint union of blocks from $\mathcal{H}$. Note that $\mathcal{H}$ is a trivial refinement of itself. Likewise, $\{\{1, \ldots, N\}\}$ and $\{\{i\} \mid 1 \leq i \leq N\}$ are considered to be trivial partitions because the former aggregates all nodes to a single one, while the latter does not aggregate anything.

## II. Related Work

The preservation of *uncontrolled* dynamics up to sums of variables has been addressed by the notion of forward equivalence [17] for linear and nonlinear systems of ordinary differential equations (ODEs). By extending the approach from [17] to controlled systems, the present work can be related to the aggregation of control systems from late 1960s [23]; while [23] relates to control equivalence, a method for computing is not provided.

More closely related is the bisimulation/abstraction of control systems [24], [25], [26] where the largest bisimulation gives rise to the smallest reduced control system, which coincides with the original one up to a given observation map. The main similarity with the bisimulation approaches is that control equivalence computes such an observation map, for which, in particular, the reduced system can be shown to be a so-called *consistent implementable abstraction* of the original one [27].

To our knowledge, a (nonunique dimension-based) computation of observation maps has been addressed in the case with unbounded control domains [27]. Being described by systems of linear equations, it relies on matrix transformations and has, thus, a cubic time complexity in the number of nodes. Control equivalence, instead, is described by structural properties [28], [29] that are used in the computation of graph isomorphisms [30, Definition 5]. Hence, while considering a smaller class of observation maps than bisimulation, control equivalence avoids matrix transformations and enjoys a quasilinear time complexity in the number of edges; moreover, it considers bounded control domains.

A less closely related work is that of balanced truncation and its extensions, see for instance [31, Chapter 9] and [32]. This is because the respective reductions are not exact and the control domains are unbounded; in addition, they have cubic time complexity [31]. Bisimulation approaches which have been also studied for nonlinear systems [24], [25], [33], [34], are, however, computationally more demanding than the respective linear counterparts.

The present work also relates too approaches from network science. We mention in particular spectral coarse graining [5], which merges nodes with the same or similar eigenvector components and that was shown to be effective for maintaining the synchronizability by simulation experiments. From control engineering perspective, synchronizability can be tied to forward invariance which is complementary to control equivalence and bisimulation [35]. Less closely related are topology- and degree-based graining approaches, see [36], [37], and [38] and references therein.

## III. Control Equivalence

For two vectors $m \in \mathbb{R}^K$ and $M \in \mathbb{R}^K$ with respective $l$th entries $m_l \leq M_l$ for every $l = 1, \ldots, K$, consider the real interval $[m_l; M_l] \subseteq \mathbb{R}$ and the $K$-dimensional cube

$$[m; M] = \prod_{l=1}^{K} [m_l; M_l] \subseteq \mathbb{R}^K.$$

We consider control inputs $u(\cdot) \in [m; M]$, meaning that $u : \mathbb{R} \to [m; M]$ is a measurable function with range $[m; M]$.

*Definition 1 (Optimal costs):* Given a matrix $A \in \mathbb{R}^{N \times N}$ and a measurable control input $u(\cdot) \in [m; M]$, we denote the value functional at time $T \in \mathbb{R}$ when starting at $x[0] \in \mathbb{R}^N$ as follows:

$$J_u(x[0], T) = F\left(x^u(T)\right) + \int_0^T R\left(t, x^u(t), u(t)\right) dt$$

where $x^u$ is the solution of (1), with initial condition $x[0]$, while $F$ is the final cost function, and $R = S + Q$ is the running cost function. With this, the optimal costs at time $T$ when starting at $x[0] \in \mathbb{R}^N$ are given by the following value functions:

$$V^{\inf}(x[0], T) = \inf\{J_u(x[0], T) \mid u : \mathbb{R} \to [m; M]\}$$
$$V^{\sup}(x[0], T) = \sup\{J_u(x[0], T) \mid u : \mathbb{R} \to [m; M]\}$$

where the control inputs $u(\cdot)$ are measurable.

In the following, we shall use the simple network provided in Fig. 1 to explain the main concepts.

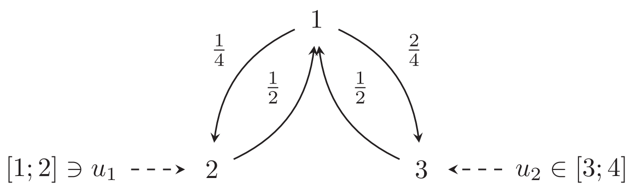


Fig. 1. Simple controlled network which will be used as a running example. Solid lines describe entries of the adjacency matrix, whereas dashed lines signify control inputs.

*Example, Part 1:* Equation (1) of the network from Fig. 1 is as follows:

$$\begin{pmatrix}\partial_t x_1\\ \partial_t x_2\\ \partial_t x_3\end{pmatrix} = \underbrace{\begin{pmatrix}0 & \frac{1}{2} & \frac{1}{2}\\ \frac{1}{4} & 0 & 0\\ \frac{1}{2} & 0 & 0\end{pmatrix}}_{A=} \begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix} + \underbrace{\begin{pmatrix}0 & 0\\ 1 & 0\\ 0 & 1\end{pmatrix}}_{B=} \begin{pmatrix}u_1\\ u_2\end{pmatrix}.$$

As next, we introduce control equivalence. For a set $H$, we denote its cardinality by $|H|$.

*Definition 2 (Control Equivalence):* Consider the system of ODEs in (1). A control equivalence of this system is a partition $\mathcal{H}$ whose aggregation matrix $L = L(\mathcal{H})$ and the corresponding right inverse $\bar{L} = \bar{L}(\mathcal{H})$ satisfy $LA = LA\bar{L}L$. Specifically, $L \in \mathbb{R}^{n\times N}$ and $\bar{L} \in \mathbb{R}^{N\times n}$ are such that

1) The $i$th row of $L$ encodes the $i$th block of $\mathcal{H}$ by setting $L_{i,j} = 1$ if $j \in H_i$ and zero otherwise;
2) The $i$th column of $\bar{L}$ encodes the $i$th block of $\mathcal{H}$ by setting $\bar{L}_{j,i} = 1/|H_i|$ if $j \in H_i$ and zero otherwise.

When the partition $\mathcal{H}$ is clear from the context, we omit the dependence on $\mathcal{H}$ in $L(\mathcal{H})$ and $\bar{L}(\mathcal{H})$.

*Example, Part 2:* In the case of the network from Fig. 1, it is easy to see that the trivial partition $\{\{1,2,3\}\}$ is not a control equivalence, which corresponds to the impossibility of aggregating all nodes together. On the other hand, the partition $\mathcal{H} = \{H_1, H_2\} = \{\{2,3\},\{1\}\}$ is a control equivalence (intuitively: it is possible to aggregate the nodes 2 and 3). This can be seen by checking that the matrices

$$L = \begin{pmatrix}0 & 1 & 1\\ 1 & 0 & 0\end{pmatrix} \qquad \text{and} \qquad \bar{L} = \begin{pmatrix}0 & 1\\ \frac{1}{2} & 0\\ \frac{1}{2} & 0\end{pmatrix}$$

satisfy $LA = LA\bar{L}L$.

Given a partition $\mathcal{H} = \{H_1, \ldots, H_n\}$ of nodes $\{1,\ldots,N\}$, we always assume without loss of generality that its first $k$ blocks $H_1,\ldots,H_k$ contain all at least one driver node, while the remaining blocks $H_{k+1},\ldots,H_n$ have no driver nodes.

*Definition 3 (Reduced System):* Consider the system of ODEs in (1). Then, the reduced system underlying a partition $\mathcal{H}$ is given by (3) where

1) $\hat{A} = LA\bar{L}$;
2) For any $l \le k$, the $l$th column of $\hat{B} \in \mathbb{R}^{n\times k}$ is the $l$th unit vector in $\mathbb{R}^n$;
3) For any $l \le k$, the $l$th control satisfies $\hat{u}_l(\cdot) \in [\hat{m}_l; \hat{M}_l]$ with

$$\hat{m}_l = \sum_{l'\in\mathcal{K}(H_l)} m_{l'} \qquad \text{and} \qquad \hat{M}_l = \sum_{l'\in\mathcal{K}(H_l)} M_{l'}$$

where $\mathcal{K}(H_l) \subseteq \{1,\ldots,K\}$ are the control indices that steer the driver nodes contained in block $H_l$.

Note that the sets $\mathcal{K}(H_1), \mathcal{K}(H_2), \ldots$ form a partition of $\{1,\ldots,K\}$ (i.e., of the driver nodes). We used this fact to well-define the bounds $\hat{m}$ and $\hat{M}$ for the macroinputs.

*Example, Part 3:* Continuing the example, we note that block $H_1 = \{2,3\}$ has two driver nodes, while block $H_2 = \{1\}$ has none. With this, we obtain the following:

$$\begin{pmatrix}\partial_t \hat{x}_1\\ \partial_t \hat{x}_2\end{pmatrix} = \underbrace{\begin{pmatrix}0 & \frac{3}{4}\\ \frac{1}{2} & 0\end{pmatrix}}_{\hat{A}=} \begin{pmatrix}\hat{x}_1\\ \hat{x}_2\end{pmatrix} + \underbrace{\begin{pmatrix}1\\ 0\end{pmatrix}}_{\hat{B}=} \begin{pmatrix}\hat{u}_1\end{pmatrix}$$

where $\hat{u}_1(\cdot) = u_1(\cdot) + u_2(\cdot) \in [4;6]$, while $\hat{x}_1 = x_2 + x_3$, and $\hat{x}_2 = x_1$. We also note that $\hat{x}_2 = x_1$ is not a driver node, reflecting the fact that $H_2 = \{1\}$ has no driver nodes.

In order to associate a cost to a reduced system, the final and running costs have to satisfy an additional property. More specifically, for a partition of nodes $\mathcal{H}$, we call the final cost $F$ and running cost $R$ *constant on* $\mathcal{H}$ if $F(x) = F(y)$ and $R(t,x,u) = R(t,y,v)$ for all $x, y \in \mathbb{R}^N$, and $u, v \in \mathbb{R}^K$ that satisfy $Lx = Ly$ and $LBu = LBv$, for $L = L(\mathcal{H})$. In such a case, we study the well-defined reduced costs $\hat{R}$ and $\hat{F}$ for the reduced system defined for $\hat{x} \in \mathbb{R}^n$ and $\hat{u} \in [\hat{m}; \hat{M}]$ via the following:

$$\hat{R}(t,\hat{x},\hat{u}) := R(t,x,u) \qquad \text{and} \qquad \hat{F}(\hat{x}) := F(x)$$

for arbitrary $x$ and $u$ such that $\hat{x} = Lx$ and $\hat{B}\hat{u} = LBu$ (equivalently, $\hat{x}_i = \sum_{j\in H_i} x_j$ and $\hat{u}_l = \sum_{l'\in\mathcal{K}(H_l)} u_{l'}$ for all $1 \le i \le n$ and $1 \le l \le k$).

*Example, Part 4:* We continue the example by picking as running cost and final cost, respectively, $R = 0$ and $F(x) = x_2 + x_3$. Then, $R$ and $F$ are constant on $\mathcal{H} = \{\{2,3\},\{1\}\}$ and we obtain $\hat{R}(\hat{x}) = 0$ and $\hat{F}(\hat{x}) = \hat{x}_1$. With these cost functions, the value functionals have the following form:

$$J_u(x[0],T) = F(x^u(T)) = x_2^u(T) + x_3^u(T).$$

*Definition 4 (Optimal cost of Reduced System):* Let $\mathcal{H}$ be an arbitrary partition, and assume that $F$ and $R$ are, respectively, final and running costs that are constant on $\mathcal{H}$. Then, for a control input $\hat{u}(\cdot) \in [\hat{m}; \hat{M}]$, the value functional at time $T \in \mathbb{R}$ when starting at $\hat{x}[0] \in \mathbb{R}^n$ is given as follows:

$$\hat{J}_{\hat{u}}(\hat{x}[0],T) = \hat{F}\left(\hat{x}^{\hat{u}}(T)\right) + \int_0^T \hat{R}\left(t, \hat{x}^{\hat{u}}(t), \hat{u}(t)\right) dt$$

where $\hat{x}^{\hat{u}}$ is the solution of (3), with initial condition $\hat{x}[0]$. With this, the optimal costs of the reduced system (3) at time $T$ when starting at $\hat{x}[0] \in \mathbb{R}^n$ are given by the reduced value functions as follows:

$$\hat{V}^{\inf}(\hat{x}[0],T) = \inf\{\hat{J}_{\hat{u}}(\hat{x}[0],T) \mid \hat{u} : \mathbb{R} \to [b; \hat{M}]\}$$

$$\hat{V}^{\sup}(\hat{x}[0],T) = \sup\{\hat{J}_{\hat{u}}(\hat{x}[0],T) \mid \hat{u} : \mathbb{R} \to [b; \hat{M}]\}$$

where the control inputs $\hat{u}(\cdot)$ are measurable.

We are now in a position to state our first result, which is key in proving that the original and the reduced control system admit identical optimal costs if and only if $\mathcal{H}$ is a control equivalence.

*Theorem 1 (Optimality-Preservation):* Let $\mathcal{H}$ be a control equivalence of the system (1) with final and running costs that are constant on $\mathcal{H}$.

1) For any control of the original system $u : \mathbb{R} \to [m; M]$, there exists a control of the reduced system $\hat{u} : \mathbb{R} \to [\hat{m}; \hat{M}]$ that satisfies the following:

$$J_u(x[0],T) = \hat{J}_{\hat{u}}(\hat{x}[0],T)$$

for every $T \in \mathbb{R}$ and every $x[0] \in \mathbb{R}^N$, where $\hat{x}[0] \in \mathbb{R}^n$ is given by $\hat{x}[0] = Lx[0]$.

2) Conversely, for any control of the reduced system $\hat{u} : \mathbb{R} \to [\hat{m}; \hat{M}]$, there exists a control of the original system $u : \mathbb{R} \to [m; M]$ that satisfies the following:

$$J_u(x[0], T) = \hat{J}_{\hat{u}}(\hat{x}[0], T)$$

for every $T \in \mathbb{R}$ and every $x[0] \in \mathbb{R}^N$, where $\hat{x}[0] \in \mathbb{R}^n$ is given by $\hat{x}[0] = Lx[0]$. In addition, the control $u$ can be chosen as follows:

$$u_{l'}(\cdot) = \begin{cases} m_{l'} + \dfrac{M_{l'} - m_{l'}}{\hat{M}_l - \hat{m}_l}(\hat{u}_l(\cdot) - \hat{m}_l), & \text{if } \hat{m}_l < \hat{M}_l \\ m_{l'} = M_{l'}, & \text{otherwise} \end{cases}$$

for every $l \in \{1, \ldots, k\}$ and $l' \in \mathcal{K}(H_l)$.

Armed with Theorem 1, we infer our main technical result.

*Theorem 2 (Characterization of Optimality-Preservation):* Consider the system in (1), and let $\mathcal{H}$ be a partition of the nodes. Then, $\mathcal{H}$ is a control equivalence if and only if

$$V^{\inf}(x[0], T) = \hat{V}^{\inf}(\hat{x}[0], T)$$

$$V^{\sup}(x[0], T) = \hat{V}^{\sup}(\hat{x}[0], T)$$

for all final and running costs that are constant on $\mathcal{H}$, for every $T \in \mathbb{R}$, and every $x[0] \in \mathbb{R}^N$, where $\hat{x}[0] \in \mathbb{R}^n$ is determined by $\hat{x}[0] = Lx[0]$.

*Example, Part 5:* In the case of our example, and with the cost functions as in Part 4, we obtain the following:

1) For any $u_1(\cdot) \in [1; 2]$ and $u_2(\cdot) \in [3; 4]$, there exists $\hat{u}(\cdot) \in [4; 6]$ such that for any $x[0] \in \mathbb{R}^3$ and $\hat{x}[0] = Lx[0]$, for every $T$ we have that

$$x_2^u(T) + x_3^u(T) = \hat{x}_1^{\hat{u}}(T).$$

2) For any $\hat{u}_1(\cdot) \in [4; 6]$, there exist $u_1(\cdot) \in [1; 2]$ and $u_2(\cdot) \in [3; 4]$ such that for any $\hat{x}[0] \in \mathbb{R}^2$ and $x[0] \in \mathbb{R}^3$ with $\hat{x}[0] = Lx[0]$, we have that

$$x_2^u(T) + x_3^u(T) = \hat{x}_1^{\hat{u}}(T).$$

Specifically, we can pick

$$u_1(t) = 1 + \tfrac{1}{2}(\hat{u}_1(t) - 4), \qquad u_2(t) = 3 + \tfrac{1}{2}(\hat{u}_1(t) - 4).$$

We end the section by stating that control equivalence can be computed in quasilinear time in the number of edges. As the trivial partition made of singletons is a control equivalence, the refining procedure in the next theorem is guaranteed to terminate.

*Theorem 3 (Computation of Control Equivalence):* Consider system (1) and let $\mathcal{H}'$ be a partition of the nodes. Then, the coarsest control equivalence refining $\mathcal{H}'$, denoted by $\mathcal{H}$, and the underlying reduced system $\partial_t \hat{x}(t) = \hat{A}\hat{x}(t) + \hat{B}\hat{u}(t)$, can be computed in at most $O(E \log N)$ steps, where $E$ is the number of nonzero entries of $A \in \mathbb{R}^{N \times N}$.

*Example, Part 6:* If we apply the partition refinement from [17] on the partition $\mathcal{H}' = \{\{1, 2, 3\}\}$, the algorithm will refine the block $\{1, 2, 3\}$ to the blocks $\{1\}, \{2, 3\}$, providing us with the previously seen partition $\mathcal{H} = \{\{1\}, \{2, 3\}\}$.

We remark that the partition $\mathcal{H}'$ in Theorem 3 can be used to steer the search for control equivalences. Indeed, if one is interested in finding a reduction, which does not aggregate node 1 with any other node, one could pick $\mathcal{H}' = \{\{1\}, \{2, 3\}\}$. With this choice, the partition refinement algorithm either splits the block $\{2, 3\}$ into $\{2\}, \{3\}$ or not. From the foregoing discussion, we know that the latter will be the case because $\{\{1\}, \{2, 3\}\}$ is a control equivalence. On the contrary, if one uses $\mathcal{H}' = \{\{1, 2\}, \{3\}\}$ (say, because one wants to obtain a reduced model, which does not aggregate node $x_3$ with any other node), then, the algorithm would split the block $\{1, 2\}$ and return the (trivial) control equivalence $\{\{1\}, \{2\}, \{3\}\}$, which does not aggregate any node.

## IV. Evaluation

The interpretation of network dynamics (1) depends on the domain from which network $A$ stems from. For instance, in case of knowledge dynamics [43, Section 7.3], $x_i$ measures the monotonically increasing knowledge of agent $i$, whereas $a_{i,j}$ measures the knowledge transfer rate from agent $i$ to agent $j$. Assuming that the gross return of agent $i$ is proportional to her knowledge growth rate, one then studies the so-called utility function $\frac{\partial_t x_i}{x_i} - d_i$, where $d_i$ is the degree of agent $i$. As for other examples, we mention infection and social influence models, see [43] for more details.

Following [4], we abstract away from the actual interpretation, seeking to evaluate the potential of control equivalence by computing it for different networks from the literature [4], [39], and [40]. The respective domains include biology, ecology, engineering, online social networks, and the web. Their sizes range from 47 to 82 169 nodes. The networks show a wide spectrum of controllability ranging from 1.28% to 99.59% of driver nodes in the network as provided by the algorithm from [4], which yields a minimal set of driver nodes for network controllability. We considered initial partitions that separated the set of driver nodes from the rest of the network. In this way one can identify the macronodes in the coarse-grained network that represent original driver nodes. As shown in column *Red.*, computing control equivalences took often a few milliseconds, with up to 3.6 s in one case (*Slashdot*). Only the two intraorganizational networks were weighted, while the others, including all networks from [4], were not. For consistency, and given that the algorithm for driver node computation from [4] considers only unweighted networks, we replaced nonzero weights with weight 1 for the two intraorganizational networks.

Overall, these benchmark networks show a full range of reducibility by control equivalence measured as $\rho = n/N$, i.e., the ratio between the number $n$ of macronodes and the number $N$ of original nodes. This ranges from 0.05 in the case of regulatory networks, to 0.992 in a road network, or no reduction in an intraorganizational one (Table I). There is a high correlation (0.87) between the overall reduction ($\rho$) and the reduction of driver nodes, indicated by $\rho_D = k/K$, i.e., the percentage ratio between the number $k$ of macro driver nodes and the number $K$ of original driver nodes. The former is obtained via Definition 3 and by initializing the reduction algorithm from Theorem 3 with an initial partition containing two blocks, one containing all driver nodes and its complement (as mentioned earlier, the set of driver nodes is computed using [4] and is minimal in size). With this in place, we note that in some networks, very small values of $k$ were found (as $k < 5$), where $K$ is between one and three orders of magnitude larger. This suggests a considerably more effective computation of the control law on the coarse-grained network, owing to the much smaller size of decision variables involved. Interestingly, the best reduction in the number of driver nodes, i.e., $\rho_D < 0.10$, occur in networks where the original density of driver nodes $K/N$ is considerable, i.e., over 0.40.

Metabolic and regulatory networks have been previously highlighted to be difficult to control because of the large ratios $K/N$, as occurs in the models of yeast and E. coli (*type* Metabolic in Table I) [4]. Interestingly, here we find that for some of such models the coarse-grained networks turned out to be considerably more controllable. In particular the E. coli coarse-grained networks only require two driver macronodes instead of more than 300 nodes in the original networks. WWW networks exhibit controllability ratios ($K/N$) between 0.10 and 0.76. We note that the corresponding coarse-grained networks tend to have lower controllability ratios ($k/n$). For example, wikipedia_link_bat_smg goes from an orginal controllability ratio of 0.64, to 0.13 in the coarse grained network.

TABLE I
REDUCTIONS OF PUBLISHED NETWORKS

| *Name* | *Type* | $N$ | $n$ | $n/N$ | $K$ | $k$ | $k/K$ | $K/N$ | *Red.(ms)* |
|---|---|---|---|---|---|---|---|---|---|
| Kohonen | Citation | 3773 | 2657 | 70.42 | 2114 | 1132 | 53.55 | 56.03 | 53 |
| subelj_cora | Citation | 23167 | 9243 | 39.90 | 10210 | 281 | 2.75 | 44.07 | 333 |
| CitHepTh | Citation | 27771 | 19780 | 71.23 | 5994 | 650 | 10.98 | 21.58 | 1139 |
| CitHepPh | Citation | 34547 | 24944 | 72.20 | 8030 | 819 | 10.20 | 23.24 | 1780 |
| s208st | Electronic Circuits | 123 | 102 | 82.93 | 29 | 13 | 44.83 | 23.58 | 4 |
| s420st | Electronic Circuits | 253 | 213 | 84.19 | 59 | 28 | 47.46 | 23.32 | 3 |
| s838st | Electronic Circuits | 513 | 437 | 85.19 | 119 | 60 | 50.42 | 23.20 | 5 |
| seagrass | Food Web | 50 | 43 | 86.00 | 13 | 7 | 53.85 | 26.00 | 6 |
| ythan | Food Web | 136 | 92 | 67.65 | 69 | 31 | 44.93 | 50.74 | 3 |
| grassland | Food Web | 89 | 30 | 33.71 | 46 | 10 | 21.74 | 51.69 | 8 |
| littlerock | Food Web | 184 | 54 | 29.35 | 99 | 4 | 4.04 | 53.80 | 82 |
| maayan-faa | Infrastructure | 1227 | 1049 | 85.49 | 363 | 224 | 61.71 | 29.58 | 10 |
| opsahl-open-flights | Infrastructure | 2940 | 2433 | 82.76 | 872 | 479 | 54.93 | 29.66 | 65 |
| US-powergrid-4941 | Infrastructure | 4942 | 4684 | 94.78 | 575 | 448 | 77.91 | 11.63 | 56 |
| tntp-Chicago-Regional | Infrastructure | 12980 | 12969 | 99.92 | 215 | 214 | 99.53 | 1.66 | 135 |
| p2p-Gnutella08 | Internet | 6302 | 5256 | 83.4 | 4106 | 3107 | 75.67 | 65.15 | 83 |
| p2p-Gnutella06 | Internet | 8718 | 7738 | 88.65 | 5033 | 4079 | 81.04 | 57.73 | 141 |
| p2p-Gnutella05 | Internet | 8847 | 7801 | 88.18 | 5111 | 4106 | 80.33 | 57.77 | 130 |
| p2p-Gnutella04 | Internet | 10877 | 9871 | 90.75 | 6004 | 5023 | 83.66 | 55.20 | 162 |
| p2p-Gnutella-24 | Internet | 26519 | 20238 | 76.32 | 18965 | 12860 | 67.81 | 71.51 | 350 |
| Consulting | Intra-Organizational | 47 | 47 | 100.00 | 2 | 2 | 100.00 | 4.26 | 10 |
| Manufacturing | Intra-Organizational | 78 | 74 | 94.87 | 1 | 1 | 100.00 | 1.28 | 7 |
| rhesusbrain1 | Brain Connections | 243 | 233 | 95.88 | 23 | 14 | 60.87 | 99.59 | 12 |
| celegans-neuronal | Neuronal | 298 | 265 | 88.93 | 49 | 20 | 40.82 | 16.44 | 102 |
| coliInter-NoAutoReg | Metabolic | 420 | 40 | 9.52 | 314 | 2 | 0.64 | 74.76 | 3 |
| coliInter-full | Metabolic | 425 | 49 | 11.53 | 309 | 2 | 0.65 | 72.71 | 4 |
| TRNYeast2-Costanzo | Regulatory | 689 | 95 | 13.79 | 565 | 43 | 7.61 | 82.00 | 7 |
| maayan-Stelzl | Protein-Protein | 1707 | 1282 | 75.10 | 765 | 425 | 55.50 | 44.82 | 19 |
| TRNYeast1-Balaji | Regulatory | 4441 | 1983 | 44.65 | 4282 | 1840 | 42.97 | 96.42 | 57 |
| yeast2017-full | Regulatory | 6854 | 6647 | 96.98 | 6648 | 6441 | 96.88 | 96.99 | 318 |
| yeast2019-full | Regulatory | 6887 | 6705 | 97.36 | 6670 | 6488 | 97.27 | 96.85 | 412 |
| ownership | Regulatory | 7254 | 361 | 4.98 | 5950 | 157 | 2.64 | 82.02 | 43 |
| moreno-innovation | Social | 242 | 214 | 88.43 | 29 | 8 | 27.59 | 11.98 | 4 |
| CSphd | Social | 1883 | 247 | 13.12 | 1176 | 100 | 8.50 | 62.45 | 12 |
| ego-twitter | Social | 23371 | 2620 | 11.64 | 22432 | 2221 | 9.90 | 95.98 | 299 |
| ego-gplus | Social | 23629 | 2442 | 10.33 | 23497 | 2339 | 9.95 | 99.44 | 312 |
| Epinions | Social | 75880 | 37630 | 49.59 | 41627 | 7398 | 17.90 | 54.86 | 1555 |
| Slashdot | Social | 82169 | 65444 | 79.65 | 3737 | 3689 | 98.72 | 4.55 | 3650 |
| linux | Source Code | 30838 | 3767 | 12.22 | 20049 | 261 | 1.30 | 65.01 | 534 |
| libre-film-trust | Trust | 875 | 418 | 47.77 | 359 | 93 | 25.91 | 41.03 | 10 |
| WikiVote | Trust | 7116 | 2321 | 32.62 | 4736 | 3 | 0.06 | 66.55 | 167 |
| librec-ciao-dvd | Trust | 4659 | 3064 | 65.77 | 3247 | 1758 | 54.14 | 69.69 | 83 |
| moreno_blogs | WWW | 1225 | 887 | 72.41 | 436 | 124 | 28.44 | 35.59 | 45 |
| dimacs10_polblogs | WWW | 1225 | 1180 | 96.33 | 126 | 85 | 67.46 | 10.29 | 67 |
| wikipedia_link_gag | WWW | 2930 | 1101 | 37.58 | 1185 | 58 | 4.89 | 40.44 | 290 |
| EPA | WWW | 4272 | 682 | 15.96 | 3285 | 156 | 4.75 | 76.90 | 37 |
| wikipedia_link_csb | WWW | 5562 | 2264 | 40.70 | 2971 | 226 | 7.61 | 53.42 | 475 |
| California | WWW | 6176 | 1052 | 17.03 | 4489 | 99 | 2.21 | 72.68 | 40 |
| wikipedia_link_mi | WWW | 7997 | 2180 | 27.26 | 4825 | 149 | 3.09 | 60.34 | 278 |
| wbcsstanford | WWW | 9436 | 5305 | 56.22 | 3653 | 796 | 21.79 | 38.71 | 130 |
| cfinder_google | WWW | 15764 | 8659 | 54.93 | 5313 | 1488 | 28.01 | 33.70 | 472 |
| wikipedia_link_bat_smg | WWW | 21901 | 6842 | 31.24 | 14047 | 902 | 6.42 | 64.14 | 413 |

We take all networks from [4] with at most 100 000 nodes, and expand them with a selection of networks from well-known public repositories like Konect [41] and the Sparse Matrix Collection [40]. Brain Connections has been taken from [42], while the two Regulatory yeast models were taken from [43]. and $N$ and $n$ is, respectively, the size of the original and reduced network. $K$ is the minimal number of driver nodes of the original network, computed using [4], while $k$ is the number of driver nodes of the reduced network obtained via Theorem 3 and Definition 3 The runtime for the computation of control equivalences is reported in *Red*. We used an 8-cores Apple M1 Pro 2021. All reductions are optimality-preserving by Theorem 1 and 2.

## V. CONCLUSION

In this work, we have introduced control equivalence, an optimality-preserving model reduction technique for linear control systems, which can be computed in quasilinear time complexity in the number of nonzero matrix entries. We have conducted a large-scale evaluation on networks from public repositories, showing in particular that networks that are controlled by a minimal set of driver nodes often allow for further substantial optimality-preserving reductions. Future work will consider approximate notions, which are stable to noise in parameters. To this end, we will exploit the fact that control equivalence is characterized by the same algebraic conditions as forward equivalence for which an approximate extension has been proposed [34], [44]. Moreover, extensions to nonlinear control systems will be studied.

## Appendix
## Proofs

*Theorem 1:* Let $L = L(\mathcal{H})$ be the aggregation matrix of the control equivalence $\mathcal{H}$. As in Definition 3, we assume without loss of generality that $H_1, \ldots, H_k$ contain all at least one driver node, while $H_{k+1}, \ldots, H_n$ have no driver nodes.

For the proof of 1), pick any $u(\cdot) \in [m; M]$ and note that

$$\begin{aligned}\partial_t Lx(t) &= L\left(Ax(t) + Bu(t)\right) = LA\bar{L}Lx(t) + LBu(t)\\ &= \hat{A}Lx(t) + LBu(t) = \hat{A}\hat{x}(t) + \hat{B}\hat{u}(t)\\ &= \partial_t \hat{x}(t)\end{aligned}$$

where the second identity holds because $\mathcal{H}$ is a control equivalence, while the third, and the last identity follow from the definition of the reduced model. Since the costs $F$ and $R$ are constant on $\mathcal{H}$, the above calculation yields 1).

Part 2) follows by picking any $\hat{u}(\cdot) \in [\hat{m}; \hat{M}]$, define $u(\cdot)$ using the formula in the part 2) and by reading the above calculation backward. Specifically, the construction of $u(\cdot)$ from $\hat{u}(t)$ ensures that $\hat{B}\hat{u}(t) = LBu(t)$. □

*Theorem 2:* In the case $\mathcal{H}$ is a control equivalence, Theorem 1 readily implies that the reduction is optimality-preserving. To show the converse, let us assume that the reduced model underlying a given partition $\mathcal{H}$ is optimality-preserving for any final and running cost, which are constant on $\mathcal{H}$. Assume without loss of generality that $H_1, \ldots, H_k$ contain all at least one driver node, while $H_{k+1}, \ldots, H_n$ have no driver nodes. We, then, fix arbitrary $T \geq 0$ and $u(\cdot) \in [m; M]$. Write $\|\cdot\|$ for the Euclidean norm and fix the final cost $F = 0$ that is obviously constant on $\mathcal{H}$ and yields $\hat{F} = 0$. Denoting by $x^u$ the solution of the original system induced by $u$, we consider the running cost as follows:

$$R(t, x, v) = \|Lx - Lx^u(t)\| + \|LBv - LBu(t)\|$$

for any $v \in [m; M]$. It is immediate to see that $R$ is constant on $\mathcal{H}$. As $R(t, x^u(t), u(t)) = 0$ for every $t \geq 0$, we have that

$$0 = \int_0^T R\left(t, x^u(t), u(t)\right) dt = J_u(x[0], T).$$

Moreover, $J_{u'}(x[0], T) \geq 0$ for any $u'(\cdot) \in [m; M]$ as $R$ is nonnegative, and so

$$0 \leq V^{\inf}(x[0], T) \leq J_u(x[0], T) = 0.$$

By assumption, $\hat{V}^{\inf}(\hat{x}[0], T) = V^{\inf}(x[0], T) = 0$. Since the optimal control is measurable by Filippov's theorem [45, Section 4.5], there thus exists $\hat{u}(\cdot) \in [\hat{m}; \hat{M}]$ such that

$$\hat{J}_{\hat{u}}(\hat{x}[0], T) = \hat{V}^{\inf}(\hat{x}[0], T) = 0.$$

Writing explicitly the reduced value functional, the above equation gives the following:

$$\begin{aligned}0 &= \hat{J}_{\hat{u}}(\hat{x}[0], T)\\ &= \int_0^T \left(\|\hat{x}^{\hat{u}}(t) - Lx^u(t)\| + \|\hat{B}\hat{u}(t) - LBu(t)\|\right) dt\end{aligned}$$

where $\hat{x}^{\hat{u}}(t)$ solves $\partial_t \hat{x}^{\hat{u}}(t) = \hat{A}\hat{x}^{\hat{u}}(t) + \hat{B}\hat{u}(t)$. Hence, $\hat{x}^{\hat{u}}(t) - Lx^u(t)$ and $\hat{B}\hat{u}(t) - LBu(t)$ are zero almost everywhere on $[0; T]$. This and the fundamental theorem of calculus then imply that $\hat{x}^{\hat{u}} = Lx^u$ everywhere. Moreover, there exists a sequence of points $(t_\nu)_\nu$ converging to zero that satisfies for all $\nu \geq 0$ as follows:

$$\begin{aligned}\hat{A}\hat{x}^{\hat{u}}(t_\nu) + \hat{B}\hat{u}(t_\nu) &= \partial_t \hat{x}^{\hat{u}}(t_\nu)\\ &= \partial_t Lx^u(t_\nu)\\ &= LAx^u(t_\nu) + LBu(t_\nu)\end{aligned}$$

yielding specifically $\hat{A}\hat{x}^{\hat{u}}(t_\nu) = LAx^u(t_\nu)$ for all $\nu \geq 0$. Since this yields $\hat{A}\hat{x}[0] = LAx[0]$ and $x[0]$ was chosen arbitrarily, we infer that $\mathcal{H}$ is a control equivalence. □

*Theorem 3:* Thanks to the fact that the algebraic description of control equivalence (i.e., $LA = LA\bar{L}L$) and the formula for $\hat{A}$ are as in the case of forward equivalence (and $\hat{B}$ is known), the statement readily follows from [17]. □

## References


[1] Y.-Y. Liu and A.-L. Barabási, "Control principles of complex systems," *Rev. Modern Phys.*, vol. 88, Sep. 2016, Art. no. 035006.

[2] L. Xiang, F. Chen, W. Ren, and G. Chen, "Advances in network controllability," *IEEE Circuits Syst. Mag.*, vol. 19, no. 2, pp. 8–32, Apr.-Jun. 2019.

[3] C.-T. Lin, "Structural controllability," *IEEE Trans. Autom. Control*, vol. 19, no. 3, pp. 201–208, Jun. 1974.

[4] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[5] Z. Yuan, C. Zhao, Z. Di, W.-X. Wang, and Y.-C. Lai, "Exact controllability of complex networks," *Nature Commun.*, vol. 4, no. 1, 2013, Art. no. 2447.

[6] J. Gao, Y.-Y. Liu, R. M. D'Souza, and A.-L. Barabási, "Target control of complex networks," *Nature Commun.*, vol. 5, no. 1, 2014, Art. no. 5415, doi: 10.1038/ncomms6415.

[7] G. Li et al., "Minimum-cost control of complex networks," *New J. Phys.*, vol. 18, no. 1, Dec. 2015, Art. no. 013012.

[8] A. E. Motter, "Networkcontrology," *Chaos: Interdiscipl. J. Nonlinear Sci.*, vol. 25, no. 9, 2015, Art. no. 097621.

[9] T. Kailath and P. Hall, *Linear Systems, Ser. Information and System Sciences Series*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1980.

[10] F. Pasqualetti, S. Zampieri, and F. Bullo, "Controllability metrics, limitations and algorithms for complex networks," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 1, pp. 40–52, Mar. 2014.

[11] J. Sun and A. E. Motter, "Controllability transition and nonlocality in network control," *Phys. Rev. Lett.*, vol. 110, May 2013, Art. no. 208701.

[12] L.-Z. Wang, Y.-Z. Chen, W.-X. Wang, and Y.-C. Lai, "Physical controllability of complex networks," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 40198, doi: 10.1038/srep40198.

[13] G. Yan, J. Ren, Y.-C. Lai, C.-H. Lai, and B. Li, "Controlling complex networks: How much energy is needed?," *Phys. Rev. Lett.*, vol. 108, May 2012, Art. no. 218703.

[14] A. Li, S. P. Cornelius, Y.-Y. Liu, L. Wang, and A.-L. Barabási, "The fundamental advantages of temporal networks," *Science*, vol. 358, no. 6366, pp. 1042–1046, 2017.

[15] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[16] I. Klickstein, A. Shirin, and F. Sorrentino, "Energy scaling of targeted optimal control of complex networks," *Nature Commun.*, vol. 8, no. 1, 2017, Art. no. 15145.

[17] L. Cardelli, M. Tribastone, M. Tschaikowski, and A. Vandin, "Maximal aggregation of polynomial dynamical systems," *Proc. Nat. Acad. Sci.*, vol. 114, no. 38, pp. 10 029–10 034, 2017. [Online]. Available: http://www.pnas.org/content/114/38/10029

[18] L. Cardelli, M. Tribastone, M. Tschaikowski, and A. Vandin, "Symbolic computation of differential equivalences," in *POPL*, R. Bodík and R. Majumdar, Eds. New York, NY, USA: ACM, 2016, pp. 137–150.

[19] R. Paige and R. Tarjan, "Three partition refinement algorithms," *SIAM J. Comput.*, vol. 16, no. 6, pp. 973–989, 1987.

[20] P. C. Kanellakis and S. A. Smolka, "CCS expressions, finite state processes, and three problems of equivalence," in *Proc. 2nd Annu. ACM Symp. Princ. Distrib. Comput.*, in Series PODC '83, 1983, pp. 228–240.

[21] S. Derisavi, H. Hermanns, and W. H. Sanders, "Optimal state-space lumping in Markov chains," *Inf. Process. Lett.*, vol. 87, no. 6, pp. 309–315, 2003.

[22] A. Valmari and G. Franceschinis, "Simple O(M LOGN) time Markov chain lumping," in *Proc. 16th Int. Conf. Tools Algorithms Construction Anal. Syst.*, 2010, pp. 38–52.

[23] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Trans. Autom. Control*, vol. 13, no. 3, pp. 246–253, Jun. 1968.

[24] A. J. v. d. Schaft, "Equivalence of dynamical systems by bisimulation," *IEEE Trans. Autom. Control*, vol. 49, no. 12, pp. 2160–2172, Dec. 2004.

[25] G. J. Pappas and S. Simic, "Consistent abstractions of affine control systems," *IEEE Trans. Autom. Control*, vol. 47, no. 5, pp. 745–756, May 2002.

[26] P. Tabuada and G. J. Pappas, "Abstractions of Hamiltonian control systems," *Automatica*, vol. 39, no. 12, pp. 2025–2033, 2003.

[27] G. J. Pappas, G. Lafferriere, and S. Sastry, "Hierarchically consistent control systems," *IEEE Trans. Autom. Control*, vol. 45, no. 6, pp. 1144–1160, Jun. 2000.

[28] L. Cardelli, M. Tribastone, M. Tschaikowski, and A. Vandin, "Efficient syntax-driven lumping of differential equations," in *Proc. Int. Conf. Tools Algorithms Construction Anal. Syst.*, 2016, pp. 93–111.

[29] S. Tognazzi, M. Tribastone, M. Tschaikowski, and A. Vandin, "Differential equivalence yields network centrality," in *Proc. Int. Symp. Leveraging Appl. Formal Methods*. Springer, 2018, pp. 186–201.

[30] S. G. Hartke and A. Radcliffe, "Mckay's canonical graph labeling algorithm," *Communicating Math.*, vol. 479, pp. 99–111, 2009.

[31] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*. Philadelphia, PA, USA: SIAM, 2005.

[32] G. Casadei, C. C.-d. Wit, and S. Zampieri, "Model reduction based approximation of the output controllability gramian in large-scale networks," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 4, pp. 1778–1788, Dec. 2020.

[33] L. Cardelli, M. Tribastone, M. Tschaikowski, and A. Vandin, "Symbolic computation of differential equivalences," *Theor. Comput. Sci.*, vol. 777, pp. 132–154, 2019.

[34] L. Cardelli, M. Tribastone, M. Tschaikowski, and A. Vandin, "Guaranteed error bounds on approximate model abstractions through reachability analysis," in *Proc. Quantitative Eval. Syst.: 15th Int. Conf.*, 2018, pp. 104–121.

[35] S. Tognazzi, M. Tribastone, M. Tschaikowski, and A. Vandin, "Differential equivalence for linear differential algebraic equations," *IEEE Trans. Autom. Control.*, vol. 67, no. 7, pp. 3484–3493, Jul. 2022.

[36] P. Wang and S. Xu, "Spectral coarse grained controllability of complex networks," *Physica A: Stat. Mechan. Appl.*, vol. 478, pp. 168–176, 2017.

[37] N. Stanley, R. Kwitt, M. Niethammer, and P. J. Mucha, "Compressing networks with super nodes," *Sci. Rep.*, vol. 8, no. 1, 2018, Art. no. 10892.

[38] Q.-L. Yang, L.-F. Wang, G.-T. Zhao, and G. Guo, "A coarse graining algorithm based on m-order degree in complex network," *Physica A: Stat. Mechan. Appl.*, vol. 558, 2020, Art. no. 124879.

[39] T. A. Davis and Y. Hu, "The university of Florida sparse matrix collection," *ACM Trans. Math. Softw.*, vol. 38, no. 1, pp. 1–25, 2011.

[40] J. Kunegis, "KONECT–The koblenz network collection," in *Proc. Int. Conf. World Wide Web Companion*, 2013, pp. 1343–1350. [Online]. Available: http://dl.acm.org/citation.cfm?id=2488173

[41] L. Harriger, M. P. v. d. Heuvel, and O. Sporns, "Rich club organization of macaque cerebral cortex and its role in network communication," *PLoS One*, vol. 7, no. 9, 2012, Art. no. e46497.

[42] P. T. Monteiro, T. Pedreira, M. Galocha, M. C. Teixeira, and C. Chaouiya, "Assessing regulatory features of the current transcriptional network of saccharomyces cerevisiae," *Sci. Rep.*, vol. 10, no. 1, 2020, Art. no. 17744, doi: 10.1038/s41598-020-74043-7.

[43] A. K. Naimzada, S. Stefani, and A. Torriero, *Networks, Topology and Dynamics: Theory and Applications to Economics and Social Systems*, vol. 613. Berlin, Germany: Springer, 2008.

[44] L. Cardelli, G. Squillace, M. Tribastone, M. Tschaikowski, and A. Vandin, "Formal lumping of polynomial differential equations through approximate equivalences," *J. Log. Algebr. Methods Prog.*, vol. 134, 2023, Art. no. 100876.

[45] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2011.