Compositional Analysis of Hybrid Systems: An Accelerated ADMM Approach

Murat Cubuktepe
The University of Texas at Austin
Austin, Texas, USA
mcubuktepe@utexas.edu

Mohamadreza Ahmadi
The University of Texas at Austin
Austin, Texas, USA
mrahmadi@utexas.edu

Ufuk Topcu
The University of Texas at Austin
Austin, Texas, USA
utopcu@utexas.edu

Brandon Hencey
Air Force Research Laboratory
Dayton, Ohio, USA
brandon.hencey@us.af.mil

ABSTRACT

We present a distributed analysis algorithm for large-scale hybrid systems based on the accelerated alternating method of multipliers (ADMM). We consider interconnected hybrid systems that are composed of continuous dynamics coupled with discrete dynamics. In particular, we focus on discrete dynamics that are defined over finite alphabets, e.g., deterministic finite state machines (DFSMs). For such classes of systems, we propose a method based on dissipativity theory for compositional analysis that allows us to study stability, passivity and input-output norms. Furthermore, for systems with large number of states, we demonstrate how accelerated ADMM can be used to carry out the computations in an scalable and distributed manner. The proposed methodology is illustrated by examples.

CCS CONCEPTS

• **Computer systems organization** → **Heterogeneous (hybrid) systems**; • **Computing methodologies** → *Computational control theory*; *Distributed algorithms*; • **Hardware** → Finite state machines;

KEYWORDS

Distributed Computation, Hybrid Systems, ADMM, Dissipativity, Finite State Machines

ACM Reference Format:
Murat Cubuktepe, Mohamadreza Ahmadi, Ufuk Topcu, and Brandon Hencey. 2018. Compositional Analysis of Hybrid Systems: An Accelerated ADMM Approach. In *Proceedings of Hybrid Systems: Computation and Control (HSCC'18).* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

HSCC'18, April 2018, Porto, Portugal
2018. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Over the past decades, we have witnessed a dramatic increase in research on hybrid and cyber-physical systems [1, 2, 17, 18, 31]. Examples of such systems in real-world can be found in robotics [15], biological networks [20], and in power systems [41].

The literature is rich in analysis and verification methods for hybrid systems [3, 21, 27, 42]. Despite the available tools for the analysis and verification, scalability still poses a challenge. Therefore, there has been a surge in compositional analysis techniques. These methods, in general, decompose the analysis problem of a large-scale hybrid system into smaller sub-problems, which can reduce the computational burden significantly. It was shown that dissipativity theory can be used as a tool for decompositional stability and detectability analysis [38]. This result was further extended in [23] to present sufficient conditions for passivity and stability analysis of a class of interconnected hybrid systems with sums of storage functions. Some well-posedness issues and input-ouput notions for interconnected hybrid systems were discussed in [30]. Nonetheless, one issue with the above compositional methods is that they often do not provide a computational framework to find the certificates and rely on ad-hoc analytical techniques to analyze the overall system.

In another vein, several compositional analysis techniques were proposed based on encoding the hybrid executions in a logic amenable to satisfiability checking (see the survey [29]). [19] proposes a method based on bounded error approximations of the hybrid dynamics and the satisfiability checking was carried out using the tool Z3. [6] brings forward a verification method at the intersection of software model checking and hybrid systems reachability, which decomposes the discrete and the continuous dynamics. However, the latter approaches based on SMT formulation are undecidable for general hybrid systems [5] and convergence is not guaranteed.

In this paper, we propose a methodology based on accelerated ADMM for distributed analysis of hybrid systems composed of continuous dynamics coupled with a system defined over finite-alphabets [34–36]. In particular, the discrete dynamics can be in the form of deterministic finite state machines, which can be used to model software. The framework we formulate is established upon dissipativity theory and decomposes the analysis problem of the overall hybrid system into smaller local sub-problems for subsystems. This method takes advantage of a global storage function which is the sum of local storage functions. To carry out the computation in a distributed manner, we use accelerated alternating method of multipliers (ADMM) [12], which is a variant of ADMM [10]. To use accelerated ADMM, which has a faster convergence rate compared to ADMM, we utilize *smoothing techniques* [9, 25], which has been used in to improve the convergence rate of similar first order methods. We also discuss the effects of *restarting* accelerated ADMM, which has shown to improve the convergence rate of similar accelerated algorithms [9, 26]. We show that accelerated ADMM outperforms conventional ADMM that was used for computational analysis of continuous dynamical systems [22]. We illustrate the proposed method by two numerical examples.

This paper is structured as follows. In the following section, we present the problem formulation and define notions of hybrid Lyapunov and storage functions. In Section 3, we propose a method based on dissipativity for compositional analysis of the class of hybrid systems under study. The proposed methodology is illustrated by two examples in Section 5. Finally, Section 6 concludes the paper and gives directions for future research.

Notation: $\mathbb{R}_{\geq 0}$ denotes the set $[0, \infty)$. $\|\cdot\|$ denotes the Euclidean vector norm on $\mathbb{R}^n$. The set of integers are denoted by $\mathbb{Z}$. For a function $f : A \to B$, $f \in L^p(A, B)$, $1 \leq p < \infty$, implies that $\left(\int_A |f(t)|^p dt\right)^{\frac{1}{p}} < \infty$ and $\sup_{t \in A} |f(t)| < \infty$ for $p = \infty$. For symmetric matrices $A_1, \ldots, A_n$, $diag(A_1, \ldots, A_n)$ denotes the diagonalized matrix

$$diag(A_1, \ldots, A_n) = \begin{bmatrix} A_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & A_n \end{bmatrix}.$$

For a vector $s \in \mathbb{R}^{n_s}$, $s \equiv 0$ denotes the element-wise equality to zero.

2 PROBLEM FORMULATION

We propose a method based on dissipativity theory for stability and performance analysis of interconnected hybrid systems. The class of hybrid systems under study are composed of a continuous subsystem in connection with a subsystem defined over a finite alphabet. The main motivation for considering this class of hybrid systems is to model applications where a continuous system is controlled by a software. In Section 4, we demonstrate how accelerated ADMM can be used to undertake the computations in a distributed manner.

Formally, we consider the following class of hybrid systems

$$\mathcal{G} : \begin{cases} \mathcal{C} : \begin{cases} \dot{x}(t) = f\left(x(t), w(t); p(t)\right) \\ y(t) = h\left(x(t); p(t)\right) \end{cases} \\ \mathcal{D} : \begin{cases} q(t^+) = g\left(q(t), u(t), x(t)\right) \\ p(t) = l\left(q(t), u(t)\right) \end{cases} \\ q(t_0) = q_0, \ x(t_0) = x_0. \end{cases} \tag{1}$$

where $x \in \mathbb{R}^n$ and $q \in \mathcal{Q} \subset \mathbb{N}$ represent continuous and discrete states. In the continuous module $\mathcal{C}$, $f(\cdot, \cdot; p) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $f(0, q; p) \equiv 0$, $\forall (q, p) \in \mathcal{Q} \times \mathcal{P}$, is a family of mappings with index $p \in \mathcal{P} \subset \mathbb{N}$ and similarly $h(\cdot; p) : \mathbb{R}^n \to \mathbb{R}^{n_y}$ is a family of output mappings. $y \in \mathbb{R}^{n_y}$ and $w \in \mathbb{R}^m$ are the continuous outputs and inputs, respectively. In the discrete module $\mathcal{D}$, $g : \mathcal{Q} \times \mathcal{U} \times \mathbb{R}^n \to \mathcal{Q}$ and $l : \mathcal{Q} \times \mathcal{U} \to \mathcal{Q}$. $p \in \mathcal{P}$ and $u \in \mathcal{U}$ are the discrete outputs and inputs, respectively. The sets associated with the discrete module $\mathcal{Q}$, $\mathcal{P}$ and $\mathcal{U}$ are assumed to be finite. In the sequel, we abuse the notation and use q^+ to represent $q(t^+)$.

The discrete module $\mathcal{D}$ can characterize a rich class of systems defined over finite alphabets [36] and can be used to model systems ranging from quantizers to deterministic finite state machines. The hybrid system (1) can also be studied in the context of hybrid automata [16]. However, note that for hybrid automata, analysis tools such as Lyapunov functions or storage functions are not available in general.

We can study the input-output and stability properties of system (1) by using a dissipativity-type and Lyapunov-type argument. To this end, we use the notion of hybrid Lyapunov or storage function, which is described as follows.

Definition 1 (Hybrid Lyapunov Function). *A function $V : \mathbb{R}^n \times \mathcal{Q} \to \mathbb{R}_{\geq 0}$ such that $V(0, q) = 0$, $\forall q \in \mathcal{Q}$, $V \in C^1(\mathbb{R}^n)$ is called a* hybrid Lyapunov function *for system* (1) *with $u \equiv 0$ and $w \equiv 0$, if it satisfies the following inequalities*

$$V(x, q) > 0, \quad \forall x \in \mathbb{R}^n \setminus \{0\}, \ \forall q \in \mathcal{Q}, \tag{2}$$

$$\left(\frac{\partial V(x, q)}{\partial x}\right)^T f(x, 0; p) < 0, \quad \forall x \in \mathbb{R}^n, \ \forall q \in \mathcal{Q}, \ \forall p \in \mathcal{P}, \tag{3}$$

and

$$V(x, q^+) - V(x, q) \leq 0, \quad \forall x \in \mathbb{R}^n, \forall q \in \mathcal{Q}. \tag{4}$$

Theorem 1. *The hybrid system* (1) *is asymptotically stable if there exists a hybrid Lyapunov function.*

Proof. See Appendix A. □

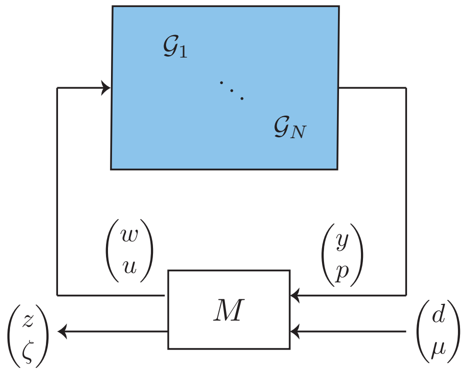

Figure 1: Interconnected system with hybrid inputs $(d, \mu)^T$ and hybrid outputs $(z, \zeta)^T$.

Definition 2 (Hybrid Storage Function). *A function $V : \mathbb{R}^n \times \mathcal{Q} \to \mathbb{R}_{\geq 0}$ such that $V \in C^1(\mathbb{R}^n)$ is called a* hybrid storage function *for system* (1)*, if it satisfies the following inequalities*

$$V(x, q) \geq 0, \quad \forall x \in \mathbb{R}^n, \forall q \in \mathcal{Q}, \tag{5}$$

$$\left(\frac{\partial V(x,q)}{\partial x}\right)^T f(x, w; p) \leq W_c(w, y), \quad \forall x \in \mathbb{R}^n,\ \forall q \in \mathcal{Q},\ \forall p \in \mathcal{P} \tag{6}$$

and

$$V(x, q^+) - V(x, q) \leq W_d(u, p), \quad \forall x \in \mathbb{R}^n, \forall q \in \mathcal{Q} \tag{7}$$

where the integrable functions $W_c : \mathbb{R}^m \times \mathbb{R}^{n_y} \to \mathbb{R}$ and $W_d : \mathcal{Q} \times \mathcal{U} \to \mathbb{R}$ are the continuous and the discrete supply rates, respectively.

Theorem 2. *The hybrid system* (1) *is dissipative with respect to the supply rates W_c and W_d, if there exists a hybrid storage function.*

Proof. The dissipativity of the continuous dynamics is standard and follows from integrating (6). The dissipativity of the discrete dynamics follows from Theorem 3 and Lemma 2 in [37]. □

3 INTERCONNECTION OF HYBRID SYSTEMS

We consider interconnected systems as illustrated in Fig. 1, where the subsystems $\{\mathcal{G}_i\}_{i=1}^N$ are known and have dynamics in the form of (1). We associate each subsystem with a set of functions $\{f_i, h_i, g_i, l_i\}$ and $x_i \in \mathbb{R}^{n_i}$, $q_i \in \mathcal{Q}_i$, $w_i \in \mathbb{R}^{n_w^i}$, $u_i \in \mathcal{U}_i$, $y_i \in \mathbb{R}^{n_y^i}$ and $p_i \in \mathcal{P}_i$. The static interconnection is characterized by a matrix $M \in \mathbb{R}^{n_w} \times \mathbb{R}^{n_y}$ where $n = \sum_{i=1}^N n_i$, $n_w = \sum_{i=1}^N n_w^i$ and $n_y = \sum_{i=1}^N n_y^i$. That is, M satisfies

$$\begin{bmatrix} w \\ z \\ u \\ \zeta \end{bmatrix} = M \begin{bmatrix} y \\ d \\ p \\ \mu \end{bmatrix}, \tag{8}$$

where $d \in \mathbb{R}^{n_d}$ and $z \in \mathbb{R}^{n_z}$ are the continuous exogenous inputs and outputs, respectively. Similarly, $\mu \in \mathcal{M} \subset \mathbb{Z}^{n_\mu}$ and $\mu \in \mathcal{Z} \subset \mathbb{Z}^{n_\zeta}$ are the discrete exogenous inputs and outputs, respectively. We assume this interconnection is well-posed, i.e., for all $d \in L_{2e}$ and initial condition $x(0) \in \mathbb{R}^n$, there exist unique $z, w, y \in L_{2e}$ that causally depend on d for all $p \in \mathcal{P}$. Furthermore, we define

$$M = \left[\begin{array}{c|c} M_c & \\ \hline & M_d \end{array}\right],$$

where $M_c \in \mathbb{R}^{n_w+n_y} \times \mathbb{R}^{n_w+n_y}$ and $M_d \in \mathbb{Z}^{n_\mu+n_\zeta} \times \mathbb{Z}^{n_\mu+n_\zeta}$.

The continuous local and global supply rates, $W_c^i(w_i, y_i)$ and $W_c(d, z)$, respectively, are defined by quadratic functions. That is,

$$W_c(d, z) = \begin{bmatrix} d \\ z \end{bmatrix}^T S \begin{bmatrix} d \\ z \end{bmatrix}. \tag{9}$$

Analogously, for discrete subsystems, we have $W_d^i(p_i, u_i)$ and $W_d(p, u)$ given by quadratic functions

$$W_d(p, u) = \begin{bmatrix} u \\ p \end{bmatrix}^T R \begin{bmatrix} u \\ p \end{bmatrix}. \tag{10}$$

We next show that certifying the dissipativity of an overall interconnected system can be concluded if each of the subsystems satisfy the local dissipativity property. Let

$$\mathcal{L}_i = \Bigg\{(S_i,\ R_i) \mid \mathcal{G}_i \text{ is dissipative w.r.t. } \begin{bmatrix} w_i \\ y_i \end{bmatrix}^T S_i \begin{bmatrix} w_i \\ y_i \end{bmatrix}, \text{ and } \begin{bmatrix} u_i \\ p_i \end{bmatrix}^T R_i \begin{bmatrix} u_i \\ p_i \end{bmatrix}\Bigg\}, \tag{11}$$

$$\mathcal{L}_c = \left\{S, \{S_i\}_{i=1}^N \mid \begin{bmatrix} M_c \\ 1_{n_y} \end{bmatrix}^T P_c^T Q_c P_c \begin{bmatrix} M_c \\ 1_{n_y} \end{bmatrix} < 0\right\}, \tag{12}$$

and

$$\mathcal{L}_d = \left\{R, \{R_i\}_{i=1}^N \mid \begin{bmatrix} M_d \\ 1_{n_p} \end{bmatrix}^T P_d^T Q_d P_d \begin{bmatrix} M_d \\ 1_{n_p} \end{bmatrix} < 0\right\}, \tag{13}$$

wherein $Q_c = diag(S_1, \ldots, S_N, -S)$, $Q_d = diag(R_1, \ldots, R_N, -R)$, and P_c and P_d are permutation matrices defined by

$$\begin{bmatrix} w_1 \\ y_1 \\ \vdots \\ w_N \\ y_N \\ d \\ z \end{bmatrix} = P_c \begin{bmatrix} w \\ z \\ y \\ d \end{bmatrix}, \text{ and } \begin{bmatrix} u_1 \\ p_1 \\ \vdots \\ u_N \\ p_N \\ \mu \\ \zeta \end{bmatrix} = P_d \begin{bmatrix} u \\ \zeta \\ p \\ \mu \end{bmatrix}. \tag{14}$$

Proposition 1. *Consider the interconnection of N subsystems as given in* (8) *with the global supply rates* (9) *and* (10). *If there exists* $\{S_i\}_{i=1}^N$ *and* $\{R_i\}_{i=1}^N$ *satisfying*

$$(S_i, R_i) \in \mathcal{L}_i, \; i = 1, \ldots, N, \tag{15}$$

and

$$(S_1, \ldots, S_N, -S) \in \mathcal{L}_c, \tag{16}$$

$$(R_1, \ldots, R_N, -R) \in \mathcal{L}_d, \tag{17}$$

then the interconnected system is dissipative with respect to the global supply rates W_d *and* W_c. *A storage function certifying global dissipativity is* $V(x, q) = \sum_{i=1}^N V_i(x_i, q_i)$, *where* V_i *is the storage function certifying dissipativity of subsystem* i *as in* $\mathcal{L}_i$.

Proof. See Appendix B. □

Proposition 1 provides the means to decompose the analysis of interconnected subsystems to smaller problems that are computationally more amenable. However, even the above discussed decompositional analysis method can be computationally involved for large-scale hybrid systems. In the following, we propose a method based on accelerated ADMM to carry out such computations in a distributed manner.

4 COMPUTATIONAL FORMULATION USING ACCELERATED ADMM

For small-scale systems, we can solve the optimization problem outlined in Proposition 1 using publicly available SDP solvers like MOSEK [4], SeDuMi [33] or SDPT3 [39]. But, these SDP solvers do not scale well for larger problems, as they use interior point methods, which requires solving a system of equations in each iteration. However, the structure in our problem allows us to decompose the constraints in (5), (6) and (7), which allows us a distributed algorithm. Specifally, ADMM [10] allows us to decompose convex optimization problems into a set of smaller problems. A generic convex optimization problem

$$\begin{aligned} \text{minimize} \quad & f(l) \\ \text{subject to} \quad & l \in C, \end{aligned} \tag{18}$$

where $l \in \mathbb{R}^n$, f is a convex function, and C is a convex set, can be written in ADMM form as

$$\begin{aligned} \text{minimize} \quad & f(l) + g(v) \\ \text{subject to} \quad & l = v, \end{aligned} \tag{19}$$

where g is the indicator function of C.

The problem we want to find a compositional formulation, which is outlined in Proposition 1 is

$$\begin{aligned} \underset{\{S_i\}_{i=1}^N, \{R_i\}_{i=1}^N, \{V_i\}_{i=1}^N}{\text{minimize}} \quad & \eta \\ \text{subject to (5),(6),(7), (16), and (17)} \end{aligned} \tag{20}$$

For example, η can be the induced norm of $\frac{\|z\|_{L^2}}{\|d\|_{L^2}}$ or $\frac{\|\zeta\|_{l^2}}{\|\mu\|_{l^2}}$ we want to minimize. Using the above form, the problem in (20) can be written in ADMM form with $f(l)$ is defined as sum of η and the indicator function of (5),(6) and (7), and $g(v)$ is defined as the indicator function of (16) and (17). Then, the scaled form of ADMM algorithm for problem in (19) is

$$\begin{aligned} l^{k+1} &= \arg\min_{l_i} f(l) + (\rho/2)||l - v^k + z^k||_2^2, \\ v^{k+1} &= \arg\min_{v} g(v) + (\rho/2)||l^{k+1} - v + z^k||_2^2, \\ z^{k+1} &= u^k + l^{k+1} - v^{k+1}, \end{aligned}$$

where l and v are the vectorized form of the matrices $\{S_i\}_{i=1}^N$, $\{V_i\}_{i=1}^N$, $\{R_i\}_{i=1}^N$, z is the scaled dual variable and $\rho > 0$ is the *penalty parameter*. Since $f(l)$ is separable for each subsystem, the ADMM algorithm can be parallelized as follows:

$$\begin{aligned} l_i^{k+1} &= \arg\min_{l_i} f_i(l) + (\rho/2)||l_i - v_i^k + z_i^k||_2^2, \\ v^{k+1} &= \arg\min_{v} g(v) + (\rho/2)||l^{k+1} - v + z^k||_2^2, \\ z^{k+1} &= z^k + l^{k+1} - v^{k+1}, \end{aligned}$$

Under mild assumptions, the ADMM algorithm converges [10], but the convergence is only asymptotic in general, therefore it may require many iterations to achieve sufficient accuracy.

4.1 Accelerated ADMM

Seceral algorithms in [7, 8, 11, 14, 26] shows that acceleration schemes can improve the performance significantly. These methods achieve $O(\frac{1}{k^2})$ convergence after k iterations, which is shown to be optimal for a first order method [24]. However, they usually require the function $f(l)$ to be differentiable with a known Lipschitz constant on the $\nabla f(l)$, which does not exist when the problem is constrained. For the case when $f(l)$ or $g(v)$ is not strongly convex or smooth, smoothing approaches have been used [9, 25] to improve convergence. However, to the best of our knowledge, these methods have not been applied in compositional analysis for hybrid systems.

Consider the following perturbation of the problem in (20)

$$\begin{aligned}\underset{\{S_i\}_{i=1}^N,\{R_i\}_{i=1}^N,\{V_i\}_{i=1}^N}{\text{minimize}} \quad & \eta + \mu \, d_i(S_i, R_i, V_i,) \\ \text{subject to } & (5),(6),(7), (16), \text{ and } (17)\end{aligned} \tag{21}$$

for some fixed smoothing parameter $\mu > 0$ and a strongly convex function d that satisfies

$$d(l) \geq d(l_0) + \frac{1}{2}||l - l_0||_2^2 \tag{22}$$

for some point $l_0 \in \mathbb{R}^n$. Specifically, we choose $d_i = \|S_i\|_F + \|V_i\|_F + \|R_i\|_F$, where $\|\cdot\|_F$ is the Frobenius norm. For some problems, it is shown that for small enough μ, the approximate problem (21) is equivalent to the original problem [9].

When $f(l)$ and $g(v)$ are strongly convex, the ADMM algorithm can be modified with an acceleration step to achieve $O(\frac{1}{k^2})$ convergence after k iterations [12]. Then, the accelerated ADMM algorithm is

$$\begin{aligned}
l_i^k &= \arg\min_{l_i} f_i(l) + (\rho/2)||l_i - \bar{v}_i^k + \bar{z}_i^k||_2^2, \\
v^k &= \arg\min_{v} g(v) + (\rho/2)||l^k - v + \bar{z}^k||_2^2, \\
z^k &= \bar{z}^k + l^k - v^k, \\
\alpha_{k+1} &= \frac{1 + \sqrt{1 + 4\alpha_k^2}}{2} \\
\bar{v}^{k+1} &= v_k + \frac{\alpha_k - 1}{\alpha_{k+1}}(v^k - v^{k-1}) \\
\bar{z}^{k+1} &= z^k + \frac{\alpha_k - 1}{\alpha_{k+1}}(z^k - z^{k-1}),
\end{aligned}$$

where ρ is a positive constant that satisfies $\rho \leq \mu$, and $\alpha_1 = 1$.

Note that l update can be carried out in parallel while achieving $O(\frac{1}{k^2})$ convergence, which cannot be achieved by the standard ADMM or accelerated proximal methods if there are constraints in the problem.

In general, we do not have access to the Lipschitz constant or strongly convexity parameter in the feasible region of the subproblems because of the constraints, which may reduce the performance of the accelerated method [9]. One approach to deal with the case of unknown Lipschitz constant or strongly convexity parameter is so-called *restart* method, which is used in [9, 26], and it is shown that restart methods can improve the convergence rate significantly. To apply the method, we *restart* the algorithm, i.e, we set the acceleration parameter $\alpha_k = 1$ after a certain number of iterations while using the point in iteration k as the starting point for the restart, which resets the acceleration parameter, and reruns the accelerated ADMM algorithm from the next starting point. Examples in [9] show that the restarting methods can greatly improve performance, but they note that the restart method requires tuning for different problems to optimize the performance, i.e, the performance can vary significantly with different restart schemes.

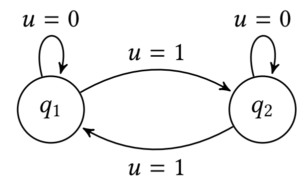

Figure 2: $\mathcal{D}_i$ in the Example I.

5 NUMERICAL EXPERIMENTS

In this section, we illustrate the proposed distributed analysis method with two examples, where we compare the convergence rate of ADMM with accelerated ADMM and several restart methods. We implemented both standard ADMM and accelerated ADMM algorithms in MATLAB using the CVX toolbox [13] and MOSEK [4] to solve SDP problems.

5.1 Example I

We illustrate our approach in the following example with the subsystems $\mathcal{G}_i$ with $i = 1, 2$:

$$\mathcal{G}_0 : \begin{cases}
f(x_1, \begin{pmatrix} w_2 \\ w_3 \end{pmatrix}; 0) = \begin{bmatrix} -4 & 3 \\ 2 & -6 \end{bmatrix} x_1 + I_2 \begin{bmatrix} w_2 + d \\ w_3 \end{bmatrix}, \\
h(x_1; 0) = w_1 = 0.5 \begin{bmatrix} 1 & 1 \end{bmatrix} x_1, \\
f(x_1, \begin{pmatrix} w_2 \\ w_3 \end{pmatrix}; 1) = \begin{bmatrix} -5 & 4 \\ 1 & -4 \end{bmatrix} x_1 + I_2 \begin{bmatrix} w_2 + d \\ w_3 \end{bmatrix}, \\
h(x_1; 1) = w_1 = 0.5 \begin{bmatrix} 1 & 1 \end{bmatrix} x_1,
\end{cases}$$

$$\mathcal{G}_1 : \begin{cases}
f(x_2, w_1; 0) = \begin{bmatrix} -7 & 1 \\ 2 & -3 \end{bmatrix} x_2 + \begin{bmatrix} 1 \\ 1 \end{bmatrix} w_1, \\
h(x_2; 0) = \begin{bmatrix} w_2 \\ w_3 \end{bmatrix} = 0.5 I_2 x_2, \\
f(x_2, w_1; 1) = \dot{x}_2 = \begin{bmatrix} -5 & 2 \\ 3 & -3 \end{bmatrix} x_2 + \begin{bmatrix} 1 \\ 1 \end{bmatrix} w_1, \\
h(x_2; 1) = \begin{bmatrix} w_2 \\ w_3 \end{bmatrix} = 0.5 I_2 x_2,
\end{cases}$$

and $\mathcal{D}_i$ is shown in Figure 2 with two states q_1 and q_2, inputs $\mathscr{U} = \{0, 1\}$ and outputs $\mathscr{P} = \{0, 1\}$. The output function p is defined as $p(q_j, u) : u$.

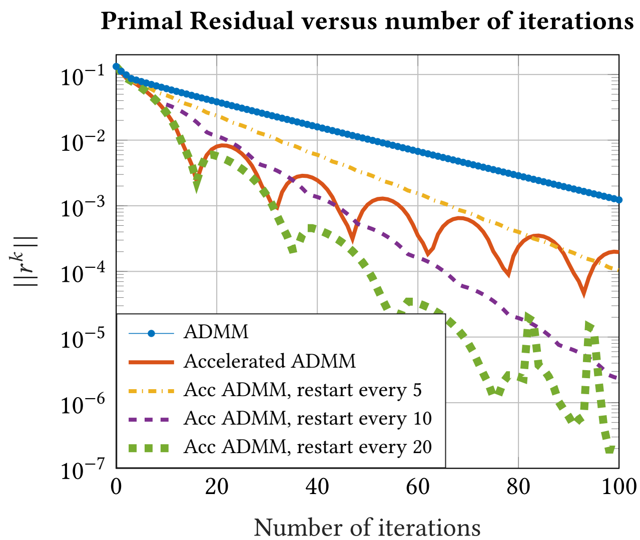

Figure 3: Norm of primal residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

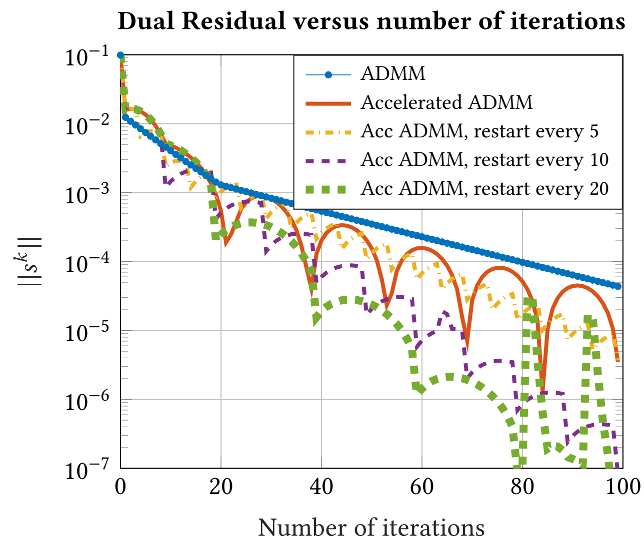

Figure 4: Norm of dual residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

We apply the compositional approach underlined by the problem in (21) to synthesize a controller to minimize the H_∞-norm between the output $y = w_3$ and input d.

Since the continuous dynamics are linear, we consider quadratic Lyapunov functions for subsystems. For each $i = 1, 2$, let

$$V(x_i) = \begin{pmatrix} x \\ q \end{pmatrix}^T \begin{bmatrix} P_i & r_i \\ r_i^T & \lambda_i \end{bmatrix} \begin{pmatrix} x \\ q \end{pmatrix}.$$

The iterative methods were initialized using $V_i^0 = S_i^0 = R_i^0 = U_i^0 = I$. For each method, we plot the norm of *primal residual* in Figure 3, which is defined as $r^k = l^k - v^k$, and it is the residual for primal feasibility. Also, we show the norm of the *dual residual* $s^k = \rho(v^k - v^{k-1})$ in Figure 4, which can be viewed as a residual for the dual feasibility condition.

The accelerated ADMM achieves superior convergence in primal and dual residuals compared to ADMM. However, restarting the algorithm in every 10 or 20 iterations makes the convergence of the primal residual and dual residual significantly faster than accelerated ADMM without restart. In this example, restarting in every 5 iterations does not improve the convergence significantly, however all of the accelerated methods outperforms ADMM. After 100 iterations with the accelerated ADMM, the minimum H_∞-norm between the output y and input d is 0.1411 with the following local Lyapunov functions:

$$P_1 = \begin{bmatrix} 0.0912 & 0.0500 & 0 \\ 0.0500 & 0.1556 & 0 \\ 0 & 0 & 7.9 \cdot 10^{-5} \end{bmatrix},$$

$$P_2 = \begin{bmatrix} 0.0029 & 0.0010 & 0 \\ 0.0010 & 0.0007 & 0 \\ 0 & 0 & 6.8 \cdot 10^{-5} \end{bmatrix}.$$

5.2 Example II

We consider a modified version of the example in [40] as illustrated in Fig 5. For $i = 1, 2, 3$, the subsystems $\mathcal{G}_i$ are characterized as follows:

$$\mathcal{G}_1 : \begin{cases} f(x_1, \begin{pmatrix} w_2 \\ w_4 \end{pmatrix}; 0) = \begin{bmatrix} -1 & 0 \\ 2 & -2 \end{bmatrix} x_1 + I_2 \begin{bmatrix} w_2 \\ w_4 \end{bmatrix}, \\ h(x_1; 0) = w_1 = 0.5 \begin{bmatrix} 1 & 1 \end{bmatrix} x_1, \\ f(x_1, \begin{pmatrix} w_2 \\ w_4 \end{pmatrix}; 1) = \begin{bmatrix} -2 & 0 \\ 2 & -3 \end{bmatrix} x_1 + I_2 \begin{bmatrix} w_2 \\ w_4 \end{bmatrix}, \\ h(x_1; 1) = w_1 = 0.5 \begin{bmatrix} 1 & 1 \end{bmatrix} x_1, \\ f(x_1, \begin{pmatrix} w_2 \\ w_4 \end{pmatrix}; 2) = \begin{bmatrix} -4 & 0 \\ 2 & -4 \end{bmatrix} x_1 + I_2 \begin{bmatrix} w_2 \\ w_4 \end{bmatrix}, \\ h(x_1; 2) = w_1 = 0.5 \begin{bmatrix} 1 & 1 \end{bmatrix} x_1, \end{cases}$$

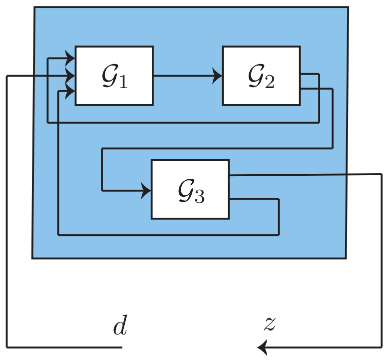

Figure 5: The interconnected system in Example II.

$$\mathcal{G}_2:\begin{cases} f(x_2,w_1;0)=\begin{bmatrix}-8 & 0\\ 12 & -2\end{bmatrix}x_2+\begin{bmatrix}1\\1\end{bmatrix}w_1,\\ h(x_2;0)=\begin{bmatrix}w_2\\w_3\end{bmatrix}=0.5I_2x_2,\\ f(x_2,w_1;1)=\begin{bmatrix}-7 & 1\\ 2 & -3\end{bmatrix}x_2+\begin{bmatrix}1\\1\end{bmatrix}w_1,\\ h(x_2;1)=\begin{bmatrix}w_2\\w_3\end{bmatrix}=0.5I_2x_2,\\ f(x_2,w_1;2)=\begin{bmatrix}-7 & 0\\ 6 & -3\end{bmatrix}x_2+\begin{bmatrix}1\\1\end{bmatrix}(w_1+d),\\ h(x_2;2)=\begin{bmatrix}w_2\\w_3\end{bmatrix}=0.5I_2x_2,\end{cases}$$

$$\mathcal{G}_3:\begin{cases} f(x_3,w_3;0)=\begin{bmatrix}-1 & 0\\ 2 & -2\end{bmatrix}x_3+\begin{bmatrix}1\\1\end{bmatrix}w_3,\\ h(x_3;0)=w_4=0.4\begin{bmatrix}1 & 1\end{bmatrix}x_3,\\ f(x_3,w_3;1)=\begin{bmatrix}-2 & 0\\ 2 & -3\end{bmatrix}x_3+\begin{bmatrix}1\\1\end{bmatrix}w_3,\\ h(x_3;1)=w_4=0.4\begin{bmatrix}1 & 1\end{bmatrix}x_3,\\ f(x_3,w_3;2)=\begin{bmatrix}-3 & 0\\ 2 & -4\end{bmatrix}x_3+\begin{bmatrix}1\\1\end{bmatrix}w_3,\\ h(x_3;2)=w_4=0.4\begin{bmatrix}1 & 1\end{bmatrix}x_3\end{cases}$$

and $\mathcal{D}_i$ is shown in Figure 6 with three states q_1, q_2, and q_3, inputs $\mathcal{U} = \{0,1\}$ and outputs $\mathcal{P} = \{0,1,2\}$. The output function p is defined as

$$p(q_i,u):\begin{cases}u, & \text{for } i\in\{1,2\}\\ 2, & \text{for } i=3.\end{cases}$$

We initialize the methods using $V_i^0 = S_i^0 = R_i^0 = U_i^0 = I$ like in Example 1. Similarly in Example I, accelerated ADMM

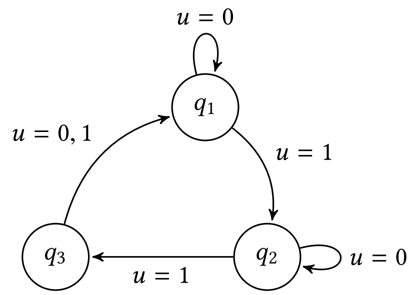

Figure 6: $\mathcal{D}_i$ in the Example II.

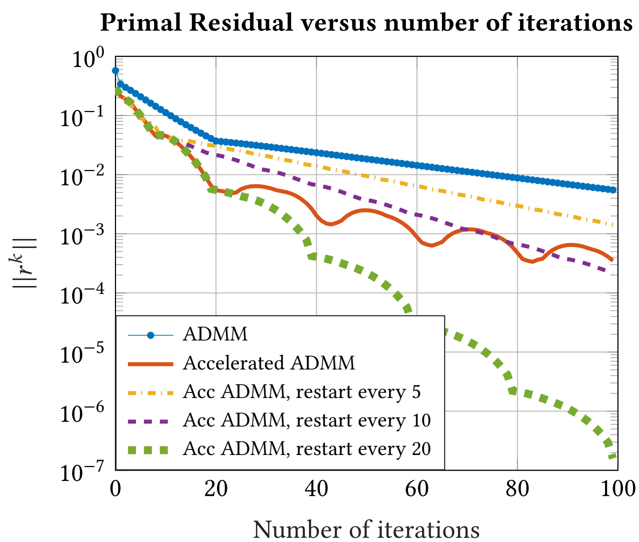

Figure 7: Norm of primal residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

achieves superior convergence in primal residual and dual residual compared to ADMM, which can be seen in Figures 7 and 8. Also, restarting the method at every 20 iteration improves the primal and dual convergence significantly, however restarting at every 5 or 10 iterations does improve the convergence of residuals significantly over the accelerated ADMM, which shows that different restarting methods can perform differently in various problems. However, all of the accelerated methods outperform ADMM. After 100 iterations with accelerated ADMM, the minimum induced-norm between the output from input u to p is $\eta = 1$ with the following local storage functions:

$$P_1 = \begin{bmatrix}0.0815 & 0.0192 & 0\\ 0.0192 & 0.0078 & 0\\ 0 & 0 & 1.7\cdot 10^{-4}\end{bmatrix},$$

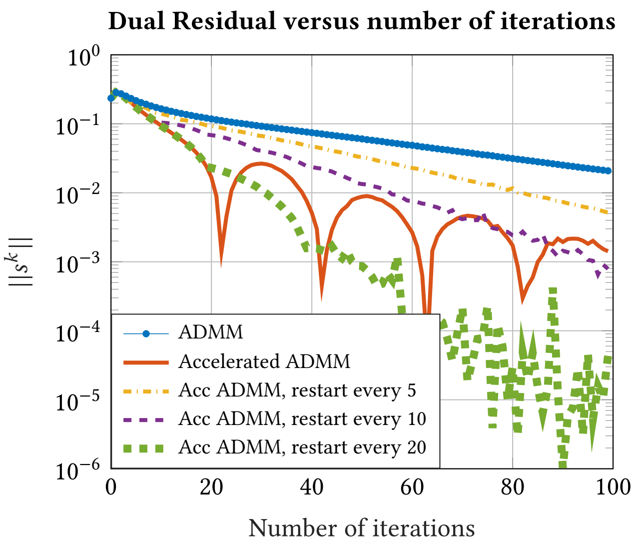

Figure 8: Norm of dual residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

$$P_2 = \begin{bmatrix} 0.1142 & 0.0685 & 0 \\ 0.0685 & 0.0489 & 0 \\ 0 & 0 & 1.7 \cdot 10^{-4} \end{bmatrix},$$

$$P_3 = \begin{bmatrix} 0.4546 & 0.1119 & 0 \\ 0.1119 & 0.0458 & 0 \\ 0 & 0 & 2.1 \cdot 10^{-4} \end{bmatrix}.$$

We also consider finding the minimal H_∞-norm between the output $y = w_3$ input d. For each method that we discussed, we plot the norm of primal residual in Figure 9, and the norm of the dual residual in Figure 10.

Similar to the previous examples, accelerated ADMM achieves superior convergence in residuals compared to ADMM and restarting the method at every 20 iteration improves the primal and dual convergence significantly. Other restart methods and the accelerated ADMM also outperforms the regular ADMM, but restarting at every 20 iteration gives the fastest rate of convergence in both cases. After 100 iterations with the accelerated ADMM, the value of the minimum norm between the output $y = w_3$ input d is $\eta = 0.2049$ with the following local storage functions:

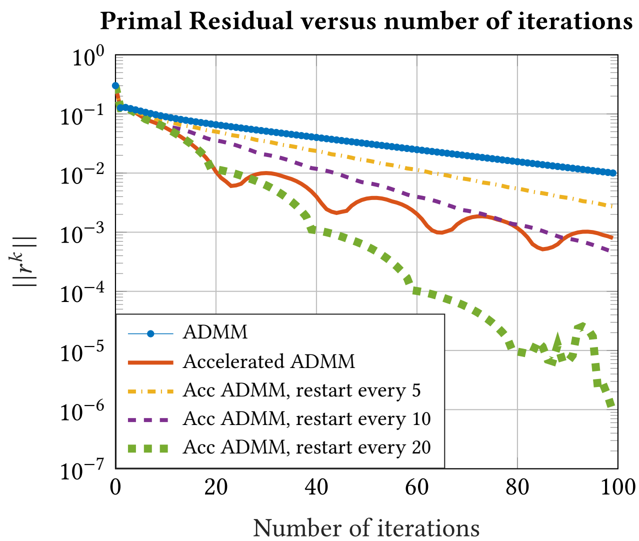

Figure 9: Norm of primal residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

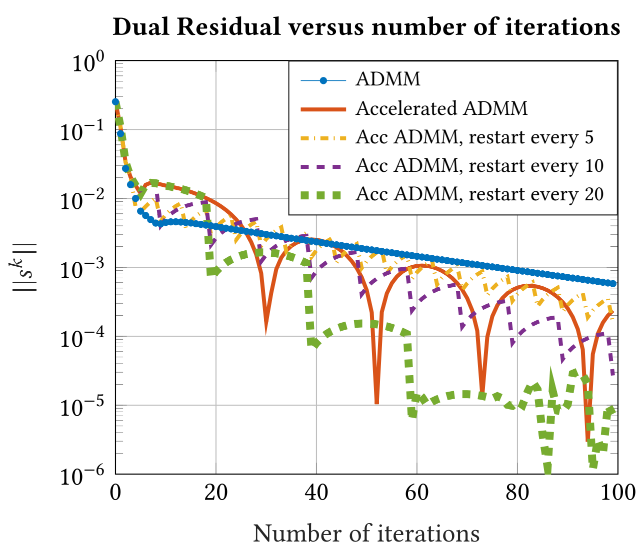

Figure 10: Norm of dual residual versus number of iterations for the decentralized analysis problem with standard and accelerated ADMM with various restart methods.

$$P_1 = \begin{bmatrix} 0.1568 & 0.0366 & 0 \\ 0.0366 & 0.0147 & 0 \\ 0 & 0 & 1.6 \cdot 10^{-4} \end{bmatrix},$$

$$P_2 = \begin{bmatrix} 0.2166 & 0.1297 & 0 \\ 0.1297 & 0.0925 & 0 \\ 0 & 0 & 1.7 \cdot 10^{-4} \end{bmatrix},$$

$$P_3 = \begin{bmatrix} 0.0380 & 0.0099 & 0 \\ 0.0099 & 0.0042 & 0 \\ 0 & 0 & 6.2 \cdot 10^{-5} \end{bmatrix}.$$

6 CONCLUSIONS AND FUTURE WORK

We propose a method for compositional analysis of large scale hybrid systems, for which an underlying interconnection topology is given. For such systems, we decompose the global analysis problem into a number of smaller local analysis problems using dissipativity theory. Furthermore, we proposed a distributed optimization method with smoothing techniques, which enables to employ accelerated ADMM. Numerical results show that the accelerated ADMM method with different restart methods significantly improves the convergence rate compared to standard ADMM.

In the examples studied in this paper, we considered linear continuous dynamics. The generalization to polynomial continuous dynamics can be formulated based on sum-of-squares optimization [28]. Moreover, one interesting analysis problem to future research is compositional safety verification. In this respect, in [32], a method is brought forward based on compositional barrier certificates for continuous systems and the dual decomposition method was used for implementation. Finally, distributed synthesis of control laws for large-scale hybrid systems can be studied using the dissipativity framework presented in this study.

A PROOF OF THEOREM 1

Inequality (2) implies that V is positive definite and $V(0,q) = 0,\ \forall q \in \mathcal{Q}$. We define the time intervals $T_k = (t_k, t_{k+1})$ where the continuous dynamics follows C in (1) with $p \in \mathscr{P}$. Similarly, t_k^+ corresponds to the discrete jump instant at t_k. By continuity, we can re-write (3) as

$$\left(\frac{\partial V(x,q)}{\partial x}\right)^T f(x,0;p) \le -\rho_p, \quad \forall x \in \mathbb{R}^n,\ \forall q \in \mathcal{Q},\ \forall p \in \mathscr{P}, \tag{23}$$

where $\rho_p,\ p \in \mathscr{P}$, is a small positive number. Moreover, differentiating V with respect to time and integrating it from t_0 to t_n and noting that the discrete jumps happen on sets of measure zero, yields

$$\int_{t_0}^{t_n} \frac{dV}{dt}\, dt = \sum_{k=0}^{n-1} \int_{t_k}^{t_{k+1}} \left(\frac{\partial V(x,q)}{\partial x}\right)^T f(x,0;p_k)\, dt + \sum_{k=0}^{n-1} \left(V\left(x(t_{k+1}), q^+(t_{k+1})\right) - V\left(x(t_{k+1}), q(t_{k+1})\right)\right). \tag{24}$$

From (4), we infer

$$\sum_{k=0}^{n-1} \left(V\left(x(t_{k+1}), q^+(t_{k+1})\right) - V\left(x(t_{k+1}), q(t_{k+1})\right)\right) \le 0,$$

since it is the finite sum of non-negative terms. Hence,

$$\int_{t_0}^{t_n} \frac{dV}{dt}\, dt \le \sum_{k=0}^{n-1} \int_{t_k}^{t_{k+1}} \left(\frac{\partial V(x,q)}{\partial x}\right)^T f(x,0;p_k)\, dt$$

Using (23), the right-hand side of above inequality satisfies

$$\sum_{k=0}^{n-1} \int_{t_k}^{t_{k+1}} \left(\frac{\partial V(x,q)}{\partial x}\right)^T f(x,0;p_k)\, dt \le -\int_{t_0}^{t_n} \rho\, dt = -\rho(t_n - t_0), \tag{25}$$

where $\rho = \min_{p \in \mathscr{P}} \rho_p$. That is,

$$\int_{t_0}^{t_n} \frac{dV}{dt}\, dt = V(x(t_n), q(t_n)) - V(x_0, q_0) \le -\rho(t_n - t_0)$$

Re-organizing the terms gives

$$V(x(t_n), q(t_n)) \le V(x_0, q_0) + \rho t_0 - \rho t_n.$$

From (2), we know that $V(x(t_n), q(t_n)) > 0$; therefore, there exists a $t \ge \frac{\rho_0 + V(x_0, q_0)}{\rho}$ such that $V(x(t), q(t)) = 0$. Additionally, since (4) holds, we have $V(x, q^+(t)) \le V(x, q(t))x$. Thus, $x(t) = 0$ for all $q \in \mathcal{Q}$.

B PROOF OF PROPOSITION 1

Multiplying the inequality in (12) from left by $\begin{bmatrix} y \\ d \end{bmatrix}^T$ and right by $\begin{bmatrix} y \\ d \end{bmatrix}$, we obtain

$$\sum_{i=1}^{N} \begin{bmatrix} w_i \\ y_i \end{bmatrix}^T S_i \begin{bmatrix} w_i \\ y_i \end{bmatrix} - \begin{bmatrix} d \\ z \end{bmatrix}^T S \begin{bmatrix} d \\ z \end{bmatrix} \le 0. \tag{26}$$

Similarly, multiplying the inequality in (13) from left by $\begin{bmatrix} p \\ \mu \end{bmatrix}^T$ and right by $\begin{bmatrix} p \\ \mu \end{bmatrix}$ gives

$$\sum_{i=1}^{N} \begin{bmatrix} u_i \\ p_i \end{bmatrix}^T R_i \begin{bmatrix} u_i \\ p_i \end{bmatrix} - \begin{bmatrix} \mu \\ \zeta \end{bmatrix}^T R \begin{bmatrix} \mu \\ \zeta \end{bmatrix} \le 0. \tag{27}$$

Moreover, because $(S_i, R_i) \in \mathcal{L}_i$, the exists storage functions $V_i(x,q),\ i = 1, 2, \ldots, N$ such that

$$\left(\frac{\partial V_i(x_i,q_i)}{\partial x_i}\right)^T f_i(x_i, w_i; p_i) - \begin{bmatrix} w_i \\ y_i \end{bmatrix}^T S_i \begin{bmatrix} w_i \\ y_i \end{bmatrix} \le 0,$$

and

$$V(x_i, q_i^+) - V(x_i, q_i) - \begin{bmatrix} u_i \\ p_i \end{bmatrix}^T R_i \begin{bmatrix} u_i \\ p_i \end{bmatrix} \le 0.$$

If we sum over $i = 1$ to N the above dissipation inequalities and use (26) and (27), we infer

$$\sum_{i=1}^{N} \left(\frac{\partial V_i(x_i,q_i)}{\partial x_i}\right)^T f_i(x_i, w_i; p_i) \le \begin{bmatrix} d \\ z \end{bmatrix}^T S \begin{bmatrix} d \\ z \end{bmatrix},$$

and

$$\sum_{i=1}^{N} \left(V(x_i, q_i^+) - V(x_i, q_i)\right) \le \begin{bmatrix} \mu \\ \zeta \end{bmatrix}^T R \begin{bmatrix} \mu \\ \zeta \end{bmatrix},$$

which implies that the overall system is dissipative with the storage function $V(x, q) = \sum_{i=1}^{N} V_i(x_i, q_i)$.

REFERENCES

[1] A. A. Ahmadi and R. M. Jungers. 2016. Lower bounds on complexity of Lyapunov functions for switched linear systems. *Nonlinear Analysis: Hybrid Systems* 21 (2016), 118 – 129.

[2] M. Ahmadi, H. Mojallali, and R. Wisniewski. 2014. Guaranteed cost H_∞ controller synthesis for switched systems defined on semi-algebraic sets. *Nonlinear Analysis: Hybrid Systems* 11 (2014), 37 – 56.

[3] Rajeev Alur. 2011. Formal Verification of Hybrid Systems. In *Proceedings of the Ninth ACM International Conference on Embedded Software (EMSOFT '11)*. ACM, New York, NY, USA, 273–278.

[4] Erling D Andersen and Knud D Andersen. 2012. The MOSEK optimization software. *EKA Consulting ApS, Denmark* (2012).

[5] K. Bae and S. Gao. 2017. Modular SMT-Based Analysis of Nonlinear Hybrid Systems. In *Formal Methods in Computer-Aided Design*. Vienna, Austria.

[6] S. Bak and S. Chaki. 2016. Verifying cyber-physical systems by combining software model checking with hybrid systems reachability. In *2016 International Conference on Embedded Software (EMSOFT)*. 1–10.

[7] Amir Beck and Marc Teboulle. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* 2, 1 (2009), 183–202.

[8] Stephen Becker, Jérôme Bobin, and Emmanuel J Candès. 2011. NESTA: A fast and accurate first-order method for sparse recovery. *SIAM Journal on Imaging Sciences* 4, 1 (2011), 1–39.

[9] Stephen R Becker, Emmanuel J Candès, and Michael C Grant. 2011. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical programming computation* 3, 3 (2011), 165–218.

[10] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* 3, 1 (2011), 1–122.

[11] Annie I Chen and Asuman Ozdaglar. 2012. A fast distributed proximal-gradient method. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*. IEEE, 601–608.

[12] Tom Goldstein, Brendan O'Donoghue, Simon Setzer, and Richard Baraniuk. 2014. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences* 7, 3 (2014), 1588–1623.

[13] Michael Grant and Stephen Boyd. 2014. CVX: Matlab Software for Disciplined Convex Programming, version 2.1. http://cvxr.com/cvx. (March 2014).

[14] Elaine T Hale, Wotao Yin, and Yin Zhang. 2008. Fixed-point continuation for \ell_1-minimization: Methodology and convergence. *SIAM Journal on Optimization* 19, 3 (2008), 1107–1130.

[15] K. Akbari Hamed and R. D. Gregg. 2017. Decentralized Feedback Controllers for Robust Stabilization of Periodic Orbits of Hybrid Systems: Application to Bipedal Walking. *IEEE Transactions on Control Systems Technology* 25, 4 (2017), 1153–1167.

[16] T. A. Henzinger. 1996. The Theory of Hybrid Automata. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science (LICS '96)*. IEEE Computer Society, Washington, DC, USA, 278–. http://dl.acm.org/citation.cfm?id=788018.788803

[17] N. Jansen, M. Cubuktepe, and U. Topcu. 2017. Synthesis of shared control protocols with provable safety and performance guarantees. In *2017 American Control Conference (ACC)*. 1866–1873.

[18] M. Kamgarpour, T. A. Wood, S. Summers, and J. Lygeros. 2017. Control synthesis for stochastic systems given automata specifications defined by stochastic sets. *Automatica* 76, Supplement C (2017), 177 – 182.

[19] R. Lal and P. Prabhakar. 2016. Compositional analysis of networked hybrid systems using bounded error approximations. In *8th NASA Formal Methods Symposium*. Minneapolis, MN.

[20] Patrick Lincoln and Ashish Tiwari. 2004. *Symbolic Systems Biology: Hybrid Modeling and Analysis of Biological Networks*. Springer Berlin Heidelberg, Berlin, Heidelberg, 660–672.

[21] C. Livadas and N. A. Lynch. 1998. *Formal verification of safety-critical hybrid systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 253–272.

[22] C. Meissen, L. Lessard, M. Arcak, and A. K. Packard. 2015. Compositional performance certification of interconnected systems using ADMM. *Automatica* 61 (2015), 55 – 63.

[23] R. Naldi and R. G. Sanfelice. 2014. Sufficient conditions for passivity and stability of interconnections of hybrid systems using sums of storage functions. In *2014 American Control Conference*. 1432–1437.

[24] Y. Nesterov. 1983. A method of solving a convex programming problem with convergence rate $O(\frac{1}{k^2})$. *Soviet Mathematics Doklady* 27 (1983), 372–376.

[25] Yu Nesterov. 2005. Smooth minimization of non-smooth functions. *Mathematical programming* 103, 1 (2005), 127–152.

[26] Yu Nesterov. 2013. Gradient methods for minimizing composite functions. *Mathematical Programming* 140, 1 (2013), 125–161.

[27] A. Papachristodoulou and S. Prajna. 2009. Robust Stability Analysis of Nonlinear Hybrid Systems. *IEEE Trans. Automat. Control* 54, 5 (May 2009), 1035–1041.

[28] P. Parrilo. 2000. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. Ph.D. Dissertation. California Institute of Technology.

[29] A. Platzer. 2011. *Logic and Compositional Verification of Hybrid Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 28–43.

[30] R. G. Sanfelice. 2011. Interconnections of Hybrid Systems: Some Challenges and Recent Results. *Journal of Nonlinear Systems and Applications* 2, 1-2 (2011), 111–121.

[31] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada. 2017. Secure State Estimation for Cyber-Physical Systems Under Sensor Attacks: A Satisfiability Modulo Theory Approach. *IEEE Trans. Automat. Control* 62, 10 (2017), 4917–4932.

[32] C. Sloth, G. J. Pappas, and R. Wisniewski. 2012. Compositional Safety Analysis Using Barrier Certificates. In *Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control (HSCC '12)*. ACM, New York, NY, USA, 15–24. https://doi.org/10.1145/2185632.2185639

[33] Jos F Sturm. 1999. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software* 11, 1-4 (1999), 625–653.

[34] D. C. Tarraf. 2011. On a control-oriented notion of finite state approximation. In *2011 IEEE International Symposium on Computer-Aided Control System Design (CACSD)*. 1385–1390.

[35] D. C. Tarraf. 2014. An Input-Output Construction of Finite State ρ/μ Approximations for Control Design. *IEEE Trans. Automat. Control* 59, 12 (2014), 3164–3177.

[36] Danielle C Tarraf, Alexandre Megretski, and Munther A Dahleh. 2008. A framework for robust stability of systems over finite alphabets. *IEEE Trans. Automat. Control* 53, 5 (2008), 1133–1146.

[37] D. C. Tarraf, A. Megretski, and M. A. Dahleh. 2008. A Framework for Robust Stability of Systems Over Finite Alphabets. *IEEE Trans. Automat. Control* 53, 5 (2008), 1133–1146.

[38] A. R. Teel. 2010. Asymptotic stability for hybrid systems via decomposition, dissipativity, and detectability. In *49th IEEE Conference on Decision and Control (CDC)*. 7419–7424.

[39] Kim-Chuan Toh, Michael J Todd, and Reha H Tütüncü. 1999. SDPT3 - A MATLAB software package for semidefinite programming, version

1.3. *Optimization methods and software* 11, 1-4 (1999), 545–581.

[40] Ufuk Topcu, Andrew K Packard, and Richard M Murray. 2009. Compositional stability analysis based on dual decomposition. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*. IEEE, 1175–1180.

[41] L. C. Totu, R. Wisniewski, and J. Leth. 2017. Demand Response of a TCL Population Using Switching-Rate Actuation. *IEEE Transactions on Control Systems Technology* 25, 5 (2017), 1537–1551.

[42] J. Zhao and D. J. Hill. 2008. Dissipativity Theory for Switched Systems. *IEEE Trans. Automat. Control* 53, 4 (2008), 941–953.