Bounded Invariant Verification for Time-Delayed Nonlinear Networked Dynamical Systems

Zhenqi Huang, Chuchu Fan, Sayan Mitra
{zhuang25,cfan10,mitras}@illinois.edu

Abstract

We present a technique for bounded invariant verification of nonlinear networked dynamical systems with delayed interconnections. The underlying problem in precise bounded-time verification lies with computing bounds on the sensitivity of trajectories (or solutions) to changes in initial states and inputs of the system. For large networks, computing this sensitivity with precision guarantees is challenging. We introduce the notion of input-to-state (IS) discrepancy of each module or subsystem in a larger nonlinear networked dynamical system. The IS discrepancy bounds the distance between two solutions or trajectories of a module in terms of their initial states and their inputs. Given the IS discrepancy functions of the modules, we show that it is possible to effectively construct a reduced (low dimensional) time-delayed dynamical system, such that the trajectory of this reduced model precisely bounds the distance between the trajectories of the complete network with changed initial states. Using the above results we develop a sound and relatively complete algorithm for bounded invariant verification of networked dynamical systems consisting of nonlinear modules interacting through possibly delayed signals. Finally, we introduce a local version of IS discrepancy and show that it is possible to compute them using only the Lipschitz constant and the Jacobian of the dynamic function of the modules.

I. Introduction

Numerical simulations are extensively used for analyzing nonlinear dynamical systems, yet for models with uncertainty in initial states, parameters and inputs, finite number of simulations alone cannot give proofs for invariant properties. Several recent papers [1]–[5] present simulation-based techniques for proving or disproving properties of such models. The details vary to some extent, but the common theme is to combine finite number of numerical simulations with symbolic static analysis to compute over-approximations of the infinitely many behaviors of the system that may arise from these uncertainties. In other words, the knowledge obtained from the static analysis is used to cover infinitely many behaviors of the system from finite simulation data.

For a dynamical system $\dot{\mathbf{x}} = f(\mathbf{x})$ and a particular initial state $\mathbf{x}$, let $\xi_{\mathbf{x}}$ be the numerically computed trajectory from $\mathbf{x}$ for a certain time bound T. From the continuous dependence of $\xi_{\mathbf{x}}$ on $\mathbf{x}$, we know that all the trajectories from a neighborhood of $\mathbf{x}$ will be close to $\xi_{\mathbf{x}}$. With more information about the sensitivity of the trajectories to the initial state we get better quantitative bounds on the distance between neighboring trajectories. This enables us to compute a tube around $\xi_{\mathbf{x}}$, that contains all possible trajectories from the neighborhood of $\mathbf{x}$. The precision or the conservativeness of this bound impacts the quality of the over-approximation, and therefore, the performance of verification with the above strategy.

For example, the Lipschitz constant of the dynamic function f gives a bound on the distance between the neighboring trajectories that grows exponentially with time. Although checking Lipschitz continuity is generally undecidable, for certain classes of functions Lipschitz constants can be inferred from elementary functions [6]. Stronger notions like sensitivity [1], [7], incremental Lyapunov functions [8], and contraction metrics for dynamical system are used in [9] to obtain more practically useful bounds.

Generalizing several of these properties, in [3] the authors introduced the notion of *discrepancy function* as a continuous function (of the distance between initial states and time) characterizing the convergence or divergence rates of trajectories. It was shown that if a nonlinear, switched, or hybrid system model is annotated with appropriate discrepancy function(s) then the above approach gives a sound and relatively complete algorithm for verifying bounded time invariants.

Until recently, there were no general techniques for computing discrepancy functions (or for that matter, sensitivity, contraction metrics and incremental Lyapunov functions) from the syntactic description of a dynamical system. One typically assumes a template polynomial for the candidate function and then solves an optimization problem to find the coefficients. Finding these annotations becomes increasingly difficult for larger models in which many components interact [10].

In this paper, we address this problem by proposing a compositional approach for automatically computing discrepancy functions for dynamical systems that are created by composing many modules that interact over a (possibly delayed) network. Consider a networked dynamical system $\mathcal{A}$ consisting of several interacting subsystems or modules $\mathcal{A}_1, \ldots, \mathcal{A}_N$. That is, the input signals of a subsystem $\mathcal{A}_i$ are driven by the outputs (or states) of some set of other components. Let's say that each $\mathcal{A}_i$ is n-dimensional which makes $\mathcal{A}$ nN-dimensional. Our solution has several parts. First, building up on our previous work [4], [5] we introduce new type of input-to-state discrepancy function (IS discrepancy) for each subsystem $\mathcal{A}_i$. An IS discrepancy for $\mathcal{A}_i$ (together with its witnesses) gives a bound on the distance between two trajectories as a function of (a) their initial states

Zhenqi Huang's contact information for submission: telephone: +1 2175502866, email: zhuang25@illinois.edu

and (b) the inputs they experience. Using IS discrepancy of the modules we syntactically construct a reduced N-dimensional dynamical system M. If the interconnections in the original network $\mathcal{A}$ has delays then so do the interconnections in the reduced network. We show that the trajectories of M give a discrepancy function for $\mathcal{A}$. We adopt the technique of [11] to this compositional setting for computing a local version of IS discrepancy that uses only the Lipschitz constants and the Jacobian matrices of the modules. This approach of [11] bounds the distance between two trajectories as an exponential function of the eigenvalues of the Jacobian matrix.

Input-to-state stability (ISS), its variants and characterization in terms of necessary and sufficient conditions have been one of the major advances of nonlinear control theory in the last two decades [12]–[14]. Incremental ISS has been used to construct discrete symbolic models that approximately bisimulate continuous systems [15], [16]. Under additional assumptions about the stability of the overall system, such as a small-gain condition, it has been shown that a large system can be reduced to a smaller system with similar behaviors [17], [18]. Our work is the first to connect these ideas to simulation-based safety verification of composed systems. Moreover, our technique does not rely on any stability assumptions of the composed system.

There has been recent work on verification of delayed differential equations (see, for example, [19] and the references therein). The algorithm in [19] iteratively computes validated simulations of delayed differential equations as sequences of Taylor over-approximations of time intervals, and verifies safety property using SMT solvers. Part of its technique can be used in our approach as a mean for computing validated numerical simulations for delayed differential equations. Our approach, in addition, involves systematically generating numerical simulations to refine reach sets and dynamically reasoning about sensitivity of interconnecting networks.

Summary of contributions: (i) We introduce a method for constructing an approximation of a networked dynamical system with time delay ($\mathcal{A}$) using the IS discrepancy functions of the components (Definition 4). Specifically, we use the collection of IS discrepancy functions for the subsystems to define a family of dynamical systems $M(\delta)$, where the parameter δ defines the initial state of M. This approach for delayed networks extends our result for delay-free networks [4]. The extension involves generalizing the model of networked dynamical systems to cover interconnections with delays, which leads to delayed differential equations, and the construction of the corresponding delayed reduced system.

(ii) We show that $M(\delta)$ has a unique trajectory μ, and that any trajectory $\xi_{\mathbf{x}}$ of $\mathcal{A}$ point-wise bloated by the value of $\mu(t)$ contains the reach set of all the trajectories of $\mathcal{A}$ starting from a δ-ball around $\mathbf{x}$ (Theorem 7). Thus, by simulating $\mathcal{A}$ and (the smaller) $M(\delta)$ we can compute bounded-time reach set over-approximations of $\mathcal{A}$.

(iii) We also show that by choosing appropriately small δ's the over-approximations computed by the above method can be made arbitrarily precise; modulo the precision of the numerical simulations (Theorem 10).

(iv) We give an algorithm for computing a local version of IS discrepancy function along a trajectory $\xi_{\mathbf{x}}$ of $\mathcal{A}$ using only the Lipschitz constant and Jacobian matrix of the dynamic mapping of each module.

(v) Using the above results we develop an algorithm for bounded safety verification of nonlinear dynamical systems that iteratively refines initial set partitions (Algorithm 1). We show that the algorithm is sound and is guaranteed to terminate whenever the model is robustly safe or unsafe with respect to a given unsafe set. Our experiments with a prototype implementation of the algorithm show that the approach achieve best performance when verifying networks of stable modules with light inter-modular couplings.

II. Preliminaries

For a natural number $n \in \mathbb{N}$, $[n]$ is the set $\{1, 2, \ldots, n\}$. For a sequence A of objects of any type with n elements, we refer to the i^{th} element, $i \leq n$ by A_i.

We will use the notations from the hybrid Input/Output automaton (HIOA) framework for modeling compositions of dynamical systems [20], [21]. In this framework, named variables are used to refer to state and input components of models. Let V be a finite set of real-valued variables. A *valuation* $\mathbf{v}$ *for* V is a function mapping each variable name to its value in $\mathbb{R}$. The set of valuations for V is denoted by $\mathit{Val}(V)$. For any function $f : A \to B$ and a set $S \subseteq A$, $f \lceil S$ is the restriction of f to S. That is, $(f \lceil S)(s) = f(s)$ for each $s \in S$. So, for a variable $v \in V$ and a valuation $\mathbf{v} \in \mathit{Val}(V)$, $\mathbf{v} \lceil v$ is the function mapping $\{v\}$ to the value $\mathbf{v}(v)$. Trajectories capture the continuous evolution of variable valuations over time. A *trajectory* for V is a continuous function $\tau : \mathbb{R}_{\geq 0} \to \mathit{Val}(V)$. The set of all possible trajectories for V is denoted by $Traj(V)$. For any function $f : C \to [A \to B]$ and a set $S \subseteq A$, $f \downarrow S$ is the restriction of the range of f to S. That is, for any $c \in C$, $(f \downarrow S)(c) = f(c) \lceil S$. In particular, for a variable $v \in V$ and a trajectory $\tau \in Traj(V)$, $\tau \downarrow v$ is the trajectory of v defined by τ.

The results in this paper cover systems with time-delayed trajectories. For notational convenience, we identify trajectories and their delayed versions as follows. For any trajectory $\xi : \mathbb{R}_{\geq 0} \to \mathit{Val}(X)$, we can uniquely define another function $\bar{\xi} : \mathbb{R} \to \mathit{Val}(X)$ such that (i) $\bar{\xi} \lceil \mathbb{R}_{\geq 0} = \xi$, and (ii) for any $t < 0$, $\bar{\xi}(t) = \xi(0)$. That is, $\bar{\xi}$ has identical valuations as ξ over the entire domain of ξ, and takes a constant valuation $\xi(0)$ everywhere else. Sometimes we may need to evaluate ξ at $t < 0$; this actually means the valuation of $\bar{\xi}$ at t, which is by definition $\bar{\xi}(t) = \xi(0)$.

Valuations can be viewed as vectors in $\mathbb{R}^{|V|}$ dimensional space by fixing some arbitrary ordering on variables. For a vector $\mathbf{v} \in \mathbb{R}^n$, $\|\mathbf{v}\|$ is ℓ^2-norm of the vector of variable values. $B_\delta(\mathbf{v}) \subseteq \mathit{Val}(V)$ is the closed ball of valuations with radius δ

centered at $\mathbf{v}$. The notions of continuity, differentiability, and integration are lifted to functions defined over sets of valuations in the usual way.

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *Lipschitz* if there exists a constant $L \geq 0$—called the *Lipschitz constant*—such that for all $a_1, a_2 \in \mathbb{R}^n$, $\|f(a_1) - f(a_2)\| \leq L\|a_1 - a_2\|$. A continuous function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a *class $\mathcal{K}$ function* if $\alpha(0) = 0$ and it is strictly increasing. Class $\mathcal{K}$ functions are closed under composition and inversion. A class $\mathcal{K}$ function α is a *class $\mathcal{K}_\infty$* function if $\alpha(x) \to \infty$ as $x \to \infty$. A continuous function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is called a *class $\mathcal{KL}$ function* if for any t, $\beta(x, t)$ is a class $\mathcal{K}$ function in x and for any x, $\beta(x, t) \to 0$ as $t \to \infty$.

III. Models and Composition

A. Dynamical System Modules

Large system models are built-up by composing smaller component models. In this section, we define component models and the composition operation. A *dynamical system module* is specified by a collection of ordinary differential equations (ODEs), possibly with inputs, and a set of initial states. For reducing notational overhead, we identify output variables with state variables in this paper but our results can be extended to systems where outputs are distinct in a straightforward manner.

Definition 1. *A* dynamical module $\mathcal{A}$ *is a tuple* $\langle X, U, \Theta, f \rangle$ *where*

(i) X *is a set of variables called the* state variables*; valuations of* X *are called* states*;*
(ii) U *is a set of variables called the* input variables *that are distinct from the state variables;*
(iii) $\Theta \subseteq Val(X)$ *is a compact set of* initial states*;*
(iv) $f : \mathbb{R}_{\geq 0} \times Val(X) \times Val(U) \to Val(X)$ *is called the* dynamic mapping*. In addition,* f *is continuous in the first argument and Lipschitz continuous with respect to the other two arguments.*

The *space of input signal* $\mathcal{U}$ is the set $Traj(U)$ of all continuous trajectories of the input variables U. For an input signal $\eta \in \mathcal{U}$ and an initial state $\theta \in \Theta$, the *solution* (or trajectory) of $\mathcal{A}$ is a trajectory of $\xi_{\theta,\eta} : \mathbb{R}_{\geq 0} \to Val(X)$ such that (i) $\xi(0) = \theta$, and (ii) for any $t \in \mathbb{R}_{\geq 0}$, the derivative of ξ satisfies the differential equation

$$\dot{\xi}(t) = f(t, \xi(t), \eta(t)). \tag{1}$$

As in Equation (1), we will suppress the subscripts of ξ when the dependence on the initial state and the input trajectory are clear from context.

Remark 1. *Safety verification problems commonly involve computing the set of trajectories up to some bounded time $T > 0$. The (global) Lipschitz assumption of the dynamic mapping f and the piecewise continuity of η guarantee that the differential equation (1) admits a unique global solution for interval $[0, T]$ with any time bound $T > 0$ and any initial state $\theta \in Val(X)$. In case the trajectories of the system are confined to an invariant set $S \subseteq Val(X)$, the results in this paper would hold for a locally Lipschitz continuous f with L as the Lipschitz constant over S.*

A module $\mathcal{A}$ without inputs ($U = \varnothing$) is said to be *closed*; otherwise, $\mathcal{A}$ is *open*. The set of all trajectories of $\mathcal{A}$ with respect to a set of initial states $\Theta' \subseteq \Theta$ and a set of input signals $\mathcal{U}' \subseteq \mathcal{U}$ is denoted by $Traj(\mathcal{A}, \Theta', \mathcal{U}')$. We will drop the argument $\mathcal{U}'$ for closed modules. The components of modules $\mathcal{A}$ and $\mathcal{A}_i$ are denoted by $X_\mathcal{A}, U_\mathcal{A}, \Theta_\mathcal{A}, f_\mathcal{A}$ and X_i, U_i, Θ_i, f_i, respectively.

Example 1. *FitzHugh-Nagumo (FHN) model [22] is a generic model for excitable media and can be applied to cardiac cells. The model has two state variables $X = \{x, v\}$, where x models the membrane voltage of a cardiac cell and v is an internal variable. Initially, the variables x, v take values in the ranges $[1.4, 1.6]$ and $[0, 0.2]$ respectively. The input variables for an FHN model is $U = \{u_1, u_2, u_3\}$, where u_1, u_2 model the voltages of two neighboring cells and u_3 models the input pulse form a pacemaker. The dynamic of the model is:*

$$\begin{aligned} \dot{x} &= -x^3 + 1.1x^2 - 0.12x - v + 0.01u_1 + 0.01u_2 + u_3 \\ \dot{v} &= 0.005x - 0.01v \end{aligned} \tag{2}$$

Next, for a closed module $\mathcal{A}$ we define reachable states and safety. A state $\mathbf{x} \in Val(X)$ is *T-reachable* if there exists a trajectory $\xi \in Traj(\mathcal{A}, \Theta)$ and a time $t \leq T$ such that the trajectory $\xi(t) = \mathbf{x}$. The set of T-reachable states is denoted by $\mathsf{Reach}_\mathcal{A}(\Theta, T) = \{\xi(t) : t \leq T, \xi \in Traj(\mathcal{A}, \Theta)\}$.

Definition 2. *For $\epsilon \geq 0$ and time $T \geq 0$, and an open unsafe set $\mathbb{U} \subseteq Val(X)$, $\mathcal{A}$ is* ϵ-robustly safe up to T with respect to $\mathbb{U}$ *if $B_\epsilon(\mathsf{Reach}_\mathcal{A}(\Theta, T)) \cap \mathbb{U} = \varnothing$. If there exists some $\epsilon > 0$ for which this condition holds, then $\mathcal{A}$ is* robustly safe up to T with respect to $\mathbb{U}$.

B. Networked Dynamical Systems with Delays

Formally, the composition operation takes a collection of modules and defines a new dynamical system by plugging-in or identifying the input variables of one subsystem with state variables of another. The resulting system may still have input variables that are not identified with any of the state variables. A collection of dynamical modules $\mathcal{A} = \{\mathcal{A}_i\}_{i\in[N]}$ are *compatible* if they do not share any of the state variables. That is, for any $i, j \in [N]$, $X_i \cap X_j = \varnothing$. This condition merely prevents any unforeseen identification of states and inputs. The collection is *closed* if $\cup_{i\in[N]} U_i \subseteq \cup_{i\in[N]} X_i$; that is, as a whole, the network has no free input variable. In this paper, we will develop compositional techniques for analyzing closed networks.

In general, when the output signals from one module are plugged into the input of another module, the latter may receive these inputs with some delay. Delays arise from transmission, processing, and buffering. In this paper, we assume that all the signals from one module to another are identically delayed. The signals from the first module to a third module may be delayed by a different amount from the delay experience by the second module. This is a reasonable assumption when the delays are dominated by module-level effects as opposed to the individual variable related effects.

For a collection of N modules $\{\mathcal{A}_i\}_{i\in[N]}$, a *delay matrix* $d \in {\mathbb{R}_{\geq 0}}^{N\times N}$ gives for each ordered pair (i, j) a non-negative time-delay constant $d(i, j)$. All the input signal from module $\mathcal{A}_i$ to module $\mathcal{A}_j$ experience the identical delay of $d(i, j)$. Now that we have defined dynamical modules and the delay matrix for interconnecting them, we proceed to defining dynamic networks with delays:

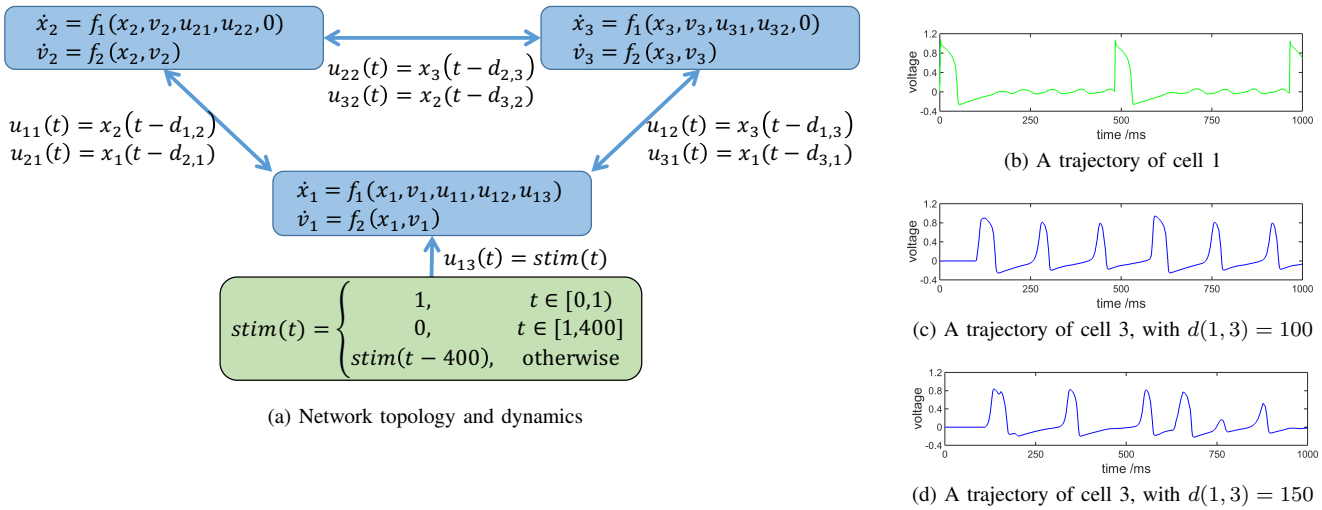
Definition 3. *For a compatible and closed collection of dynamical modules $\{\mathcal{A}_i\}_{i\in[N]}$, and a delay matrix $d \in {\mathbb{R}_{\geq 0}}^{N\times N}$, a* networked dynamical system*, denoted by $\mathcal{A} = \|_d\{\mathcal{A}_i\}_{i\in[N]}$, is the tuple $\langle X, \Theta, \mathcal{T}\rangle$, where*

(i) $X = \cup_{i\in[N]} X_i$,
(ii) $\Theta = \{\theta \in Val(X) \mid \theta \lceil X_i \in \Theta_i,$ *for each* $i\}$, *and*
(iii) $\mathcal{T}$ *is a set of trajectories of* $\mathcal{A}$ *such that* $\xi \in \mathcal{T}$ *if and only if for each* $i \in [N]$, $t \in \mathbb{R}_{\geq 0}$,

$$\dot{\xi}_i(t) = f_i(t, \xi_i(t), \eta_i(t)), \tag{3}$$

where $\xi_i = \xi \downarrow X_i$ *and for each input variable* $u \in U_i \cap X_j$ *from* j *to* i, $\eta_i(t) \lceil u = \xi_j(t - d(i, j)) \lceil u$.

Recall that for any $i, j \in [N]$ and any time $t < d(i, j)$ we defined $\xi_j(t - d(i, j)) = \xi_j(0)$ in Section II. For a networked dynamical system $\mathcal{A}$ defined above, let $d_M = \max_{i,j} d(i, j)$ be the maximum delay and $\mathcal{C} = C([-d_M, 0], \mathbb{R}^{|X|})$ be the space of continuous functions from $[-d_M, 0]$ to $\mathbb{R}^{|X|}$. The trajectories of the network defined in Equation (3) are solutions of the *delayed differential equation* [23] $\dot{\xi}(t) = f(t, H_{\xi,d_M,t})$ with $f : \mathbb{R} \times \mathcal{C} \to \mathbb{R}^{|X|}$ and $H_{\xi,d_M,t}(s) = \xi(t + s), \forall s \in [-d_M, 0]$ such that f is an aggregate of all the N Equations (3). Since each f_i is continuous in t and Lipschitz in the other arguments, we can verify that f is continuous in t and Lipschitz in $H_{\xi,d_M,t}$[1]. From Theorem 2.1, 2.2 in [23], for any initial state $\theta \in \Theta$, a networked dynamical system $\mathcal{A}$ has a unique trajectory ξ. With the set of trajectories $Traj(\mathcal{A}, \Theta)$ as defined above, the set of bounded time reachable states $\mathsf{Reach}_{\mathcal{A}}(\Theta, T)$ of the closed networked dynamical system $\mathcal{A}$ is defined analogously to that of closed dynamical modules.



(a) Network topology and dynamics

(b) A trajectory of cell 1

(c) A trajectory of cell 3, with $d(1, 3) = 100$

(d) A trajectory of cell 3, with $d(1, 3) = 150$

Fig. 1: A networked dynamical system with three cells and a pacemaker and its sample trajectories

[1]The norm of $H \in \mathcal{C}$ is defined as $\|H\| = \sup_{t\in[-d_M,0]} \|H(t)\|$.

Example 2. *We consider a network of cardiac cells $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ stimulated by a pacemaker, as illustrated in Figure 1(a). The cardiac cells follow the FHN model presented in Example 1. The network is defined by associating input of each cell $\mathcal{A}_i$ to the state of another $\mathcal{A}_j$ with a nonnegative delay $d_{i,j}$. The pacemaker is modeled as a rectangular wave generator with period 450 and duty cycle 2. We simulate the network by assigning the delays $d(1,2) = d(2,1) = 70$, $d(2,3) = d(3,2) = 40$ and $d(3,1) = 100$ with two different delay values of $d(1,3)$. A trajectory of cell 1 with $d(1,3) = 100$ is plotted in Figure 1(b). Two trajectories of cell 3 with $d(1,3) = 100$ and $d(1,3) = 150$ are plotted in (c) and (d) respectively. Not only the trajectory of cell 3 is shifted in time due to different delay value, but the shape of the trajectories also changes due to the combined inputs from cell 1 and 2.*

IV. Input-to-State Discrepancy

In this section, we introduce the notion of input-to-state (IS) discrepancy functions for dynamical system modules and present three approaches for finding them. In Section V, we will use IS discrepancy to develop the approaches for computing over-approximations of reachable states. We will present a variation of IS discrepancy function in Section VII that can be computed along a trajectory relative easily.

A. Input to State Discrepancy

Roughly, input-to-state discrepancy (IS discrepancy) of a module $\mathcal{A}$ bounds the distance between two trajectories of $\mathcal{A}$ in terms of different initial states and different inputs.

Definition 4. *(IS discrepancy) For a dynamical module $\mathcal{A} = \langle X, U, \Theta, f\rangle$, a continuous function $V : Val(X)^2 \to \mathbb{R}_{\geq 0}$ is an* input-to-state discrepancy function *if*

(i) *$\exists$ class-$\mathcal{K}$ functions $\underline{\alpha}, \overline{\alpha}$ such that for any $\mathbf{x}, \mathbf{x}' \in Val(X)$, $\underline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|) \leq V(\mathbf{x}, \mathbf{x}') \leq \overline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|)$, and*

(ii) *$\exists \beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ of class-$\mathcal{K}$ in the first argument and a class-$\mathcal{K}$ function γ such that for any pair of initial states $\theta, \theta' \in \Theta$, and pair of input trajectories $\eta, \eta' \in Traj(U)$, and $t \in \mathbb{R}_{\geq 0}$,*

$$V(\xi(t), \xi'(t)) \leq \beta(\|\theta - \theta'\|, t) + \int_0^t \gamma(\|\eta(s) - \eta'(s)\|)ds, \tag{4}$$

where $\xi = Traj(\mathcal{A}, \theta, \eta)$ and $\xi' = Traj(\mathcal{A}, \theta', \eta')$.

$(\underline{\alpha}, \overline{\alpha}, \beta, \gamma)$ are called the witnesses *of the IS discrepancy function.*

In the rest of the paper, we make a technical assumption that $\underline{\alpha}^{-1}$ and γ are Lipschitz, and $\beta(\cdot, \cdot)$ has a Lipschitz continuous derivative in the second argument. These assumptions enable us to construct a reduced model with well-defined trajectories in Section V-A.

Remark 2. *The discrepancy function (and its witnesses) bounds the maximum distance between two trajectories in terms of the ℓ^2 distance between their input signals and their initial states. This type of discrepancy function is motivated by the notion of incremental integral input-to-state stability of dynamical modules [14], except that we do not require the $\beta(\cdot, t)$ in (4) converges to 0 as $t \to \infty$. We made technical assumptions on the Lipschitz continuity of the witness functions. In Section VII, we propose an algorithm that computes local version of discrepancy functions and witnesses that satisfy all these assumptions.*

B. Finding IS discrepancy

In what follows, we describe three well-known methods for obtaining input-to-state stability proofs and here we note that they can be used to find IS discrepancy functions. In Remark 3, we discuss the limitations of these heuristics. An automatic and on-the-fly algorithm for finding a local version of IS discrepancy will be introduced in Section VII.

Lipschitz Dynamics: For any dynamical module $\mathcal{A}$ with Lipschitz continuous dynamic mapping f and for any bounded time, we can find an IS discrepancy function of $\mathcal{A}$ for that time bound. This version of IS discrepancy will be sufficient for bounded safety proofs. We note that the problem of computing Lipschitz constant for a function is generally undecidable. However, for many elementary functions such as trigonometric functions and polynomial functions, the Lipschitz constant can be computed for any compact set [6]. There are also some heuristics to estimate Lipschitz constant for general functions [24].

Proposition 1. *Suppose the dynamic mapping f of module $\mathcal{A}$ is Lipschitz in both arguments with Lipschitz constant L. For any time bound $T > 0$, $V(\mathbf{x}, \mathbf{x}') \triangleq \|\mathbf{x} - \mathbf{x}'\|$ is a discrepancy function with witnesses $(\underline{\alpha}, \overline{\alpha}, \beta, \gamma)$ where $\underline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|) = \overline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|) = \|\mathbf{x} - \mathbf{x}'\|$, $\beta(\|\theta - \theta'\|, t) = e^{Lt}\|\theta - \theta'\|$ and $\gamma(\|\mathbf{u} - \mathbf{u}'\|) = Le^{LT}\|\mathbf{u} - \mathbf{u}'\|$.*

Stable Linear Dynamics: Suppose $\mathcal{A}$ has dynamic mapping $f(\mathbf{x},\mathbf{u}) = C\mathbf{x} + D\mathbf{u}$, where C is a $n \times n$ matrix and D is a $n \times m$ matrix. If C is asymptotically stable, then its input-free trajectories converge exponentially and we can get an IS dependency function with exponentially convergent witness β.

Proposition 2. *For linear module $\mathcal{A}$ with dynamic mapping $f(\mathbf{x},\mathbf{u}) = C\mathbf{x} + D\mathbf{u}$ with asymptotically stable matrix C, there exists $\lambda > 0$, such that $V(\mathbf{x},\mathbf{x}') = \|\mathbf{x}-\mathbf{x}'\|$ is a discrepancy function with witnesses $(\underline{\alpha},\overline{\alpha},\beta,\gamma)$ where $\underline{\alpha}(\|\mathbf{x}-\mathbf{x}'\|) = \overline{\alpha}(\|\mathbf{x}-\mathbf{x}'\|) = \|\mathbf{x}-\mathbf{x}'\|$, $\beta(\|\theta-\theta'\|,t) = e^{-\lambda t}\|\theta-\theta'\|$ and $\gamma(\|\mathbf{u}-\mathbf{u}'\|) = \|D\|\|\mathbf{u}-\mathbf{u}'\|$.*

The positive constant λ can be computed by solving Lyapunov equation [25].

Incremental Integral ISS: The notion of incremental integral input-to-state stability (incremental integral ISS) of dynamical modules [14] is a generalization of the standard notion of input-to-state stability [8], [12], [13]. A Lyapunov like theorem of proving incremental integral ISS as well as a converse Lyapunov theorem are presented in [14]. Given a proof of an incremental integral ISS property of a module, we automatically get its IS discrepancy function (with witnesses).

Definition 5. *A dynamical module $\mathcal{A}$ is called incremental-integral-input-to-state stable (δiISS) if there exists a class-$\mathcal{K}_\infty$ function α, a class-$\mathcal{KL}$ function β and a class-$\mathcal{K}$ function γ such that, for any initial states $\theta,\theta' \in \Theta_\mathcal{A}$, for any input signal $\eta,\eta' \in U_\mathcal{A}$ and any $t > 0$,*

$$\alpha(\|\xi(t)-\xi'(t)\|) \leq \beta(\|\theta-\theta'\|,t) + \int_0^t \gamma(\|\eta(s)-\eta'(s)\|)ds. \tag{5}$$

where $\xi = Traj(\mathcal{A},\theta,\eta)$ and $\xi' = Traj(\mathcal{A},\theta',\eta')$.

Proposition 3. *Given an incremental integral ISS module $\mathcal{A}$ with (α,β,γ) as in Definition 5, then $V(\mathbf{x},\mathbf{x}') = \alpha(\|\mathbf{x}-\mathbf{x}'\|)$ is a discrepancy function with witnesses $(\underline{\alpha},\overline{\alpha},\beta,\gamma)$ where $\underline{\alpha}(\|\mathbf{x}-\mathbf{x}'\|) = \overline{\alpha}(\|\mathbf{x}-\mathbf{x}'\|) = \alpha(\|\mathbf{x}-\mathbf{x}'\|)$, and β,γ given above.*

Remark 3. *Proposition 1 establishes an IS discrepancy function only using the Lipschitz constant of the dynamic mapping f. Although, computing Lipschitz constants of arbitrary functions is undecidable in general, for certain classes of polynomial functions it can be estimated [6]. However, even if the dynamics of the module is stable, IS discrepancy functions obtained by this method have witnesses β and γ that grow exponentially with time. Incremental-integral-input-to-state stability (Definition 5) can be proved by constructing an incremental Lyapunov function [14]. For some dynamical models with physical implications, the incrememtal Lyapunov function may capture the energy of the system. However, constructing the incremental Lyapunov function is in general a hard problem. For modules with linear dynamics, IS discrepancy can be computed automatically using Proposition 2.*

In Section VII, we will propose a local version of IS discrepancy function, which can be computed automatically from the Lipschitz constant and the Jacobian matrix of the dynamic mapping f.

V. Small Approximations from IS Discrepancy

In this section we construct scalar (one-dimensional) approximations for an $|X_i|$-dimensional dynamical module $\mathcal{A}_i = \langle X_i, U_i, \Theta_i, f_i\rangle$ (Definition 6) in a network, using input-to-state discrepancy functions. For the sake of a cleaner presentation, we develop the results for a network consisting of two $|X_1|$ and $|X_2|$-dimensional modules $\mathcal{A}_1$ and $\mathcal{A}_2$ with a two-dimensional approximation. The general results follow by straightforward extension and are stated in Section V-D.

A. IS Approximation of $\|_d\{\mathcal{A}_1,\mathcal{A}_2\}$

Consider closed networked dynamical system $\mathcal{A} = \|_d\{\mathcal{A}_1,\mathcal{A}_2\}$ composed of two modules $\mathcal{A}_1 = \langle X_1,U_1,\Theta_1,f_1\rangle$ and $\mathcal{A}_2 = \langle X_2,U_2,\Theta_2,f_2\rangle$ and a delay matrix $d \in {\mathbb{R}_{\geq 0}}^{2\times 2}$. The input signals U_2 of $\mathcal{A}_2$ are obtained from X_1 delayed by $d(2,1)$ and the input signals U_1 of $\mathcal{A}_1$ are obtained from X_2 delayed by $d(1,2)$. Let V_i be an IS discrepancy function for $\mathcal{A}_i, i \in \{1,2\}$ with witness $(\underline{\alpha}_i,\overline{\alpha}_i,\beta_i,\gamma_i)$. For any pair of initial states θ,θ' in $\Theta_\mathcal{A}$, let $\xi = Traj(\mathcal{A},\theta)$ and $\xi' = Traj(\mathcal{A},\theta')$ be the unique trajectories of the network $\mathcal{A}$ starting from θ and θ' respectively. We define $\theta_i = \theta \lceil X_i$, $\theta_i' = \theta' \lceil X_i$, $\xi_i = \xi \downarrow X_i$ and $\xi_i' = \xi' \downarrow X_i$. From Definition 3, the restriction of ξ_i and ξ_i' to X_i are trajectories of $\mathcal{A}_i$ from θ_i and θ_i'. From Definition 4, for every $t \in [0,T]$ the following holds:

$$\begin{aligned} V_1(\xi_1(t),\xi_1'(t)) &\leq \beta_1(\|\theta_1-\theta_1'\|,t) + \int_0^t \gamma_1(\|\xi_2(s-d(1,2))-\xi_2'(s-d(1,2))\|)ds, \\ V_2(\xi_2(t),\xi_2'(t)) &\leq \beta_2(\|\theta_2-\theta_2'\|,t) + \int_0^t \gamma_2(\|\xi_1(s-d(2,1))-\xi_1'(s-d(2,1))\|)ds. \end{aligned} \tag{6}$$

Recall that when a trajectory ξ is evaluated at $t < 0$, we use the valuation $\xi(t) = \xi(0)$. Next, we introduce the key notion of a family of IS approximations for $\mathcal{A}$. Each approximation is parameterized by nonnegative reals $\delta_1,\delta_2 \geq 0$ and is a closed networked dynamical system M with two main variables m_1 and m_2. As we shall see in Theorem 7, at any time t, m_1 gives an upper-bound on the distance between two state trajectories of $\mathcal{A}_1$ that start from neighboring states at most δ_1 apart. Similarly,

m_2 gives an upper-bound on neighboring trajectories of $\mathcal{A}_2$. Of course, the distance between two neighboring state trajectories of $\mathcal{A}_1$ depends on (a) their initial states and (b) on the inputs they experience. These inputs in turn are delayed versions of the state trajectories of $\mathcal{A}_2$. So, the dynamics of m_1 (and m_2) takes into account the impact of both of these factors using the witnesses of the IS discrepancy functions. Since β_1 and β_2 witnesses bound the impact of initial states on the discrepancy as a function of time, the dynamics of m_1 (and m_2) are time varying.

Definition 6. *(IS approximation or reduced model) For a pair of nonnegative constants (δ_1, δ_2), the (δ_1, δ_2)-*IS approximation *of a networked dynamical system $\mathcal{A} = \|_d\{\mathcal{A}_1, \mathcal{A}_2\}$ is a closed networked dynamical system $M = \langle X_M, \Theta_M, \mathcal{T}_M\rangle$ where*

(i) $X_M = \{m_1, m_2\}$,
(ii) $\Theta_M = \{\theta\}$ *where* $\theta \lceil m_i = \beta_i(\delta_i, 0)$, *for* $i \in \{1, 2\}$, *and*
(iii) for any trajectory μ of X_M, $\mu \in \mathcal{T}_M$ if and only if

$$\begin{aligned}\dot{\mu}_1(t) &= \dot{\beta}_1(\delta_1, t) + \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(t - d(1,2)))\\ \dot{\mu}_2(t) &= \dot{\beta}_2(\delta_2, t) + \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_1(t - d(2,1))).\end{aligned} \tag{7}$$

Here $\mu_i = \mu \downarrow m_i$ and $\dot{\beta}_i(\delta_i, t)$ is a short hand for $\frac{\partial}{\partial t}\beta_i(\delta_i, t)$.

The dynamics of the closed reduced network M is defined syntactically in terms of the IS-discrepancy functions of the dynamic modules $\mathcal{A}_1$ and $\mathcal{A}_2$ and the delay matrix. The witness functions $\underline{\alpha}_1^{-1}, \gamma_1$, etc., are Lipschitz continuous and Lipschitz functions are closed under composition. Therefore, the delay differential equation (7) has a unique solution. Since M has a single initial state, it is completely deterministic. Note that both the initial state and the dynamics of M depend on the choice of the parameters δ_1 and δ_2. In Theorem 7 we relate m_1 and m_2 with the divergence between trajectories of $\mathcal{A}_1$ and $\mathcal{A}_2$. Specifically, if μ is the trajectory of a (δ_1, δ_2)-IS approximation then $\mu_i = \mu \downarrow m_i(t)$ gives an upper bound on the distance between the trajectories of $\mathcal{A}_i$ starting from initial states that are at most δ_i apart.

B. Over-approximation with IS Discrepancy

For any pair of non-negative reals $\delta = (\delta_1, \delta_2)$ and any state $\mathbf{x} \in \mathit{Val}(X_\mathcal{A})$, we define

$$B_\delta(\mathbf{x}) = B_{\delta_1}(\mathbf{x} \lceil X_1) \times B_{\delta_2}(\mathbf{x} \lceil X_2)$$

as the product of the δ_i-balls around $\mathbf{x} \lceil X_i$. Given a pair of discrepancy functions $V = (V_1, V_2)$ for $\mathcal{A}_1$ and $\mathcal{A}_2$, a state $\mathbf{m} \in \mathit{Val}(X_M)$ of M naturally defines a sublevel set $L_V(\mathbf{m}) \subseteq \mathit{Val}(X_A) \times \mathit{Val}(X_A)$:

$$L_V(\mathbf{m}) = \{(\mathbf{x}, \mathbf{x}') \mid \forall i \in \{1, 2\},\ V_i(\mathbf{x} \lceil X_i, \mathbf{x}' \lceil X_i) \leq \mathbf{m} \lceil m_i\}.$$

This set is the intersection of the $(\mathbf{m} \lceil m_i)$-*sublevel sets* of V_i. For a state $\mathbf{x} \in \mathit{Val}(X_\mathcal{A})$ of $\mathcal{A}$ and a state $\mathbf{m} \in \mathit{Val}(X_M)$ we define

$$B^V_\mathbf{m}(\mathbf{x}) = \{\mathbf{x}' \in \mathit{Val}(X_\mathcal{A}) \mid (\mathbf{x}, \mathbf{x}') \in L_V(\mathbf{m})\}$$

as the subset of states of $\mathcal{A}$ for which $(\mathbf{x}, \mathbf{x}')$ lies in the sublevel set defined by $\mathbf{m}$.

We will now prove a sequence of results that ultimately established in Lemma 6 that the trajectory of M gives an upper bound on the discrepancy of $\mathcal{A}$. That is, at any time $t \geq 0$

$$\begin{aligned}\beta_1(\|\theta_1 - \theta_1'\|, t) + \int_0^t \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds \leq \mu_1(t)\\ \beta_2(\|\theta_2 - \theta_2'\|, t) + \int_0^t \gamma_2(\|\xi_1(s - d(2,1)) - \xi_1'(s - d(2,1))\|)ds \leq \mu_2(t).\end{aligned} \tag{8}$$

From the construction of M, we observe that at time $t = 0$, the above inequalities hold. Moreover, the first derivatives of the right-hand sides upper bound those of the left-hand sides at time $t = 0$. However, this property at $t = 0$ does not immediately generalize to all $t > 0$. In our proof, we first construct a strict upper bound of the left-hand sides of (8) that holds for all t, and then show that this bound converges to $\mu(\cdot)$.

First, for any positive $\epsilon > 0$, we construct a pair of ϵ-factor trajectories $(\mu_{1\epsilon}, \mu_{2\epsilon})$ with derivatives ϵ-close to the trajectory of μ in (7) and show that these trajectories strictly upper-bound the discrepancy functions of V_1 and V_2.

For any $\delta_1, \delta_2 \geq 0$ and any $\epsilon > 0$, a pair of trajectories $\mu_{1\epsilon}, \mu_{2\epsilon} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are defined as solutions to the differential equations:

$$\begin{aligned}\dot{\mu}_{1\epsilon}(t) &= \dot{\beta}_1(\delta_1, t) + \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{2\epsilon}(t - d(1,2))) + \epsilon, \text{ and}\\ \dot{\mu}_{2\epsilon}(t) &= \dot{\beta}_2(\delta_2, t) + \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{1\epsilon}(t - d(2,1))) + \epsilon,\end{aligned} \tag{9}$$

with $\mu_{1\epsilon}(0) = \beta_1(\delta_1, 0) + \epsilon$ and $\mu_{2\epsilon}(0) = \beta_2(\delta_2, 0) + \epsilon$. The right-hand side of Equation (9) is Lipschitz, and therefore, the solutions $\mu_{1\epsilon}$ and $\mu_{2\epsilon}$ are well-defined differentiable functions of time. For any two initial states of $\mathcal{A}$, θ, θ', we define two differentiable functions $g_1, g_2 : \mathbb{R}_{\geq 0} \to \mathbb{R}$:

$$\begin{aligned} g_1(t) &= \mu_{1\epsilon}(t) - \beta_1(\delta_1, t) - \int_0^t \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds, \\ g_2(t) &= \mu_{2\epsilon}(t) - \beta_2(\delta_2, t) - \int_0^t \gamma_2(\|\xi_1(s - d(2,1)) - \xi_1'(s - d(2,1))\|)ds. \end{aligned} \tag{10}$$

Recall that $\xi = Traj(\mathcal{A}, \theta)$ and $\xi' = Traj(\mathcal{A}, \theta')$ are the trajectories of $\mathcal{A}$ starting from θ and θ'. The following proposition states that if $g_1(t), g_2(t)$ are positive for all $t \geq 0$, then the distance between trajectories ξ and ξ' are bounded by $(\mu_{1\epsilon}, \mu_{2\epsilon})$.

Proposition 4. *Consider any non-negative pair $\delta = (\delta_1, \delta_2)$ and initial states $\theta, \theta' \in \Theta_{\mathcal{A}}$ such that $\theta' \in B_\delta(\theta)$. Let $\xi = Traj(\mathcal{A}, \theta)$ and $\xi' = Traj(\mathcal{A}, \theta')$. Then, for any $\epsilon > 0, t \geq 0$, if $g_1(t), g_2(t) > 0$, then*

$$V_1(\xi_1(t), \xi_1'(t)) < \mu_{1\epsilon}(t), \text{ and } V_2(\xi_2(t), \xi_2'(t)) < \mu_{2\epsilon}(t).$$

Proof. Here we prove the bound for V_1; the bound for V_2 follows by symmetry. For any $t \geq 0$, since $g_1(t) > 0$, from Equation (10) we have

$$\mu_{1\epsilon}(t) > \beta_1(\delta_1, t) + \int_0^t \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds. \tag{11}$$

From $\theta' \in B_\delta(\theta)$, we have $\|\theta_1 - \theta_1'\| \leq \delta_1$. Since $\beta_1(\cdot, t)$ is a class-$\mathcal{K}$ function, it follows that

$$\beta_1(\delta_1, t) \geq \beta_1(\|\theta_1 - \theta_1'\|, t).$$

Thus, Equation (11) becomes

$$\mu_{1\epsilon}(t) > \beta_1(\|\theta_1 - \theta_1'\|, t) + \int_0^t \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds.$$

By applying Equation (6), it follows that

$$\mu_{1\epsilon}(t) > \beta_1(\|\theta_1 - \theta_1'\|, t) + \int_0^t \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds \geq V_1(\xi_1(t), \xi_1'(t)).$$

□

The next lemma establishes that we can drop the assumption about the positivity of g_1 and g_2 and still arrive at the conclusion of Proposition 4.

Lemma 5. *Consider any non-negative pair $\delta = (\delta_1, \delta_2)$, and initial states $\theta, \theta' \in \Theta_A$ such that $\theta' \in B_\delta(\theta)$. Let $\xi = Traj(\mathcal{A}, \theta)$ and $\xi' = Traj(\mathcal{A}, \theta')$. Then, for any $\epsilon > 0, t \geq 0$,*

$$V_1(\xi_1(t), \xi_1'(t)) < \mu_{1\epsilon}(t) \text{ and } V_2(\xi_2(t), \xi_2'(t)) < \mu_{2\epsilon}(t).$$

Proof. By Proposition 4, it suffices to prove that for all $t \geq 0$, $g_1(t), g_2(t) > 0$. At $t = 0$, by (10)

$$g_1(0) = \beta_1(\delta_1, 0) + \epsilon - \beta_1(\delta_1, 0) = \epsilon > 0.$$

Similarly, $g_2(0) > 0$. Suppose for the sake of contradiction that $t_a > 0$ is the first time when $g_1(t), g_2(t) > 0$ is violated. From the continuity of g_1, g_2, we have that the both of the following conditions hold:

(i) $g_1(t), g_2(t) > 0$ for all $t \in [0, t_a)$, and
(ii) $g_1(t_a) = 0$ or $g_2(t_a) = 0$.

Without loss of generality, we assume $g_1(t_a) = 0$. From the Mean value theorem, we know that there exists some time $t_b \in (0, t_a)$ such that

$$\dot{g}_1(t_b) = \frac{g_1(0) - g_1(t_a)}{0 - t_a} \leq -\frac{\epsilon}{t_a} < 0. \tag{12}$$

We can bound the derivative $\dot{g}_1(t_b)$ as:

$$\dot{g}_1(t_b) = \dot{\mu}_{1\epsilon}(t_b) - \frac{d}{dt}\left(\beta_1(\delta_1, t_b) + \int_0^{t_b} \gamma_1(\|\xi_2(s - d(1,2)) - \xi_2'(s - d(1,2))\|)ds\right).$$

Plugging the right-hand side of Equation (9) into the above equation, it follows:

$$\begin{aligned} \dot{g}_1(t_b) &= \dot{\beta}_1(\delta_1, t_b) + \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{2\epsilon}(t_b - d(1,2))) + \epsilon - \dot{\beta}_1(\delta_1, t_b) - \gamma_1(\|\xi_2(t_b - d(1,2)) - \xi_2'(t_b - d(1,2))\|) \\ &= \epsilon + \left(\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{2\epsilon}(t_b - d(1,2))) - \gamma_1(\|\xi_2(t_b - d(1,2)) - \xi_2'(t_b - d(1,2))\|)\right). \end{aligned} \tag{13}$$

From condition (i), we know $g_2(t_b - d(1,2)) > 0$. It follows from Proposition 4 that

$$\mu_{2\epsilon}(t_b - d(1,2)) > V_2(\xi_2(t_b - d(1,2)), \xi_2'(t_b - d(1,2))). \tag{14}$$

From Definition 4, we have $V_2(\xi_2(t_b), \xi_2'(t_b)) \geq \underline{\alpha}_2(\|\xi_2(t_b - d(1,2)) - \xi_2'(t_b - d(1,2))\|)$. Equation (14) yields $\mu_{2\epsilon}(t_b - d(1,2)) > \underline{\alpha}_2(\|\xi_2(t_b - d(1,2)) - \xi_2'(t_b - d(1,2))\|)$. Because $\gamma \circ \underline{\alpha}_2^{-1}$ is a class-$\mathcal{K}$ function, it follows that

$$\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{2\epsilon}(t_b - d(1,2))) \geq \gamma_1(\|\xi_2(t_b - d(1,2)) - \xi_2'(t_b - d(1,2))\|).$$

Combining the above equation with Equation (13), we have that $\dot{g}_1(t_b) \geq \epsilon > 0$, which contradicts to Equation (12). □

Lemma 5 shows that for any $\epsilon > 0$, the ϵ-factor trajectories $\mu_{1\epsilon}$ and $\mu_{2\epsilon}$ give strict upper bounds on the distance between corresponding trajectories of the original modules $\mathcal{A}_1$ and $\mathcal{A}_2$. In the following lemma, we show that as $\epsilon \to 0$, $\mu_{i\epsilon}$ converges to the trajectory $\mu \downarrow m_i$; recall that μ is the trajectory of the IS approximation M. It follows that the trajectory μ indeed bounds the divergence of any trajectories of $\mathcal{A}$.

Lemma 6. *Consider any non-negative pair $\delta = (\delta_1, \delta_2)$ and initial states $\theta, \theta' \in \Theta_{\mathcal{A}}$ such that $\theta' \in B_\delta(\theta)$. Let $\xi = Traj((\mathcal{A}, \theta))$, $\xi' = Traj((\mathcal{A}, \theta'))$, and μ be the trajectory of $\mathcal{A}$'s (δ_1, δ_2)-IS approximation M. Then for all $t \geq 0$,*

$$(\xi(t), \xi'(t)) \in L_V(\mu(t)).$$

Proof. For brevity we write $\mu \downarrow m_i$ as μ_i. By integrating both sides of (7) with initial condition $\mu_1(0) = \beta_1(\delta_1, 0)$ and $\mu_2(0) = \beta_2(\delta_2, 0)$, we have,

$$\begin{aligned} \mu_1(t) = &\quad \beta_1(\delta_1, t) + \textstyle\int_0^t \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(s - d(1,2)))ds, \\ \mu_2(t) = &\quad \beta_2(\delta_2, t) + \textstyle\int_0^t \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_1(s - d(2,1)))ds, \end{aligned} \tag{15}$$

Similarly, by integrating Equation (9), we have

$$\begin{aligned} \mu_{1\epsilon}(t) = &\quad \beta_1(\delta_1, t) + \textstyle\int_0^t \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{2\epsilon}(s - d(1,2)))ds + \epsilon(1+t) \\ \mu_{2\epsilon}(t) = &\quad \beta_2(\delta_1, t) + \textstyle\int_0^t \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{1\epsilon}(s - d(2,1)))ds + \epsilon(1+t). \end{aligned} \tag{16}$$

Define $h(t) \triangleq \|\mu_1(t) - \mu_{1\epsilon}(t)\| + \|\mu_2(t) - \mu_{2\epsilon}(t)\|$. Plugging in Equation (15) and (16) into the definition of $h(t)$, we have:

$$\begin{aligned} h(t) \quad \leq \quad & 2\epsilon(t+1) + \textstyle\int_0^t \|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(s - d(1,2))) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(s - d(1,2)))\|ds \\ & + \textstyle\int_0^t \|\gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{2\epsilon}(s - d(2,1))) - \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_2(s - d(2,1)))\|ds \end{aligned} \tag{17}$$

We will bound the two integrals of above inequality in (18) and (19). The property of delay function implies that for any $t \in [0, d]$, $\mu(t - d) = \mu(0)$. By splitting the integral intervals, we have

$$\begin{aligned} & \textstyle\int_0^t \|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(s - d(1,2))) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(s - d(1,2)))\|ds \\ = \quad & \textstyle\int_0^{d(1,2)} \|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(0)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(0))\|ds + \int_0^{t-d(1,2)} \|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(s)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(s))\|ds \\ \leq \quad & d(1,2)\|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(0)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(0))\| + \textstyle\int_0^t \|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(s)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(s))\|ds \end{aligned} \tag{18}$$

By using the same trick, we rewrite the last term of (17):

$$\begin{aligned} & \textstyle\int_0^t \|\gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{2\epsilon}(s - d(2,1))) - \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_2(s - d(2,1)))\|ds \\ \leq \quad & d(2,1)\|\gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{2\epsilon}(0)) - \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_2(0))\|ds + \textstyle\int_0^t \|\gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{2\epsilon}(s)) - \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_2(s))\|ds \end{aligned} \tag{19}$$

From the Lipschitz property of $\gamma_1 \circ \underline{\alpha}_2^{-1}$ and $\gamma_2 \circ \underline{\alpha}_1^{-1}$, we can find a constant $L > 0$ such that $\|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_{1\epsilon}(t)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_1(t))\| \leq L\|\mu_1(t) - \mu_{1\epsilon}(t)\|$ and $\|\gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_{2\epsilon}(t)) - \gamma_2 \circ \underline{\alpha}_1^{-1}(\mu_2(t))\| \leq L\|\mu_2(t) - \mu_{2\epsilon}(t)\|$. Using the Lipschitz condition with (18) and (19), Equation (17) becomes

$$h(t) \leq 2\epsilon(t + 1 + Ld_M) + \int_0^t Lh(s)ds,$$

with $d_M = \max\{d(1,2), d(2,1)\}$. By Gronwall-Bellman inequality [26], it follows that

$$h(t) \leq 2\epsilon(t + 1 + Ld_M) + 2\epsilon L \int_0^t (s + 1 + Ld_M)e^{L(t-s)}ds. \tag{20}$$

It follows that for any $t \in \mathbb{R}_{\geq 0}$, the integral $\int_0^t (s+1)e^{L(t-s)}ds$ is bounded. Thus $h(t) \to 0$ as $\epsilon \to 0$, which implies both $\|\mu_1(t) - \mu_{1\epsilon}(t)\| \to 0$ and $\|\mu_2(t) - \mu_{2\epsilon}(t)\| \to 0$ as $\epsilon \to 0$. Using Lemma 5, and taking the limit of $\epsilon \to 0$, it follows that

$$V_1(\xi_1(t), \xi_1'(t)) \leq \mu_1(t) \text{ and } V_2(\xi_2(t), \xi_2'(t)) \leq \mu_2(t).$$

That is, for any $t \geq 0$, $(\xi(t), \xi'(t)) \in L_V(\mu(t))$. □

Theorem 7 states that the reach set of any (large) networked dynamical system $\mathcal{A} = \mathcal{A}_1 \| \mathcal{A}_2$ from a set of states can be over-approximated by bloating an individual execution ξ of $\mathcal{A}$ by a factor that is entirely determined by (a) IS discrepancy functions V_1 and V_2 of its (smaller) modules, and (b) the trajectory μ of the reduced (2-dimensional) dynamical system M that is its IS approximation.

Theorem 7. *Consider a networked dynamical system $\mathcal{A} = \|_d\{\mathcal{A}_1, \mathcal{A}_2\}$ with IS discrepancy functions $V = (V_1, V_2)$. Let $\xi = Traj(\mathcal{A}, \theta)$ for some initial state $\theta \in \Theta_{\mathcal{A}}$. For any nonnegative pair $\delta = (\delta_1, \delta_2)$ suppose μ is the trajectory of the (δ_1, δ_2)-IS approximation M. Then, for any $T \geq 0$*

$$\mathsf{Reach}_{\mathcal{A}}(B_\delta(\theta), T) \subseteq \bigcup_{t \in [0,T]} B^V_{\mu(t)}(\xi(t)).$$

Proof. This follows from the bounds on the distance between trajectories established above. For any $\mathbf{x} \in \mathsf{Reach}_{\mathcal{A}}(B_\delta(\theta), T)$, there exists an initial state $\theta' \in B_\delta(\theta)$, a trajectory $\xi' = Traj(\mathcal{A}, \theta')$ and a time $t \in [0, T]$ such that $\xi'(t) = \mathbf{x}$. It follows by Lemma 6 that $(\xi(t), \xi'(t)) \in L_V(\mu(t))$, and therefore, $\mathbf{x} \in B^V_{\mu(t)}(\xi(t))$. □

Theorem 7 establishes an over-approximation of the set of reachable states from a δ-ball $B_\delta(\theta)$. To compute the set of reachable states from a compact initial set Θ_A, we can proceed as usual by first computing a δ-cover of Θ_A, and then computing the union of reach sets of the covers using the Theorem.

Remark 4. *Theorem 7 does not require $\mathcal{A}$ to be stable or any global property to hold for the IS discrepancy functions. To use Theorem 7 we need to (a) find IS discrepancy functions for the modules, and (b) compute individual trajectories ξ of complete network $\mathcal{A}$ and μ of M. Fortunately, for large classes of nonlinear dynamical systems there exist scalable numerical techniques for (b). This is one of the motivations of this work. For linear and some special classes of nonlinear systems (a) can be solved automatically (see Section IV-B). In Section VII we will present an algorithmic approach for computing local versions of the discrepancy function.*

C. Precision of the IS Approximation

The error in the over-approximation obtained from the reduced model is determined by the trajectories of the reduced model. In the following, we show that the over-approximation error can be made arbitrarily small with sufficiently small but positive δ_1, δ_2.

Lemma 8. *Consider any $T > 0$, $t \in [0, T]$, and a sequence of pairs of positive reals $\delta^k = (\delta^k_1, \delta^k_2)$ converging to $(0, 0)$. For the trajectory (μ^k) of the corresponding (δ^k_1, δ^k_2)-IS approximation M^k, $|(\mu^k \downarrow m_i)(t)| \to 0$ for $i \in \{1, 2\}$.*

Proof. Fix a $T > 0$ and $\delta^k = (\delta^k_1, \delta^k_2)$. This defines the (δ^k_1, δ^k_2)-IS approximation M^k and its trajectory μ^k. We will prove that for all $t \in [0, T]$,

$$|(\mu^k \downarrow m_1)(t)| + |(\mu^k \downarrow m_2)(t)| \to 0,$$

as $\delta^k \to 0$. From here on, we drop the superscript k and use the notations setup earlier: $\mu_i = \mu \downarrow m_i$, etc.

From the first row in Equation (15), applying the same trick as (18), we have that

$$|\mu_1(t)| \leq \beta_1(\delta_1, t) + d(1,2)|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(0))| + \int_0^t |\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(s))| ds \tag{21}$$

From the Lipschitz property of $\gamma_1 \circ \underline{\alpha}_2^{-1}$, there exists some $L > 0$, such that $|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(s)) - \gamma_1 \circ \underline{\alpha}_2^{-1}(0)| \leq L|\mu_2(s) - 0|$. Since $\gamma_1 \circ \underline{\alpha}_2^{-1}$ is of class-$\mathcal{K}$, it follows that

$$|\gamma_1 \circ \underline{\alpha}_2^{-1}(\mu_2(s))| \leq L|\mu_2(s)|.$$

From Equation (15), we observe that for $i \in \{1, 2\}$, $\mu_i(t)$ are nonnegative scalars. We drop the absolute value symbols $|\cdot|$. Then Equation (21) reduces to

$$|\mu_1(t)| \leq \beta_1(\delta_1, t) + d(1,2)L\beta_1(\delta_1, 0) + \int_0^t L|\mu_2(s)| ds. \tag{22}$$

Since $\beta_1(\delta_1, t)$ is bounded in compact intervals, define

$$C_1^T(\delta_1) = \sup_{t \in [0,T]} \beta_1(\delta_1, t), \tag{23}$$

as the upper bound of the function $\beta_1(\cdot, t)$ in interval $t \in [0, T]$. It follows from Equation (22) that

$$|\mu_1(t)| \leq C_1^T(\delta_1) + d(1,2)L\beta_1(\delta_1, 0) + \int_0^t L|\mu_2(s)| ds. \tag{24}$$

Starting from the second row of Equation (15), and following similar steps from Equation (21)-(24), we have:

$$|\mu_2(t)| \leq C_2^T(\delta_2) + d(2,1)L\beta_2(\delta_2, 0) + \int_0^t L|\mu_1(s)|ds. \tag{25}$$

Summing up Equation (24) and (25), by applying the Gronwall-Bellman inequality, we have

$$|\mu_1(t)| + |\mu_2(t)| \leq (C_1^T(\delta_1) + C_2^T(\delta_2) + d(1,2)L\beta_1(\delta_1, 0) + d(2,1)L\beta_2(\delta_2, 0))e^{Lt}.$$

For $i \in \{1,2\}$, $\beta_i(\cdot,\cdot)$ is a class-$\mathcal{K}$ function in the first argument, thus $\beta_i(\delta_i^k, 0) \to 0$ and $C_i(\delta_i^k) \to 0$ as $\delta_i^k \to 0$. It follows that, $|\mu_1^k(t)| + |\mu_2^k(t)| \to 0$ as $\delta^k \to 0$. □

Proposition 9 follows from the fact that for $i \in \{1,2\}$, for any $\mathbf{x}, \mathbf{x}' \in X_i$, $\underline{\alpha}_i(\mathbf{x} - \mathbf{x}') \leq V_i(\mathbf{x}, \mathbf{x}')$ (Definition 4).

Proposition 9. *For dynamical system $\mathcal{A}$ with discrepancy function $V = (V_1, V_2)$, fix any $\mathbf{x} \in X_{\mathcal{A}}$. For any $\epsilon > 0$, there exists $r > 0$, such that $B_r^V(\mathbf{x}) \subseteq B_\epsilon(\mathbf{x})$.*

Using Lemma 8 and Proposition 9, the next theorem bounding the error on the over-approximation of the reach set follows in a straightforward fashion.

Theorem 10. *Consider a closed dynamical system $\mathcal{A} = ||_d\{\mathcal{A}_1, \mathcal{A}_2\}$ with IS discrepancy $V = (V_1, V_2)$. Let $\xi = Traj(\mathcal{A}, \theta)$ for some initial state $\theta \in \Theta_{\mathcal{A}}$. For any $\epsilon > 0$, there exists a positive pair $\delta_1, \delta_2 > 0$ such that , for $\mathcal{A}$'s (δ_1, δ_2)-IS approximation M, for any $T \geq 0$*

$$\bigcup_{t \in [0,T]} B_{\mu(t)}^V(\xi(t)) \subseteq B_\epsilon(\mathsf{Reach}_{\mathcal{A}}(B_\delta(\theta), T)),$$

where μ is the unique trajectory of M.

If the over-approximation obtained from Theorem 7 is not precise enough to prove safety, then, we can refine the parameters δ_1 and δ_2. Then we can compute $B_{\mu(t)}^V(\xi(t))$ for each of the smaller partitions with higher precision. This is the standard approach used in simulation-based verification [5], [27], [28]. In the next section, we present this algorithm in more detail.

D. Generalizing to Arbitrary Networked Dynamical Systems

The approximation and its analysis presented in the previous section can be extended to closed networked dynamical systems. Let $\{\mathcal{A}_i\}_{i \in [N]}$ be a compatible closed collection of dynamic modules and $d \in \mathbb{R}_{\geq 0}^{N \times N}$ be a delay matrix. Let $\mathcal{A} = ||_d\{\mathcal{A}_i\}_{i \in [N]}$ be the corresponding closed dynamic network. For any pair of trajectories $\xi, \xi' \in Traj(\mathcal{A}, \Theta)$, and for each $i \in [N]$, we define $\xi_i = \xi \downarrow X_i$ and $\xi_i' = \xi' \downarrow X_i$. The next definition gives a natural generalization of Definition 4 to account for dynamical modules that take inputs from more than one module in the network.

Definition 7. *Let $\mathcal{A}_i = \langle X_i, U_i, \theta_i, f_i \rangle$ be a dynamic module receiving input signals from a collection of modules $\{\mathcal{A}_j\}_{j \in U_i \cap X_j}$. An input-to-state discrepancy function for $\mathcal{A}_i$ (in this environment) is a continuous function $V_i : Val(X)^2 \to \mathbb{R}_{\geq 0}$ with class-$\mathcal{K}$ witnesses $(\underline{\alpha}_i, \overline{\alpha}_i, \beta_i)$ and $\{\gamma_{ij}\}_{(i,j):U_i \cap X_j \neq \varnothing}$, such that*

(i) for any $\mathbf{x}, \mathbf{x}' \in Val(X_i)$, $\underline{\alpha}_i(\|\mathbf{x} - \mathbf{x}'\|) \leq V_i(\mathbf{x}, \mathbf{x}') \leq \overline{\alpha}_i(\|\mathbf{x} - \mathbf{x}'\|)$, and
(ii) for any pair of initial states $\theta, \theta' \in \Theta_i$ and input signals $\eta_{ij}, \eta_{ij}' \in Traj(U_i \cap X_j)$ from module $\mathcal{A}_j$ to $\mathcal{A}_i$, for each $t \geq 0$

$$V_i(\xi_i(t), \xi_i'(t)) \leq \beta_i(\|\theta - \theta'\|, t) + \int_0^t \sum_{j : X_j \cap U_i \neq \varnothing} \gamma_{ij}(\|\eta_{ij}(s) - \eta_{ij}'(s)\|)ds,$$

where ξ_i and ξ_i' are the trajectories of $\mathcal{A}_i$ from initial state θ (θ' respectively) and with input signal η_{ij} (η_{ij}' respectively) from module $\mathcal{A}_j$.

The set $\{\mathcal{A}_j | X_j \cap U_i \neq \varnothing\}$ is the set of modules that provide inputs to module $\mathcal{A}_i$. Definition 7 allows each $\mathcal{A}_j$ providing an input to $\mathcal{A}_i$ to have a different witness γ_{ij} to the discrepancy function V_i. This flexibility also permits the input signals from different modules to experience delays to different degrees. Generalizing Definition 6, the IS approximation of $\mathcal{A}$ is a N-dimensional closed deterministic dynamical system M.

Definition 8. *For any $\delta = (\delta_1, \ldots, \delta_N) \in \mathbb{R}_{\geq 0}^N$, the δ-IS approximation of $\mathcal{A} = ||_d\{\mathcal{A}_i\}_{i \in [N]}$ is a dynamical network $M = \langle X_M, \Theta_M, \mathcal{T}_M \rangle$, where*

(i) $X_M = \{m_1, m_2, \ldots, m_N\}$,
(ii) $\Theta_M = \{\theta\}$, where $\theta \lceil m_i = \beta_i(\delta_i, 0)$, for each $i \in [N]$, and
(iii) for any trajectory μ of X_M, $\mu \in \mathcal{T}_M$ if and only if

$$\dot{\mu}_i(t) = \dot{\beta}_i(\delta_i, t) + \sum_{j : X_j \cap U_i \neq \varnothing} \gamma_{ij} \circ \underline{\alpha}_j^{-1}(\mu_j(t - d(i,j))),$$

where $\mu_i = \mu \downarrow m_i$.

The dynamics of the closed reduced network M is defined syntactically in terms of the IS-discrepancy functions of the constituent modules $\{\mathcal{A}_i\}_{i\in[N]}$ and the delay matrix d. Once again, from the Lipschitz continuity of the right hand side it follows that the delay differential equation has a unique solution. The IS approximation M of the original dynamical network $\mathcal{A}$ is an N-dimensional system. The construction of M is defined syntactically only using (a) information of individual modules (IS discrepancy functions etc.) and (b) the topology and delay matrix for composition. An N-dimensional analogue of Theorems 7 and 10 giving approximations of Reach_A in terms of $\mu(\cdot)$ can be proven essentially following the same steps.

VI. From Simulations to Proofs

Theorem 7 gives us a recipe for verifying bounded time invariants (or safety properties) of closed networks, only by numerically computing trajectories of the whole network $\mathcal{A}$ and the trajectories of the statically computed reduced IS approximation M. That is, detailed static analysis of the complete model $\mathcal{A}$ becomes unnecessary in this method. In this section, we develop this idea further and provide the details of the verification algorithm.

A. Verification Algorithm

The algorithm involves simulations or numerical computation of trajectories of dynamical systems. We proceed by first defining what we mean by simulations. Given a closed networked dynamical system $\mathcal{A}$ without delay ($d(i,j) = 0$ for all i, j), an initial state θ, let $\xi = Traj(\mathcal{A}, \theta)$ be the actual trajectory of $\mathcal{A}$ from the initial state θ. For a step size $\tau > 0$, validated ODE solvers such as [29]–[31] compute a sequence of boxes $R_1, R_2, \ldots, R_l \subseteq Val(X_A)$, such that for each $k \in [l]$, $t \in [(k-1)\tau, k\tau]$, $\xi(t) \in R_k$. For a desired error bound $\epsilon > 0$, by reducing the step size τ, the diameter of R_k can be made smaller than ϵ. For delayed differential equations, there are existing techniques for numerical simulation [19], [32]–[34]. Several of these reduce the problem to simulating ODEs [35]. Specifically, the simulations generated in [19] have validated error bounds. Our safety verification algorithm assumes the availability of a subroutine $Simulate$ for computing numerical simulations that meet the following definition.

Definition 9. *Consider a closed network $\mathcal{A}$, an initial state θ, a time bound T, an error bound $\epsilon > 0$, and time step $\tau > 0$. Let $\xi = Traj(\mathcal{A}, \theta)$ denote the trajectory of $\mathcal{A}$ from the initial state θ. A* $(\theta, T, \epsilon, \tau)$-simulation trace *is a finite sequence $\phi = \langle R_0, t_0\rangle, \langle R_1, t_1\rangle, \ldots, \langle R_l, t_l\rangle$ where*

(i) $t_0 = 0$, $t_l = T$, *and* $\forall\ k \in [l]$, $t_k - t_{k-1} \leq \tau$,
(ii) $\forall k \in [l]$ *and* $\forall t \in [t_{k-1}, t_k]$, $\xi(t) \in R_k$, *and*
(iii) $\forall k \in [l]$, $dia(R_k) \leq \epsilon$.

In Algorithm 1 the subroutine $Simulate(\mathcal{A}, \theta, T, \epsilon, \tau)$ (line 5,17) computes a $(\theta, T, \epsilon, \tau)$-simulation of the network $\mathcal{A}$ as defined above. For the completeness of the algorithm, it is required that for any precision parameters $\epsilon, \tau > 0$, a simulation trace fulfilling this condition can be computed.

Algorithm 1: Bounded verification algorithm

input: $\mathcal{A}, \mathbb{U}, \epsilon_0, \delta_0, T, \{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ij}\}\}$
1 $\delta \leftarrow \delta_0; \epsilon \leftarrow \epsilon_0; \mathcal{R} \leftarrow \varnothing$;
2 $\mathcal{C} \leftarrow Partition(\Theta_{\mathcal{A}}, \delta, \epsilon)$;
3 **while** $\mathcal{C} \neq \varnothing$ **do**
4 **for** $(\theta, \delta, \epsilon) \in \mathcal{C}$ **do**
5 $\psi \leftarrow Simulate(\mathcal{A}, \theta, T, \epsilon, \tau)$;
6 $S \leftarrow BloatWithISD(\psi, d, \delta, \epsilon, \tau, \{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ij}\}\})$;
7 **if** $S \cap \mathbb{U} = \varnothing$ **then**
8 $\mathcal{C} \leftarrow \mathcal{C} \backslash \{(\theta, \delta, \epsilon)\}; \mathcal{R} \leftarrow \mathcal{R} \cup S$;
9 **else if** $\exists k, R_k \subseteq \mathbb{U}$ **then**
10 **return** (UNSAFE, $\mathcal{R}$)
11 **else**
12 $\mathcal{C} \leftarrow \mathcal{C} \backslash \{(\theta, \delta, \epsilon)\}$;
13 $\mathcal{C} \leftarrow \mathcal{C} \cup Partition(\Theta_{\mathcal{A}} \cap B_\delta(\theta), (\frac{\delta_1}{2}, \ldots, \frac{\delta_N}{2}), \frac{\epsilon}{2})$;
14 **return** (SAFE, $\mathcal{R}$);
15 **Subroutine** $BloatWithISD(\psi, d, \delta, \epsilon, \tau, \{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ij}\}\})$ **is**
16 $M \leftarrow ISApprox(\delta, d, \{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ij}\}\})$;
17 $\phi \leftarrow Simulate(M, \theta_M, T, \epsilon, \tau)$;
18 $\rho \leftarrow SupByTime(\phi)$;
19 $S \leftarrow B^V_\rho(\psi)$;
20 **return** S;

The algorithm takes several inputs: (a) the system model $\mathcal{A} = \|_d\{\mathcal{A}_i\}_{i\in[N]}$, (b) the unsafe set $\mathbb{U} \subseteq \mathit{Val}(X_\mathcal{A})$, (c) initial partition parameter δ_0, simulation parameter ϵ_0, and the time bound T, and (d) a collection $\{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ji}\}\}$ of IS discrepancy functions and witnesses for the modules in $\mathcal{A}$.

The algorithm proceeds as follows: The set $\mathcal{C}$ computed in line 2 is a finite set of triples $\{(\theta_c, \delta, \epsilon)\}_{c\in|\mathcal{C}|}$, such that $\{\theta_c\}_{c\in|\mathcal{C}|}$ is a δ-cover of the initial set $\Theta_\mathcal{A}$. Each θ_c is associated with a precision parameter $\epsilon > 0$ and positive real-valued vector δ. For each triple $(\theta, \delta, \epsilon)$ in the cover set $\mathcal{C}$, the algorithm first computes a simulation trace $\psi = \langle R_0, t_0\rangle, \ldots, \langle R_p, t_p\rangle$ containing the trajectory $\xi = Traj(\mathcal{A}, \theta)$ of the (large) networked dynamical system $\mathcal{A}$ (line 5). Then the subroutine *BloatWithISD* bloats the trace ψ to get a tube S (line 6). We claim that the tube S contains the set of states of $\mathcal{A}$ reachable from the set of initial state $B_\delta(\theta)$. Then the algorithm checks whether this tube is safe or unsafe. If neither of these cases can be deduced then the part of the initial set $B_\delta(\theta)$ is further refined by adding a new partition to $\mathcal{C}$.

The subroutine *BloatWithISD* is detailed in line 15-20. First, an δ-IS approximation M (line 16) is constructed following Definition 8. The same *Simulate* subroutine is then used to compute a simulation trace $\phi = \langle Q_0, t_0\rangle, \ldots, \langle Q_p, t_p\rangle$ of the trajectory $\mu = Traj(M, \theta_M)$, of the (smaller) reduced model M. Here we assume the time stamps of the sequence ψ match up with that of ϕ. This can be achieved by using a fixed step solver or through repeated simulations using smaller step sizes. The sequence ρ computed in line 18 is a sequence of pairs $\langle r_0, t_0\rangle, \ldots, \langle r_p, t_p\rangle$, where for each $k \in [p]$, r_k is a nonnegative real defined as $r_k = \sup_{\mathbf{m}\in Q_k} \|\mathbf{m}\|$. In line 19, the sequence of sets $\{R_k\}$ is bloated by the sequence of factors $\{r_k\}$ element-wise to construct a tube S.

B. Analysis of the Verification Algorithm

We establish the soundness and relative completeness of the algorithm in Theorems 11 and 12. Practical scalability of the algorithm is discussed in Section VIII.

Theorem 11. *Algorithm 1 is sound. That is, if it returns* SAFE *then $\mathcal{A}$ is safe up to T, and if it returns* UNSAFE *then $\mathcal{A}$ is unsafe.*

Proof. Suppose the algorithm returns SAFE. For any cover $(\theta, \delta, \epsilon) \in \mathcal{C}$, S computed in line 19 is the union of a sequence of regions $\{B^V_{r_k}(R_k)\}$, where $\cup_{k\in[p]} R_k$ contains the trajectory from θ and the sequence of r_k upper bounds the trajectory of δ-IS approximation. It follows from Theorem 7, that $\mathsf{Reach}_\mathcal{A}(B_\delta(\theta), T) \subseteq S$. Thus if $S \cap \mathbb{U} = \varnothing$, then $\mathsf{Reach}_\mathcal{A}(B_\delta(\theta)) \cap \mathbb{U} = \varnothing$. The algorithm returns SAFE, if all such covers are checked safe. It is straightforward to check that $\mathcal{C}$ captures a finite cover of the initial set $\bigcup_{(\theta,\delta,\epsilon)\in\mathcal{C}} B_\delta(\theta) \supseteq \Theta_\mathcal{A}$. Therefore, we conclude $\mathsf{Reach}_\mathcal{A}(\Theta_\mathcal{A}, T) \cap \mathbb{U} = \varnothing$.

Otherwise if the algorithm returns UNSAFE, there exists at least one set R_k contained in the unsafe set $\mathbb{U}$. It follows that for the trajectory $\xi = Traj(\mathcal{A}, \theta)$ and some $t \in [t_{k-1}, t_k]$, $\xi(t) \in \mathbb{U}$. □

Theorem 12. *Algorithm 1 is relatively complete. That is, if $\mathcal{A}$ is robustly safe then Algorithm 1 terminates and returns* SAFE *and if $\mathcal{A}$ is unsafe then it terminates and returns* UNSAFE.

Proof. Suppose that for some $\epsilon' > 0$, $\mathcal{A}$ is ϵ'-robustly safe up to time T. It follows from Definition 2 that

$$B_{\epsilon'}(\mathsf{Reach}_\mathcal{A}(\Theta_\mathcal{A}, T)) \cap \mathbb{U} = \varnothing.$$

It follows that line 8 is never executed. For any $\theta \in \Theta$, we will show that for small enough refinement parameters $\delta, \epsilon > 0$, for any k, $dia(B^V_{r_k}(R_k)) < \epsilon'$. From Proposition 9, we can show that there exists $e > 0$ such that for $r_k < e$,

$$B^V_{r_k}(R_k) \subseteq B_{\epsilon'/3}(R_k). \tag{26}$$

From Lemma 8, there exists a $\delta > 0$, for all $t \in [0, T]$, and all $i \in [N]$, $|\mu(t)| \leq e/2$. For a simulation trace $\phi = \langle M_0, t_0\rangle, \ldots, \langle M_q, t_q\rangle$ (line 17), and $\epsilon \leq e/2$, it follows from Definition 9 that for all $k \in [q]$, the diameter $dia(M_k) \leq e/2$. Thus for any $k \in [q]$, $r_k = \sup_{\mathbf{m}\in M_k} |\mathbf{m}| \leq \epsilon + \sup_{t\in[t'_{l-1}, t'_l]} |\mu(t)| \leq e/2 + e/2 = e$. It follows from Equation (26) that by choosing $\delta \leq d$ and $\epsilon \leq e/2$, we have $B^V_{r_k}(R_k) \subseteq B_{\epsilon'/3}(R_k)$. Notice that the diameter of R_k is bounded by the refinement parameter ϵ. By choosing $\epsilon = \min\{\epsilon'/3, e\}$, it follows that

$$dia(B_{\epsilon'/3}(R_k)) \leq \epsilon'/3 + dia(R_k) \leq \epsilon'/3 + \epsilon'/3 < \epsilon'.$$

Thus, after a number of $\max\{\log(\frac{\epsilon_0}{\min\{\epsilon'/3, e\}}), \log\frac{\delta_0}{d}\}$ refinements, the parameters δ, ϵ are small enough to guarantee that $S \cap \mathbb{U} = \varnothing$. Thus the algorithm returns SAFE.

On the other hand, suppose A is unsafe with respect to an open unsafe set $\mathbb{U}$. There exists an initial state θ, a time $t \geq 0$ and a $\epsilon' > 0$ such that $B_(\epsilon'\xi(\theta, t)) \subseteq \mathbb{U}$. For the same number of refinement as the robustly safe case, it can be shown that there exists an $B^V_{r_k}(R_k) \subseteq B_{\epsilon'}(\xi(\theta, t))$. It follows that the algorithm returns UNSAFE. □

VII. On-the-fly Computation of Local IS Discrepancy

The IS discrepancy defined in Definition 5 are global in the sense that the inequalities hold for arbitrary trajectories in the state space uniformly. While we noted several existing methods for statically computing such global IS discrepancy functions in Section 2, these methods are guaranteed to work only for certain classes of stable or linear models. Furthermore, even where these methods work, the bounds they provide can be overtly conservative which negatively affects the performance of Algorithm 1. To overcome this limitation, we will define a notion of IS discrepancy function (Definition 10) that only needs to hold for trajectories in a compact set K. This notion turns out to be adequate for bounded time verification because the said compact set bound K can be inferred from the initial neighborhood of the system using a coarse estimate of the reach set. The local version of discrepancy was introduced and applied in [11], [28] and similar results have been obtained using other norms on the Jacobian matrix in [36]. Here we extend the idea to input-to-state discrepancy functions, that is, Algorithm 1 does not use the final argument $\{V_i, \underline{\alpha}_i, \overline{\alpha}_i, \beta_i, \{\gamma_{ij}\}\}$ but a local version of IS discrepancy computed from the Lipschitz constants and Jacobian matrices of dynamical mappings f_i of the modules.

A. Local IS Discrepancy

Consider a dynamical module $\mathcal{A} = \langle X, U, \Theta, f\rangle$, where the dynamic mapping f is Lipschitz continuous and has a continuous Jacobian. For a compact set $K \subseteq \mathit{Val}(X)$ and time interval $[t_0, t_1]$, a trajectory ξ is $(K, [t_0, t_1])$-*bounded* if $\{\xi(t) : t \in [t_0, t_1]\} \subseteq K$. In other words, all the states of trajectory ξ in interval $[t_0, t_1]$ are contained by the set K. For a compact set $\mathcal{I} \subseteq \mathit{Val}(U)$, the notion of $(\mathcal{I}, [t_0, t_0])$-bounded input signals is similarly defined. Next, we introduce restrictions on Definition 4 to compact neighborhoods.

Definition 10 (Local IS Discrepancy)**.** *For a dynamical module $\mathcal{A} = \langle X, U, \Theta, f\rangle$, a compact subset $K \subseteq \mathit{Val}(X)$, a compact subset $\mathcal{I} \subseteq \mathit{Val}(U)$ and time points $0 \le t_0 \le t_1$, the function $V : K^2 \to \mathbb{R}_{\ge 0}$ with class-$\mathcal{K}$ witnesses $(\underline{\alpha}, \overline{\alpha}, \beta, \gamma)$ is the $(K, \mathcal{I}, [t_0, t_1])$-local IS discrepancy function of $\mathcal{A}$ if*

(i) for any $\mathbf{x}, \mathbf{x}' \in K$, $\underline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|) \le V(\mathbf{x}, \mathbf{x}') \le \overline{\alpha}(\|\mathbf{x} - \mathbf{x}'\|)$, and

(ii) for any pair of initial states $\theta, \theta' \in \Theta$ and $(\mathcal{I}, [t_0, t_1])$-bounded input signals $\eta, \eta' \in Traj(U)$, if $\xi = Traj(\mathcal{A}, \theta, \eta)$ and $\xi' = Traj(\mathcal{A}, \theta', \eta')$ are $(K, [t_0, t_1])$-bounded, then the following holds at any time $t \in [t_0, t_1]$,

$$V(\xi(t), \xi'(t)) \le \beta(\|\theta - \theta'\|, t - t_0) + \int_{t_0}^{t} \gamma(\|\eta(s) - \eta'(s)\|)ds. \tag{27}$$

The above definition relaxes Definition 4 so that the right-hand side of (28) just upper bounds distance between all trajectories contained in set K for an interval $[t_0, t_1]$ with input signal bounded by $\mathcal{I}$. A similar version of IS approximation can be built using $(K, \mathcal{I}, [t_0, t_1])$-local IS discrepancy functions instead of the standard IS discrepancy functions, whose trajectories can be used to over-approximate all $(K, [t_0, t_1])$-bounded trajectories with $(\mathcal{I}, [t_0, t_1])$-bounded input signals. The definition of local IS discrepancy can be generalized to the module with input from multiple modules similarly to Definition 7.

For a dynamic network $\|_d\{\mathcal{A}_1, \mathcal{A}_2\}$ and any nonnegative pair $\delta = (\delta_1, \delta_2)$, we can build an IS approximation following Definition 6 with $(K_i, \mathcal{I}_i, [t_0, t_1])$-local discrepancy functions of module $\mathcal{A}_i$ for $i \in \{1, 2\}$. We call this reduced model $(\delta, K, \mathcal{I}, [t_0, t_1])$-*IS approximation*, where $K_i = K \restriction X_i, \mathcal{I}_i = \mathcal{I} \restriction U_i$ for $i \in \{1, 2\}$.

Lemma 13. *For any trajectories ξ, ξ' of dynamic network $\mathcal{A} = \|_d\{\mathcal{A}_1, \mathcal{A}_2\}$, we denote $\xi_i = \xi \downarrow X_i$, $\xi'_i = \xi \downarrow X_i$, $\eta_i = \xi \downarrow U_i$ and $\eta'_i = \xi \downarrow U_i$, $\delta_i = \|\xi_i(t_0) - \xi'_i(t_0)\|$ for $i \in \{1, 2\}$. Let M be the $(\delta, K, \mathcal{I}, [t_0, t_1])$-IS approximation with trajectory μ. The following two statements hold:*

(i) If ξ_i is $(K_i, [t_0, t_1])$-bounded and $\eta_i(t - d(i, j))$ is $(\mathcal{I}_i, [t_0, t_1])$-bounded for each $i, j \in \{1, 2\}$ with $i \ne j$, then, for all $t \ge [t_0, t_1]$,

$$(\xi(t), \xi'(t)) \in L_V(\mu(t)).$$

(ii) For any $\epsilon > 0$, there exists $\delta > 0$ such that the resulting trajectory μ of the $(\delta, K, \mathcal{I}, [t_0, t_1])$-IS approximation satisfies $\|\mu(t)\| \le \epsilon$ for all $t \in [t_0, t_1]$.

The proof of the above lemma follows from arguments that are identical to those used to prove Lemmas 6 and 8. The lemma essentially states that the local IS discrepancy functions can be used instead of the (global) IS discrepancy functions of the modules as long as we can somehow compute compact sets $K, \mathcal{I}$ that conservatively over-approximate the reach sets and the input sets of the modules over short time intervals.

B. Computing Local IS Discrepancy

For any compact subset of the state space $K \subseteq \mathit{Val}(X)$, any compact set $\mathcal{I} \subseteq \mathit{Val}(U)$ and time interval $[t_0, t_1]$, Lemma 15 (below) will provide a way to compute a $(K, \mathcal{I}, [t_0, t_1])$-IS discrepancy of any dynamical module $\mathcal{A}$ only using the Jacobian matrix of the dynamic mapping f. Before proving the lemma, we state the following generalized form of the Mean value theorem (see [11] for a detailed proof).

Proposition 14. *For any differentiable function $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^n$, any $t \in \mathbb{R}$, any $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$, any $\mathbf{u}, \mathbf{u}' \in \mathbb{R}^p$, the following holds:*

$$f(t,\mathbf{x}',\mathbf{u}') - f(t,\mathbf{x},\mathbf{u}) = \left(\int_0^1 J_X(t,\mathbf{x}+(\mathbf{x}'-\mathbf{x})s,\mathbf{u}')ds\right)(\mathbf{x}'-\mathbf{x}) + \left(\int_0^1 J_U(t,\mathbf{x},\mathbf{u}+(\mathbf{u}'-\mathbf{u})\tau)d\tau\right)(\mathbf{u}'-\mathbf{u}),$$

where $J_X = \frac{\partial f(t,\mathbf{x},\mathbf{u})}{\partial X}$ and $J_U = \frac{\partial f(t,\mathbf{x},\mathbf{u})}{\partial U}$ are the Jacobian matrices of f with respect to x and u.

Lemma 15. *For a dynamical module $\mathcal{A}$, compact set $K \subseteq Val(X)$ and time points $0 \le t_0 \le t_1$, a $(K, \mathcal{I}, [t_0,t_1])$-local IS discrepancy function is $V(\mathbf{x},\mathbf{x}') \triangleq \|\mathbf{x}-\mathbf{x}'\|$ with witnesses*

$$\beta(y, t-t_0) \triangleq e^{a(t-t_0)}y, \text{ and } \gamma(y) \triangleq by, \tag{28}$$

$$\text{with } a = \sup_{\substack{t\in[t_0,t_1]\\ \mathbf{x}\in K\\ \mathbf{u}\in\mathcal{I}}} \lambda_1\left(\frac{J_X+J_X^\top}{2}\right) + \frac{1}{2}, \quad b' = \sup_{\substack{t\in[t_0,t_1]\\ \mathbf{x}\in K\\ \mathbf{u}\in\mathcal{I}}} \|J_U\|, \text{ and } b = b' \sup_{t\in[t_0,t_1]} e^{a(t-t_0)}.$$

Here $\lambda_1(A)$ denoted the largest eigenvalue of matrix A.

The Jacobian matrix J_X depends on the state $\mathbf{x}$, the input $\mathbf{u}$ and time t, and the constant a is the maximum eigenvalue of the symmetric part of the Jacobian $(\frac{J_X+J_X^T}{2})$ over the compact ranges K, $\mathcal{I}$ and $[t_0,t_1]$ for $\mathbf{x}$, $\mathbf{u}$, and t. Similarly, b' is the 2-norm of the Jacobian matrix J_U over K and $\mathcal{I}$ and finally b is obtained from a and b'.

Proof. Let ξ and ξ' be two $(K,[t_0,t_1])$-bounded trajectories with input signals η, η'. Define $y(t) = \xi'(t) - \xi(t)$ and $v(t) = \eta'(t) - \eta(t)$. For a time $t \in [t_0,t_1]$, using Proposition 14, we have

$$\begin{aligned}\dot{y}(t) &= f(t,\xi'(t),\eta'(t)) - f(t,\xi(t),\eta(t))\\ &= \left(\int_0^1 J_X(t,\xi(t)+sy(t),\eta'(t))ds\right)y(t) + \left(\int_0^1 J_U(t,\xi(t),\eta(t)+\tau v(t))d\tau\right)v(t).\end{aligned}$$

We write $J_X(t,\xi(t)+sy(t),\eta'(t))$ as $J_X(t,s)$ or simply as J_X when the dependence on t and s is clear from context. Similarly, $J_U(t,\xi(t),\eta(t)+\tau v(t))$ is written as $J_U(t,\tau)$ or J_U. Then the differentiating $\|y(t)\|^2$ with respect to t:

$$\begin{aligned}\frac{d}{dt}\|y(t)\|^2 &= \frac{d}{dt}(y(t)^\top y(t)) = \dot{y}(t)^\top y(t) + y(t)^\top \dot{y}(t)\\ &= y(t)^\top\left(\int_0^1 (J_X^\top + J_X)ds\right)y(t) + v(t)^\top\left(\int_0^1 J_U^\top d\tau\right)y(t) + y(t)^\top\left(\int_0^1 J_U d\tau\right)v(t)\\ &\le y(t)^\top\left(\int_0^1 J_X^\top + J_X ds\right)y(t) + y(t)^\top y(t) + \left(\left(\int_0^1 J_U d\tau\right)v(t)\right)^\top\left(\left(\int_0^1 J_U d\tau\right)v(t)\right).\end{aligned} \tag{29}$$

Let $\lambda_{max}(S) = \sup_{x\in K}\lambda_1(\frac{J_X+J_X^\top}{2})$ be the upper bound of the eigenvalues of the symmetric part of J_X over S, so $J_X^\top + J_X \preceq 2\lambda_{max}(S)I$. Since f has continuous Jacobian, then a, b' defined in (15) are finite. Thus, (29) becomes:

$$\frac{d}{dt}\|y(t)\|^2 \le (2\lambda_{max}(S)+1)\|y(t)\|^2 + \|\left(\int_0^1 J_U d\tau\right)v(t)\|^2 \le 2a\|y(t)\|^2 + b'\|v(t)\|^2.$$

For $t \in [t_0,t_1]$ integrating both sides of the above inequality from t_0 to t and using the property of square root $\sqrt{x+y} \le \sqrt{x}+\sqrt{y}$, we have

$$\|y(t)\|^2 \le e^{2a(t-t_0)}\left(\|y(t_0)\|^2 + \int_{t_0}^t b'\|v(\tau)\|^2 d\tau\right) \implies \|y(t)\| \le e^{a(t-t_0)}\|y(t_0)\| + \int_{t_0}^t b\|v(\tau)\|d\tau.$$

Thus the lemma follows. □

For compact sets $K \subseteq Val(X)$, $\mathcal{I} \subseteq Val(U)$ and interval $[t_0,t_1]$, the above lemma gives a local IS discrepancy function over K, $\mathcal{I}$ and a specified time interval and it can be computed from the Jacobian matrices of the dynamical mapping f. If all the eigenvalues of the symmetric Jacobian matrix $\frac{J_X+J_X^\top}{2}$ are less than $-\frac{1}{2}$, from (28), the witness function β is exponentially decreasing which may result in a tight over-approximation of the reach set. The result can be generalized to modules with input from multiple modules in a natural way.

C. Over-approximation of Reach Set based on Local IS Discrepancy

We introduce a subroutine $OnTheFlyBolating(\psi, \{J_i\}, \{L_i\}, \delta, \epsilon, \tau)$ which replaces the subroutine $BloatWithISD()$ (line 6) in Algorithm 1. The subroutine takes a simulation trace $\psi = \langle R_0, t_0\rangle, \ldots, \langle R_p, t_p\rangle$, the Jacobian matrices of each module $\{J_i\}$, the Lipschitz constants of each module $\{L_i\}$ and the precision parameters (δ, ϵ, τ) as input, and returns a bloated tube $S = \{S_i\}_{i\in[p]}$. For each $k \in [p]$, the $BloatWithLipschitz$ (line 2) first computes the compact bound on the reach set R using only Lipschitz constants. Then using this bound R (as the set denoted by K above), it computes compact sets which bound the state trajectories and input trajectories over the time interval $[t_{k-1}, t_k]$. In line 3, the local IS discrepancy function of all modules as established in Lemma 3 are computed. Then a set S_k is computed by bloating the simulation R_k with the trajectory of a reduced model built with the local discrepancy functions. Following the same arguments of Theorem 7, we can prove that the tube S contains all trajectories from a δ-ball of the initial state θ of simulation trace ψ. Therefore, the soundness and completeness of Algorithm 1 stated in Theorem 11 and 12 are maintained.

Algorithm 2: Subroutine $OnTheFlyBloating(\psi, \{J_i\}, \{L_i\}, \delta, \epsilon, \tau)$.

input: ψ,$\{J_i\}$,$\{L_i\}$,δ, ϵ, τ
1 **for** $k \in [p]$ **do**
2 $\quad R \leftarrow BloatWithLipschitz(\delta, R_k, \{L_i\}, [t_{k-1}, t_k])$;
3 $\quad \{(\beta_i, \{\gamma_{ij}\})\} \leftarrow ComputeLocalISD(R, [t_{k-1}, t_k], \{J_i\})$;
4 $\quad S_k \leftarrow BloatWithLISD(\delta, R_k, \{\beta_i\}, \{\gamma_{ij}\}, [t_{k-1}, t_k])$;
5 $\quad \delta \leftarrow dia(S_k)$;
6 **return** S ;

VIII. Experimental Validation

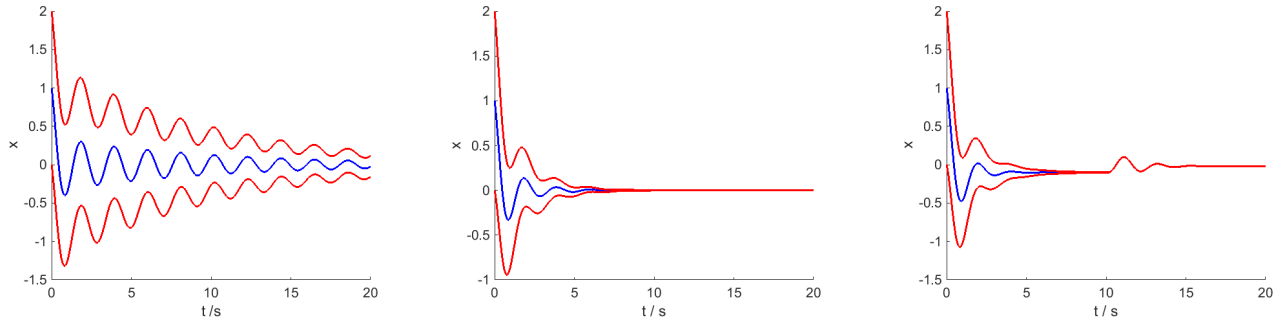
We discuss experimental results from a prototype implementation of Algorithm 1 in Matlab. We verify bounded time invariant properties of several linear and nonlinear networked dynamical systems. First, we give a brief overview of the different examples. Each module in the linear synchronization examples (see [37] for detail) is a 4-dimensional linear dynamical system, and the overall system is composed of several modules in different topologies. Through communicating with neighbors, the modules in the network aim to reach synchronization to a common solution. We compute reach sets for several such networks, both with and without delays. The nonlinear water tank network example is a modified version of the one presented in [38]. In this example, each module captures the water levels in a group of tanks, that depends on the flows from other tanks. The nonlinear cardiac cell network with delayed interconnections was shown in Examples 1 and 2.

For the synchronization examples, the IS-discrepancy functions and witnesses are computed statically using Proposition 2. For the remaining nonlinear examples, the discrepancies are computed on-the-fly using Algorithm 2. In the results presented here, the time bound is $T = 20$ seconds and the running time is based on executing the procedure on an Intel i5-3470 CPU. The columns in Table I present (i) the system being verified, (ii) the number of total state variables, (iii) the number of modules, (iv) the number of covers for the initial set, (v) the total number of simulation boxes generated, and (vi) the running time of the algorithm. Our experimental results show that the running time roughly scales linearly with the total number of simulation boxes generated for both the original system $\mathcal{A}$ and its IS approximation $\mathcal{M}$. The number of simulation boxes generated is proportional to the product of the total number of covers and the time bound. Fixing a compact initial set, the number of covers generated depends on the level of precision needed to prove (or disprove) safety, which further relying on the distance between the unsafe set to the reachable states. From the linear synchronization examples, we also observe that verifying time-delayed networks takes a longer time than delay-free networks of the same dimensions, even if the algorithm produces similar number of simulations boxes. One source of this difference is that simulating a delayed differential equation is more expensive than simulating an ODE.

Sample reach tubes computed for the linear synchronization example are shown in Figures 2 and 3. The blue trace (center) is the simulation of the actual system and the red lines give the upper and lower-bounds computed from the reduced model. These are networks of two 4-dimensional modules with linear dynamics $\dot{x}_1 = A_1x_1 + B_1u_1$ and $\dot{x}_2 = A_2x_2 + B_2u_2$. The modules are connected with an inter-modular gain $g > 0$ and a delay $d \geq 0$, such that $u_1(t) = gx_2(t-d)$ and $u_2(t) = gx_1(t-d)$. For a network with Hurwitz A_1 and A_2, as seen in Figure 2, the diameter of the computed reach tube (for the same size of the initial set) grows quickly with the gain, but less so with delay. The computed reach tubes of networks with an unstable module are illustrated in Fig. 3. From the construction of the reduced model (Definition 6), if a module is unstable with $\dot{\beta}_i(\delta, t) \geq 0$, the corresponding state component μ_i of the reduced model is unstable. Thus, even though the whole network may be stable, the reduced model will be unstable. In Fig. 3, the reach tube in (a), which is corresponding to a smaller delay, is slightly thinner than the one in (b). From the trajectories (blue curves) in Fig. 3, the whole network corresponding to (a) is stable however the reduced model does not capture this. This shows that our approach performs relatively well when either the individual modules are stable or the inter-modular gains are small.

Systems	# V	# N	# C	# sim	RT (s)
Linear synchronization I	16	4	128	45440	129.2
Linear synchronization II	24	6	128	45649	135.1
Linear synchronization with Delay I	16	4	64	23168	103.1
Linear synchronization with Delay II	16	4	128	46304	205.8
Nonlinear Water Tank I	10	2	128	45184	127.4
Nonlinear Water Tank II	30	6	128	47232	140.0
Nonlinear cardiac cell network with delay I	6	3	16	6134	22.0
Nonlinear cardiac cell network with delay II	16	8	24	13563	42.5

TABLE I: Experimental results. The columns represent: (1) the system being verified, (2) # state variables, (3) # modules, (4) # covers of initial set, (5) # total simulation boxes, and (6) run time.



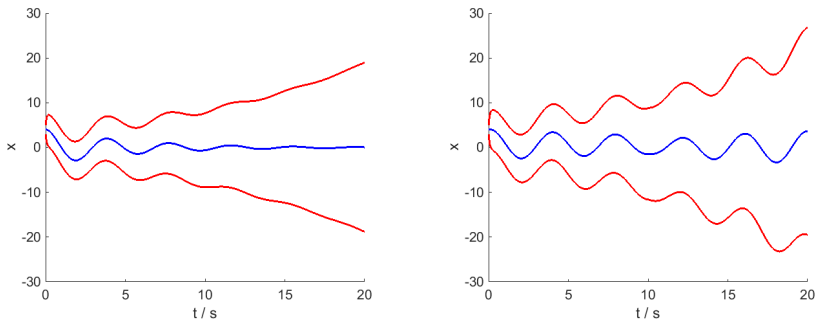
(a) Reach tube for network with delay $d = 0$ and gain $g = 1$

(b) Reach tube for network with delay $d = 0$ and gain $g = 0.5$

(c) Reach tube for network with delay $d = 10s$ and gain $g = 0.5$

Fig. 2: Reach tubes for network with stable modules projected on one component of x_1 . A_1 and A_2 are both Hurwitz. The blue curve is a trajectory, and the red curves outline a reach tube from a neighborhood.

Our implementation uses the built-in ODE solver of Matlab and these illustrative experiments assume that the error bounds of the computation in line 5 and 17 are met by Matlab's ODE solver, but in reality this may not hold. However, from a sequence of sample simulation points, one can construct a sequence of simulation boxes satisfying the rigorous requirements of Definition 9 using methods such as the one presented in [39]. Alternatively, for one could use a validated ODE solver such as CAPD [29] as in the C2E2 verificaiton tool [27].



(a) Reach tube for network with delay 0

(b) Reach tube for network with delay $10s$

Fig. 3: Reach tubes for network with an unstable module projected on one component of x_1. A_2 is Hurwitz while A_1 is not. The blue curve is a trajectory, and the red curves outline a reach tube from a neighborhood.

IX. Conclusions

The technique we present for proving bounded time safety properties of (possibly unstable) nonlinear dynamical networks with delayed interconnection uses numerical simulations and IS discrepancy functions for the modules. IS discrepancy of a module $\mathcal{A}_i$, bounds the distance between two (possibly diverging) trajectories of $\mathcal{A}_i$ in terms of their initial states and inputs. It is closely related to the notion of input-to-state stability that is well studied in control theory, but an important distinction is that it does not require the subsystems or the overall system to be stable. Consequently, our construction of the low dimensional dynamical network $M(\delta)$ with the same delayed interconnection that gives a bound on the divergence of trajectories of $\mathcal{A}$, does not rely on any global properties like small-gain of the interconnection nor stability of $\mathcal{A}$, but instead only uses the individual IS discrepancy functions and the numerical simulations of $\mathcal{A}$ and $M(\delta)$. Further, we also show that by choosing appropriately small δ's the overapproximations can be made arbitrarily precise; and therefore our verification algorithm is sound and relatively complete. To make this technique practical, we introduce a local version of IS discrepancy functions and present an algorithm to compute them using Lipschitz constant and the Jacobian of the dynamic functions of the modules. The results suggest some interesting future directions. We plan to extend the results to switched or hybrid systems. An orthogonal direction is to use similar compositional analysis to synthesize dynamical networks that satisfy safety properties.

References

[1] A. Donzé, "Breach, a toolbox for verification and parameter synthesis of hybrid systems," in *Computer Aided Verification: 22nd International Conference, Edinburgh, UK, July 15-19*. Springer, 2010, pp. 167–170.

[2] Y. Annpureddy, C. Liu, G. Fainekos, and S. Sankaranarayanan, "S-taliro: A tool for temporal logic falsification for hybrid systems," in *Tools and Algorithms for the Construction and Analysis of Systems: 17th International Conference, March 26–April 3*. Springer, 2011, pp. 254–257.

[3] P. Duggirala, S. Mitra, and M. Viswanathan, "Verification of annotated models from executions," in *International Conference on Embedded Software*, 2013.

[4] Z. Huang and S. Mitra, "Proofs from simulations and modular annotations," in *Proceedings of the 17th international conference on Hybrid systems: computation and control*. ACM, 2014, pp. 183–192.

[5] Z. Huang, C. Fan, A. Mereacre, S. Mitra, and M. Kwiatkowska, "Invariant verification of nonlinear hybrid automata networks of cardiac cells," in *Computer Aided Verification*. Springer, 2014, pp. 373–390.

[6] M. Jha and S. Raskhodnikova, "Testing and reconstruction of lipschitz functions with applications to data privacy," *SIAM Journal on Computing*, vol. 42, no. 2, pp. 700–731, 2013.

[7] Z. Han and P. J. Mosterman, "Towards sensitivity analysis of hybrid systems using simulink," in *Proceedings of the 16th international conference on Hybrid systems: computation and control*. ACM, 2013, pp. 95–100.

[8] D. Angeli, "A lyapunov approach to incremental stability properties," *Automatic Control, IEEE Transactions on*, vol. 47, no. 3, pp. 410–421, 2002.

[9] E. M. Aylward, P. A. Parrilo, and J.-J. E. Slotine, "Stability and robustness analysis of nonlinear systems via contraction metrics and sos programming," *Automatica*, vol. 44, no. 8, pp. 2163–2170, 2008.

[10] U. Topcu, A. Packard, and R. Murray, "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*, Dec 2009, pp. 1175–1180.

[11] C. Fan and S. Mitra, "Bounded verification with on-the-fly discrepancy computation," in *13th International Symposium on Automated Technology for Verification and Analysis*. Springer, 2015, pp. 446–463.

[12] E. D. Sontag, "Comments on integral variants of iss," *Systems & Control Letters*, vol. 34, no. 1-2, pp. 93 – 100, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167691198000036

[13] D. Angeli, E. D. Sontag, and Y. Wang, "A characterization of integral input-to-state stability," *Automatic Control, IEEE Transactions on*, vol. 45, no. 6, pp. 1082–1097, 2000.

[14] D. Angeli, "Further results on incremental input-to-state stability," *Automatic Control, IEEE Transactions on*, vol. 54, no. 6, pp. 1386–1391, 2009.

[15] A. Girard, G. Pola, and P. Tabuada, "Approximately bisimilar symbolic models for incrementally stable switched systems," *Automatic Control, IEEE Transactions on*, vol. 55, no. 1, pp. 116–126, 2010.

[16] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508–2516, 2008.

[17] P. Tabuada, A. D. Ames, A. Julius, and G. J. Pappas, "Approximate reduction of dynamic systems," *Systems & Control Letters*, vol. 57, no. 7, pp. 538–545, 2008.

[18] M. A. Islam, A. Murthy, A. Girard, S. A. Smolka, and R. Grosu, "Compositionality results for cardiac cell dynamics," in *Proceedings of the 17th international conference on Hybrid systems: computation and control*. ACM, 2014, pp. 243–252.

[19] L. Zou, M. Fränzle, N. Zhan, and P. N. Mosaad, "Automatic verification of stability and safety for delay differential equations," in *Computer Aided Verification*. Springer, 2015, pp. 338–355.

[20] S. Mitra, *A verification framework for hybrid systems*, 2007, vol. 68, no. 12, PhD dissertation, MIT.

[21] D. K. Kaynar, N. Lynch, R. Segala, and F. Vaandrager, *The Theory of Timed I/O Automata*. Morgan Claypool, November 2005, also available as Technical Report MIT-LCS-TR-917.

[22] J. Nagumo, S. Arimoto, and S. Yoshizawa, "An active pulse transmission line simulating nerve axon," *Proceedings of the IRE*, vol. 50, no. 10, pp. 2061–2070, 1962.

[23] Y. Kuang, *Delay differential equations: with applications in population dynamics*. Academic Press, 1993.

[24] G. Wood and B. Zhang, "Estimation of the lipschitz constant of a function," *Journal of Global Optimization*, vol. 8, no. 1, pp. 91–103, 1996.

[25] P. Benner, J.-R. Li, and T. Penzl, "Numerical solution of large-scale lyapunov equations, riccati equations, and linear-quadratic optimal control problems," *Numerical Linear Algebra with Applications*, vol. 15, no. 9, pp. 755–777, 2008.

[26] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 1992.

[27] P. Duggirala, S. Mitra, M. Viswanathan, and M. Potok, "C2E2: A verification tool for stateflow models," in *Tools and Algorithms for the Construction and Analysis of Systems*, ser. Lecture Notes in Computer Science, vol. 9035. Springer Berlin Heidelberg, 2015, pp. 68–82.

[28] P. S. Duggirala, C. Fan, S. Mitra, and M. Viswanathan, "Meeting a powertrain verification challenge," in *Computer Aided Verification - 27th International Conference, CAV 2015, San Francisco, CA, USA, July 18-24, 2015, Proceedings, Part I*, 2015, pp. 536–543. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-21690-4_37

[29] P. Zgliczynski, CAPD: "Computer assisted proofs in dynamics," 2002. Available: http://www.capd.ii.uj.edu.pl/

[30] N. S. Nedialkov, K. R. Jackson, and G. F. Corliss, "Validated solutions of initial value problems for ordinary differential equations," *Applied Mathematics and Computation*, vol. 105, no. 1, pp. 21–68, 1999.

[31] O. Bouissou and M. Martel, "Grklib: a guaranteed runge kutta library," in *Scientific Computing, Computer Arithmetic and Validated Numerics, 2006. SCAN 2006. 12th GAMM-IMACS International Symposium on*. IEEE, 2006, pp. 8–8.
[32] A. Bellen and M. Zennaro, *Numerical methods for delay differential equations*. Oxford University Press, 2013.
[33] L. F. Shampine and S. Thompson, "Solving ddes in matlab," *Applied Numerical Mathematics*, vol. 37, no. 4, pp. 441–458, 2001.
[34] K. Engelborghs, T. Luzyanina, and D. Roose, "Numerical bifurcation analysis of delay differential equations using dde-biftool," *ACM Transactions on Mathematical Software (TOMS)*, vol. 28, no. 1, pp. 1–21, 2002.
[35] L. F. Shampine and S. Thompson, "Numerical solution of delay differential equations," in *Delay Differential Equations*. Springer, 2009, pp. 1–27.
[36] J. Maidens and M. Arcak, "Reachability analysis of nonlinear systems using matrix measures," *Automatic Control, IEEE Transactions on*, vol. 60, no. 1, pp. 265–270, Jan 2015.
[37] L. Scardovi and R. Sepulchre, "Synchronization in networks of identical linear systems," *Automatica*, vol. 45, no. 11, pp. 2557–2562, 2009.
[38] X. Chen, E. Abrahám, and S. Sankaranarayanan, "Taylor model flowpipe construction for non-linear hybrid systems," in *Real-Time Systems Symposium (RTSS), 2012 IEEE 33rd*. IEEE, 2012, pp. 183–192.
[39] Z. Huang and S. Mitra, "Computing bounded reach sets from sampled simulation traces," in *Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control*. ACM, 2012, pp. 291–294.