Automatica 164 (2024) 111637

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Contract composition for dynamical control systems: Definition and verification using linear programming[☆]

Miel Sharf [a,*], Bart Besselink [b], Karl Henrik Johansson [a]

[a] Division of Decision and Control Systems, KTH Royal Institute of Technology, and Digital Futures, 10044 Stockholm, Sweden
[b] Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, 9700 AK Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 29 October 2022
Received in revised form 16 October 2023
Accepted 19 February 2024
Available online 29 March 2024

Keywords:
Formal methods
Contracts
Linear programming
Modular design
Graph theory
Interconnection topology

ABSTRACT

Designing large-scale control systems to satisfy complex specifications is hard in practice, as most formal methods are limited to systems of modest size. Contract theory has been proposed as a modular alternative, in which specifications are defined by assumptions on the input to a component and guarantees on its output. However, current contract-based methods for control systems either prescribe guarantees on the state of the system, going against the spirit of contract theory, or are not supported by efficient computational tools. In this paper, we present a contract-based modular framework for discrete-time dynamical control systems. We extend the definition of contracts by allowing the assumption on the input at a time k to depend on outputs up to time $k-1$, which is essential when considering feedback systems. We also define contract composition for arbitrary interconnection topologies, and prove that this notion supports modular design, analysis and verification. This is done using graph theory methods, and specifically using the notions of topological ordering and backward-reachable nodes. Lastly, we present an algorithm for verifying vertical contracts, which are claims of the form "the conjunction of given component-level contracts implies given contract on the integrated system". These algorithms are based on linear programming, and scale linearly with the number of components in the interconnected network. A numerical example is provided to demonstrate the scalability of the presented approach, as well as the modularity achieved by using it.

1. Introduction

In recent years, engineering systems have become larger and more complex than ever, as the number of different components and subsystems is rapidly increasing due to the prominence of the "system-of-systems" design philosophy. At the same time, these systems are subject to specifications with constantly increasing intricacy, including safety and performance specifications. As a result, the validation and verification process, which must be conducted before deployment, has become exponentially more difficult. Recently, several attempts have been made to adapt contract theory, which is a modular approach for software design, to dynamical control systems. In this paper, we present a modular approach for contract-based design of dynamical control systems by defining a "contract algebra", considering the composition of contracts on different components with a general interconnection topology. We prove that our definition supports independent design, analysis, and verification of the components or subsystems. We also prescribe linear programming (LP)-based tools for verifying that a given contract on the integrated system is implied by a collection of component-level contracts.

1.1. Background

Modularity is a widely accepted philosophy of system design. Identifying a natural partition of a large-scale system into smaller modules enables independent and parallel work on the different components by different teams, as well as outsourcing part of the work to a subcontractor. Modular design also supports future modifications in the design, as only the updated components need to be re-verified rather than the entire system. For these reasons, a wide range of literature advocates for designing large-scale systems using as much modularity as possible, see Baldwin and Clark (2006) and Huang and Kusiak (1998) for discussions on modular design in engineering systems. The opposite approach, in which a single designer integrates all parts of the system, is known as integral design (Ulrich, 1995).

Traditional control system specifications do not always enable modular verification. Safety is most commonly defined via

[☆] The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Dimitra Panagou under the direction of Editor Christos G. Cassandras.

[*] Corresponding author.

E-mail addresses: sharf@kth.se (M. Sharf), b.besselink@rug.nl (B. Besselink), kallej@kth.se (K.H. Johansson).

https://doi.org/10.1016/j.automatica.2024.111637
0005-1098/© 2024 Elsevier Ltd. All rights reserved.

controlled invariant sets (Blanchini & Miani, 2008), but those can only handle rudimentary safety specifications, and cannot be applied modularly. Existing modular techniques, such as dissipativity theory (Willems, 1972a, 1972b), can only handle limited performance specifications, and cannot be used for safety. In contrast, formal methods in control provide verification methods and correct-by-design synthesis procedures for specifications given by temporal logic formulae (Belta, Yordanov, & Gol, 2017; Tabuada, 2009). However, they typically scale exponentially with the dimension of the system, and are thus applicable only to systems of modest size. Also, most works on scalable distributed and decentralised control methods, such as Rantzer (2015) and Šiljak and Zečević (2005), are not modular, as they require a single authority with complete knowledge of the system model to design the decentralised or distributed controllers.

Lately, several modular approaches have been proposed to tackle problems in the design of dynamical control systems. One example is composition-compatible notions of abstraction and simulation, attempting to "modularise" formal methods in control (Saoud, Jagtap, Zamani, & Girard, 2018; Zamani & Arcak, 2018). Another approach attempts to relate controlled-invariant sets and reachability analysis on the subsystem-level to controlled-invariant sets and reachability analysis on the composite system-level (Chen, Herbert, Vashishtha, Bansal, & Tomlin, 2018; Smith, Nilsson, & Ozay, 2016). A third approach, and the focus of this paper, is contract theory. Contract theory is the most prominent modular design philosophy in software engineering (Benveniste et al., 2018; Meyer, 1992). A contract is a specification that explicitly defines assumptions on the input and desired guarantees on the output of each software component. Contract theory supports this with tools for verification as well as for comparing and composing contracts, where the latter two are crucial for enabling modularity.

Motivated by these features, there is an increased interest in using contract theory in the realm of dynamical control systems. The works of Nuzzo et al. apply contract theory to the "cyber" aspects of cyber–physical systems, see Nuzzo, Sangiovanni-Vincentelli, Bresolin, Geretti, and Villa (2015), Nuzzo, Xu, Ozay, Finn, Sangiovanni-Vincentelli, Murray, Donzé, and Seshia (2014) and references therein. More recently, other attempts have been made to apply it to dynamical control systems. So-called parametric assume-guarantee contracts are introduced in Kim, Arcak, and Seshia (2017), see also Chen, Anderson, Kalsi, Ames, and Low (2021), building on linear temporal logic specifications for discrete-time systems. Such systems are also studied in the works (Eqtami & Girard, 2019; Ghasemi, Sadraddini, & Belta, 2020; Girard, Iovine, & Benberkane, 2022; Saoud, Girard, & Fribourg, 2018), relying on set-invariance techniques. In these works, however, guarantees on the system state are often included. This is a limitation in contract theory, as the state of the system is an internal variable that should not be a part of its interface. A similar observation holds for Saoud, Girard, and Fribourg (2021), which considers a very general framework for contracts for continuous-time systems. Also considering continuous-time systems, Besselink, Johansson, and Schaft (2019) and Shali, Heidema, van der Schaft, and Besselink (2022) employ verification techniques based on geometric control theory, whereas Shali, van der Schaft, and Besselink (2023) builds on behavioural systems theory. Works targeting contract-based controller synthesis include Ghasemi, Sadraddini, and Belta (2022), Kim, Arcak, and Seshia (2015) and Liu, Saoud, Jagtap, Dimarogonas, and Zamani (2022).

A key disadvantage of many of the above works is that the proposed contract theory is not supported by efficient computational tools. The work (Sharf, Besselink, Molin, Zhao, & Johansson, 2021), however, presents preliminary results on a contract theory that is supported by computational tools based on linear programs for verifying satisfaction. These are extended in Sharf, Besselink, and Johansson (2021). Still, only very rudimentary notions for composition of these LP-based contracts, a crucial element in building a full contract theory supporting modular design, are given in Sharf, Besselink, et al. (2021). The current paper aims to fill this gap.

1.2. Contributions

This paper develops a modular and compositional framework based on contract theory for discrete-time control systems and has the following main contributions.

First, we present contracts in which the assumptions on the input at time k are allowed to depend on the values of the system output up to time $k-1$. These contracts arise naturally when considering feedback control and considerably extend existing methods in the literature, where assumptions (on the input) are not allowed to be "responsive" to the output.

Second, we define contract composition for arbitrary network interconnections and prove that the composition supports modular design, analysis, and verification, in the sense that the verification of component-level contracts guarantees that the composite system satisfies its global contract. These results are first achieved for networks without feedback loops (Definition 4.2 and Algorithm 1), and are later generalised to arbitrary well-posed network interconnections (Definition 5.3 and Algorithm 2).

Third, we ensure that the contract framework of this paper is fully supported by efficient computational tools and provide LP-based algorithms for verification and composition. We prove that the presented algorithms are always correct, non-conservative, and scale linearly with the number of components in the integrated system.

The paper is organised as follows. Section 2 presents the class of discrete-time systems considered in the paper. Section 3 introduces generalised contracts, as well as a formal definition of the problems discussed in the paper. Section 4 considers networks without feedback, and Section 5 considers general well-posed networks. Section 6 applies these methods in a numerical example.

Notation. Let $\mathbb{N} = \{0, 1, \ldots\}$ be the set of natural numbers. For $n_1, n_2 \in \mathbb{N}$, we let $\mathcal{I}_{n_1,n_2} = \{n_1, \ldots, n_2\}$ if $n_1 \leq n_2$, and $\mathcal{I}_{n_1,n_2} = \emptyset$ otherwise. The collection of discrete-time signals $\mathbb{N} \to \mathbb{R}^d$ will be denoted by $\mathcal{S}^d$. We say that $d_1 \in \mathcal{S}^{n_{d_1}}$ is a subsignal of $d \in \mathcal{S}^{n_d}$ if there exists a permutation matrix $P \in \mathbb{R}^{n_d \times n_d}$ such that $\begin{pmatrix} d_1(k) \\ d_2(k) \end{pmatrix} = Pd(k)$ for all $k \in \mathbb{N}$, for some signal $d_2 \in \mathcal{S}^{n_d - n_{d_1}}$. We refer to d_2 as the complementary subsignal to d_1. For a signal $v \in \mathcal{S}^m$ and $k_1, k_2 \in \mathbb{N}$, we denote the vector containing $v(k_1), v(k_1+1), \ldots, v(k_2)$ as $v(k_1 : k_2) \in \mathbb{R}^{(k_2-k_1+1)m}$. A set-valued map $f : X \rightrightarrows Y$ between two sets X, Y associates a subset $f(x) \subseteq Y$ to any element $x \in X$. Moreover, X^n is the set of n-tuples of elements of X. For vectors $u, v \in \mathbb{R}^n$, we write $u \leq v$ if and only if $u_i \leq v_i$ holds for any coordinate $i \in \mathcal{I}_{1,n}$.

2. Systems and composition

We first define the class of systems we consider, which are seen as operators on the set of all possible signals.

Definition 2.1. A (dynamical) system Π with input $d \in \mathcal{S}^{n_d}$ and output $y \in \mathcal{S}^{n_y}$ is a set-valued map $\Pi : \mathcal{S}^{n_d} \rightrightarrows \mathcal{S}^{n_y}$. In other words, for any input trajectory $d \in \mathcal{S}^{n_d}$, $\Pi(d)$ is the set of all corresponding output trajectories.

Here, we consider set-valued maps rather than functions to also consider cases in which an input trajectory can have more than one associated output trajectory, e.g., due to initial conditions or non-determinism.

Example 2.1. Consider the class of systems governed by

$$\begin{aligned} x(k+1) &\in \mathcal{F}(x(k), d(k)),\ \forall k \in \mathbb{N}, \quad x(0) \in \mathcal{X}_0, \\ y(k) &\in \mathcal{H}(x(k), d(k)),\ \forall k \in \mathbb{N}, \end{aligned} \tag{1}$$

where $x \in \mathcal{S}^{n_x}$ is the state of the system, $\mathcal{X}_0$ is a set of admissible initial conditions, and $\mathcal{F}, \mathcal{H} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \rightrightarrows \mathbb{R}^{n_x}$ are set-valued maps defining the state evolution and observation, respectively. This class of systems is included within Definition 2.1 and contains all systems with both linear and non-linear (time-invariant) dynamics, as well as perturbed, unperturbed or uncertain dynamics. Thus, the formalism of Definition 2.1 includes many systems often considered within the scope of control theory.

Systems governed by (1) are causal, i.e., the output up to time k is independent of inputs beyond time k. As causality will be the key property allowing us to define composition in Section 5. we explicitly define it for general systems as in Definition 2.1:

Definition 2.2. Let $\Pi : \mathcal{S}^{n_d} \rightrightarrows \mathcal{S}^{n_y}$ be a system with input $d \in \mathcal{S}^{n_d}$ and output $y \in \mathcal{S}^{n_y}$ and let $d_1 \in \mathcal{S}^{n_{d_1}}$ be a subsignal of d. The system Π is *causal* with respect to d_1 if, for any time k, $y(k)$ does not depend on $d_1(k+1), d_1(k+2), \dots$. It is *strictly causal* with respect to d_1 if, for any time k, $y(k)$ is also independent of $d_1(k)$. If Π is causal with respect to d, we simply say it is causal, without mentioning a subsignal.

Remark 2.1. Causality with respect to d, à la Definition 2.2, is equivalent to the standard definition of causality using truncation operators (Desoer & Vidyasagar, 2009). Equivalently, there is a one-to-one correspondence between causal systems $\Pi : d \mapsto y$ and sets of timewise set-valued maps $\{\Pi_k\}_{k \in \mathbb{N}}$ mapping $d(0:k)$ to $y(k)$.

As we will be interested in networks of systems as in Definition 2.1, we briefly recall some graph theory. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set of vertices (or nodes) $\mathcal{V}$, and a set of edges $\mathcal{E}$, which are pairs of vertices. In this paper, we consider *directed* graphs. If $i, j \in \mathcal{V}$, the edge e from i to j is denoted $i \to j \in \mathcal{E}$, and we say that i is e's tail, and j its head.

Now, consider multiple components interconnected according to a network with associated graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, thus forming a composite system. The input to the ith component, d_i, is given by

$$d_i(k) = \textstyle\sum_{j \in \mathcal{V}} F_{ij} y_j(k) + E_i d^{\text{ext}}(k), \tag{2}$$

where $d^{\text{ext}} \in \mathcal{S}^{n_{d^{\text{ext}}}}$ is an external input to the composite system, which is distributed over the components according to the matrices $\{E_i\}_{i \in \mathcal{V}}$. The matrix $F_{ij} \in \mathbb{R}^{n_{d_i} \times n_{y_j}}$ captures the influence that component $j \in \mathcal{V}$ has on the ith component and, as such, it is nonzero if and only if $j \to i \in \mathcal{E}$.

To define an external output, we choose a set $\mathcal{V}^{\text{out}} \subset \mathcal{V}$ of "output components" that contribute to the external output y^{ext} as

$$y^{\text{ext}}(k) = \textstyle\sum_{i \in \mathcal{V}} H_i y_i(k), \tag{3}$$

where the matrix H_i is nonzero if and only if $i \in \mathcal{V}^{\text{out}}$.

This allows us to formally define system composition.

Definition 2.3. Consider systems $\{\Pi_i\}_{i \in \mathcal{V}}$ interconnected according to the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with associated matrices $\{F_{ij}\}_{i,j \in \mathcal{V}}$ and $\{E_i, H_i\}_{i \in \mathcal{V}}$. The composite system, denoted by $\bigotimes_{i \in \mathcal{V}} \Pi_i$, is a system with input d^{ext} and output y^{ext}, defined by the following set-valued function: We say that $y^{\text{ext}} \in \bigotimes_{i \in \mathcal{V}} \Pi_i(d^{\text{ext}})$ if there exist signals $d_i \in \mathcal{S}^{n_{d_i}}$ and $y_i \in \mathcal{S}^{n_{y_i}}$ such that:

(i) $y_i \in \Pi_i(d_i)$ for all $i \in \mathcal{V}$;
(ii) (2) and (3) hold for all $i \in \mathcal{V}$ and all $k \in \mathbb{N}$.

In the remainder, we will sometimes refer to the composite system $\bigotimes_{i \in \mathcal{V}} \Pi_i$ as the network. We note that a cycle in $\mathcal{G}$ corresponds to a feedback loop in the composite system. Namely, if Π_i is a system that is part of a cycle, its input d_i depends on its output y_i, potentially after passing through other systems in the network.

Remark 2.2. Definition 2.3 states composition in terms of the consistency relations (2) and (3), which can be made concrete for instance for systems of the form (1). However, the definition also obfuscates the problem of algebraic loops, which might exist even when only considering causal systems. More precisely, any algebraic loop corresponds to a cycle in $\mathcal{G}$ traversing through the nodes $i_1, \dots, i_l$, where the corresponding systems $\Pi_{i_1}, \dots, \Pi_{i_l}$ are causal (but not strictly causal). A thorough investigation of algebraic loops will be considered in Section 5, in which networks with feedback will be considered.

We will be interested in guaranteeing a specification on the composite system $\bigotimes_{i \in \mathcal{V}} \Pi_i$ on the basis of specifications on the components $\{\Pi_i\}_{i \in \mathcal{V}}$. These specifications will be formalised in terms of assume/guarantee contracts, which are the topic of the next section.

3. Contracts and problem formulation

This section defines a class of contracts for (discrete-time) dynamical systems, and gives a detailed formulation of the problem statement.

3.1. Recursively-defined contracts

We start by defining recursively-defined contracts as a class of assume/guarantee contracts.

Definition 3.1. A *recursively-defined* (RD) *contract* is a pair $(\mathcal{D}, \Omega)$ of sets inside $\mathcal{S}^{n_d + n_y}$ of the form

$$\mathcal{D} = \left\{ \left(\begin{smallmatrix} d(\cdot) \\ y(\cdot) \end{smallmatrix}\right) : d(k) \in A_k\left(\begin{smallmatrix} d(0:k-1) \\ y(0:k-1) \end{smallmatrix}\right), \forall k \right\}, \tag{4}$$

$$\Omega = \left\{ \left(\begin{smallmatrix} d(\cdot) \\ y(\cdot) \end{smallmatrix}\right) : \left[\begin{smallmatrix} d(k) \\ y(k) \end{smallmatrix}\right] \in G_k\left(\begin{smallmatrix} d(0:k-1) \\ y(0:k-1) \end{smallmatrix}\right), \forall k \right\}, \tag{5}$$

for some set-valued functions $A_k : (\mathbb{R}^{n_d} \times \mathbb{R}^{n_y})^k \rightrightarrows \mathbb{R}^{n_d}$ and $G_k : (\mathbb{R}^{n_d} \times \mathbb{R}^{n_y})^k \rightrightarrows \mathbb{R}^{n_d} \times \mathbb{R}^{n_y}$. The sets $\mathcal{D}$ and Ω are called assumptions and guarantees, respectively.

The relevance of these contracts becomes clear through the following definition, which shows how an RD contract acts as a specification on a dynamical system.

Definition 3.2. The system Π satisfies the RD contract $\mathcal{C} = (\mathcal{D}, \Omega)$, denoted as $\Pi \vDash \mathcal{C}$, if the following implication holds for any $d \in \mathcal{S}^{n_d}, y \in \mathcal{S}^{n_y}$: if $y \in \Pi(d)$ and $(d, y) \in \mathcal{D}$ hold, then $(d, y) \in \Omega$.

Thus, at any time k, the assumptions $\mathcal{D}$ define a class of inputs $d(k)$ that a system may expect, whereas the guarantees Ω specify the required output $y(k)$. We emphasise that RD contracts put assumptions on the input $d(k)$ in terms of the previous inputs and outputs, and guarantees on the output $y(k)$ in terms of the previous inputs, the previous outputs, and the current input. Note that although the assumptions are written as $(d, y) \in \mathcal{D}$, they only restrict $d(k)$. The assumptions are allowed to "react" to $y(0:k-1)$, but cannot restrict it in any form. This feature is particularly relevant in systems with feedback, as illustrated next.

Example 3.1. Consider a dynamical system with input $d(\cdot) \in \mathcal{S}^2$ and output $y(\cdot) \in \mathcal{S}^1$. The input $d(\cdot)$ has two subsignals d_1, d_2. The signal $d_1(\cdot) \in \mathcal{S}^1$ is a disturbance that should be rejected, and the signal $d_2(\cdot) \in \mathcal{S}^1$ is a control input. We assume $d_1(\cdot)$ is small, and that $d_2(\cdot)$ is the output of a proportional controller with gain K and a small actuation error. We wish to guarantee that $y(\cdot)$ is close enough to zero. This specification can be expressed as the RD contract $\mathcal{C} = (\mathcal{D}, \Omega)$ with

$$\mathcal{D} = \{(d, y) : |d_1(k)| \le \epsilon_1, |d_2(k) - Ky(k-1)| \le \epsilon_2, \forall k\},$$
$$\Omega = \{(d, y) : |y(k)| \le \epsilon_3, \forall k\}.$$

Alternatively, one could specify asymptotic behaviour, e.g., convergence of $y(\cdot)$ to zero with rate $\sigma \in (0, 1)$:

$$\Omega = \{(d, y) : |y(k)| \le \sigma|y(k-1)|, \forall k\}.$$

Remark 3.1. The fact that the assumptions in RD contracts are allowed to depend on "past" outputs is a significant extension with respect to existing contracts in the literature, such as Saoud et al. (2021), in which the assumptions cannot "react" to the output. This extension is vital for dealing with feedback systems, but also in other cases where the "environment" can "react" to the system. For example, if an autonomous vehicle is driving on a two-lane highway along a human driver, it is reasonable to assume that the human cannot switch lanes when the two vehicles share a longitudinal position along different lanes (e.g., when the autonomous vehicle is overtaking the human driver), but can do so at other times. This condition restricts the behaviour of the environment (human driver) as a function of the position of the system (autonomous vehicle), and therefore cannot be incorporated into the existing frameworks in literature.

At the same time, the "recursively-defined" structure in (4) and (5) restricts the class of signals in the assumptions and guarantees when compared to the contracts in Kim et al. (2017) and Saoud et al. (2021). However, in addition to being naturally linked to system causality, this structure will allow us to develop efficient computational tools for RD contracts. Such tools are lacking for the abstract contracts in Kim et al. (2017) and Saoud et al. (2021).

The notion of refinement allows for comparing contracts.

Definition 3.3. Let $\mathcal{C} = (\mathcal{D}, \Omega)$ and $\mathcal{C}' = (\mathcal{D}', \Omega')$ be two RD contracts on the same system. We say that $\mathcal{C}$ refines $\mathcal{C}'$ (and write $\mathcal{C} \preccurlyeq \mathcal{C}'$) if $\mathcal{D} \supseteq \mathcal{D}'$ and $\Omega \cap \mathcal{D}' \subseteq \Omega' \cap \mathcal{D}'$.

Colloquially, $\mathcal{C} \preccurlyeq \mathcal{C}'$ if $\mathcal{C}$ assumes less than $\mathcal{C}'$, but guarantees more. In particular, we have that the implication

$$\Pi \vDash \mathcal{C} \text{ and } \mathcal{C} \preccurlyeq \mathcal{C}' \implies \Pi \vDash \mathcal{C}' \tag{6}$$

holds, strengthening the interpretation that $\mathcal{C}$ is a stricter requirement than $\mathcal{C}'$.

Later in the paper, we will consider a class of RD contracts that are amendable to efficient computational tools using linear programming.

Definition 3.4. A *linear time-invariant* (LTI) RD contract $\mathcal{C} = (\mathcal{D}, \Omega)$ of assumption depth $m_A \in \mathbb{N}$ and guarantee depth $m_G \in \mathbb{N}$ is given by matrices $\{\mathfrak{A}^r\}_{r=0}^{m_A}$, $\{\mathfrak{G}^r\}_{r=0}^{m_G}$ and vectors $\mathfrak{a}^0, \mathfrak{g}^0$ of appropriate sizes, where

$$\mathcal{D} = \left\{ \left(\begin{smallmatrix} d \\ y \end{smallmatrix}\right) : \mathfrak{A}^0 d(k) + \sum_{r=1}^{m_A} \mathfrak{A}^r \left[\begin{smallmatrix} d(k-r) \\ y(k-r) \end{smallmatrix}\right] \le \mathfrak{a}^0,\ \forall k \ge m_A \right\}$$
$$\Omega = \left\{ \left(\begin{smallmatrix} d \\ y \end{smallmatrix}\right) : \sum_{r=0}^{m_G} \mathfrak{G}^r \left[\begin{smallmatrix} d(k-r) \\ y(k-r) \end{smallmatrix}\right] \le \mathfrak{g}^0,\ \forall k \ge m_G \right\}. \tag{7}$$

Example 3.2. LTI RD contracts allow to express constraints of the form $(d(k), y(k)) \in \mathcal{P}$ for some polyhedral set $\mathcal{P}$. In fact, the set $\mathcal{P}$ is allowed to depend (linearly) on past signals. For further examples of LTI RD contracts, we refer to Section 6.

Thus, LTI RD contracts replace the set-valued functions A_k and G_k in (4) and (5) with inequalities involving a linear combination of the signal over a moving time window of length m_A and m_G, respectively. These linear contracts will enable the development of LP-based algorithms for contract operations.

Remark 3.2. We may assume that $m_A, m_G \ge 1$, as contracts of depth 0 are also contracts of depth 1.

For any LTI RD contract of the form (7), we consider two associated piecewise-linear functions $\alpha : (\mathbb{R}^{n_d})^{m_A+1} \times (\mathbb{R}^{n_y})^{m_A} \to \mathbb{R}$ and $\gamma : (\mathbb{R}^{n_d+n_y})^{m_G+1} \to \mathbb{R}$, given by

$$\begin{aligned} \alpha\left(\begin{smallmatrix} d(0:m_A) \\ y(0:m_A-1) \end{smallmatrix}\right) &= \max_i \alpha_i\left(\begin{smallmatrix} d(0:m_A) \\ y(0:m_A-1) \end{smallmatrix}\right), \\ \gamma\left(\begin{smallmatrix} d(0:m_G) \\ y(0:m_G) \end{smallmatrix}\right) &= \max_i \mathrm{e}_i^\top \left(\sum_{r=0}^{m_G} \mathfrak{G}^r \left[\begin{smallmatrix} d(k-r) \\ y(k-r) \end{smallmatrix}\right] - \mathfrak{g}^0 \right), \end{aligned} \tag{8}$$

where e_i is the ith column of the identity matrix and

$$\alpha_i\left(\begin{smallmatrix} d(0:m_A) \\ y(0:m_A-1) \end{smallmatrix}\right) = \mathrm{e}_i^\top \left(\mathfrak{A}^0 d(k) + \sum_{r=1}^{m_A} \mathfrak{A}^r \left[\begin{smallmatrix} d(k-r) \\ y(k-r) \end{smallmatrix}\right] - \mathfrak{a}^0 \right).$$

Using this notation, the contract (7) can be written as

$$\mathcal{D} = \left\{ \left(\begin{smallmatrix} d(\cdot) \\ y(\cdot) \end{smallmatrix}\right) : \alpha\left(\begin{smallmatrix} d(k-m_A:k) \\ y(k-m_A:k-1) \end{smallmatrix}\right) \le 0,\ \forall k \ge m_A \right\},$$
$$\Omega = \left\{ \left(\begin{smallmatrix} d(\cdot) \\ y(\cdot) \end{smallmatrix}\right) : \gamma\left(\begin{smallmatrix} d(k-m_G:k) \\ y(k-m_G:k) \end{smallmatrix}\right) \le 0,\ \forall k \ge m_G \right\}.$$

Lastly, let us define the notion of extendibility converting assumptions on $d(\cdot)$ by assumptions on $d(0:n)$ for times $n \in \mathbb{N}$. It manifests the self-consistency of the set of assumptions, in the sense that a signal satisfying the assumptions up to time k can be extended beyond time k while still satisfying the assumptions, see also a similar notion in Sharf, Besselink, et al. (2021).

Definition 3.5. Let $\mathcal{D} \subseteq \mathcal{S}^{n_d} \times \mathcal{S}^{n_y}$ be a set of the form (4). The set $\mathcal{D}$ is *extendable* if the following condition holds for any $k \in \mathbb{N}$ and any signals $d(\cdot), y(\cdot)$ defined at times $\{0, \dots, k\}$:

$$d(\ell+1) \in A_\ell\left(\begin{smallmatrix} d(0:\ell) \\ y(0:\ell) \end{smallmatrix}\right), \forall \ell \in \mathcal{I}_{0,k-1} \Rightarrow A_k\left(\begin{smallmatrix} d(0:k-1) \\ y(0:k-1) \end{smallmatrix}\right) \ne \emptyset.$$

Remark 3.3. For LTI contracts, we abuse the notation and say that α is extendable if $\mathcal{D}$ is, where (8) holds.

3.2. Contract composition and vertical contracts

Our aim is to guarantee specifications on the composite system $\bigotimes_{i \in \mathcal{V}} \Pi_i$ on the basis of specifications on the components. In particular, if the RD contract $\mathcal{C}_{\mathrm{tot}}$ is the desired specification on the composite system, we aim to answer the following question: given that the components Π_i satisfy component-level contracts $\mathcal{C}_i$, does $\bigotimes_{i \in \mathcal{V}} \Pi_i$ satisfy $\mathcal{C}_{\mathrm{tot}}$? To answer this question, we consider the composition of contracts.

Contract composition is considered by the meta-theory of Benveniste et al. (2018) for abstract contracts, relying on two modularity principles. Namely, given a collection of abstract contracts $\{\mathcal{C}_i\}_{i \in \mathcal{V}}$, the contract composition $\bigotimes_{i \in \mathcal{V}} \mathcal{C}_i$ is defined to satisfy the two following postulates:

(A) Its guarantees are the conjunction of the guarantees of all the $\mathcal{C}_i$-s.

(B) Its assumptions are defined as the largest set with the following property: for any i, the conjunction of these assumptions with the guarantees of $\mathcal{C}_j$ for all $j \neq i$ imply the assumptions of $\mathcal{C}_i$.

This definition supports modular design. Namely, Benveniste et al. (2018) show that if components $\{\Sigma_i\}_{i\in\mathcal{V}}$ satisfy $\mathcal{C}_i$ for $i \in \mathcal{V}$, then the composite system $\bigotimes_{i\in\mathcal{V}} \Sigma_i$ satisfies the composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$.

Unfortunately, this meta-theoretical definition cannot be directly applied to RD contracts for dynamical control systems for two main reasons. First, the definition in Benveniste et al. (2018) makes no distinction between external and internal variables, leading to situations in which the set of assumptions for the composed contract refers to the value of internal variables. Similarly, composition is only defined when the network output y^{ext} is composed of *all* "local" outputs y_i. Second, Benveniste et al. (2018) do not propose any computational tools for composition, e.g., a way to verify that a given contract on a composite system is refined by the composition of component-level contracts. The goal of this paper is to address both of these problems, specifically for contracts on (causal) dynamical control systems. This goal is explicitly formulated in the following problem statements:

Problem 3.1. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, RD contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$, and a set $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$ of output nodes, define the composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$, with input d^{ext} and output y^{ext} in a way compatible with postulates (A) and (B), while only using the external input d^{ext} and output y^{ext}.

We also show our definition satisfies the universal property of composition, namely, for causal systems Π_i,

$$\Pi_i \vDash \mathcal{C}_i \text{ for } i \in \mathcal{V} \implies \bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \bigotimes_{i\in\mathcal{V}} \mathcal{C}_i. \tag{9}$$

Once composition is defined, we can address the connection between contracts on different levels of abstraction, which will be referred to as vertical contracts:

Definition 3.6. Consider a network with a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a set $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$ of output nodes. A vertical contract is a statement of the form $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$, with $\mathcal{C}_{\text{tot}}$ an RD contract on the composite system and $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$ component-level RD contracts.

Problem 3.2. Find a computationally viable algorithm checking if a vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds.

If Problems 3.1 and 3.2 can be solved, we have the following key result.

Theorem 3.1. *Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an output set $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$, and component-level RD contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$. Assume that Problem 3.1 is solved, i.e., $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$ is defined and satisfies (9). Let $\mathcal{C}_{\text{tot}}$ be an RD contract on the composite system for which the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds. Then, if the systems $\{\Pi_i\}_{i\in\mathcal{V}}$ satisfy $\Pi_i \vDash \mathcal{C}_i$ for all $i \in \mathcal{V}$, we have $\bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \mathcal{C}_{\text{tot}}$.*

Proof. This follows directly from (6) and (9) as we have $\bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$, see also Proposition 1 of Sharf, Besselink, et al. (2021). □

The relevance of Theorem 3.1 is that it enables modularity. Namely, in order to guarantee that the composite system satisfies $\mathcal{C}_{\text{tot}}$, it is sufficient to verify $\Pi_i \vDash \mathcal{C}_i$ for each component (under the assumption that the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds). This can be verified completely *independently*, i.e., verification of $\Pi_i \vDash \mathcal{C}_i$ for a given $i \in \mathcal{V}$ does require knowledge of neither the other systems in the composite system nor the interconnection structure (see Definition 3.2). This also means that $\bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \mathcal{C}_{\text{tot}}$ is guaranteed for any components that satisfy their respective contracts. In particular, if a component Π_i is replaced by a component Π_i', it is sufficient to verify $\Pi_i' \vDash \mathcal{C}_i$ to guarantee that the composite system still satisfies $\mathcal{C}_{\text{tot}}$.

This paper focuses on defining and verifying the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$. LP-based tools for verifying contract satisfaction on component-level $\Pi_i \vDash \mathcal{C}_i$ can be found in Sharf, Besselink, and Johansson (2021).

Before presenting solutions to Problems 3.1 and 3.2, we make an important remark about the output set $\mathcal{V}^{\text{out}}$. RD contracts allow the assumption to depend on previous outputs, and these assumptions should still be manifested in the composition. Thus, relevant "local" outputs y_i must be available as a part of the "global" output y^{ext}:

Assumption 3.1. For any $i \in \mathcal{V}$, if the assumption on the external input d^{ext} explicitly depends on the output y_i in the RD contract $\mathcal{C}_i$, then $i \in \mathcal{V}^{\text{out}}$.

In other words, if the component-level assumption on the external input depends on the output y_i, then y_i should be a part of the output of the composite system. This assures that the assumptions of the composition will not depend on any internal variables.

4. Networks without feedback

In this section, we propose solutions to Problems 3.1 and 3.2 for composite systems whose interconnection is characterised by a directed acyclic graph, i.e., networks without feedback. We first introduce directed acyclic graphs, define composition for networks without feedback, and then show that the correctness of vertical contracts can be verified using LP-enabled tools.

4.1. Directed acyclic graphs

Before defining directed acyclic graphs, consider a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and recall that, for an edge $e = i \to j \in \mathcal{E}$, i and j are e's tail and head, respectively. A path is a sequence of edges $e_1, e_2, \dots, e_l \in \mathcal{E}$ such that e_r's head is e_{r+1}'s tail for all $r \in \mathcal{I}_{1,l-1}$. The path is called a cycle if e_l's head is e_1's tail. For a node $i \in \mathcal{V}$, the node $j \in \mathcal{V}$ is *backward-reachable* from i if there exists a path from j to i. The collection of all backward-reachable nodes from $i \in \mathcal{V}$ is denoted $\text{BR}(i)$. We also denote $\text{BR}_+(i) = \text{BR}(i) \cup \{i\}$.

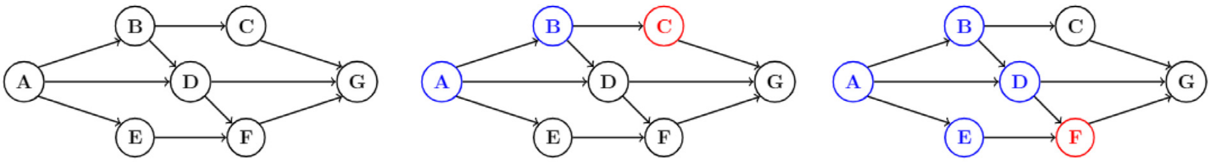
A directed acyclic graph (DAG) is a directed graph $\mathcal{G}$ containing no cycles. DAGs play a vital role in algorithm design and analysis of many problems, e.g., the shortest-path problem (Cormen, Leiserson, Rivest, & Stein, 2009). This is largely due to the tool of topological ordering:

Definition 4.1. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with N nodes. A topological ordering is a map $\sigma : \mathcal{I}_{1,N} \to \mathcal{V}$ such that:

(i) If $p, q \in \mathcal{I}_{1,N}$ satisfy $p \neq q$, then $\sigma(p) \neq \sigma(q)$.
(ii) If $p, q \in \mathcal{I}_{1,N}$ satisfy $\sigma(p) \to \sigma(q) \in \mathcal{E}$, then $p < q$.

Namely, a graph has a topological ordering if and only if it is a DAG. There are linear-time algorithms for finding a topological ordering of a DAG, and for checking whether a graph is a DAG, e.g., relying on depth-first search [Cormen et al. (2009), p. 613–614]. We will repeatedly apply the following lemma connecting backward-reachability and topological ordering.

Lemma 4.1. *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a DAG with topological ordering $\sigma : \mathcal{I}_{1,N} \to \mathcal{V}$. For any $q \in \mathcal{I}_{1,N}$, we have that $\text{BR}(\sigma(q)) \subseteq \{\sigma(1), \dots, \sigma(q-1)\}$.*



(a) The graph $\mathcal{G}$, which is a DAG. (b) The node C (red) and $\mathrm{BR}(C)$ (blue) (c) The node F (red) and $\mathrm{BR}(F)$ (blue)

Fig. 1. An example of a DAG $\mathcal{G}$ and two backward-reachable sets. The graph $\mathcal{G}$ has a total of 11 different topological orderings, including ABCDEFG, ABDEFCG and AEBDFCG. Here, the notation ABCDEGF is shorthand for a topological ordering σ defined as $\sigma(1) = \mathrm{A}$, $\sigma(2) = \mathrm{B}$, etc.

Proof. Follows from the part (ii) of Definition 4.1. □

Pictorially, a topological ordering is an ordering of the vertices on a horizontal line such that all edges go from left to right. Fig. 1 gives an example of a DAG, together with some sets $\mathrm{BR}(i)$ and topological orderings.

In the remainder of this section, we consider composite systems whose underlying interconnection structure is given by a DAG. We refer to such composite systems as networks without feedback.

4.2. Contract composition

We wish to define the composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$ as to satisfy postulates (A) and (B), while only using the external input d^{ext} and the external output y^{ext}. Postulate (A), defining the guarantees of the composition, will be adapted by requiring the existence of signals $(d_i, y_i) \in \Omega_i$ for $i \in \mathcal{V}$ such that the consistency relations (2) and (3) hold. As for postulate (B), instead of considering all components $j \neq i$, it suffices to consider components j which precede i (in the sense of backward reachability). Indeed, these are the only components whose output can affect input d_i, as there are no feedback loops.

Definition 4.2. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a DAG, $\mathcal{V}^{\mathrm{out}} \subseteq \mathcal{V}$ a set of output nodes, and $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ be component-level RD contracts, so that Assumption 3.1 holds. The composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i = (\mathcal{D}_\otimes, \Omega_\otimes)$, having input $d^{\mathrm{ext}}(\cdot)$ and output $y^{\mathrm{ext}}(\cdot)$, is defined as follows:

- $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \mathcal{D}_\otimes$ if for any signals $\{d_i(\cdot), y_i(\cdot)\}_{i\in\mathcal{V}}$ satisfying the input-consistency constraints (2) and output-consistency constraints (3), the following implication holds for all $i \in \mathcal{V}$: if $(d_j, y_j) \in \Omega_j$ holds for all $j \in \mathrm{BR}(i)$, then $(d_i, y_i) \in \mathcal{D}_i$.
- $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \Omega_\otimes$ if there are signals $\{d_i(\cdot), y_i(\cdot)\}_{i\in\mathcal{V}}$ such that $(d_i, y_i) \in \Omega_i$ holds for $i \in \mathcal{V}$, and the input- and output-consistency constraints (2) and (3) hold.

It can be shown that this composition of RD contracts is itself an RD contract, and in particular, the input signal y^{ext} is a free variable in $\mathcal{D}_\otimes$. We next prove that the universal property of composition is satisfied:

Theorem 4.1. *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a DAG with output set $\mathcal{V}^{\mathrm{out}} \subseteq \mathcal{V}$ and consider component-level RD contracts $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ satisfying Assumption 3.1. If $\{\Pi_i\}_{i\in\mathcal{V}}$ are systems such that $\Pi_i \vDash \mathcal{C}_i$ holds for any $i \in \mathcal{V}$, then $\bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$.*

Proof. We must show that if $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \mathcal{D}_\otimes$ and $y^{\mathrm{ext}} \in \bigotimes_{i\in\mathcal{V}} \Pi_i(d^{\mathrm{ext}})$ both hold, then $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \Omega_\otimes$. As the graph $\mathcal{G}$ is a DAG, we can find a topological ordering $\sigma : \mathcal{I}_{1,N} \to \mathcal{V}$ of $\mathcal{G}$ satisfying Definition 4.1. By Definition 2.3, there exist signals $\{d_i\}_{i\in\mathcal{V}}$ and $\{y_i\}_{i\in\mathcal{V}}$ such that $y_i \in \Pi_i(d_i)$. We prove that $(d_i, y_i) \in \Omega_i$ holds for $i \in \mathcal{V}$, implying that $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \Omega_\otimes$. We do so by writing $i = \sigma(q)$ for $q \in \mathcal{I}_{1,N}$ and using induction on q.

We first consider the basis $i = \sigma(1)$. By Lemma 4.1, $\mathrm{BR}(i) = \emptyset$. Thus, by the definition of the matrices F_{ij}, we have that $d_i = E_i d^{\mathrm{ext}}$, and the assumption that $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \mathcal{D}_\otimes$ together with Definition 4.2 imply that $(d_i, y_i) \in \mathcal{D}_i$. Hence, $(d_i, y_i) \in \Omega_i$ as $y_i \in \Pi_i(d_i)$ and $\Pi_i \vDash \mathcal{C}_i$. For the induction step, we write $i = \sigma(q)$ and assume $(d_j, y_j) \in \Omega_j$ holds for all $j = \sigma(p)$ for $p \in \mathcal{I}_{1,q-1}$. In particular, $(d_j, y_j) \in \Omega_j$ holds for any $j \in \mathrm{BR}(i)$ by Lemma 4.1. As $(d^{\mathrm{ext}}, y^{\mathrm{ext}}) \in \mathcal{D}_\otimes$, we conclude that $(d_i, y_i) \in \mathcal{D}_i$ by Definition 4.2. We therefore see that $(d_i, y_i) \in \Omega_i$ using $y_i \in \Pi_i(d_i)$, as $\Pi_i \vDash \mathcal{C}_i$. □

Remark 4.1. Definition 4.2 considers the assumptions of the composite contract as pairs $(d^{\mathrm{ext}}, y^{\mathrm{ext}})$ satisfying a certain implication. If no such pair exist, so that $\mathcal{D}_\otimes = \emptyset$, one might say that the contracts are *incompatible*, using the terminology of Benveniste et al. (2018). One example of such case is when a certain contract guarantees that some signal v has $|v(k)| \leq 2$, but another contract assumes that $|v(k)| \leq 1$, i.e., the guarantee of the former is not strict enough for the latter.

4.3. Vertical contracts

We now consider Problem 3.2 for networks without feedback. We build LP-based tools for verifying vertical contracts of the form $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$ for LTI RD contracts. Let $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ for $i \in \mathcal{V}$ be component-level LTI RD contracts, and let $\mathcal{C}_{\mathrm{tot}} = (\mathcal{D}_{\mathrm{tot}}, \Omega_{\mathrm{tot}})$ be a given LTI RD contract on the composite system. Assume $\mathcal{C}_i, \mathcal{C}_{\mathrm{tot}}$ have assumption depth $m_i^A, m_{\mathrm{tot}}^A$ and guarantee depth $m_i^G, m_{\mathrm{tot}}^G$, respectively. Denoting the associated piecewise-linear functions as $\alpha_i, \alpha_{\mathrm{tot}}, \gamma_i, \gamma_{\mathrm{tot}}$, we write:

$$
\begin{aligned}
\mathcal{D}_i &= \left\{ \begin{pmatrix} d_i \\ y_i \end{pmatrix} : \alpha_i\begin{pmatrix} d_i(k-m_i^A:k) \\ y_i(k-m_i^A:k-1) \end{pmatrix} \leq 0,\ \forall k \geq m_i^A \right\}, \\
\Omega_i &= \left\{ \begin{pmatrix} d_i \\ y_i \end{pmatrix} : \gamma_i\begin{pmatrix} d_i(k-m_i^G:k) \\ y_i(k-m_i^G:k) \end{pmatrix} \leq 0,\ \forall k \geq m_i^G \right\}, \qquad (10) \\
\mathcal{D}_{\mathrm{tot}} &= \left\{ \begin{pmatrix} d^{\mathrm{ext}} \\ y^{\mathrm{ext}} \end{pmatrix} : \alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(k-m_{\mathrm{tot}}^A:k) \\ y^{\mathrm{ext}}(k-m_{\mathrm{tot}}^A:k-1) \end{pmatrix} \leq 0,\ \forall k \geq m_{\mathrm{tot}}^A \right\}, \\
\Omega_{\mathrm{tot}} &= \left\{ \begin{pmatrix} d^{\mathrm{ext}} \\ y^{\mathrm{ext}} \end{pmatrix} : \gamma_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(k-m_{\mathrm{tot}}^G:k) \\ y^{\mathrm{ext}}(k-m_{\mathrm{tot}}^G:k) \end{pmatrix} \leq 0,\ \forall k \geq m_{\mathrm{tot}}^G \right\}.
\end{aligned}
$$

We denote $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i = (\mathcal{D}_\otimes, \Omega_\otimes)$. Our goal is to find a computationally-viable method for verifying that $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$ holds. The vertical contract is equivalent to the set inclusions $\mathcal{D}_\otimes \supseteq \mathcal{D}_{\mathrm{tot}}$ and $\Omega_\otimes \cap \mathcal{D}_{\mathrm{tot}} \subseteq \Omega_{\mathrm{tot}} \cap \mathcal{D}_{\mathrm{tot}}$, which can be rewritten as the following implications for the signals $d^{\mathrm{ext}}, y^{\mathrm{ext}}, \{d_j, y_j\}_{j\in\mathcal{V}}$ satisfying the consistency relations (2), (3):

- Given any $i \in \mathcal{V}$, if $(d^{\mathrm{ext}}(\cdot), y^{\mathrm{ext}}(\cdot)) \in \mathcal{D}_{\mathrm{tot}}$ and $(d_j(\cdot), y_j(\cdot)) \in \Omega_j$ hold for all $j \in \mathrm{BR}(i)$, then $(d_i, y_i) \in \mathcal{D}_i$.
- If $(d^{\mathrm{ext}}(\cdot), y^{\mathrm{ext}}(\cdot)) \in \mathcal{D}_{\mathrm{tot}}$ and $(d_i(\cdot), y_i(\cdot)) \in \Omega_i$ hold for all $i \in \mathcal{V}$, then $(d^{\mathrm{ext}}(\cdot), y^{\mathrm{ext}}(\cdot)) \in \Omega_{\mathrm{tot}}$.

Using extendibility (Definition 3.5), we reformulate these as implications on signals defined on a bounded time interval.

Theorem 4.2. *Consider a composite system with DAG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and output set $\mathcal{V}^{\mathrm{out}}$. Let $\{\mathcal{C}_i\}_{i\in\mathcal{V}}, \mathcal{C}_{\mathrm{tot}}$ be RD contracts as in (10),* *where Assumption 3.1 holds. Under mild extendibility assumptions,*[1] $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$ *holds if and only if the following implications hold for any signals* $d_i, y_i, d^{\mathrm{ext}}, y^{\mathrm{ext}}$ *satisfying the consistency relations (2), (3):*

(i) For any $i \in \mathcal{V}$*, if*

$$\alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^A_i}$$

$$\gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^G_j,m^A_i},\ \forall j \in \mathrm{BR}(i),$$

all hold, then $\alpha_i\begin{pmatrix} d_i(0:m^A_i) \\ y_i(0:m^A_i-1)\end{pmatrix} \le 0.$

(ii) If

$$\alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^G_{\mathrm{tot}}},$$

$$\gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^G_j,m^G_{\mathrm{tot}}},\ \forall j \in \mathcal{V},$$

all hold, then $\gamma_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(0:m^G_{\mathrm{tot}}) \\ y^{\mathrm{ext}}(0:m^G_{\mathrm{tot}})\end{pmatrix} \le 0.$

The proof is found in Appendix. Colloquially, condition (i) states that the assumptions of the composition $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i$ assumes less than $\mathcal{C}_{\mathrm{tot}}$, and condition (ii) states that the composition guarantees more than $\mathcal{C}_{\mathrm{tot}}$. We emphasise that Theorem 4.2 does not require the explicit construction of the composite contract $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i$ to guarantee the vertical contract $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$.

The theorem allows one to verify the vertical contract for a network without feedback $\mathcal{G}$ by verifying $|\mathcal{V}|+1$ implications. Importantly, even though the definition of a vertical contract is in terms of signals, the implications involve only a finite number of variables. Specifically, each of them can be cast as an LP in the variables $d^{\mathrm{ext}}(0:m), y^{\mathrm{ext}}(0:m), \{d_j(0:m), y_j(0:m)\}_{j\in\mathcal{V}}$ where m is the maximum over all depths.

Namely, for $i \in \mathcal{V}$, let ϱ_i be the solution to

$$\begin{aligned} \max\ & \alpha_i\begin{pmatrix} d_i(0:m^A_i) \\ y_i(0:m^A_i-1)\end{pmatrix} && && (11a)\\ \text{s.t. } & \alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^A_i}, \\ & \gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^G_j,m^A_i}, \forall j \in \mathrm{BR}(i), \\ & \text{(2) at time } \ell \text{ and node } j, && \forall \ell = \mathcal{I}_{0,m^A_i}, \forall j \in \mathrm{BR}_+(i), \\ & \text{(3) at time } \ell, && \forall \ell \in \mathcal{I}_{0,m^A_i}, \end{aligned}$$

and let ϱ_Ω be equal to

$$\begin{aligned} \max\ & \gamma_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(0:m^G_{\mathrm{tot}}) \\ y^{\mathrm{ext}}(0:m^G_{\mathrm{tot}})\end{pmatrix} && && (11b)\\ \text{s.t. } & \alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^G_{\mathrm{tot}}}, \\ & \gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^G_j,m^G_{\mathrm{tot}}}, \forall j \in \mathcal{V}, \\ & \text{(2) at time } \ell \text{ and node } j, && \forall \ell \in \mathcal{I}_{0,m^G_{\mathrm{tot}}}, \forall j \in \mathcal{V}, \\ & \text{(3) at time } \ell, && \forall \ell \in \mathcal{I}_{0,m^G_{\mathrm{tot}}}. \end{aligned}$$

Algorithm 1 Verifying Vertical Contracts for Networks without Feedback

Input: A DAG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, output set $\mathcal{V}^{\mathrm{out}} \subseteq \mathcal{V}$, component-level RD LTI contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$, and an RD LTI contract $\mathcal{C}_{\mathrm{tot}}$ on the composite system of the form (10) such that Assumption 3.1 holds.
Output: A Boolean variable $\mathbf{b}_{\otimes,\preccurlyeq}$.

```
Compute $\{\varrho_i\}_{i\in\mathcal{V}}, \varrho_\Omega$ by solving the LPs (11).
if $\{\varrho_i\}_{i\in\mathcal{V}}, \varrho_\Omega$ are all non-positive then
    Return $\mathbf{b}_{\otimes,\preccurlyeq}$ = true.
else
    Return $\mathbf{b}_{\otimes,\preccurlyeq}$ = false.
```

These programs give rise to an algorithm determining whether $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$, see Algorithm 1. It is an LP-based verification method, solving a total of $|\mathcal{V}|+1$ LPs. They can be solved using standard optimisation software. The correctness of the algorithm is stated in the following corollary:

Corollary 4.1. *Under the assumptions of Theorem 4.2, Algorithm 1 is always correct, i.e.,* $\bigotimes_{i\in\mathcal{V}}\mathcal{C}_i \preccurlyeq \mathcal{C}_{\mathrm{tot}}$ *holds if and only if the algorithm returns* $\mathbf{b}_{\otimes,\preccurlyeq}$ *= true.*

Proof. Follows from Theorem 4.2 and the following principle: Given functions $f, g : \mathcal{X} \to \mathbb{R}$ defined on an arbitrary space, the implication $f(x) \le 0 \implies g(x) \le 0$ holds if and only if $\max\{g(x) : x \text{ s.t. } f(x) \le 0\} \le 0$. □

Example 4.1. We demonstrate the LP framework for a cascade of RD contracts, for which the assumptions do not depend on the output variables. The network is given by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $\mathcal{V} = \{1, 2\}$ and $\mathcal{E} = \{1 \to 2\}$, where node 1 corresponds to an open-loop controller and node 2 corresponds to the system to be controlled. Thus, $\mathrm{BR}(1) = \emptyset$ and $\mathrm{BR}(2) = \{1\}$. Moreover, $\mathcal{V}^{\mathrm{out}} = \{2\}$, so $d^{\mathrm{ext}} = d_1$, $d_2 = y_1$ and $y^{\mathrm{ext}} = y_2$. We verify that $\mathcal{C}_1 \otimes \mathcal{C}_2 \preccurlyeq \mathcal{C}_{\mathrm{tot}}$ by checking three implications:

- The assumptions of $\mathcal{C}_{\mathrm{tot}}$ imply the assumptions of $\mathcal{C}_1$. This is equivalent to $\varrho_1 \le 0$, where ϱ_1 is equal to

$$\begin{aligned} \max_{d_1}\ & \alpha_1\big(d_1(0:m^A_1)\big) \\ \text{s.t. } & \alpha_{\mathrm{tot}}\big(d_1(\ell - m^A_{\mathrm{tot}} : \ell)\big) \le 0, && \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^A_1}. \end{aligned}$$

- The assumptions of $\mathcal{C}_{\mathrm{tot}}$, plus the guarantees of $\mathcal{C}_1$, imply the assumptions of $\mathcal{C}_2$. This is equivalent to $\varrho_2 \le 0$, where ϱ_2 is equal to

$$\begin{aligned} \max_{d_i}\ & \alpha_2\big(d_2(0:m^A_2)\big) \\ \text{s.t. } & \alpha_{\mathrm{tot}}\big(d_1(\ell - m^A_{\mathrm{tot}} : \ell)\big) \le 0, && \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^A_2} \\ & \gamma_1\begin{pmatrix} d_1(\ell-m^G_1:\ell) \\ d_2(\ell-m^G_1:\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^G_1,m^A_2}. \end{aligned}$$

- The assumption of $\mathcal{C}_{\mathrm{tot}}$, plus guarantees of $\mathcal{C}_1$ and $\mathcal{C}_2$, imply the guarantees of $\mathcal{C}_{\mathrm{tot}}$. This is equivalent to $\varrho_{\mathrm{tot}} \le 0$, where ϱ_{tot} is equal to

$$\begin{aligned} \max_{d_i,y_2}\ & \gamma_{\mathrm{tot}}\begin{pmatrix} d_1(0:m^G_{\mathrm{tot}}) \\ y_2(0:m^G_{\mathrm{tot}})\end{pmatrix} \\ \text{s.t. } & \alpha_{\mathrm{tot}}\big(d_1(\ell - m^A_{\mathrm{tot}} : \ell)\big) \le 0, && \forall \ell \in \mathcal{I}_{m^A_{\mathrm{tot}},m^A_{\mathrm{tot}}} \\ & \gamma_1\begin{pmatrix} d_1(\ell-m^G_1:\ell) \\ d_2(\ell-m^G_1:\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^G_1,m^A_{\mathrm{tot}}} \\ & \gamma_2\begin{pmatrix} d_2(\ell-m^G_2:\ell) \\ y_2(\ell-m^G_2:\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m^G_2,m^A_{\mathrm{tot}}}. \end{aligned}$$

[1] The functions

$$\max\{\max_{\ell\le m^A_i}\alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix}, \max_{j\in\mathrm{BR}(i)} \max_{\ell\le m^A_i} \gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix}\}$$

for $i \in \mathcal{V}$, are extendable, as is the function

$$\max\{ \max_{\ell\le m^G_{\mathrm{tot}}}\alpha_{\mathrm{tot}}\begin{pmatrix} d^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell) \\ y^{\mathrm{ext}}(\ell-m^A_{\mathrm{tot}}:\ell-1)\end{pmatrix}, \max_{i\in\mathcal{V}} \max_{\ell\le m^G_{\mathrm{tot}}}\gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix}\}.$$

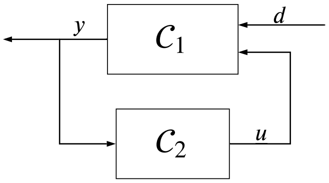


Fig. 2. Feedback Composition of two contracts.

Indeed, the first and second implications above are the implication (i) in Theorem 4.2 for the vertices $1, 2 \in \mathcal{V}$ respectively, and the third implication above is the implication (ii) from Theorem 4.2.

Remark 4.2. The LP problems above depend on the depths of the RD LTI contracts. One could consider a contract with multiple assumptions or guarantees defined by different depths. In that case, the problems (11) should be amended as follows: Whenever we use the contract for defining constraints, we add different constraints for each assumption or guarantee, having different relevant times ℓ. Whenever we use the contract for defining the cost function, replace it with the maximum of all corresponding piecewise-linear functions.

Together, Theorems 4.1 and 4.2 enable the efficient modular verification of composite systems without feedback. Namely, if each component Π_i satisfies its local contract $\mathcal{C}_i$, and the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds, then the composite system $\bigotimes_{i\in\mathcal{V}} \Pi_i$ is guaranteed to satisfy $\mathcal{C}_{\text{tot}}$. Here, verification of the vertical contract can be done through Algorithm 1, whereas (LP-based) algorithms for verifying contract satisfaction are given in Sharf, Besselink, and Johansson (2021).

5. Networks with feedback

The previous section focused on networks without feedback. In this section, we generalise our results to general networks with feedback, e.g., the connection of a feedback controller to a system.

5.1. Causality and algebraic loops

Before delving into the definition of the composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$, we must understand its basic limitations. We demonstrate them in an example.

Example 5.1. Consider the network in Fig. 2, with RD contracts $\mathcal{C}_1 = (\mathcal{D}_1, \Omega_1)$ and $\mathcal{C}_2 = (\mathcal{D}_2, \Omega_2)$ given as

$$\mathcal{D}_1 = \{(d(\cdot), u(\cdot), y(\cdot)) : |d(k)| \le 1\ |u(k)| \le 1, \forall k\},$$
$$\Omega_1 = \{(d(\cdot), u(\cdot), y(\cdot)) : y(k) = (d(k) + u(k)) + 1, \forall k\},$$
$$\mathcal{D}_2 = \{(y(\cdot), u(\cdot)) : |y(k)| \le 1, \forall k\},$$
$$\Omega_2 = \{(y(\cdot), u(\cdot)) : u(k) = y(k) + 1, \forall k\}.$$

If the composition $\mathcal{C}_1 \otimes \mathcal{C}_2$ could be defined, and (d, y) is an input–output pair satisfying its guarantees, we should have $(d, u, y) \in \Omega_1, (y, u) \in \Omega_2$ for some signal u, i.e., for any $k \in \mathbb{N}$, we would have $y(k) = d(k) + u(k) + 1$ and $u(k) = y(k) + 1$. The only solution to these equations is the constant signal $d(k) = -2$, which is not compatible with $\mathcal{D}_1$. Hence, $\mathcal{C}_1 \otimes \mathcal{C}_2$ cannot be defined meaningfully in this case.

The inconsistency in Example 5.1 arises from contradicting specifications. More precisely, the guarantees of $\mathcal{C}_1$ constrain $y(k)$ in terms of $u(k)$, and the guarantees of $\mathcal{C}_2$ constrain $u(k)$ in terms of $y(k)$, resulting in an algebraic loop creating ill-posed constraints. This situation can be avoided if we demand that $\mathcal{C}_1$ would constrain $y(k)$ using only $d(0:k), y(0:k-1), u(0:k-1)$ and not using $u(k)$, which can be understood as a strict causality-type demand on the contract $\mathcal{C}_1$ with respect to the control input u. This motivates the following definition:

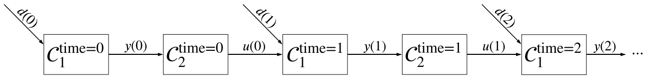

$d(0)$
$d(1)$
$d(2)$
$\mathcal{C}_1^{\text{time}=0}$
$y(0)$
$\mathcal{C}_2^{\text{time}=0}$
$u(0)$
$\mathcal{C}_1^{\text{time}=1}$
$y(1)$
$\mathcal{C}_2^{\text{time}=1}$
$u(1)$
$\mathcal{C}_1^{\text{time}=2}$
$y(2)$
...

Fig. 3. An infinite cascade composition, equivalent to the feedback connection in Fig. 2 if $\mathcal{C}_1$ is RD and $\mathcal{C}_2$ is SRD(u).

Definition 5.1. Let $\mathcal{C} = (\mathcal{D}, \Omega)$ be an RD contract of the form (4), (5) with input $d \in \mathcal{S}^{n_d}$ and output $y \in \mathcal{S}^{n_y}$. For a subsignal d_{sub} of d, $\mathcal{C}$ is *strictly* recursively defined with respect to d_{sub}, denoted SRD(d_{sub}), if for any time k, the condition defining $\mathcal{C}$'s guarantees at time k, $\begin{bmatrix} d(k) \\ y(k) \end{bmatrix} \in G_k\begin{pmatrix} d(0:k-1) \\ y(0:k-1) \end{pmatrix}$, is independent of $d_{\text{sub}}(k)$.[2]

As explained above, the ill-posedness issue in Example 5.1 could not occur if $\mathcal{C}_1$ was SRD(u) and $\mathcal{C}_2$ was RD. Indeed, there is a clear "order of constraining" guaranteeing well-posedness: in the sequence $y(0), u(0), y(1), u(1), \ldots$, each element is constrained using the preceding elements, but not using the following elements. This "order of constraining" is illustrated in Fig. 3, replacing the feedback composition by an infinite cascade composition. This approach can be generalised to more intricate networks. Suppose there exists an "order of constraining" given by $y_{i_1}(0), \ldots, y_{i_N}(0), y_{i_1}(1), \ldots, y_{i_N}(1), y_{i_1}(2), \ldots$. Then $y_{i_1}(k)$ is constrained by $\{y_{i_q}(0:k-1)\}_{q=1}^N$, so $\mathcal{C}_{i_1}$ must be SRD with respect to $y_{i_2}, \ldots, y_{i_N}$. Similarly, $y_{i_2}(k)$ is only constrained by $\{y_{i_q}(0:k-1)\}_{q=1}^N$ and $y_{i_1}(k)$, implying that $\mathcal{C}_2$ is SRD with respect to $y_{i_3}, \ldots, y_{i_N}$.

In Section 5.2, we will define the contract composition $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$ for RD contracts while assuming an "order of constraining" exists. The remainder of this section is devoted to formalise the idea of "order of constraining". We start by translating strict causality to the language of graph theory:

Definition 5.2. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and component-level RD contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$, we say an edge $e = i \to j \in \mathcal{E}$ is *strictly causal* if $\mathcal{C}_j$ is SRD(y_i). We let $\mathcal{E}_{\text{sc}}$ be the set of strictly causal edges, and $\mathcal{E}_{\text{nsc}} = \mathcal{E} \setminus \mathcal{E}_{\text{sc}}$ be the set of non-strictly causal edges.

In other words, the edge $i \to j$ is non-strictly causal if the guarantee on $y_j(k)$ can depend on $y_i(k)$. Mimicking the argument for networks without feedback, $y_i(k)$ is constrained by $y_j(0:k-1)$ if j is backward-reachable from i in $\mathcal{G}$, i.e., if $j \in \text{BR}(i)$. Similarly, $y_i(k)$ is constrained by $y_j(k)$ if j is backward reachable from i while only using non-strictly causal edges (i.e., in $\mathcal{G}_{\text{nsc}}$). For convenience, we denote the backward-reachable set from i in $\mathcal{G}_{\text{nsc}}$ as $\text{BR}_{\text{nsc}}(i)$. In particular, (the lack of) contract-theoretic algebraic loops corresponds to (the lack of) cycles in $\mathcal{G}_{\text{nsc}}$, leading to the following assumption:

Assumption 5.1. Any cycle in the graph $\mathcal{G}$ contains at least one strictly causal edge, i.e., $\mathcal{G}_{\text{nsc}}$ is a DAG.

[2] If d'_{sub} is the complementary subsignal to d_{sub}, this is equivalent to the existence of set-valued functions $\tilde{G}_k$ such that

$$\begin{bmatrix} d(k) \\ y(k) \end{bmatrix} \in G_k\begin{pmatrix} d(0:k-1) \\ y(0:k-1) \end{pmatrix} \iff \begin{bmatrix} d'_{\text{sub}}(k) \\ y(k) \end{bmatrix} \in \tilde{G}_k\begin{pmatrix} d(0:k-1) \\ y(0:k-1) \end{pmatrix}.$$

5.2. Contract composition

From now on, we fix a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N nodes, a set of output nodes $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$, and component-level RD contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$ satisfying Assumptions 3.1 and 5.1. For each $i \in \mathcal{V}$, we write the contract $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ as:

$$\begin{aligned} \mathcal{D}_i &= \left\{ \begin{pmatrix} d_i(\cdot) \\ y_i(\cdot) \end{pmatrix} : d_i(k) \in A_{k,i}\begin{pmatrix} d_i(0:k-1) \\ y_i(0:k-1) \end{pmatrix}, \forall k \right\}, \\ \Omega_i &= \left\{ \begin{pmatrix} d_i(\cdot) \\ y_i(\cdot) \end{pmatrix} : \begin{bmatrix} d_i(k) \\ y_i(k) \end{bmatrix} \in G_{k,i}\begin{pmatrix} d_i(0:k-1) \\ y_i(0:k-1) \end{pmatrix}, \forall k \right\} \end{aligned} \tag{12}$$

for set-valued maps $A_{k,i}, G_{k,i}$, where the interconnection is defined by (2) and (3). Drawing inspiration from the infinite cascade composition seen in Fig. 3 and postulates (A) and (B), we define the composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$ as follows:

Definition 5.3. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph, $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$ a set of output nodes, and $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ be component-level RD contracts, so that Assumptions 3.1 and 5.1 both hold. The composite contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i = (\mathcal{D}_\otimes, \Omega_\otimes)$, having input $d^{\text{ext}}(\cdot)$ and output $y^{\text{ext}}(\cdot)$, is defined as follows:

- $(d^{\text{ext}}, y^{\text{ext}}) \in \mathcal{D}_\otimes$ if for any signals $\{d_j, y_j\}_{j\in\mathcal{V}}$ satisfying the consistency constraints (2) and (3), the following implication holds for any time $k \in \mathbb{N}$ and any $i \in \mathcal{V}$: If
$$\begin{aligned} &\begin{bmatrix} d_j(\ell) \\ y_j(\ell) \end{bmatrix} \in G_{\ell,j}\begin{pmatrix} d_j(0:\ell-1) \\ y_j(0:\ell-1) \end{pmatrix}, && \forall \ell \in \mathcal{I}_{0,k}, \\ & && \forall j \in \text{BR}_{\text{nsc}}(i) \\ &\begin{bmatrix} d_j(\ell) \\ y_j(\ell) \end{bmatrix} \in G_{\ell,j}\begin{pmatrix} d_j(0:\ell-1) \\ y_j(0:\ell-1) \end{pmatrix}, && \forall \ell \in \mathcal{I}_{0,k-1}, \\ & && \forall j \in \text{BR}_+(i)\backslash\text{BR}_{\text{nsc}}(i) \end{aligned}$$
all hold, then $d_i(k) \in A_{k,i}\begin{pmatrix} d_i(0:k-1) \\ y_i(0:k-1) \end{pmatrix}$.
- $(d^{\text{ext}}, y^{\text{ext}}) \in \Omega_\otimes$ if there exist signals $\{d_j, y_j\}_{j\in\mathcal{V}}$ such that the consistency relations (2) and (3), and $(d_j, y_j) \in \Omega_j$ hold for all $j \in \mathcal{V}$.

Essentially, Definition 5.3 mimics Definition 4.2 by replacing the network with feedback with an infinite network without feedback. This is done by replacing the contracts $\mathcal{C}_i$, with constraints defined over the entire time horizons, by "timewise" contracts $\mathcal{C}_i^{\text{time}=k}$ constraining signals at time k. This intuition is key in proving that the counterpart to Theorem 4.1 holds in the feedback case.

Theorem 5.1. *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with output set $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$ and consider component-level RD contracts $\mathcal{C}_i$ as in (12), satisfying Assumptions 3.1 and 5.1. If $\{\Pi_i\}_{i\in\mathcal{V}}$ are causal systems such that $\Pi_i \vDash \mathcal{C}_i$ holds for any $i \in \mathcal{V}$, then $\bigotimes_{i\in\mathcal{V}} \Pi_i \vDash \bigotimes_{i\in\mathcal{V}} \mathcal{C}_i$.*

To prove this result, we first consider the following lemma, linking the timewise contracts $\mathcal{C}_i^{\text{time}=k}$ and $\mathcal{C}_i$:

Lemma 5.1. *Let $\mathcal{C} = (\mathcal{D}, \Omega)$ be an RD contract of the form (4), (5), where $\mathcal{D}$ is extendable, and let Π be a causal system satisfying $\mathcal{C}$. Let $\hat{d}(\cdot) \in \mathcal{S}^{n_d}$ be any input signal, and let $\hat{y} \in \Pi(\hat{d})$. For any time n, if $\hat{d}(k) \in A_k\begin{pmatrix} \hat{d}(0:k-1) \\ \hat{y}(0:k-1) \end{pmatrix}$ holds for $k \in \mathcal{I}_{0,n}$, then $\begin{bmatrix} \hat{d}(n) \\ \hat{y}(n) \end{bmatrix} \in G_n\begin{pmatrix} \hat{d}(0:n-1) \\ \hat{y}(0:n-1) \end{pmatrix}$.*

In other words, satisfying the RD contract $\mathcal{C}$ is equivalent to satisfying all timewise contracts $\mathcal{C}^{\text{time}=k}$.

Proof. We will construct signals d, y such that $d(0:n) = \hat{d}(0:n)$ and $y(0:n) = \hat{y}(0:n)$, $(d, y) \in \mathcal{D}$, and $y \in \Pi(d)$. We thus conclude from $\Pi \vDash \mathcal{C}$ that $(d, y) \in \Omega$, which yields the result by writing the guarantees at time n and using $d(0:n) = \hat{d}(0:n)$ and $y(0:n) = \hat{y}(0:n)$.

We now construct d and y. Following Remark 2.1, we denote the timewise set-valued maps $d(0:k) \mapsto y(k)$ as Π_k. We define $d(k)$ and $y(k)$ by induction on k. We first define $d(0:n) = \hat{d}(0:n)$ and $y(0:n) = \hat{y}(0:n)$, so that both $y(k) \in \Pi_k(d(0:k))$ and $d(k) \in A_k\begin{pmatrix} d(0:k-1) \\ y(0:k-1) \end{pmatrix}$ hold for $k \in \mathcal{I}_{0,n}$. Now, assume $d(0:k), y(0:k)$ have been defined so that both $y(j) \in \Pi_j(d(0:j))$ and $d(j) \in A_j\begin{pmatrix} d(0:j-1) \\ y(0:j-1) \end{pmatrix}$ hold for $j \in \mathcal{I}_{0,k}$. By extendibility, the set $A_{k+1}\begin{pmatrix} d(0:k) \\ y(0:k) \end{pmatrix}$ is non-empty, and we choose $d(k+1)$ as one of its elements, as well as some $y(k+1) \in \Pi_{k+1}(d(0:k+1))$. By construction, we have $(d, y) \in \mathcal{D}$, $y \in \Pi(d)$, $d(0:n) = \hat{d}(0:n)$ and $y(0:n) = \hat{y}(0:n)$. □

Given Lemma 5.1, the proof of Theorem 5.1 is nearly identical to the proof of Theorem 4.1, and will be omitted due to lack of space. The only difference is that we are using the timewise contracts $\mathcal{C}_i^{\text{time}=k}$ instead of the RD contracts $\mathcal{C}_i$, and that gap is bridged by Lemma 5.1.

5.3. Vertical contracts

We shift our attention to Problem 3.2. As before, we build LP-based tools for verifying vertical contracts $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ for LTI RD contracts. We fix component-level LTI RD contracts $\mathcal{C}_i = (\mathcal{D}_i, \Omega_i)$ for $i \in \mathcal{V}$ and an LTI RD contract $\mathcal{C}_{\text{tot}} = (\mathcal{D}_{\text{tot}}, \Omega_{\text{tot}})$ on the composite system, such that Assumption 3.1 holds. We let $\alpha_i, \alpha_{\text{tot}}, \gamma_i, \gamma_{\text{tot}}$ be the corresponding piecewise-linear functions so that (10) holds, and we denote $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i = (\mathcal{D}_\otimes, \Omega_\otimes)$. As before, the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ is equivalent to the set inclusions $\mathcal{D}_\otimes \supseteq \mathcal{D}_{\text{tot}}$ and $\Omega_\otimes \cap \mathcal{D}_{\text{tot}} \subseteq \Omega_{\text{tot}} \cap \mathcal{D}_{\text{tot}}$.

Theorem 5.2. *Consider a composite system with graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and output set $\mathcal{V}^{\text{out}}$. Let $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$, $\mathcal{C}_{\text{tot}}$ be LTI RD contracts as in (10), where Assumptions 3.1 and 5.1 hold. Denote $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i = (\mathcal{D}_\otimes, \Omega_\otimes)$. Under mild extendibility assumptions*[1]*, the following claims hold:*

- *$\mathcal{D}_\otimes \subseteq \mathcal{D}_{\text{tot}}$ holds if and only if the following implication holds for all $i \in \mathcal{V}$. For any signals $d_i, y_i, d^{\text{ext}}, y^{\text{ext}}$, defined at times $\{0, 1, \ldots, m_i^A\}$, if the consistency relations (2) and (3) hold, and*
$$\begin{aligned} &\alpha_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(\ell-m_{\text{tot}}^A:\ell) \\ y^{\text{ext}}(\ell-m_{\text{tot}}^A:\ell-1) \end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_i^A} \\ &\gamma_j\begin{pmatrix} d_j(\ell-m_j^G:\ell) \\ y_j(\ell-m_j^G:\ell) \end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_i^A}, \\ & && \forall j \in \text{BR}_{\text{nsc}}(i), \\ &\gamma_j\begin{pmatrix} d_j(\ell-m_j^G:\ell) \\ y_j(\ell-m_j^G:\ell) \end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_i^A-1}, \\ & && \forall j \in \text{BR}(i) \setminus \text{BR}_{\text{nsc}}(i), \end{aligned}$$
all hold, then $\alpha_i\begin{pmatrix} d_i(0:m_i^A) \\ y_i(0:m_i^A-1) \end{pmatrix} \le 0$.
- *$\Omega_\otimes \cap \mathcal{D}_{\text{tot}} \subseteq \Omega_{\text{tot}} \cap \mathcal{D}_{\text{tot}}$ holds if and only if the following implication holds. For any signals $d_i, y_i d^{\text{ext}}, y^{\text{ext}}$ defined at times $\{0, 1, \ldots, m_{\text{tot}}^G\}$, if the consistency relations (2) and (3) hold, and*
$$\begin{aligned} &\alpha_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(\ell-m_{\text{tot}}^A:\ell) \\ y^{\text{ext}}(\ell-m_{\text{tot}}^A:\ell-1) \end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_{\text{tot}}^G} \\ &\gamma_j\begin{pmatrix} d_j(\ell-m_j^G:\ell) \\ y_j(\ell-m_j^G:\ell) \end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_{\text{tot}}^G}, \ \forall j \in \mathcal{V} \end{aligned}$$
all hold, then $\gamma_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(0:m_{\text{tot}}^G) \\ y^{\text{ext}}(0:m_{\text{tot}}^G) \end{pmatrix} \le 0$.

In particular, the vertical contract $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds if and only if the first implication holds for all $i \in \mathcal{V}$, and the second implication holds.

The proof of Theorem 5.2 is nearly identical to that of Theorem 4.2 and is omitted due to space limitations. As in the case of networks without feedback, Theorem 5.2 replaces the set inclusion on the signal level defining the vertical contract by $|\mathcal{V}|+1$ implications between linear inequalities. As before, these can be cast as LPs.

For $i \in \mathcal{V}$, let ϱ_i denote the solution to

$$\begin{aligned}
\max\ & \alpha_i\begin{pmatrix} d_i(0{:}m_i^A) \\ y_i(0{:}m_i^A-1)\end{pmatrix} && && (13a)\\
\text{s.t.}\ & \alpha_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(\ell-m_{\text{tot}}^A{:}\ell) \\ y^{\text{ext}}(\ell-m_{\text{tot}}^A{:}\ell-1)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_i^A} \\
& \gamma_j\begin{pmatrix} d_j(\ell-m_j^G{:}\ell) \\ y_j(\ell-m_j^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_i^A} \\
& && \forall j \in \mathrm{BR}_{\text{nsc}}(i) \\
& \gamma_j\begin{pmatrix} d_j(\ell-m_j^G{:}\ell) \\ y_j(\ell-m_j^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_i^A-1} \\
& && \forall j \in \mathrm{BR}(i) \setminus \mathrm{BR}_{\text{nsc}}(i) \\
& (2) \text{ at time } \ell \text{ and node } j, && \forall \ell \in \mathcal{I}_{0,m_i^A}, \forall j \in \mathrm{BR}_{+}(i), \\
& (3) \text{ at time } \ell, && \forall \ell \in \mathcal{I}_{0,m_i^A}.
\end{aligned}$$

Next, ϱ_Ω is equal to

$$\begin{aligned}
\max\ & \gamma_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(0{:}m_{\text{tot}}^G{:}\ell) \\ y^{\text{ext}}(0{:}m_{\text{tot}}^G)\end{pmatrix} && && (13b)\\
\text{s.t.}\ & \alpha_{\text{tot}}\begin{pmatrix} d^{\text{ext}}(\ell-m_{\text{tot}}{:}\ell) \\ y^{\text{ext}}(\ell-m_{\text{tot}}{:}\ell-1)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_{\text{tot}}^G} \\
& \gamma_j\begin{pmatrix} d_j(\ell-m_j^G{:}\ell) \\ y_j(\ell-m_j^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_j^G, m_{\text{tot}}^G}, \forall j \in \mathcal{V}, \\
& (2) \text{ at time } \ell \text{ and node } j, && \forall \ell \in \mathcal{I}_{0,m_{\text{tot}}^G}, \forall j \in \mathcal{V}, \\
& (3) \text{ at time } \ell \text{ and node } j, && \forall \ell \in \mathcal{I}_{0,m_{\text{tot}}^G}.
\end{aligned}$$

They suggest an algorithm for determining whether $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ for general vertical contracts, see Algorithm 2. As Algorithm 1, it is an LP-based verification method, solving a total of $|\mathcal{V}|+1$ LPs, and the algorithm is correct:

Theorem 5.3. *Under the assumptions of Theorem 5.2, Algorithm 2 is always correct, i.e., $\bigotimes_{i\in\mathcal{V}} \mathcal{C}_i \preccurlyeq \mathcal{C}_{\text{tot}}$ holds if and only if the algorithm returns $\mathbf{b}_{\otimes,\preccurlyeq}$ = true.*

Proof. Similar to Corollary 4.1. □

Example 5.2. We elucidate the LP framework for general composite systems by demonstrating it on the feedback composition in Fig. 2. The network is given by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $\mathcal{V} = \{1, 2\}$, $\mathcal{E}_{\text{nsc}} = \{1 \to 2\}$, $\mathcal{E}_{\text{sc}} = \{2 \to 1\}$, $\mathcal{E} = \mathcal{E}_{\text{sc}} \cup \mathcal{E}_{\text{nsc}}$, and $\mathcal{V}^{\text{out}} = \{1\}$. Node 1 corresponds to the plant and node 2 to the feedback controller. In this case, $\mathrm{BR}_{\text{nsc}}(1) = \emptyset$, $\mathrm{BR}_{\text{nsc}}(2) = \{1\}$ and $\mathrm{BR}(1) = \mathrm{BR}(2) = \{1, 2\}$. Following Fig. 2, we denote the external input by d, the output of $\mathcal{C}_1$ as y, and the output of $\mathcal{C}_2$ by u. For simplicity, we consider SRD LTI contracts $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_{\text{tot}}$ for which the assumptions of $\mathcal{C}_2$ and $\mathcal{C}_{\text{tot}}$ do not depend on the previous outputs y, and the guarantee of $\mathcal{C}_1$ depends only on d and y. This assumption corresponds to a situation in which $\mathcal{C}_1$ defines an unregulated physical system, $\mathcal{C}_2$ defines a controller, and $\mathcal{C}_{\text{tot}}$ defines the closed-loop system. Thus, $d_1(\cdot) = \left(\begin{smallmatrix} d(\cdot) \\ u(\cdot)\end{smallmatrix}\right)$, $d_2(\cdot) = y(\cdot)$ and $y^{\text{ext}}(\cdot) = y(\cdot)$. In order to verify that $\mathcal{C}_1 \otimes \mathcal{C}_2 \preccurlyeq \mathcal{C}_{\text{tot}}$, we have to verify three implications:

- If the assumptions of $\mathcal{C}_{\text{tot}}$ hold until a certain time n, and the guarantees of both $\mathcal{C}_1, \mathcal{C}_2$ hold until time $n-1$, then the assumptions of $\mathcal{C}_1$ hold at time n. This is equivalent to $\varrho_1 \le 0$, where ϱ_1 is equal to

$$\begin{aligned}
\max\quad & \alpha_1\begin{pmatrix} d(0{:}m_1^A) \\ u(0{:}m_1^A-1)\end{pmatrix} \\
\text{s.t.}\quad & \alpha_{\text{tot}}(d(\ell - m_{\text{tot}}^A : \ell)) \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_1^A} \\
& \gamma_1\begin{pmatrix} d(\ell-m_1^G{:}\ell) \\ y(\ell-m_1^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_1^G, m_1^A-1} \\
& \gamma_2\begin{pmatrix} y(\ell-m_2^G{:}\ell) \\ u(\ell-m_2^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_2^G, m_1^A-1}.
\end{aligned}$$

- If the assumptions of $\mathcal{C}_{\text{tot}}$ and guarantees of $\mathcal{C}_1$ hold until some time n, and the guarantees of $\mathcal{C}_2$ hold until time $n-1$, then the assumptions of $\mathcal{C}_2$ hold at time n. This is equivalent to $\varrho_2 \le 0$, where ϱ_2 is

$$\begin{aligned}
\max\quad & \alpha_2(y(0 : m_2^A)) \\
\text{s.t.}\quad & \alpha_{\text{tot}}(d(\ell - m_{\text{tot}}^A : \ell)) \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_2^A} \\
& \gamma_1\begin{pmatrix} d(\ell-m_1^G{:}\ell) \\ y(\ell-m_1^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_1^G, m_2^A} \\
& \gamma_2\begin{pmatrix} y(\ell-m_2^G{:}\ell) \\ u(\ell-m_2^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_2^G, m_2^A-1}.
\end{aligned}$$

- The assumption of $\mathcal{C}_{\text{tot}}$, plus guarantees of $\mathcal{C}_1$ and $\mathcal{C}_2$, imply the guarantees of $\mathcal{C}_{\text{tot}}$. This is equivalent to $\varrho_{\text{tot}} \le 0$, where ϱ_{tot} is given by

$$\begin{aligned}
\max\quad & \gamma_{\text{tot}}\begin{pmatrix} d(0{:}m_{\text{tot}}^G) \\ y(0{:}m_{\text{tot}}^G)\end{pmatrix} \\
\text{s.t.}\quad & \alpha_{\text{tot}}(d(\ell - m_{\text{tot}}^A : \ell)) \le 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_{\text{tot}}^G} \\
& \gamma_1\begin{pmatrix} d(\ell-m_1^G{:}\ell) \\ y(\ell-m_1^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_1^G, m_{\text{tot}}^G} \\
& \gamma_2\begin{pmatrix} y(\ell-m_2^G{:}\ell) \\ u(\ell-m_2^G{:}\ell)\end{pmatrix} \le 0, && \forall \ell \in \mathcal{I}_{m_2^G, m_{\text{tot}}^G}.
\end{aligned}$$

Algorithm 2 Verifying Vertical Contracts for General Networks

Input: A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, output set $\mathcal{V}^{\text{out}} \subseteq \mathcal{V}$, component-level RD LTI contracts $\{\mathcal{C}_i\}_{i\in\mathcal{V}}$ and an RD LTI contract $\mathcal{C}_{\text{tot}}$ on the composite system of the form (10) such that Assumptions 3.1 and 5.1 hold.
Output: A Boolean variable $\mathbf{b}_{\otimes,\preccurlyeq}$.

```
Compute $\{\varrho_i\}_{i\in\mathcal{V}}, \varrho_\Omega$ by solving the LPs (13).
if $\{\varrho_i\}_{i\in\mathcal{V}}, \varrho_\Omega$ are all non-positive then
    Return true.
else
    Return false.
```

6. Numerical example

In this section, we apply the presented contract-based framework to autonomous vehicles in an M-vehicle platooning-like scenario. We first define the scenario and the specifications in the form of a contract. We then use the presented framework to refine the contract on the integrated M-vehicle system to a collection of contracts on the physical and control subsystems of each of the vehicles. Different values of M will be considered to demonstrate the scalability of the approach. Lastly, we demonstrate the modularity achieved by these processes by presenting options for realising the controller subsystem satisfying the corresponding contract, and show using simulation that the specifications on the integrated system are met for the case of $M = 2$ vehicles.

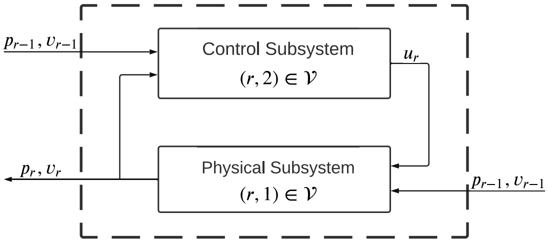


Fig. 4. Interconnection topology of the rth follower, for $r \geq 2$.

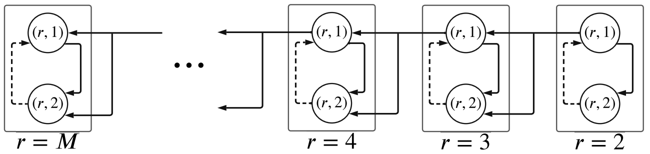


Fig. 5. The interconnection graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ for the scenario in Section 6 with M vehicles. Dashed lines correspond to strictly causal edges, and solid lines correspond to non-strictly causal edges. Each square aggregates the subsystems corresponding to vehicle #r.

6.1. Scenario description and vertical contracts

We consider M vehicles driving along a single-lane highway. The first vehicle in the group is called the leader, and the other $M-1$ vehicles, are the followers. We are given a headway $h > 0$, and a speed limit $V_{\max}^f$, and our goal is to verify that each of the followers keeps at least the given headway from its predecessor, and obeys the speed limit.

We denote the position and velocity of the ith vehicle in the group as p_i and v_i respectively. We consider all followers as one integrated system, whose input is $d^{\text{ext}} = [p_1, v_1] \in \mathcal{S}^2$ and output $y^{\text{ext}} = [p_2, v_2, \dots, p_M, v_M] \in \mathcal{S}^{2(M-1)}$. The guarantees can be written as $p_{r-1}(k) - p_r(k) - hv_r(k) \geq 0$ and $0 \leq v_r(k) \leq V_{\max}^f$ for any $k \in \mathbb{N}$ and $r \in \{2, \dots, M\}$. We assume the leader follows the first kinematic law, i.e., $p_1(k+1) = p_1(k) + \Delta t v_1(k)$ holds for any time k, where $\Delta t > 0$ is the length of the discrete time-step. We further assume the leader obeys a speed limit $V_{\max}^l$, i.e., that $0 \leq v_1(k) \leq V_{\max}^l$ holds for $k \in \mathbb{N}$. The assumptions and guarantees define a contract $\mathcal{C}_{\text{tot}}$ on the followers. For this example, we take $\Delta t = 1$ [s], $h = 2$[s, $V_{\max}^l = 110$ [km/h] and $V_{\max}^f = 100$ [km/h].

We consider each follower vehicle as the interconnection of two subsystems in feedback: a physical subsystem, including all physical components, actuators, etc.; and a control subsystem, which measures the physical subsystem and the environment, and issues a control signal to the physical components. The interconnection of the two systems composing the ith follower can be seen in Fig. 4. The following paragraphs describe the inputs, outputs, assumptions and guarantees associated with the "local" contracts on each subsystem.

First, we consider the physical subsystem, corresponding to the vertex $i = (r,1) \in \mathcal{V}$. Intuitively, the input should only include the control input u_r. However, the headway guarantee refers to the position and velocity of the $(r-1)$-th vehicle. Thus, we take the input $d_i = [p_{r-1}, v_{r-1}, u_r]$ and the output $y_i = [p_r, v_r]$. The physical subsystem is associated with a contract $\mathcal{C}_{\text{phy},r}$. We assume that the $(r-1)$-th vehicle follows the kinematic law $p_{r-1}(k+1) = p_{r-1}(k) + \Delta t v_{r-1}(k)$ for any $k \in \mathbb{N}$. Moreover, we assume the control input satisfies the following inequalities:

$$u_r \leq \frac{p_{r-1} - p_r - hv_r}{h\Delta t} + \frac{v_{r-1} - v_r}{h} - \omega_{\text{acc}},$$
$$\frac{-v_r}{\Delta t} + \omega_{\text{acc}} \leq u_r \leq \frac{V_{\max}^f - v_r}{\Delta t} - \omega_{\text{acc}}.$$

Here, ω_{acc} is a bound on the parasitic acceleration due to wind, friction, etc., which is taken as $\omega_{\text{acc}} = 0.3$ [m/s^2]. These bounds on the control input are motivated by realistic conditions, see Sharf, Besselink, and Johansson (2021), Sharf, Besselink, et al. (2021). As for guarantees, we desire that the headway and speed limit are kept, i.e., that $p_{r-1}(k) - p_r(k) - hv_r(k) \geq 0$ and $0 \leq v_r(k) \leq V_{\max}^f$ hold for all $k \in \mathbb{N}$. We also specify a guarantee that the follower satisfies $p_r(k+1) = p_r(k) + \Delta t v_r(k)$. Thus, $\mathcal{C}_{\text{phy},r}$ is an SRD contract which is strictly causal with respect to u_r, as the guarantees at time k are independent of $u_r(k)$.

Second, we consider an SRD contract $\mathcal{C}_{\text{ctr},r}$ on the control subsystem, matching the vertex $i = (r,2) \in \mathcal{V}$. The input includes the position and velocity of both the rth and the $(r-1)$-th vehicles, i.e., $d_i = [p_{r-1}, v_{r-1}, p_r, v_r]$, and its output is $y_i = u_r$. The contract assumes both vehicles follow the kinematic relations and the speed limits, i.e., that $p_{r-1}(k+1) = p_{r-1}(k) + \Delta t v_{r-1}(k)$, $p_r(k+1) = p_r(k) + \Delta t v_r(k)$, $v_r(k+1) = v_r(k) + \Delta t a_r(k)$, $0 \leq v_{r-1}(k) \leq V_{\max}^l$ and $0 \leq v_r(k) \leq V_{\max}^f$ all hold for any time $k \in \mathbb{N}$. These assumptions can be understood as working limitations for the sensors used by the subsystem to measure the environment, or as first principles used to generate a more exact estimate of the state, which is used for planning and the control law. For guarantees, the control signal u_r must satisfy at any time k:

$$u_r \leq \frac{p_{r-1} - p_r - hv_r}{h\Delta t} + \frac{v_{r-1} - v_r}{h} - \omega_{\text{acc}},$$
$$\frac{-v_r}{\Delta t} + \omega_{\text{acc}} \leq u_r \leq \frac{V_{\max}^f - v_r}{\Delta t} - \omega_{\text{acc}}. \tag{14}$$

We wish to prove that the composition of $\mathcal{C}_{\text{phy},r}$ and $\mathcal{C}_{\text{ctr},r}$ for $r \in \{2, \dots, M\}$ refines $\mathcal{C}_{\text{tot}}$, and we do so using Algorithm 2. First, the composite system is modelled by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{(r,j) : r \in \{2, \dots, M\}, j \in \{1,2\}\}$. As seen in Fig. 5, the set $\mathcal{E}_{\text{nsc}}$ includes the edges $(r,1) \to (r,2)$, as well as the edges $(r-1,1) \to (r,1)$ and $(r-1,1) \to (r,2)$ for $r \in \{3, \dots, M\}$. The set $\mathcal{E}_{\text{sc}}$ includes the edges $(r,2) \to (r,1)$ for $r \in \{2, \dots, M\}$. An illustration of $\mathcal{G}$ can be seen in Fig. 5, which shows the network has no algebraic loops. As the output of $\mathcal{C}_{\text{tot}}$ includes the position and velocity of all followers, we take $\mathcal{V}^{\text{out}} = \{(r,1) : r \in \{2, \dots, M\}\}$. Thus, running Algorithm 2 requires us to solve a total of $|\mathcal{V}| + 1 = 2M - 1$ LPs. We solve them using MATLAB's LP solver for different values of M, detailed in Table 1. In all cases, we find $\varrho_i = \varrho_\Omega = 0$ for all $i \in \mathcal{V}$, so the vertical contract $\bigotimes_{r \in \{2,\dots,M\}} (\mathcal{C}_{\text{phy},r} \otimes \mathcal{C}_{\text{ctr},r}) \preccurlyeq \mathcal{C}_{\text{tot}}$ holds. The results are further discussed below.

6.2. Demonstrating modularity via simulation

In this section, we focus on the case $M = 2$, and thus drop the index r from the contracts $\mathcal{C}_{\text{phy},r}$ and $\mathcal{C}_{\text{ctr},r}$ and from u_r. In this case, the vertical contract $\mathcal{C}_{\text{phy}} \otimes \mathcal{C}_{\text{ctr}} \preccurlyeq \mathcal{C}_{\text{tot}}$ can be interpreted as follows: if the physical and control subsystems of the single follower are designed to satisfy $\mathcal{C}_{\text{phy}}$ and $\mathcal{C}_{\text{ctr}}$, then the integrated system satisfies $\mathcal{C}_{\text{tot}}$. The two subsystem-level contracts are independent of each other, meaning these subsystems can be independently analysed, designed, verified, and tested. We demonstrate this fact by choosing a realisation for the physical subsystems, as well as two realisations for the control, and running the closed-loop system in simulation to show the guarantees hold for both control laws.

For the physical subsystem, we consider a double integrator $p_2(k+1) = p_2(k) + \Delta t v_2(k)$, $v_2(k+1) = v_2(k) + \Delta t(u(k) +$

Table 1
An analysis of the runtime of Algorithm 2 for the vertical contract detailed in Section 6. Network Time refers to the time it took to compute the sets BR, $\mathrm{BR}_{\mathrm{nsc}}$ needed to define the LPs. LP Time refers to the time it took to assemble and solve the LPs using MATLAB's own LP solver on a Dell Latitude 7400 computer with an Intel Core i5-8365U processor.

M	$\lvert\mathcal{V}\rvert$	Num. of LP	Avg. Var. Num.	Avg. constraint num.	Network time [s]	LP time [s]	Total time [s]
2	2	3	14.00	13.33	0.33	1.52	1.86
5	8	9	31.33	21.11	0.35	1.67	2.02
10	18	19	56.95	31.58	0.38	2.15	2.54
20	38	39	107.23	51.79	0.42	4.33	4.75
50	98	99	257.39	111.92	0.57	31.11	31.69
100	198	199	507.45	211.96	0.83	287.76	288.60

$\omega_a(k)$) with acceleration uncertainty. For the realisation Σ_{phy}, acceleration uncertainty is taken as i.i.d. uniformly distributed between $-\omega_{\mathrm{acc}}$ and ω_{acc}. It can be verified that $\Sigma_{\mathrm{phy}} \vDash \mathcal{C}_{\mathrm{phy}}$ using k-induction, similarly to the framework presented in Sharf, Besselink, and Johansson (2021).

For the control subsystem, the first realisation $\Sigma_{\mathrm{ctr}}^{(1)}$ is achieved by taking $u(k)$ as the minimum of the two upper bounds in (14). The second realisation chooses $u(k)$ using an MPC-like controller over a horizon of $T = 5$ steps, assuming constant velocity for the leader. More precisely, $u(k) = u^0$ is chosen by optimising $\sum_{t=1}^{T}[(v_2^t - V_{\mathrm{des}})^2 + (u^t - u^{t-1})^2]$ over the variables $\{p_1^t, v_1^t, p_2^t, v_2^t, u^t\}_{t=0}^{T}$, under the input constraints (14), the kinematic rules $p_1^{t+1} = p_1^t + \Delta v_1^t$, $v_1^{t+1} = v_1^t$, $p_2^{t+1} = p_2^t + \Delta v_2^t$ and $v_2^{t+1} = v_2^t + \Delta u^t$, and the initial constraints $p_2^0 = p_2(k)$, $p_1^0 = p_1(k)$, $v_2^0 = v_2(k)$ and $v_1^0 = v_1(k)$. For the simulation, we choose $V_{\mathrm{des}} = 90$ [km/h]. It can be verified that both systems satisfy $\mathcal{C}_{\mathrm{ctr}}$.

Both realisations $\Sigma_{\mathrm{phy}} \otimes \Sigma_{\mathrm{ctr}}^{(1)}$ and $\Sigma_{\mathrm{phy}} \otimes \Sigma_{\mathrm{ctr}}^{(2)}$ satisfy $\mathcal{C}_{\mathrm{tot}}$. We run simulations of length 300 [s]. In the simulations, the leader starts at a speed of 95 [km/h], and 80 [m] in front of the follower, having an initial speed of 98 [km/h]. The leader will roughly keep its speed for the first 100 seconds. In the next 100 seconds, it will brake and accelerate hard, repeatedly changing its velocity between 95 [km/h] and 10 [km/h]. For the last 100 seconds of the simulations, the leader slowly accelerates to about 105 [km/h], which is faster than the speed limit $V_{\max}^f$ of the follower. The trajectory of the leader can be seen in Fig. 6(a)–(b). The results of the first run are given in Fig. 6(c)–(d), and of the second in Fig. 6(e)–(f). It can be seen that in both runs, the headway between the vehicles is at least 2 [s] and the velocity of the follower is positive and does not exceed 100 [km/h], as prescribed by the guarantees.

In the first run, the controller $\Sigma_{\mathrm{ctr}}^{(1)}$ chooses the maximal possible actuation input $u(k)$ guaranteeing safe behaviour given the contracts on the physical subsystem, encouraging the follower to drive as fast as possible while guaranteeing safety. For this reason, the first 100 s are characterised by the headway approaching 2_s, as the speed of the leader (95 [km/h]) is smaller than the speed limit for the follower. The headway grows at the last 100 seconds of the simulation as the leader accelerates to about 105 [km/h], which is faster than the speed limit of the follower. In the second simulation run, the headway grows large both in the first and the last 100 s, as the MPC controller $\Sigma_{\mathrm{ctr}}^{(2)}$ attempts to keep the speed of the follower around $V_{\mathrm{des}} = 90$ [km/h].

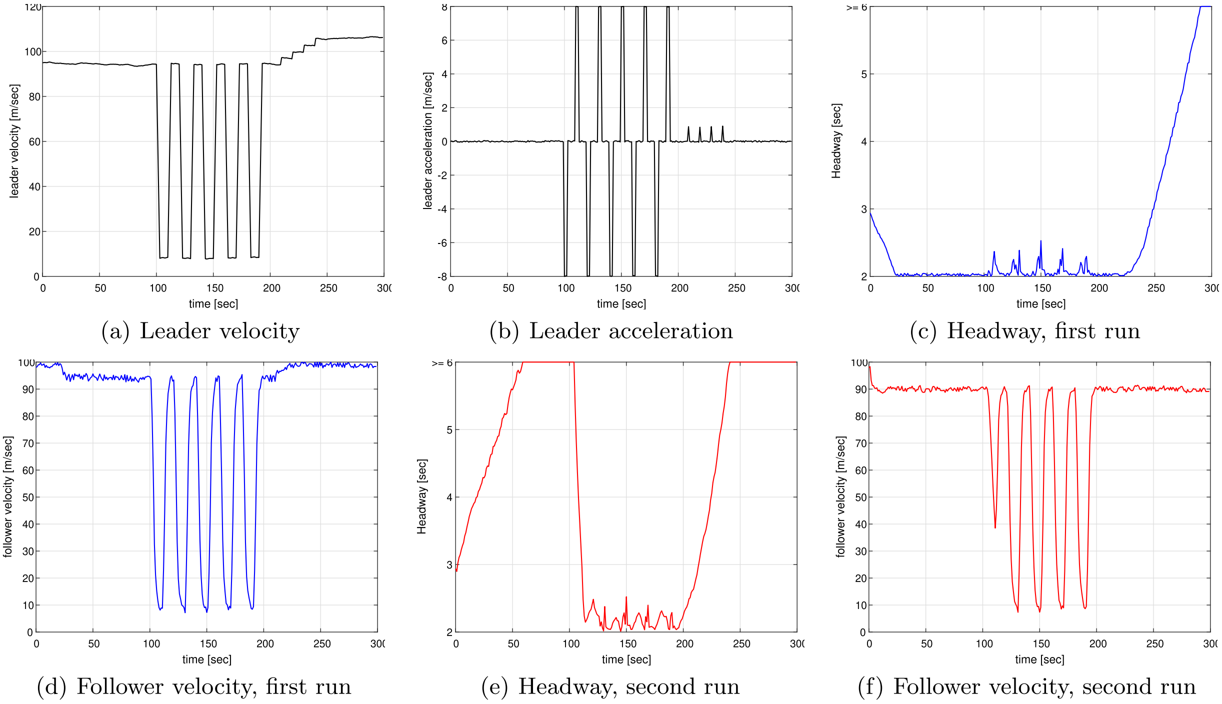
6.3. Discussion

The numerical case study presents some of the advantages of contract theory for design in general and of the presented LP-based framework for verifying vertical contracts in particular. First, Table 1 shows the approach is scalable even for an interconnection of many components. Indeed, we verify that a collection of component-level contracts refines a specification on the composite system comprising 98 components in about 30 seconds, and do the same for a system with 198 components in less than 5 minutes. We also note that contract theory supports hierarchical design, meaning that we do not need to consider hundreds of components or subsystems at the same time. In the numerical example, it is intuitive to first consider each follower on its own, and then decompose each of them further, individually and independently from the other followers. The analysis could be carried out similarly and will have similar results. This hierarchical approach also allows different abstraction levels for each step in the hierarchy. Indeed, when defining the contract for each individual follower, we only need the variables p_r, v_r. The u-variables are only needed when bisecting each follower to its two corresponding subsystems. Moreover, variables corresponding to the measurements taken by the sensors only appear if we decompose the control subsystem into smaller components, responsible for sensing and regulation. We chose not to apply the hierarchical approach in the numerical case study but instead portray the scalability of the proposed framework.

If the composite system is designed according to the principles of contract theory, modularity is achieved by design, meaning that different components or subsystems can be analysed, designed, verified, tested, updated and replaced independently of one another. In this example, if we decide to replace a follower's controller by another control law, only the control subsystem of said follower would have to be re-verified, rather than the entire autonomous vehicle or the entire platoon. In contrast, existing formal methods that do not rely on contract theory mostly consider the entire system as one entity. Thus, any change in any component of the system must be followed by a complete re-verification process of the entire system, no matter how small the component or how insignificant the change is. In general, lack of modularity is a problem which is widespread throughout control theory, with the exception of specialised techniques like retrofit control (Ishizaki, Kawaguchi, Sasahara, & Imura, 2019; Ishizaki, Sadamoto, Imura, Sandberg, & Johansson, 2018; Sadamoto, Chakrabortty, Ishizaki, & Imura, 2017). As highlighted by the example, contracts allow us to prove safety of the closed-loop system before we even know the structure of each block: the same proof of safety for a piecewise-linear controller also held for an MPC-like controller.

7. Conclusion

We considered the problem of contract-based modular design for dynamical control systems. First, we extended the existing definition of contracts to incorporate situations in which the assumption on the input at time k depends on the outputs up to time $k-1$, which are essential for composite systems with feedback. We defined contract composition for such general network interconnections, and proved the definition supports independent design of the components. We then considered vertical contracts, which are statements about the refinement of a contract on a composite system by a collection of component-level contracts. For the case of contracts defined by time-invariant inequalities, we presented efficient LP-based algorithms for verifying these vertical contracts, which scale linearly with the number of components. These results were first achieved for networks without



(a) Leader velocity

(b) Leader acceleration

(c) Headway, first run

(d) Follower velocity, first run

(e) Headway, second run

(f) Follower velocity, second run

Fig. 6. Simulation of the two-vehicle leader–follower system. The black plots correspond to the leader, the blue plots correspond to the first run of the simulation, and the red plots corresponds to the second run of the simulation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

feedback using directed acyclic graphs, and later extended to networks with feedback interconnections but no algebraic loops using causality and strict causality.

One possible avenue for future research is extending the presented contract-based framework to specifications defined using more general temporal logic formulae. Another direction to tackle is finding the optimal vertical contract, i.e., one is given a contract on a composite system, and the goal is to find a vertical contract which is cheapest to implement. Finally, we are interested in using contracts for controller synthesis, as opposed to verification as in this paper.

Acknowledgments

This work was supported in part by Denso, Swedish Research Council Distinguished Professor Grant 2017-01078 and Knut and Alice Wallenberg Foundation Wallenberg Scholar Grant.

Appendix. Proof of Theorem 4.2

This appendix is dedicated to proving Theorem 4.2:

Proof. We show that under the extendability assumptions of the theorem, the set of implications (i) for all $i \in \mathcal{V}$ is equivalent to $\mathcal{D}_{\rm tot} \subseteq \mathcal{D}_\otimes$, and implication (ii) is equivalent to $\Omega_\otimes \cap \mathcal{D}_{\rm tot} \subseteq \Omega_{\rm tot} \cap \mathcal{D}_{\rm tot}$. We start with the former equivalence.

Suppose first that the implication (i) holds for $i \in \mathcal{V}$, and take $(d^{\rm ext}(\cdot), y^{\rm ext}(\cdot)) \in \mathcal{D}_{\rm tot}$. We show that $(d^{\rm ext}(\cdot), y^{\rm ext}(\cdot)) \in \mathcal{D}_\otimes$. In other words, we show that for any $i \in \mathcal{V}$ and for any $\{d_j, y_j\}_{j\in{\rm BR}_+(i)}$ satisfying (2) and (3), if $(d_j, y_j) \in \Omega_j$ holds for $j \in {\rm BR}(i)$ then $(d_i, y_i) \in \Omega_i$. Taking arbitrary $\{d_j, y_j\}_{j\in{\rm BR}_+(i)}$ satisfying these constraints, both $\alpha_{\rm tot}\begin{pmatrix} d^{\rm ext}(k-m^A_{\rm tot}:k) \\ y^{\rm ext}(k-m^A_{\rm tot}:k-1)\end{pmatrix} \le 0$ and $\gamma_j\begin{pmatrix} d_j(k-m^G_j:k) \\ y_j(k-m^G_j:k)\end{pmatrix} \le 0$ hold for any k. Thus, by applying (i) for $d^{\rm ext}, y^{\rm ext}, d_j, y_j$ at times $k - m^A_i, \dots, k$, we yield $\alpha_i\begin{pmatrix} d_i(k-m^A_i:k) \\ y_i(k-m^A_i:k-1)\end{pmatrix} \le 0$ for $k \ge m^A_i$. In particular, we have $(d_i, y_i) \in \mathcal{D}_i$, as claimed. As the choice of $i \in \mathcal{V}$ was arbitrary, we conclude that $(d^{\rm ext}, y^{\rm ext}) \in \mathcal{D}_\otimes$.

Conversely, we assume $\mathcal{D}_\otimes \supseteq \mathcal{D}_{\rm tot}$ and show the implication (i) holds for $i \in \mathcal{V}$. We take $\{d_j, y_j\}_{j\in{\rm BR}_+(i)}, d^{\rm ext}, y^{\rm ext}$ defined up to time m^A_i, and assume they satisfy the consistency constraints (2) and (3), as well as

$$\alpha_{\rm tot}\begin{pmatrix} d^{\rm ext}(\ell-m^A_{\rm tot}:\ell) \\ y^{\rm ext}(\ell-m^A_{\rm tot}:\ell-1)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^A_{\rm tot}, m^A_i},$$
$$\gamma_j\begin{pmatrix} d_j(\ell-m^G_j:\ell) \\ y_j(\ell-m^G_j:\ell)\end{pmatrix} \le 0, \qquad \forall \ell \in \mathcal{I}_{m^G_j, m^A_i},\ j \in {\rm BR}(i).$$

By extendibility, we find signals $\{\hat{y}_j, \hat{d}_j\}$, $\hat{d}^{\rm ext}$ and $\hat{y}^{\rm ext}$ with $\hat{d}^{\rm ext}(0:m^A_i) = d^{\rm ext}(0:m^A_i)$, $\hat{y}^{\rm ext}(0:m^A_i) = y^{\rm ext}(0:m^A_i)$, and $\hat{y}_j(0:m^A_i) = y_j(0:m^A_i)$, $\hat{d}_j(0:m^A_i) = d_j(0:m^A_i)$ for any $j \in {\rm BR}_+(i)$, satisfying (2), (3), and

$$\alpha_{\rm tot}\begin{pmatrix} \hat{d}^{\rm ext}(k-m^A_{\rm tot}:k) \\ \hat{y}^{\rm ext}(k-m^A_{\rm tot}:k-1)\end{pmatrix} \le 0, \qquad \forall k \ge m^A_{\rm tot}$$
$$\gamma_j\begin{pmatrix} \hat{d}_j(k-m^G_j:k) \\ y_j(k-m^G_j:k)\end{pmatrix} \le 0, \qquad \forall k \ge m^G_j,\ \forall j \in {\rm BR}(i).$$

As $(\hat{d}^{\rm ext}, \hat{y}^{\rm ext}) \in \mathcal{D}_\otimes$, we conclude by Definition 4.2 that $(d_i, y_i) \in \mathcal{D}_i$, i.e., that $\alpha_i\begin{pmatrix} \hat{d}_i(k-m^A_i:k) \\ \hat{y}^{\rm ext}(k-m^A_i:k-1)\end{pmatrix} \le 0$ holds for any time $k \ge m^A_i$. Taking $k = m^A_i$ gives the desired result.

We now move to the second part of the theorem, showing that the implication (ii) is equivalent to $\Omega_\otimes \cap \mathcal{D}_{\rm tot} \subseteq \Omega_{\rm tot} \cap \mathcal{D}_{\rm tot}$. Assume first that (ii) holds, and take any $(d^{\rm ext}(\cdot), y^{\rm ext}(\cdot)) \in \mathcal{D}_{\rm tot} \cap \Omega_{\rm tot}$. By Definition 4.2, there exist signals $\{d_i, y_i\}_{i\in\mathcal{V}}$ satisfying the consistency constraints (2) and (3) and $(d_i, y_i) \in \Omega_i$ for $i \in \mathcal{V}$. Thus, for any k and $i \in \mathcal{V}$, both $\alpha_{\rm tot}\begin{pmatrix} d^{\rm ext}(k-m^A_{\rm tot}:k) \\ y^{\rm ext}(k-m^A_{\rm tot}:k-1)\end{pmatrix} \le 0$

and $\gamma_i \begin{pmatrix} d_i(k-m_i^G:k) \\ y_i(k-m_i^G:k) \end{pmatrix} \leq 0$ hold. The implication (ii), applied to $d^{\text{ext}}, y^{\text{ext}}, d_i, y_i$ at times $k - m_{\text{tot}}^G, \dots, k$, gives $\gamma_{\text{tot}} \begin{pmatrix} d^{\text{ext}}(k-m_{\text{tot}}^G:k) \\ y^{\text{ext}}(k-m_{\text{tot}}^G:k) \end{pmatrix} \leq 0$ holds for $k \geq m_{\text{tot}}^G$. We thus yield $(d^{\text{ext}}(\cdot), y^{\text{ext}}(\cdot)) \in \Omega_{\text{tot}}$, as desired.

Conversely, we assume $\Omega_{\otimes} \cap \mathcal{D}_{\text{tot}} \subseteq \Omega_{\text{tot}} \cap \mathcal{D}_{\text{tot}}$ and prove the implication (ii) holds. Take $d^{\text{ext}}, y^{\text{ext}}, d_i, y_i$ defined up to time m_{tot}^G, satisfying constraints (2), (3), and

$$\begin{aligned} &\alpha_{\text{tot}} \begin{pmatrix} d^{\text{ext}}(0:m_{\text{tot}}^A) \\ y^{\text{ext}}(0:m_{\text{tot}}^A-1) \end{pmatrix} \leq 0, && \forall \ell \in \mathcal{I}_{m_{\text{tot}}^A, m_{\text{tot}}^G} \\ &\gamma_i \begin{pmatrix} d_i(\ell-m_i^G:\ell) \\ y_i(\ell-m_i^G:\ell) \end{pmatrix} \leq 0, && \forall \ell \in \mathcal{I}_{m_i^G, m_{\text{tot}}^G},\ \forall i \in \mathcal{V}. \end{aligned}$$

By extendibility, we find signals $\{\hat{y}_i(\cdot), \hat{d}_i(\cdot)\}_{i \in \mathcal{V}}$, $d^{\text{ext}}(\cdot)$ and $y^{\text{ext}}(\cdot)$, such that $\hat{d}^{\text{ext}}(0 : m_{\text{tot}}) = d^{\text{ext}}(0 : m_{\text{tot}})$, $\hat{y}^{\text{ext}}(0 : m_{\text{tot}}) = y^{\text{ext}}(0 : m_{\text{tot}})$, and both $\hat{y}_i(0 : m_{\text{tot}}) = y_i(0 : m_{\text{tot}})$, $\hat{d}_i(0 : m_{\text{tot}}) = d_i(0 : m_{\text{tot}})$ hold for any $i \in \mathcal{V}$. Moreover, for any time k, both the input- and output-consistency constraints (2) and (3) hold, and,

$$\begin{aligned} &\alpha_{\text{tot}} \begin{pmatrix} \hat{d}^{\text{ext}}(k-m_{\text{tot}}^A:k) \\ \hat{y}^{\text{ext}}(k-m_{\text{tot}}^A:k-1) \end{pmatrix} \leq 0, && \forall k \geq m_{\text{tot}}^A \\ &\gamma_i \begin{pmatrix} \hat{d}_i(k-m_i^G:k) \\ \hat{y}_i(k-m_i^G:k) \end{pmatrix} \leq 0, && \forall i \in \mathcal{V}, \forall k \geq m_i^G. \end{aligned}$$

In other words, we have $(\hat{d}^{\text{ext}}, \hat{y}^{\text{ext}}) \in \mathcal{D}_{\text{tot}}$ and $(\hat{d}_i, \hat{y}_i) \in \Omega_i$ for $i \in \mathcal{V}$. Thus, $(\hat{d}^{\text{ext}}, \hat{y}^{\text{ext}}) \in \mathcal{D}_{\text{tot}} \cap \Omega_{\otimes} \subset \mathcal{D}_{\text{tot}} \cap \Omega_{\text{tot}}$, implying that $\gamma_{\text{tot}} \begin{pmatrix} \hat{d}^{\text{ext}}(k-m_{\text{tot}}^G:k) \\ \hat{y}^{\text{ext}}(k-m_{\text{tot}}^G:k) \end{pmatrix} \leq 0$ holds for any $k \geq m_{\text{tot}}^G$. Choosing $k = m_{\text{tot}}^G$ completes the proof. □

References

Baldwin, C. Y., & Clark, K. B. (2006). Modularity in the design of complex engineering systems. In *Complex engineered systems* (pp. 175–205). Springer.

Belta, C., Yordanov, B., & Gol, E. A. (2017). *Formal methods for discrete-time dynamical systems*: *vol. 89*, Springer.

Benveniste, A., Caillaud, B., Nickovic, D., Passerone, R., Raclet, J.-B., Reinkemeier, P., et al. (2018). Contracts for system design. *Foundations and Trends in Electronic Design Automation*, *12*(2–3), 124–400.

Besselink, B., Johansson, K. H., & Schaft, A. V. D. (2019). Contracts as specifications for dynamical systems in driving variable form. In *Proc. Eur. control conf.* (pp. 263–268).

Blanchini, F., & Miani, S. (2008). *Set-theoretic methods in control*. Springer.

Chen, Y., Anderson, J., Kalsi, K., Ames, A. D., & Low, S. H. (2021). Safety-critical control synthesis for network systems with control barrier functions and assume-guarantee contracts. *IEEE Transactions on Control of Network Systems*, *8*(1), 487–499.

Chen, M., Herbert, S. L., Vashishtha, M. S., Bansal, S., & Tomlin, C. J. (2018). Decomposition of reachable sets and tubes for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, *63*(11), 3675–3688.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms*. MIT Press.

Desoer, C. A., & Vidyasagar, M. (2009). *Feedback systems: input-output properties*. SIAM.

Eqtami, A., & Girard, A. (2019). A quantitative approach on assume-guarantee contracts for safety of interconnected systems. In *Proc. Eur. control conf.* (pp. 536–541).

Ghasemi, K., Sadraddini, S., & Belta, C. (2020). Compositional synthesis via a convex parameterization of assume-guarantee contracts. In *Proc. 23rd int. conf. hybrid syst.: comput. control* (pp. 1–10).

Ghasemi, K., Sadraddini, S., & Belta, C. (2022). Compositional synthesis for linear systems via convex optimization of assume-guarantee contracts. arXiv preprint arXiv:2208.01701.

Girard, A., Iovine, A., & Benberkane, S. (2022). Invariant sets for assume-guarantee contracts. In *Proc. 61st IEEE conf. decision control*.

Huang, C.-C., & Kusiak, A. (1998). Modularity in design of products and systems. *IEEE Transactions on Systems, Man, and Cybernetics*, *28*(1), 66–77.

Ishizaki, T., Kawaguchi, T., Sasahara, H., & Imura, J.-i. (2019). Retrofit control with approximate environment modeling. *Automatica*, *107*, 442–453.

Ishizaki, T., Sadamoto, T., Imura, J.-i., Sandberg, H., & Johansson, K. H. (2018). Retrofit control: Localization of controller design and implementation. *Automatica*, *95*, 336–346.

Kim, E. S., Arcak, M., & Seshia, S. A. (2015). Compositional controller synthesis for vehicular traffic networks. In *Proc. 54th IEEE conf. decision control* (pp. 6165–6171).

Kim, E. S., Arcak, M., & Seshia, S. A. (2017). A small gain theorem for parametric assume-guarantee contracts. In *Proc. 20th int. conf. hybrid syst.: comput. control* (pp. 207–216).

Liu, S., Saoud, A., Jagtap, P., Dimarogonas, D. V., & Zamani, M. (2022). Compositional synthesis of signal temporal logic tasks via assume-guarantee contracts. In *Proc. 61st IEEE conf. decision control* (pp. 2184–2189).

Meyer, B. (1992). Applying 'design by contract'. *Computer*, *25*(10), 40–51.

Nuzzo, P., Sangiovanni-Vincentelli, A. L., Bresolin, D., Geretti, L., & Villa, T. (2015). A platform-based design methodology with contracts and related tools for the design of cyber-physical systems. *Proceedings of the IEEE*, *103*(11), 2104–2132.

Nuzzo, P., Xu, H., Ozay, N., Finn, J. B., Sangiovanni-Vincentelli, A. L., Murray, R. M., et al. (2014). A contract-based methodology for aircraft electric power system design. *IEEE Access*, *2*, 1–25.

Rantzer, A. (2015). Scalable control of positive systems. *European Journal of Control*, *24*, 72–80.

Sadamoto, T., Chakrabortty, A., Ishizaki, T., & Imura, J.-i. (2017). Retrofit control of wind-integrated power systems. *IEEE Transactions on Power Systems*, *33*(3), 2804–2815.

Saoud, A., Girard, A., & Fribourg, L. (2018). On the composition of discrete and continuous-time assume-guarantee contracts for invariance. In *Proc. Eur. control conf.* (pp. 435–440).

Saoud, A., Girard, A., & Fribourg, L. (2021). Assume-guarantee contracts for continuous-time systems. *Automatica*, *134*, Article 109910.

Saoud, A., Jagtap, P., Zamani, M., & Girard, A. (2018). Compositional abstraction-based synthesis for cascade discrete-time control systems. In *Proc. 6th IFAC conf. anal. des. hybrid syst.* (pp. 13–18).

Shali, B. M., Heidema, H. M., van der Schaft, A. J., & Besselink, B. (2022). Series composition of simulation-based assume-guarantee contracts for linear dynamical systems. In *Proc. 61st IEEE conf. decision control* (pp. 2204–2209).

Shali, B. M., van der Schaft, A. J., & Besselink, B. (2023). Composition of behavioural assume-guarantee contracts. *IEEE Transactions on Automatic Control*, *68*(10), 5991–6006.

Sharf, M., Besselink, B., & Johansson, K. H. (2021). Verifying contracts for perturbed control systems using linear programming. arXiv preprint arXiv:2111.01259.

Sharf, M., Besselink, B., Molin, A., Zhao, Q., & Johansson, K. H. (2021). Assume/Guarantee contracts for dynamical systems: Theory and computational tools. In *Proc. 7th IFAC conf. anal. des. hybrid syst.*.

Šiljak, D. D., & Zečević, A. (2005). Control of large-scale systems: Beyond decentralized feedback. *The Annual Review of Control*, *29*(2), 169–179.

Smith, S. W., Nilsson, P., & Ozay, N. (2016). Interdependence quantification for compositional control synthesis with an application in vehicle safety systems. In *Proc. IEEE conf. decision control* (pp. 5700–5707).

Tabuada, P. (2009). *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media.

Ulrich, K. (1995). The role of product architecture in the manufacturing firm. *Research Policy*, *24*(3), 419–440.

Willems, J. C. (1972a). Dissipative dynamical systems part I: General theory. *Archive for Rational Mechanics and Analysis*, *45*(5), 321–351.

Willems, J. C. (1972b). Dissipative dynamical systems part II: Linear systems with quadratic supply rates. *Archive for Rational Mechanics and Analysis*, *45*(5), 352–393.

Zamani, M., & Arcak, M. (2018). Compositional abstraction for networks of control systems: A dissipativity approach. *IEEE Transactions on Control of Network Systems*, *5*(3), 1003–1015.

Miel Sharf received his Ph.D. (20') in Aerospace Engineering at the Technion - Israel Institute of Technology, Haifa, Israel. He also received his B.Sc. (13', summa cum laude) and M.Sc. (16', magna cum laude) in Mathematics from the Technion - Israel Institute of Technology.

Since March 2022, he is a researcher with Jether Energy Research, working on topics related to power grids and renewable energy. He was previously a postdoctoral researcher with the Division of Decision and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden. He is a recipient of the Springer Thesis Award, and was inducted to the 2021 class of Forbes Israel "30 under 30". His research interests include the relation between graph theory and algebraic graph theory to multi-agent systems, modular formal verification and synthesis of networked control systems, data-driven control, and security in networked systems.

Bart Besselink is an associate professor at the Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence of the University of Groningen, the Netherlands. He received the M.Sc. degree (cum laude) in Mechanical Engineering in 2008 and the Ph.D. degree in 2012, both from Eindhoven University of Technology, the Netherlands. He was a short-term visiting researcher at the Tokyo Institute of Technology, Japan, in 2012, and a post-doctoral researcher at the Department of Automatic Control and ACCESS Linnaeus Centre at KTH Royal Institute of Technology, Sweden, between 2012 and 2016. His main research interests are on mathematical systems theory for large-scale interconnected systems, with emphasis on contract-based verification and control, model reduction, and applications in intelligent transportation systems and neuromorphic computing. He is a recipient (with Xiaodong Cheng and Jacquelien Scherpen) of the 2020 Automatica Paper Prize.

Karl Henrik Johansson is Director of Digital Futures and Chaired Professor with the School of Electrical Engineering and Computer Science at KTH Royal Institute of Technology in Sweden. He received M.Sc. degree in Electrical Engineering and Ph.D. in Automatic Control from Lund University. He has held visiting positions at UC Berkeley, California Institute of Technology, Nanyang Technological University, Institute of Advanced Studies Hong Kong University of Science and Technology, Norwegian University of Science and Technology, and Zhejiang University. At KTH he directed the ACCESS Linnaeus Centre 2009–2016 and the Strategic Research Area ICT TNG 2013–2020. His research interests are in networked control systems and cyber–physical systems with applications in transportation, energy, and automation networks, areas in which he has co-authored more than 800 journal and conference papers, and supervised almost 100 postdocs and Ph.D. students. He has co-authored and edited 4 books, 33 book chapters, and 7 patents. He is President of the European Control Association and member of the IFAC Council, and has served on the IEEE Control Systems Society Board of Governors and the Swedish Scientific Council for Natural Sciences and Engineering Sciences. He has been on the Editorial Boards of Automatica, IEEE Transactions on Automatic Control, IEEE Transactions on Control of Network Systems, IET Control Theory and Applications, and European Journal of Control, and currently serves on the Editorial Boards of ACM Transactions on Internet of Things, ACM Transactions on Cyber–Physical Systems, and Annual Review of Control, Robotics, and Autonomous Systems. He was the General Chair of the ACM/IEEE Cyber–Physical Systems Week 2010 and IPC Chair of many conferences. He has received several best paper awards and other distinctions from IEEE, IFAC, and ACM. He has been awarded Swedish Research Council Distinguished Professor, Wallenberg Scholar with the Knut and Alice Wallenberg Foundation, Future Research Leader Award from the Swedish Foundation for Strategic Research, the triennial IFAC Young Author Prize, and IEEE Control Systems Society Distinguished Lecturer. He is Fellow of the IEEE and the Royal Swedish Academy of Engineering Sciences.