Contracts as specifications for dynamical systems in driving variable form

Besselink, Bart; Johansson, Karl Henrik; van der Schaft, Arjan

Published in:
Proceedings of the 18th European Control Conference, Naples, Italy

DOI:
10.23919/ECC.2019.8795736

Publication date:
2019

2019 18th European Control Conference (ECC)
Napoli, Italy, June 25-28, 2019

Contracts as specifications for dynamical systems in driving variable form

Bart Besselink, Karl H. Johansson, Arjan van der Schaft

Abstract— **This paper introduces assume/guarantee contracts on continuous-time control systems, hereby extending contract theories for discrete systems to certain new model classes and specifications. Contracts are regarded as formal characterizations of control specifications, providing an alternative to specifications in terms of dissipativity properties or set-invariance. The framework has the potential to capture a richer class of specifications more suitable for complex engineering systems. The proposed contracts are supported by results that enable the verification of contract implementation and the comparison of contracts. These results are illustrated by an example of a vehicle following system.**

I. Introduction

Specifications on dynamical (control) systems typically come in the form of requirements on stability, performance (generally expressed as a bounded gain for a suitably chosen input-output pair), or passivity. Such specifications have in common that they can be captured in the elegant framework of dissipativity as introduced in [25], see also [20], [2] and [21] for related control approaches. An alternative class of specifications can be characterized through set-invariance techniques, capturing properties such as safety, e.g., [9].

However, stricter performance requirements and increasing complexity of modern engineering systems such as intelligent transportation systems or smart manufacturing systems require the expression of control specifications that go beyond dissipativity or invariance. This observation motivates the work on formal methods in control, e.g., [23], [5], which generally requires the abstraction of continuous dynamical systems to discrete transition systems because of the need to express logic specifications such as LTL.

A different approach is taken in this paper. Namely, we present an approach for expressing rich specifications on dynamical systems directly in the continuous domain by introducing *contracts* for linear time-invariant dynamical systems, leading to the following contributions.

First, inspired by contracts for discrete systems in computer science developed in [6] and [7], we define *assume/guarantee contracts* for a class of continuous-time linear dynamical systems. A contract is a pair of dynamical systems known as the assumptions and guarantees and can be regarded as a specification on the dynamical system. Namely, a system implements a contract (i.e., satisfies the specification) when it satisfies the guarantees whenever it is interconnected with an environment that satisfies the assumptions; a definition that will be made precise by using the notion of *simulation* (see, e.g., [16], [19]) as a means for comparing system behavior.

Second, we present a result that allows for efficiently verifying whether a dynamical system implements a given contract. In particular, geometric conditions are given for contract implementation enabling the use of tools from geometric control theory, e.g., [28], [3], [24].

Third, a notion of contract *refinement* is developed as a means for comparing contracts, i.e., providing a way to formalize whether a given contract provides tighter or relaxed requirements with respect to a second contract. The definition is again inspired by results in computer science, see [4], [7].

Fourth, the contract approach is illustrated by application to a vehicle following system as used for, e.g., vehicle platooning [1]. Here, the objective is to *guarantee* a desired inter-vehicle distance under the *assumption* that the lead vehicle satisfies the kinematic relation. Crucially, knowledge of the exact dynamics of the lead vehicle is not assumed.

Through the above contributions, it is argued that assume/guarantee contracts provide various distinguishing and useful features with respect to specifications expressed using dissipativity theory or set invariance. Namely, the explicit characterization (through the assumptions) of the set of environments in which the dynamical system is expected to operate potentially allows for relaxing requirements on the system (as the guarantees might be partially ensured by the assumptions). The characterization of these environments is particularly relevant in the analysis of interconnected systems; an important topic that will be explored in future work.

Moreover, by using simulation relations as a basis for comparing system behavior (recall that the assumptions and guarantees are themselves dynamical systems), a rich class of system behaviors can be characterized including dynamic behavior. We note that the static nature of the supply rates limits the ability of traditional dissipativity theory to capture dynamic behavior. Dynamic supply rates [2] and integral quadratic constraints [14] have been introduced to address this limitation, but these approaches do again not characterize the environment in which a system operates.

Related work on contracts for dynamical systems is presented in [18], where contracts are used to capture set-invariance properties in input and output spaces. As such, the contracts in [18] do not allow for including dynamics in the specification. Richer contracts, called parametric assume/guarantee contracts, are defined in [12], allowing for expressing input-output gain properties. However, only

Bart Besselink and Arjan van der Schaft are with the Jan C. Willems Centre for Systems and Control and the Bernoulli Institute for Mathematics, Computer Science, and Artificial Intelligence, University of Groningen, The Netherlands; Email: b.besselink@rug.nl; a.j.van.der.schaft@rug.nl

Karl H. Johansson is with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Sweden; Email: kallej@kth.se

discrete-time systems are considered in this work.

The remainder of this paper is organized as follows. Section II introduces the class of systems considered in this work and develops a notion of simulation for such systems. Corresponding compositional properties are given in Section III before developing contracts as system specifications in Section IV. Section V presents an illustrative example and Section VI concludes the paper.

Notation. For a linear map $A : \mathcal{X} \to \mathcal{Y}$ with $\mathcal{X}$ and $\mathcal{Y}$ finite-dimensional vector spaces, $\operatorname{im} A$ and $\ker A$ denote the image and kernel of A, respectively. Given a linear subspace $\mathcal{V} \subset \mathcal{X} \times \mathcal{Y}$, let $\pi_{\mathcal{X}}(\mathcal{V}) = \{x \mid \exists y \text{ s.t. } (x, y) \in \mathcal{V}\}$ be the projection of $\mathcal{V}$ on $\mathcal{X}$; $\pi_{\mathcal{Y}}(\mathcal{V})$ is defined similarly.

II. Systems in driving variable form

Consider the linear dynamical system

$$\boldsymbol{\Sigma}_i : \begin{cases} \dot{x}_i = A_i x_i + G_i d_i, \\ w_i = C_i x_i, \\ 0 = H_i x_i, \end{cases} \tag{1}$$

with state $x_i \in \mathcal{X}_i$, external variable $w_i \in \mathcal{W}$, and driving variable $d_i \in \mathcal{D}_i$. Here, $\mathcal{X}_i$, $\mathcal{W}$, and $\mathcal{D}_i$ are finite-dimensional vector spaces. The system (1) is regarded as an *open* system in which the external variable w_i interacts with the environment, whereas the state x_i is the internal variable. The driving variable d_i plays the role of generator of trajectories.

Remark 1: The interpretation of the system (1) in terms of external variables and (internal) state variables is similar to the perspective taken in the behavioral approach to system theory, e.g., [27]. In fact, the form (1) (but without constraints $H_i x_i = 0$) is given in [26] as one representation of this perspective. We stress that no explicit distinction is made between inputs and outputs in (1), even though the external variable w_i could be partitioned as such (see again [27]). ◁

Remark 2: The algebraic constraints in (1) provide a flexible system description that will turn out to be useful in defining system composition (in Section III) as well as in formalizing complex specification, see the example in Section V. ◁

Due to the algebraic constraints in (1), not all initial conditions lead to feasible trajectories. This motivates the introduction of the *consistent subspace* $\mathcal{V}_i^*$ as the set of initial conditions $x_i(0)$ for which there exists (for some $d_i(\cdot)$) a trajectory $x_i(\cdot)$ that satisfies the constraints, i.e., $H_i x_i(t) = 0$ for all $t \in \mathbb{R}_+$. The consistent subspace can be characterized as the largest (with respect to subspace inclusion) subspace $\mathcal{V}_i \subset \mathcal{X}_i$ such that

$$A_i \mathcal{V}_i \subset \mathcal{V}_i + \operatorname{im} G_i, \qquad \mathcal{V}_i \subset \ker H_i, \tag{2}$$

see, e.g., [22], [13].

Following results on unconstrained systems in [16], [19] and constrained systems in [22], the notion of *simulation relation* is introduced as a means for comparing the behavior of systems $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$. Such system comparison will turn out to be crucial for expressing rich system specifications.

Definition 1: A linear subspace $\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$ satisfying $\pi_{\mathcal{X}_i}(\mathcal{S}) \subset \mathcal{V}_i^*$, $i \in \{1, 2\}$, is a simulation relation of $\boldsymbol{\Sigma}_1$ by $\boldsymbol{\Sigma}_2$ if, for all $(x_1(0), x_2(0)) \in \mathcal{S}$, the following hold:

1) for each driving function $d_1(\cdot)$ such that the corresponding state trajectory $x_1(\cdot)$ with initial condition $x_1(0)$ satisfies $x_1(t) \in \mathcal{V}_1^*$ for all $t \in \mathbb{R}_+$, there exists a driving function $d_2(\cdot)$ such that the corresponding state trajectory $x_2(\cdot)$ with initial condition $x_2(0)$ satisfies
$$\big(x_1(t), x_2(t)\big) \in \mathcal{S} \tag{3}$$
for all $t \in \mathbb{R}_+$;
2) the external variables are equal, i.e.,
$$C_1 x_1(0) = C_2 x_2(0). \tag{4}$$

Whereas simulation relations are defined in terms of system trajectories, equivalent algebraic conditions can be obtained using standard arguments from geometric control theory, e.g., [28], [3], [24]. This is formalized next.

Lemma 1: A linear subspace $\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$ satisfying $\pi_{\mathcal{X}_i}(\mathcal{S}) \subset \mathcal{V}_i^*$, $i \in \{1, 2\}$, is a simulation relation of $\boldsymbol{\Sigma}_1$ by $\boldsymbol{\Sigma}_2$ if and only if the following conditions hold for all $(x_1, x_2) \in \mathcal{S}$:

1) for all $d_1 \in \mathcal{D}_1$ such that $A_1 x_1 + G_1 d_1 \in \mathcal{V}_1^*$, there exists $d_2 \in \mathcal{D}_2$ such that $A_2 x_2 + G_2 d_2 \in \mathcal{V}_2^*$ and
$$\big(A_1 x_1 + G_1 d_1, A_2 x_2 + G_2 d_2\big) \in \mathcal{S}; \tag{5}$$
2) the external variables are equal, i.e.,
$$C_1 x_1 = C_2 x_2. \tag{6}$$

Proof: The proof follows that of [13, Proposition 3.1], see also [19, Proposition 2.9] for unconstrained systems. ■

Now, *simulation* can be defined directly using the notion of simulation relation in Definition 1.

Definition 2: A system $\boldsymbol{\Sigma}_1$ is said to be simulated by $\boldsymbol{\Sigma}_2$ (or, $\boldsymbol{\Sigma}_2$ simulates $\boldsymbol{\Sigma}_1$), denoted $\boldsymbol{\Sigma}_1 \preccurlyeq \boldsymbol{\Sigma}_2$, if there exists a simulation relation $\mathcal{S}$ of $\boldsymbol{\Sigma}_1$ by $\boldsymbol{\Sigma}_2$ satisfying $\pi_{\mathcal{X}_1}(\mathcal{S}) = \mathcal{V}_1^*$ and $\pi_{\mathcal{X}_2}(\mathcal{S}) \subset \mathcal{V}_2^*$. A simulation relation with this property will be referred to as a full simulation relation.

The following lemma gives an important property of the notion of simulation, which enables its use as a means for comparing the behavior of systems $\boldsymbol{\Sigma}_i$. The result of this lemma will be exploited later.

Lemma 2: Simulation $\preccurlyeq$ is a preorder. Namely, it is

1) reflexive, i.e., $\boldsymbol{\Sigma}_1 \preccurlyeq \boldsymbol{\Sigma}_1$ for any $\boldsymbol{\Sigma}_1$;
2) transitive, i.e., for any systems $\boldsymbol{\Sigma}_i$, $i \in \{1, 2, 3\}$ satisfying $\boldsymbol{\Sigma}_1 \preccurlyeq \boldsymbol{\Sigma}_2$ and $\boldsymbol{\Sigma}_2 \preccurlyeq \boldsymbol{\Sigma}_3$, it holds that $\boldsymbol{\Sigma}_1 \preccurlyeq \boldsymbol{\Sigma}_3$.

Proof: Reflexivity of the simulation operation follows by choosing $\mathcal{S} = \{(x_1, x_1) \mid x_1 \in \mathcal{V}_1^*\}$, which is easily verified to be a full simulation relation of $\boldsymbol{\Sigma}_1$ by itself. To prove transitivity, let $\mathcal{S}_{12}$ and $\mathcal{S}_{23}$ be (full) simulation relations of $\boldsymbol{\Sigma}_1$ by $\boldsymbol{\Sigma}_2$ and $\boldsymbol{\Sigma}_2$ by $\boldsymbol{\Sigma}_3$, respectively. Following [11], define

$$\mathcal{S}_{13} = \big\{(x_1, x_3) \,\big|\, \exists x_2 \in \mathcal{X}_2 \text{ such that } (x_1, x_2) \in \mathcal{S}_{12}, (x_2, x_3) \in \mathcal{S}_{23}\big\}. \tag{7}$$

Then, it can be checked that $\mathcal{S}_{13}$ defines a full simulation relation of $\boldsymbol{\Sigma}_1$ by $\boldsymbol{\Sigma}_3$. ■

III. System composition

In this section, we consider the interconnection of systems through their external variables. Given two systems $\mathbf{\Sigma}_i$, $i \in \{1,2\}$, of the form (1), their composition $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ is defined as the system resulting from setting

$$w_1 = w_2. \tag{8}$$

Thus, system composition is regarded as *variable sharing*; a perspective that is advocated in, e.g., [27].

Following (8), a realization of $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ is given as

$$\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2 : \begin{cases} \dot{x}^\otimes &= A^\otimes x^\otimes + G^\otimes d^\otimes, \\ w^\otimes &= C^\otimes x, \\ 0 &= H^\otimes x, \end{cases} \tag{9}$$

with state $x^\otimes = (x_1^\otimes, x_2^\otimes) \in \mathcal{X}_1 \times \mathcal{X}_2$, external variable $w^\otimes \in \mathcal{W}$, and driving variable $d^\otimes = (d_1^\otimes, d_2^\otimes) \in \mathcal{D}_1 \times \mathcal{D}_2$. The linear maps in (9) are given by

$$A^\otimes = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, \quad G^\otimes = \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \end{bmatrix}, \tag{10}$$

$$C^\otimes = \tfrac{1}{2}\begin{bmatrix} C_1 & C_2 \end{bmatrix}, \quad H^\otimes = \begin{bmatrix} H_1 & 0 \\ 0 & H_2 \\ C_1 & -C_2 \end{bmatrix}. \tag{11}$$

Note that the final constraint imposed by $H^\otimes$ restricts the external behavior of (9) to the external behavior that is common to $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$, which agrees with (8). Due to this additional constraint, the consistent subspace $\mathcal{V}^{\otimes,*}$ of $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ can not easily be expressed in terms of those of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. Instead, recall that $\mathcal{V}^{\otimes,*}$ is the largest subspace $\mathcal{V}^\otimes$ satisfying

$$A^\otimes \mathcal{V}^\otimes \subset \mathcal{V}^\otimes + \operatorname{im} G^\otimes, \qquad \mathcal{V}^\otimes \subset \ker H^\otimes. \tag{12}$$

The next result states that the composed system $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ is simulated by both $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. In fact, it is the *largest* (with respect to simulation) system with this property.

Theorem 3: Consider systems $\mathbf{\Sigma}_i$, $i \in \{1,2\}$, of the form (1) and let their composition $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ be defined as in (9)–(11). Then, the following two statements hold:

1) For $i \in \{1,2\}$, $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$ is simulated by $\mathbf{\Sigma}_i$, i.e.,
$$\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2 \preccurlyeq \mathbf{\Sigma}_i. \tag{13}$$
2) Let $\mathbf{\Sigma}$ be a system of the form (1) that is simulated by both $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. Then, it is also simulated by $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2$. Stated differently, the following holds:
$$\mathbf{\Sigma} \preccurlyeq \mathbf{\Sigma}_i,\ i \in \{1,2\} \quad \Longrightarrow \quad \mathbf{\Sigma} \preccurlyeq \mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2. \tag{14}$$

Proof: The proof can be found in [8, Appendix A]. ■

Theorem 3 thus formalizes the intuition that the interconnection of systems through variable sharing in (8) can only restrict the behavior of systems (the first statement of the theorem). Another important consequence of Theorem 3 is that the property of simulation is preserved under system composition, as stated next.

Theorem 4: Let $\mathbf{\Sigma}_i$ and $\mathbf{\Sigma}'_i$, $i \in \{1,2\}$, be systems of the form (1) such that $\mathbf{\Sigma}_1 \preccurlyeq \mathbf{\Sigma}'_1$ and $\mathbf{\Sigma}_2 \preccurlyeq \mathbf{\Sigma}'_2$. Then,

$$\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2 \preccurlyeq \mathbf{\Sigma}'_1 \otimes \mathbf{\Sigma}'_2. \tag{15}$$

Proof: From the first statement in Theorem 3 we obtain

$$\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2 \preccurlyeq \mathbf{\Sigma}_i \preccurlyeq \mathbf{\Sigma}'_i, \tag{16}$$

for $i \in \{1,2\}$ and where the final simulation relation follows from the assumption $\mathbf{\Sigma}_i \preccurlyeq \mathbf{\Sigma}'_i$. Then, transitivity of the simulation relation (see Lemma 2) gives $\mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2 \preccurlyeq \mathbf{\Sigma}'_i$, after which the application of the second statement of Theorem 3 leads to (15). ■

IV. Contracts as specifications

Consider the linear dynamical system

$$\mathbf{\Sigma} : \begin{cases} \dot{x} &= Ax + Gd, \\ w &= Cx, \\ 0 &= Hx, \end{cases} \tag{17}$$

as in (1), but with indices removed for ease of presentation. The system (17) can interact with its environment through the external variable w. To make this explicit, an environment $\mathcal{E}$ is defined to be a system of the same form, i.e.,

$$\mathcal{E} : \begin{cases} \dot{x}^e &= A^e x^e + G^e d^e, \\ w^e &= C^e x^e, \\ 0 &= H^e x^e, \end{cases} \tag{18}$$

with $x^e \in \mathcal{X}^e$ and driving variable $d^e \in \mathcal{D}^e$. Finally, its external variables w^e take values in the same space $\mathcal{W}$ as the system (17), i.e., $w^e \in \mathcal{W}$. Consequently, the interconnection of $\mathbf{\Sigma}$ and $\mathcal{E}$ can be considered by setting $w = w^e$, leading to the system $\mathcal{E} \otimes \mathbf{\Sigma}$ with external variables $w \in \mathcal{W}$.

We are interested in guaranteeing properties of $\mathbf{\Sigma}$ when interconnected with relevant environments $\mathcal{E}$. To make this explicit, two systems will be introduced. First, define *assumptions* $\mathcal{A}$ as the system

$$\mathcal{A} : \begin{cases} \dot{x}^a &= A^a x^a + G^a d^a, \\ w^a &= C^a x^a, \\ 0 &= H^a x^a, \end{cases} \tag{19}$$

with $x^a \in \mathcal{X}^a$, $d^a \in \mathcal{D}^a$, and external variables $w^a \in \mathcal{W}$. Next, *guarantees* $\mathcal{G}$ are defined as

$$\mathcal{G} : \begin{cases} \dot{x}^g &= A^g x^g + G^g d^g, \\ w^g &= C^g x^g, \\ 0 &= H^g x^g, \end{cases} \tag{20}$$

where $x^g \in \mathcal{X}^g$, $d^g \in \mathcal{D}^g$, and $w^g \in \mathcal{W}$. Note that both the assumptions (19) and guarantees (20) are systems of the same form as $\mathbf{\Sigma}$ in (17) and that they share the same space of external variables. Finally, $\mathcal{X}^a$, $\mathcal{D}^a$, $\mathcal{X}^g$, and $\mathcal{D}^g$ above are all finite-dimensional vector spaces.

The introduction of $\mathcal{A}$ and $\mathcal{G}$ allows for defining contracts.

Definition 3: A contract $\mathcal{C}$ is a pair of systems $(\mathcal{A}, \mathcal{G})$ as in (19) and (20).

The relevance of contracts is given by their use as formal specifications for systems $\mathbf{\Sigma}$ as in (17). This is made explicit using the following definition.

Definition 4: An environment $\mathcal{E}$ as in (18) is said to be compatible with the contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$ if it is simulated by the assumptions $\mathcal{A}$, i.e., $\mathcal{E} \preccurlyeq \mathcal{A}$. A system $\mathbf{\Sigma}$ as in (17) is said to be an implementation of the contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$

if the composition of $\mathbf{\Sigma}$ with any compatible environment is simulated by the guarantees $\mathcal{G}$, i.e.,

$$\mathcal{E} \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{G} \tag{21}$$

for all $\mathcal{E} \preccurlyeq \mathcal{A}$.

A contract $\mathcal{C}$ thus gives a formal specification for the external behavior of a system $\mathbf{\Sigma}$ through two aspects. First, it specifies (by the assumptions $\mathcal{A}$) the class of environments in which the system is supposed to operate. Second, it characterizes the required behavior of $\mathbf{\Sigma}$ through the guarantees $\mathcal{G}$, which the system needs to satisfy *for any* compatible environment.

Remark 3: Whereas the concept of contracts was originally proposed in the scope of software engineering in [15], Definition 3 is inspired by assume/guarantee contracts in formal methods developed in [6], see also the recent book [7] for a detailed discussion. We note that these works consider models of computation that are inherently discrete in nature, such that the theory developed in [6], [7] is not applicable to continuous dynamical systems as in (17). ◁

Whereas Definition 4 defines contract implementation using a class of environments, the verification of contract implementation can be performed on the basis of the contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$ directly, i.e., without explicitly constructing all compatible environments. This is stated next.

Lemma 5: Consider a system $\mathbf{\Sigma}$ as in (17) and a contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$. Then, $\mathbf{\Sigma}$ is an implementation of the contract if and only if

$$\mathcal{A} \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{G}. \tag{22}$$

Proof: This is a direct result of Theorem 3 and the fact that $\mathcal{E} = \mathcal{A}$ is a compatible environment. ■

An important consequence of Lemma 5 is that it allows for efficiently verifying whether a system $\mathbf{\Sigma}$ implements a given contract $\mathcal{C}$. To do so, the following theorem is instrumental.

Theorem 6: Consider a system $\mathbf{\Sigma}$ as in (17) and a contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$. Let the consistent subspaces of $\mathcal{A} \otimes \mathbf{\Sigma}$ and $\mathcal{G}$ be denoted as $\mathcal{V}^{\otimes,*}$ and $\mathcal{V}^{g,*}$, respectively. Then, a linear subspace $\mathcal{S} \subset \mathcal{X} \times \mathcal{X}^a \times \mathcal{X}^g$ satisfies $\pi_{\mathcal{X}^a \times \mathcal{X}}(\mathcal{S}) \subset \mathcal{V}^{\otimes,*}$ and $\pi_{\mathcal{X}^g}(\mathcal{S}) \subset \mathcal{V}^{g,*}$ and is a simulation relation of $\mathcal{A} \otimes \mathbf{\Sigma}$ by $\mathcal{G}$ if and only if

$$\begin{bmatrix} A^{\otimes} & 0 \\ 0 & A^g \end{bmatrix} \mathcal{S} \subset \mathcal{S} + \operatorname{im} \begin{bmatrix} G^{\otimes} & 0 \\ 0 & G^g \end{bmatrix}, \tag{23}$$

$$\begin{bmatrix} \operatorname{im} G^{\otimes} \cap \mathcal{V}^{\otimes,*} \\ 0 \end{bmatrix} \subset \mathcal{S} + \begin{bmatrix} 0 \\ \operatorname{im} G^g \cap \mathcal{V}^{g,*} \end{bmatrix}, \tag{24}$$

$$\mathcal{S} \subset \ker \begin{bmatrix} H^{\otimes} & 0 \\ 0 & H^g \\ C^{\otimes} & -C^g \end{bmatrix}, \tag{25}$$

where the linear maps $A^{\otimes}$, $G^{\otimes}$, $C^{\otimes}$, $H^{\otimes}$ form a realization of $\mathcal{A} \otimes \mathbf{\Sigma}$. The system $\mathbf{\Sigma}$ implements the contract $\mathcal{C}$ if and only if there exists a linear subspace $\mathcal{S}$ satisfying the above and, in addition, $\pi_{\mathcal{X}^a \times \mathcal{X}}(\mathcal{S}) = \mathcal{V}^{\otimes,*}$.

Proof: The proof is given in [8, Appendix B]. ■

Remark 4: Theorem 6 enables the efficient algorithmic verification of contract implementation through the use of tools from geometric control theory, e.g., [28], [24]. Namely, the so-called *invariant subspace algorithm* can compute the largest (in the sense of subspace inclusion) subspace $\mathcal{S}$ that satisfies (23) and (25). For this subspace $\mathcal{S}^*$, the condition (24) as well as $\pi_{\mathcal{X} \times \mathcal{X}^a}(\mathcal{S}) = \mathcal{V}^{\otimes,*}$ then need to be verified, which can again be done algorithmically. For an example of the use of the invariant subspace algorithm for system (bi)simulation (albeit for a different class of systems than the one studied in this paper), see [19]. ◁

Remark 5: From condition (22) in Lemma 5 and Theorem 3, it can be concluded that if $\mathbf{\Sigma}$ is an implementation of $\mathcal{C} = (\mathcal{A}, \mathcal{G})$, it is also an implementation of $(\mathcal{A}, \mathcal{A} \otimes \mathcal{G})$ (and vice versa). There is thus no restriction in replacing $\mathcal{G}$ by $\mathcal{G}' = \mathcal{A} \otimes \mathcal{G}$. ◁

A distinguishing feature of using contracts as specifications is that contracts themselves can be compared through a notion of *refinement* (see [4] for a similar definition in a more abstract setting).

Definition 5: A contract $\mathcal{C}' = (\mathcal{A}', \mathcal{G}')$ is said to refine a contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$, denoted as $\mathcal{C}' \preccurlyeq \mathcal{C}$, if the following two conditions hold:

$$\mathcal{A} \preccurlyeq \mathcal{A}', \qquad \mathcal{A} \otimes \mathcal{G}' \preccurlyeq \mathcal{G}. \tag{26}$$

The above definition allows one to reason about tightening or relaxing specifications, where we note that this involves two aspects. Namely, a contract $\mathcal{C}'$ refines $\mathcal{C}$ if it simultaneously *enlarges* the class of environments and asks for *tighter* guarantees. This observation is made explicit as follows.

Theorem 7: Let $\mathcal{C}' = (\mathcal{A}', \mathcal{G}')$ and $\mathcal{C} = (\mathcal{A}, \mathcal{G})$ be contracts such that $\mathcal{C}' \preccurlyeq \mathcal{C}$. Then, the following holds:

1) If $\mathcal{E}$ is a compatible environment for $\mathcal{C}$, then it is also a compatible environment for $\mathcal{C}'$.
2) If $\mathbf{\Sigma}$ is an implementation of $\mathcal{C}'$, then it is also an implementation of $\mathcal{C}$.

Proof: Statement 1 is a direct result of the condition $\mathcal{A} \preccurlyeq \mathcal{A}'$ in (26), as transitivity of the simulation relation (see Theorem 2) gives that $\mathcal{E} \preccurlyeq \mathcal{A}$ implies $\mathcal{E} \preccurlyeq \mathcal{A}'$ (recall Definition 4 on compatibility of an environment).

To prove statement 2, Let $\mathbf{\Sigma}$ be an implementation of $\mathcal{C}'$. Now, consider

$$\mathcal{A} \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{A}' \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{G}'. \tag{27}$$

Here, the first simulation relation is the result of Theorem 3 and $\mathcal{A} \preccurlyeq \mathcal{A}'$, whereas the second follows from the assumption that $\mathbf{\Sigma}$ implements $\mathcal{C}$, i.e., $\mathcal{A}' \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{G}'$ by Lemma 5. Using the definition of system composition and Theorem 3, we also have $\mathcal{A} \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{A}$, after which the same theorem gives $\mathcal{A} \otimes \mathbf{\Sigma} \preccurlyeq \mathcal{A} \otimes \mathcal{G}'$. The result then follows from (26) and transitivity of the simulation relation. ■

Remark 6: When regarded as a means for characterizing control specifications, contracts in Definition 3 provide an alternative to specifications expressed as dissipativity or set-invariance properties, see [25], [20], [9]. Contracts have the unique feature that they explicitly characterize the set of environments in which a system $\mathbf{\Sigma}$ is supposed to operate. In addition, the expression of contracts as (a pair of) dynamical

Fig. 1. A vehicle following system.

systems and the use of the notion of simulation for comparing system behavior allows for defining specifications in which dynamic behavior can explicitly be taken into account, potentially allowing for expressing richer specifications than in dissipativity or set-invariance theory. ◁

V. Illustrative example

To illustrate the assume/guarantee reasoning enabled by contracts, consider the vehicle following system in Figure 1. We aim to show that a given controller for the follower vehicle (with index 2) guarantees that the inter-vehicle distance to its predecessor satisfies a certain spacing policy, regardless of the behavior of the predecessor.

For the vehicle following system, the position s_i and velocity v_i, $i \in \{1, 2\}$, of both vehicles are regarded as the external variables, such that

$$w^{\mathrm{T}} = \begin{bmatrix} s_1 & v_1 & s_2 & v_2 \end{bmatrix}, \tag{28}$$

and $\mathcal{W} = \mathbb{R}^4$. We assume that the exact dynamics of the first vehicle is unknown, but that the second vehicle satisfies

$$\dot{s}_2 = v_2, \quad \dot{v}_2 = u_2, \tag{29}$$

and implements the controller

$$u_2 = h^{-1}(v_1 - v_2) + kh^{-1}(s_1 - s_2 - hv_2), \tag{30}$$

for some $h, k > 0$. Now, the objective is to show that the controller (30) guarantees tracking of the so-called constant headway spacing policy (see, e.g., [10]), i.e., that

$$s_2(t) - s_1(t) = hv_2(t) \tag{31}$$

holds for all $t \in \mathbb{R}_+$. To guarantee this property using a contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$, the requirement (31) is captured by choosing the guarantees $\mathcal{G}$ as in (20) with state $x^g = w^g$ according to (28) and such that

$$A^g = I, \quad G^g = I, \quad C^g = I, \quad H^g = \begin{bmatrix} -1 & 0 & 1 & h \end{bmatrix}. \tag{32}$$

Note that the guarantees $\mathcal{G}$ merely constrain the external variables w^g through the constraint H^g (corresponding to (31)) and that the choice for A^g, G^g, and C^g does not impose further restrictions. As a result, it is easy to show that the consistent subspace of $\mathcal{G}$ is given as $\mathcal{V}^{g,*} = \ker H^g$.

Next, even though the exact dynamics of the first vehicle is unknown, it is safe to assume that its position s_1 and velocity v_1 satisfy the kinematic relation $\dot{s}_1 = v_1$. This can be expressed by choosing the assumptions $\mathcal{A}$ in (19) as

$$A^a = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad G^a = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad C^a = I, \tag{33}$$

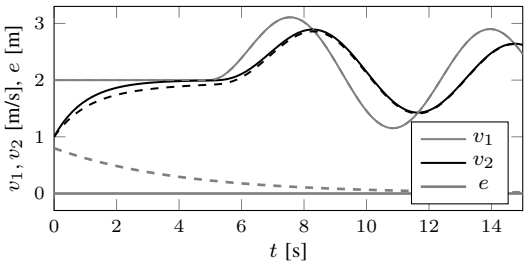

Fig. 2. Simulation of the model (29), (30), and (37) for initial conditions $\begin{bmatrix} 1 & 2 & 0 & 1 \end{bmatrix}^{\mathrm{T}} \in \ker H^g$ (solid) and $\begin{bmatrix} 1 & 2 & 0.8 & 1 \end{bmatrix}^{\mathrm{T}} \notin \ker H^g$ (dashed) and parameters $h = 1$, $k = 0.25$, $c = 0.5$. Here, $e = -s_1 + s_2 + hv_1$ and the external disturbance is chosen as $d_1(t) = 1$ for $t \in [0, 5)$ and $d_1(t) = 1 + \sin(t - 5)$ for $t \geq 5$.

and with $H^a = 0$, corresponding to the state $x^a = w^a$. We stress that the form (33) characterizes the kinematic relation of the first vehicle, but does not constrain the behavior of the second vehicle (in terms of s_2 and v_2).

Given the contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$ specified by (32) and (33), it remains to be shown that the vehicle (29) with controller (30) satisfies the contract. To this end, the closed-loop system $\mathbf{\Sigma}$ as in (17) (with state $x = w$) is represented as

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ kh^{-1} & h^{-1} & -kh^{-1} & -k - h^{-1} \end{bmatrix}, \; G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{34}$$

and with $C = I$, $H = 0$. Similar to before, we stress that the system (34) does not pose any restrictions on the behavior of the first vehicle, but only captures the dynamics (29)–(30).

In order to use Theorem 6 to verify contract implementation, the invariant subspace algorithm (see Remark 4) is used to compute the largest linear subspace $\mathcal{S}$ satisfying (23) and (25). This yields

$$\mathcal{S} = \left\{ (x^a, x, x^g) \,\middle|\, x^a = x = x^g,\, x^g \in \mathcal{V}^{g,*} \right\}. \tag{35}$$

Then, after noting that the consistent subspace of the composition $\mathcal{A} \otimes \mathcal{G}$ is given by $\mathcal{V}^{\otimes,*} = \{(x^a, x) \,|\, x^a = x\}$, it can be verified that $\mathcal{S}$ in (35) also satisfies (24). Hence, $\mathcal{S}$ is a simulation relation of $\mathcal{A} \otimes \mathbf{\Sigma}$ by $\mathcal{G}$.

We note that, by definition of contract implementation and simulation (see Definition 2), the simulation relation $\mathcal{S}$ needs to be full in order to guarantee contract implementation, see Theorem 6. However, this is not the case as

$$\pi_{\mathcal{X}^a \times \mathcal{X}}(\mathcal{S}) = \left\{ (x^a, x) \,\middle|\, x^a = x,\, x \in \mathcal{V}^{g,*} \right\} \subsetneq \mathcal{V}^{\otimes,*}. \tag{36}$$

This limitation stems from the fact that satisfaction of the spacing policy (31) is not guaranteed for initial conditions x_0 of $\mathbf{\Sigma}$ that do not satisfy the constraint $H^g x_0 = 0$. Nonetheless, for initial conditions $x_0 \in \ker H^g = \mathcal{V}^{g,*}$, the existence of the simulation relation $\mathcal{S}$ guarantees that the controlled vehicle (29)–(30) achieves tracking of the spacing policy (31) (captured in $\mathcal{G}$) *for any* preceding vehicle that satisfies the kinematic relation (captured in $\mathcal{A}$).

To illustrate this, let the dynamics of the first vehicle be given as

$$\dot{s}_1 = v_1, \quad \dot{v}_1 = -cv_1 + d_1, \tag{37}$$

for some constant $c > 0$ and external disturbance d_1. It can be verified that the dynamics (37) is simulated by the assumptions $\mathcal{A}$ in (33) (when (37) is appended with arbitrary dynamics for the second vehicle in a similar way as in (33)). Hence, the satisfaction of the spacing policy is guaranteed for initial conditions $x_0 \in \ker H^g$. This is confirmed by the results in Figure 2, which depicts (in solid lines) a time simulation of the model (29), (30), and (37) for initial conditions $x_0 \in \ker H^g$ and confirms that the spacing policy (31) is satisfied for all time $t \geq 0$ (even when the first vehicle is subject to time-varying disturbances). Finally, we note that it can in addition be shown that the subspace $\ker H^g$ is attractive, even though this is not a requirement in the contract $\mathcal{C} = (\mathcal{A}, \mathcal{G})$. This is illustrated by a simulation in Figure 2 as well (in dashed lines).

VI. Conclusions

Assume/guarantee contracts for dynamical systems are introduced in this paper as a means of characterizing control system specifications. For these contracts, a result is given that enables efficient algorithmic verification of contract satisfaction and the notion of refinement is introduced to allow for comparison of contracts.

We regard this work as the first step towards contract theory for continuous-time dynamical control systems. For such theory, tools for system composition are crucial (see [17], [7]) and future work will focus on this topic. The expression of relevant control specifications in terms of contracts will be a second topic of further research.

References

[1] A. Alam, B. Besselink, V. Turri, J. Mårtensson, and K.H. Johansson. Heavy-duty vehicle platooning towards sustainable freight transportation: A cooperative method to enhance safety and efficiency. *IEEE Control Systems Magazine*, 35(6):34–56, 2015.

[2] M. Arcak, C. Meissen, and A. Packard. *Networks of dissipative systems: Compositional certification of stability, performance, and safety*. SpringerBriefs in Control, Automation and Robotics. Springer International Publishing, Switzerland, 2016.

[3] G. Basile and G. Marro. *Controlled and conditioned invariants in linear system theory*. Prentice Hall, Englewood Cliffs, USA, 1992.

[4] S.S. Bauer, A. David, R. Hennicker, K. Guldstrand Larsen, A. Legay, U. Nyman, and A. Wasowski. Moving from specifications to contracts in component-based design. In J. de Lara and Zisman A., editors, *Fundamental Approaches to Software Engineering*, volume 7212 of *Lecture Notes in Computer Science*, pages 43–58. Springer, Berlin Heidelberg, Germany, 2012.

[5] C. Belta, B. Yordanov, and E.A. Gol. *Formal methods for discrete-time dynamical systems*, volume 89 of *Studies in Systems, Decision and Control*. Springer, Cham, Switzerland, 2017.

[6] A. Benveniste, B. Caillaud, A. Ferrari, L. Mangeruca, R. Passerone, and C. Sofronis. Multiple viewpoint contract-based specification and design. In *Proceedings of the 6th International Symposium on Formal Methods for Components and Objects, Amsterdam, the Netherlands*, volume 5382 of *Lecture Notes in Computer Science*, pages 200–225. Springer Berlin Heidelberg, Germany, 2008.

[7] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. Henzinger, and K. Larsen. Contracts for system design. *Foundations and Trends in Electronic Design Automation*, 12(2-3):124–400, 2018.

[8] B. Besselink, K.H. Johansson, and A.J. van der Schaft. Contracts as specifications for dynamical systems in driving variable form. arXiv:1810.05542, 2019.

[9] F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

[10] P.A. Ioannou and C.C. Chien. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):657–672, 1993.

[11] F. Kerber and A. van der Schaft. Compositional analysis for linear systems. *Systems & Control Letters*, 59(10):645–653, 2010.

[12] E.S. Kim, M. Arcak, and S.A. Seshia. A small gain theorem for parametric assume-guarantee contracts. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control, Pittsburgh, USA*, pages 207–216, 2017.

[13] N.Y. Megawati and A. van der Schaft. Bisimulation equivalence of differential-algebraic systems. *International Journal of Control*, pages 1–11, 2016.

[14] A. Megretski and A. Rantzer. System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control*, 42(6):819–830, 1997.

[15] B. Meyer. Applying "design by contract". *Computer*, 25(10):40–51, 1992.

[16] G.J. Pappas. Bisimilar linear systems. *Automatica*, 39(12):2035–2047, 2003.

[17] A. Sangiovanni-Vincentelli, W. Damm, and R. Passerone. Taming Dr. Frankenstein: Contract-based design for cyber-physical systems. *European Journal of Control*, 18(3):217–238, 2012.

[18] A. Saoud, G. Girard, and L. Fribourg. On the composition of discrete and continuous-time assume-guarantee contracts for invariance. In *Proceedings of the 16th European Control Conference, Limassol, Cyprus*, pages 435–440, 2018.

[19] A.J. van der Schaft. Equivalence of dynamical systems by bisimulation. *IEEE Transactions on Automatic Control*, 49(12):2160–2172, 2004.

[20] A.J. van der Schaft. *L_2-gain and passivity techniques in nonlinear control*. Communications and Control Engineering Series. Springer International Publishing, Cham, Switzerland, third edition, 2017.

[21] R. Sepulchre, M. Janković, and P. Kokotović. *Constructive nonlinear control*. Communications and Control Engineering Series. Springer-Verlag, London, Great Britain, 1997.

[22] A. van der Schaft. Equivalence of hybrid dynamical systems. In *Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems, Leuven, Belgium*, 2004.

[23] P. Tabuada. *Verification and control of hybrid systems: A symbolic approach*. Springer, New York, USA, 2009.

[24] H.L. Trentelman, A.A. Stoorvogel, and M. Hautus. *Control theory for linear systems*. Communications and Control Engineering. Springer-Verlag, London, Great Britain, 2001.

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

[26] J.C. Willems. Input-output and state-space representations of finite-dimensional linear time-invariant systems. *Linear Algebra and its Applications*, 50:581–608, 1983.

[27] J.C. Willems. The behavioral approach to open and interconnected systems. *IEEE Control Systems Magazine*, 27(6):46–99, 2007.

[28] W.M. Wonham. *Linear multivariable control: a geometric approach*. Springer-Verlag, New York, USA, second edition, 1979.