Inclusion of Frequency Domain Behaviors

Stephen Prajna
Control and Dynamical Systems
California Inst. of Technology
Pasadena, CA 91125 - USA

Pablo A. Parrilo
Automatic Control Laboratory
Swiss Federal Inst. of Technology
CH-8092, Zürich - Switzerland

Abstract

This paper addresses inclusion of behaviors and its verification. It is shown that verifying inclusion of frequency domain behaviors defined by polynomial frequency domain equalities and inequalities amounts to proving emptiness of some basic semialgebraic sets. A semidefinite programming relaxation method for solving this problem is outlined. Some applications are given to illustrate the use of the concepts.

1 Introduction

There has been a widespread research on behavioral systems theory in the recent years. As one of the paradigms in systems and control, the behavioral approach puts the emphasis on behaviors of systems, i.e., collections of possible trajectories. This is in contrast to operator theoretic approaches, which treat systems as signal processors with clear partition between input and output signals. The behavioral approach is for example treated in the seminal paper [7], and also in [3].

The significance of behavior inclusion has been recognized as early as [6], in the context of most powerful unfalsified models. It has also appeared recently in relation to implementable behaviors [8]. One behavior is included in another behavior if all possible trajectories of the former are also possible trajectories of the latter. Conditions for some special cases of behavior inclusion have been presented in [4], for behaviors that admits rational kernel representations, and also in [5], in relation to the H_∞ optimal control problem.

In the present paper, we consider behaviors that can be described by polynomial frequency domain equalities and inequalities, with possibly some parametric dependence. Through the use of inequalities and parameters, it is possible to describe behaviors of systems with parametric and linear time invariant dynamic uncertainty. Our results show that verifying behavior inclusion can be performed by proving that some basic semialgebraic sets are empty. The semidefinite programming relaxation methods introduced in [2] can be used for this purpose. This paper is concluded by application examples on robust controller verification, and model verification and selection. In particular, we elucidate on the behavior inclusion interpretations of these problems.

2 On Frequency Domain Behaviors

Consider a linear time invariant differential system, whose behavior restricted to the L_2-space is described by

$$\mathcal{B} = \{w \in L_2^n(\mathbb{R}) \mid R(\frac{d}{dt})w(t) = 0\},$$

where $L_2^n(\mathbb{R})$ denotes the space of functions mapping $\mathbb{R}$ to $\mathbb{C}^n$ that are square integrable, $R[\xi]$ is an $m \times n$ matrix of polynomials in the indeterminate ξ, and w is assumed to satisfy the differential equations above in the weak sense (cf. [3, Chapter 2]). For any $w \in \mathcal{B}$, the Fourier transform $\hat{w} \in \hat{L}_2^n(j\mathbb{R})$ exists. Moreover, it satisfies

$$R(j\omega)\hat{w}(j\omega) = 0$$

for all $\omega \in \mathbb{R}$. Therefore, to the same system we can associate the frequency domain behavior

$$\hat{\mathcal{B}} = \{\hat{w} \in \hat{L}_2^n(j\mathbb{R}) \mid R(j\omega)\hat{w}(j\omega) = 0\}.$$

By considering behaviors on $L_2^n(\mathbb{R})$ and $\hat{L}_2^n(j\mathbb{R})$, we get the following property: for any $w \in \mathcal{B}$, there exists a corresponding $\hat{w} \in \hat{\mathcal{B}}$, which can be obtained using the Fourier transform; and conversely for any $\hat{w} \in \hat{\mathcal{B}}$ we can use the inverse Fourier transform to obtain the corresponding $w \in \mathcal{B}$. We have an isomorphism between the time domain and the frequency domain behaviors, and therefore inclusion of time domain behaviors is equivalent to inclusion of frequency domain behaviors.

Remark 2.1. *Implicit in the definition of frequency domain behavior is the assumption that two functions are considered the same, if they only differ on a set of measure zero. This is due to the fact that when we apply the inverse Fourier transform to such functions, we will get the same time domain signal. Similar thing applies to the definition of time domain behavior as well.*

Next, we step further by adding parameters and frequency domain inequalities in the definition of behavior, i.e., we consider behaviors defined by

$$\hat{\mathcal{B}}(a) = \{\hat{w} \in \hat{L}_2^n(j\mathbb{R}) \mid R(j\omega, a)\hat{w}(j\omega) = 0, \;\; \hat{w}(j\omega)^*\Pi_i(j\omega, a)\hat{w}(j\omega) \geq 0, \;\; i = 1, 2, ..., k\},$$

where the parameters $a \in \mathbb{R}^p$ are assumed to be in a *basic semialgebraic set* A, whereas $R(j\omega, a)$, $\Pi_i(j\omega, a)$ are polynomial matrices in $(j\omega, a)$, and in addition $\Pi_i(j\omega, a) = \Pi_i(j\omega, a)^*$ for all $i = 1, ..., k$. By basic semialgebraic set we mean a set that is described by polynomial equalities and inequalities

$$A = \{a \in \mathbb{R}^p \;\mid\; p_{i_1}(a) = 0, p_{i_2}(a) \geq 0, p_{i_3}(a) \neq 0, \text{ where } p_{i_1}, p_{i_2}, p_{i_3} \text{ are real scalar polynomials in } a;\; i_1 = 1, ..., \ell_1;\; i_2 = 1, ..., \ell_2;\; i_3 = 1, ..., \ell_3\}. \tag{2.1}$$

Our rationale for adding parameters and frequency domain inequalities is as follows:

- the dependence of the behavior on a may represent parametric uncertainties in the model.
- the frequency domain inequalities $\hat{w}(j\omega)^*\Pi_i(j\omega,a)\hat{w}(j\omega) \geq 0$ can be used for representing dynamic uncertainties in our model, or characterizing some signal properties. As we will see in Section 4, this inequality may also represent some performance specifications for robust performance analysis.

Given two behaviors $\hat{\mathcal{B}}^1(a)$, $\hat{\mathcal{B}}^2(a)$ and their respective R^1, Π_i^1, R^2, Π_i^2, the problem of interest is to check if we have the behavior inclusion

$$\hat{\mathcal{B}}^1(a) \subseteq \hat{\mathcal{B}}^2(a) \qquad \forall a \in A. \tag{2.2}$$

In most cases, we will be interested in $\hat{\mathcal{B}}^2$ that does not depend on a. In this case, the condition above will be equivalent to $\hat{\mathcal{B}}^1 \subseteq \hat{\mathcal{B}}^2$, where $\hat{\mathcal{B}}^1$, which is given by

$$\hat{\mathcal{B}}^1 = \bigcup_{a\in A} \hat{\mathcal{B}}^1(a) = \{\hat{w} \in \hat{L}_2^n(j\mathbb{R}) \;\; | \;\; \exists a \in A \text{ such that } R^1(j\omega,a)\hat{w}(j\omega) = 0,$$
$$\hat{w}(j\omega)^*\Pi_i^1(j\omega,a)\hat{w}(j\omega) \geq 0\},$$

may represent the aggregate behavior of an uncertain system.

At this point we want to emphasize that the signal spaces in the behaviors we are considering must be the same. Not only that, the signals in one behavior must represent the same physical quantities as the signals in the other behavior. For example, it does not make sense to compare *full behaviors* $\hat{\mathcal{B}}^{1,f}(a)$ and $\hat{\mathcal{B}}^{2,f}(a)$ of systems with *latent variables*, where the latent variables in one behavior are not compatible with the latent variables in the other, even if their manifest variables represent the same physical quantities. In this case, we should first project the full behaviors to the manifest variable space in order to obtain the manifest behaviors $\hat{\mathcal{B}}^{1,m}(a)$ and $\hat{\mathcal{B}}^{2,m}(a)$, and only after that can we check inclusion of the manifest behaviors $\hat{\mathcal{B}}^{1,m}(a) \subseteq \hat{\mathcal{B}}^{2,m}(a)$. The projection is done e.g. by elimination of latent variables. In general, checking inclusion of projections of two sets is a harder problem, even on finite dimensional spaces. In this paper we will not deal with that issue, and we only consider behaviors whose latent variables can be easily eliminated.

Remark 2.2. *It is actually sufficient to perform the elimination of latent variables as mentioned above only on the second behavior $\hat{\mathcal{B}}^{2,f}(a)$, to obtain $\hat{\mathcal{B}}^{2,m}(a)$. This manifest behavior can then be lifted to the signal space of $\hat{\mathcal{B}}^{1,f}(a)$, and we can verify the behavior inclusion $\hat{\mathcal{B}}^{1,f}(a) \subseteq \hat{\mathcal{B}}^{2,m,\ell}(a)$, where $\hat{\mathcal{B}}^{2,m,\ell}(a)$ is the result of the lifting. If the inclusion holds, then inclusion of the manifest behaviors is also true, because $\hat{\mathcal{B}}^{1,m}(a) \subseteq \hat{\mathcal{B}}^{2,m}(a)$ iff $\hat{\mathcal{B}}^{1,f}(a) \subseteq \hat{\mathcal{B}}^{2,m,\ell}(a)$. See also Section 4.1 (e.g. equations (4.10) and (4.11)) and Section 4.2.*

Now, with respect to $\hat{\mathcal{B}}^1(a)$ in (2.2), we associate the following set:

$$B^1 = \{(z,\omega,a) \in \mathbb{C}^m \times \mathbb{R}^{p+1} \mid a \in A, \;\; R^1(j\omega,a)z = 0, \;\; z^*\Pi_i^1(j\omega,a)z \geq 0, \;\; i = 1,2,...,k_1\},$$

and similarly with $\hat{\mathcal{B}}^2(a)$,

$$B^2 = \{(z,\omega,a) \in \mathbb{C}^m \times \mathbb{R}^{p+1} \mid a \in A,\;\; R^2(j\omega,a)z = 0,\;\; z^*\Pi_i^2(j\omega,a)z \geq 0,\;\; i = 1,2,...,k_2\}.$$

Then we have the following result.

Proposition 2.1. *Inclusion (2.2) holds if*

$$B^1 \subseteq B^2. \tag{2.3}$$

Proof. We will show that $\hat{\mathcal{B}}^1(a) \nsubseteq \hat{\mathcal{B}}^2(a)$ for some a implies $B_1 \nsubseteq B_2$. Assume that there exists $a \in A$ and $\hat{w} \in \hat{L}_2^n(j\mathbb{R})$ such that $\hat{w} \in \hat{\mathcal{B}}^1(a)$ but $\hat{w} \notin \hat{\mathcal{B}}^2(a)$. Then there exists $W \subseteq \mathbb{R}$, such that for any $\omega \in W$, $\hat{w}(j\omega)$ does not satisfy at least one of the equalities and inequalities that define $\hat{\mathcal{B}}^2(a)$. On the other hand, since $\hat{w} \in \hat{\mathcal{B}}^1(a)$, $\hat{w}(j\omega)$ will satisfy the defining equalities and inequalities of $\hat{\mathcal{B}}^1(a)$ for all $\omega \in W$. Now choose any $\omega_0 \in W$ and let $z = \hat{w}(j\omega_0)$. Then $(z,\omega_0,a) \in B_1$ but $(z,\omega_0,a) \notin B_2$, thus $B_1 \nsubseteq B_2$. □

We would like to note that in general (2.3) is not a necessary condition for (2.2). This is shown by a simple example below.

Example 2.1. *Consider the following behaviors:*

$$\begin{aligned}
\hat{\mathcal{B}}^1 &= \{\hat{w} \in \hat{L}_2^1(j\mathbb{R}) \mid (j\omega)\hat{w}(j\omega) = 0\} \\
\hat{\mathcal{B}}^2 &= \{\hat{w} \in \hat{L}_2^1(j\mathbb{R}) \mid \hat{w}(j\omega) = 0\}
\end{aligned}$$

(which corresponds to $\frac{d}{dt}w(t) = 0$ and $w(t) = 0$). It is clear that $B^1 = \{(z,\omega) \mid (j\omega)z = 0\} \nsubseteq B^2 = \{(z,\omega) \mid z = 0\}$, yet $\hat{\mathcal{B}}^1 \subseteq \hat{\mathcal{B}}^2$ since $\hat{w}(j\omega) \in \hat{\mathcal{B}}^1$ can only differ from zero at $\omega = 0$, and is therefore in the same equivalence class as the zero function (cf. Remark 2.1). In fact, the time domain behaviors of $\frac{d}{dt}w(t) = 0$ and $w(t) = 0$ restricted to the L_2-space are the same.

However, in many cases such degeneracy does not occur, and (2.3) will be a necessary condition for (2.2). This happens for example when $\hat{\mathcal{B}}^1$ is defined only by frequency domain equalities and $R^1(j\omega,a)$ has full rank for all ω and a.

3 Proving Set Inclusion

It has been shown in the previous section that verifying behavior inclusion in the frequency domain (2.2) amounts to proving set inclusion (2.3). By imposing the requirements that A is a basic semialgebraic set and the R's and Π's are polynomials, the problem reduces to the problem of proving inclusion of basic semialgebraic sets, for which deterministic algorithmic methods exist. Among them are quantifier elimination based methods and a recently developed semidefinite programming relaxation approach [2]. In this section we will focus on the latter approach.

By writing the complex vector z as $z = z_R + j\, z_I$ and expressing the real and imaginary parts of $R^1(j\omega, a)z = 0$, $R^2(j\omega, a)z = 0$ separately, it is clear that B^1 and B^2 are isomorphic via a common mapping to two subsets of $\mathbb{R}^{2m+p+1}$ that are described by scalar polynomial equalities and inequalities. With a slight abuse of notation, let us also denote these subsets by B^1, B^2, and assume that they are given by

$$\begin{aligned} B^1 &= \{(z_R, z_I, \omega, a) \mid \tilde{R}^1_{j_1}(z_R, z_I, \omega, a) = 0, \tilde{\Pi}^1_{j_2}(z_R, z_I, \omega, a) \geq 0, p_{i_1}(a) = 0, p_{i_2}(a) \geq 0, \\ &\qquad p_{i_3}(a) \neq 0, \quad \forall j_1, j_2, i_1, i_2, i_3\}, \\ B^2 &= \{(z_R, z_I, \omega, a) \mid \tilde{R}^2_{j_3}(z_R, z_I, \omega, a) = 0, \tilde{\Pi}^2_{j_4}(z_R, z_I, \omega, a) \geq 0, p_{i_1}(a) = 0, p_{i_2}(a) \geq 0, \\ &\qquad p_{i_3}(a) \neq 0, \quad \forall j_3, j_4, i_1, i_2, i_3\}, \end{aligned}$$

where $\tilde{R}^1_{j_1}$, $\tilde{\Pi}^1_{j_2}$, $\tilde{R}^2_{j_3}$, and $\tilde{\Pi}^2_{j_4}$ are real scalar polynomials, and $p_{i_1}(a) = 0$, $p_{i_2}(a) \geq 0$, $p_{i_3}(a) \neq 0$ are from (2.1). Although not written explicitly here, we assume that we keep track of the indices.

Before presenting a sufficient and necessary condition for $B^1 \subseteq B^2$, let us introduce the following sets. For each j_3, define F_{j_3} as

$$F_{j_3} = B^1 \cap \{(z_R, z_I, \omega, a) \mid \tilde{R}^2_{j_3}(z_R, z_I, \omega, a) \neq 0\},$$

and similarly for each j_4, define G_{j_4} as

$$G_{j_4} = B^1 \cap \{(z_R, z_I, \omega, a) \mid \tilde{\Pi}^2_{j_4}(z_R, z_I, \omega, a) < 0\}.$$

Using these notations, we get the following result.

Proposition 3.1. *$B^1 \subseteq B^2$ iff*

$$\begin{aligned} F_{j_3} &= \emptyset, \\ G_{j_4} &= \emptyset, \end{aligned}$$

for all j_3, j_4.

Proof. We leave it to the reader to verify that

$$B^1 \cap \overline{B^2} = (\bigcup_{j_3} F_{j_3}) \cup (\bigcup_{j_4} G_{j_4}).$$

The proposition follows immediately since $B^1 \subseteq B^2$ iff $B^1 \cap \overline{B^2} = \emptyset$. □

The previous proposition indicates clearly the importance of set emptiness verification. Now we will give a brief outline of a semidefinite programming based relaxation method for checking if a basic semialgebraic set is empty. Readers are referred to [2] and references therein for a more thorough account. From this point onward we assume that $x \in \mathbb{R}^n$ and that all polynomials are real. First, we have the following definition of sum of squares.

Definition 3.1. *A polynomial* $p(x)$ *is a* sum of squares *if it can be written as* $p(x) = \sum q_i(x)^2$ *for some polynomials* $q_i(x)$.

Testing if a polynomial is a sum of squares can be performed using semidefinite programming and the so-called "Gram matrix" method. In particular, write the polynomial as $p(x) = \mathbf{q}(x)^T Z \mathbf{q}(x)$, where $\mathbf{q}(x)$ is some appropriately chosen set of monomials in x. If Z is positive semidefinite, then $p(x)$ is a sum of squares. Thus checking if a polynomial is a sum of squares amounts to finding $Z \geq 0$ that at the same time satisfies the equation above. This is equivalent to the feasibility problem of a linear matrix inequality (LMI) with affine constraints.
Next, from polynomial algebra we have the subsequent definitions.

Definition 3.2. *Given a finite set of polynomials* $\{p_i(x)\}$, *the* ideal *generated by* $\{p_i(x)\}$, *which is denoted by* $I(p_i)$, *is*

$$I(p_i) = \left\{ \sum_i a_i p_i \mid a_i \text{ are polynomials for all } i \right\}$$

Definition 3.3. *Given a finite set of polynomials* $\{p_i(x)\}$, *the* multiplicative monoid *generated by* $\{p_i(x)\}$, *which is denoted by* $M(p_i)$, *is the set of finite products of elements* p_i, *including the empty product (the identity).*

The following is an equivalent characterization of a *cone* generated by a finite set of polynomials in the polynomial ring.

Definition 3.4. *Given a finite set of polynomials* $\{p_i(x)\}$, *the* cone *generated by* $\{p_i(x)\}$, *which is denoted by* $P(p_i)$, *is*

$$P(p_i) = \left\{ a + \sum_{j=1}^{k} b_j q_j \mid a, b_j \text{ are sums of squares, } q_j \in M(p_i) \text{ for } j = 1, ..., k \right\}$$

All these definitions are used in the Positivstellensatz, a central result from real algebraic geometry. The theorem provides a characterization of *infeasibility certificates* for real solutions of systems of polynomial equalities and inequalities.

Theorem 3.1 (Positivstellensatz). *Let* f_j, g_k, h_ℓ *be finite sets of polynomials in* x. *Then the following properties are equivalent:*

1. *The set*
$$\{x \in \mathbb{R}^n \mid f_j(x) \geq 0, g_k(x) \neq 0, h_\ell(x) = 0, \ \ \forall j, k, \ell\} \tag{3.4}$$
is empty.

2. *There exist* $f \in P(f_j)$, $g \in M(g_k)$, $h \in I(h_\ell)$ *such that*
$$f + g^2 + h = 0. \tag{3.5}$$

It has been recently shown that Positivstellensatz refutations (i.e., f, g, h that satisfy (3.5)) can be computed using hierarchies of semidefinite programming. The idea is to choose a degree bound for the polynomials, and then affinely parameterize a family of candidate f and h. This converts the problem into the feasibility problem of some LMIs with affine constraints. This is summarized by the following theorem.

Theorem 3.2 ([2]). *Consider a basic semialgebraic set of the form (3.4). Then the search for bounded degree Positivstellensatz refutations can be done using semidefinite programming. If the set is empty and the degree bound is chosen to be large enough, then the semidefinite programs will be feasible, and the refutations can be obtained from its solution.*

The crucial property that will allow us to use this results in the behavior inclusion problem is the fact that F_{j_3} and G_{j_4} in Proposition 3.1 are basic semialgebraic sets of the form (3.4). Thus the test for their emptiness can be performed by a hierarchy of semidefinite programming relaxations.

Remark 3.1. *To fully understand our set inclusion test, it is instructive to consider the simple case of sets defined only by linear inequalities, i.e., polyhedral sets. It is well known that containment of polyhedra can be verified by solving a finite number of linear programming problems. These linear programs are* exactly *what our approach would provide as a first order relaxation, with the convexity properties of polyhedra guaranteeing that this first order relaxation actually provides the exact solution.*

4 Applications

In this section we provide some examples of problems in control theory that can be interpreted as behavior inclusion problems. Some concepts from the input-output paradigm are still present in our discussion, because the original problems are cast in that setting. Yet the behavior inclusion interpretations are free from those concepts. Readers are referred to [1] for the background of the problems.

4.1 Verification of Robust Controllers

Suppose that we have an uncertain system with parametric and dynamic linear time invariant (LTI) uncertainties as shown in Figure 1. The parametric uncertainty is represented by the parameters $a \in A$ in the description of the LTI differential system G, whereas the dynamic uncertainty is represented by $\Delta \in \mathbf{\Delta}$. The system G defines a relation between the signals as follows:

$$G(j\omega, a) \left[\hat{p}(j\omega) \;\; \hat{q}(j\omega) \;\; \hat{u}(j\omega) \;\; \hat{y}(j\omega) \;\; \hat{w}(j\omega) \;\; \hat{z}(j\omega)\right]^T = 0. \tag{4.6}$$

We also assume that the uncertainty set $\mathbf{\Delta}$ defines the following relation between p and q:

$$\begin{bmatrix} \hat{p}(j\omega) \\ \hat{q}(j\omega) \end{bmatrix}^* \Pi_\Delta(j\omega) \begin{bmatrix} \hat{p}(j\omega) \\ \hat{q}(j\omega) \end{bmatrix} \geq 0. \tag{4.7}$$

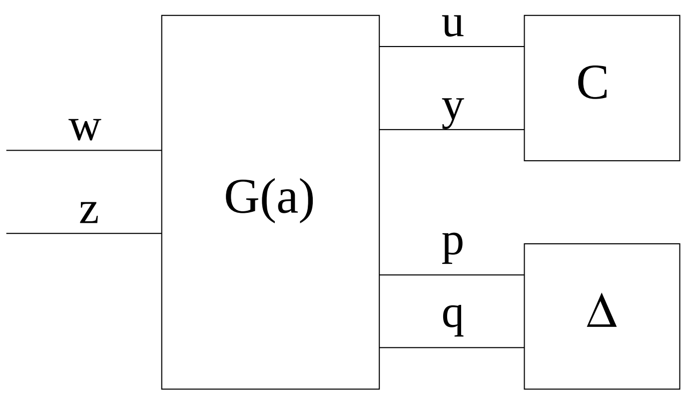

Figure 1: Controller verification problem. The figure depicts the to-be-controlled system G with parametric uncertainty $a \in A$ and dynamic linear time invariant uncertainty $\Delta \in \mathbf{\Delta}$, in interconnection with the controller C. It is of interest to know whether or not the controlled system satisfies some given performance specifications on (w, z).

For example, with n and d being scalar polynomials, $\Pi_\Delta(j\omega)$ can be

$$\Pi_\Delta(j\omega) = \begin{bmatrix} |n(j\omega)|^2 I & 0 \\ 0 & -|d(j\omega)|^2 I \end{bmatrix},$$

which is used to specify a frequency weighted norm bound

$$\|\hat{q}(j\omega)\| \leq \left| \frac{n(j\omega)}{d(j\omega)} \right| \|\hat{p}(j\omega)\|, \qquad \forall \omega \in \mathbb{R}.$$

The controller verification problem can now be stated as follows. Suppose that a controller C has been designed for this system, which relates u and y via

$$C(j\omega) \begin{bmatrix} \hat{u}(j\omega) \\ \hat{y}(j\omega) \end{bmatrix} = 0. \tag{4.8}$$

We would like to know if in the presence of uncertainties, the controlled system satisfies the performance specification Π_P, i.e., if the quadratic form

$$\begin{bmatrix} \hat{w}(j\omega) \\ \hat{z}(j\omega) \end{bmatrix}^* \Pi_P(j\omega) \begin{bmatrix} \hat{w}(j\omega) \\ \hat{z}(j\omega) \end{bmatrix} \geq 0. \tag{4.9}$$

is satisfied. Different specifications are stated by choosing different Π_P. For example,

$$\Pi_P = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$$

is used for the specification that the L_2-gain from w to z is not greater than one.

We will now show how the behavior inclusion interpretation is brought into picture. Define the first behavior, $\hat{\mathcal{B}}^1$, as

$$\hat{\mathcal{B}}^1 = \bigcup_{a \in A} \{(\hat{p}, \hat{q}, \hat{u}, \hat{y}, \hat{w}, \hat{z}) \in \hat{L}_2^n(j\mathbb{R}) \mid (4.6),\ (4.7),\ (4.8) \text{ are satisfied}\}. \tag{4.10}$$

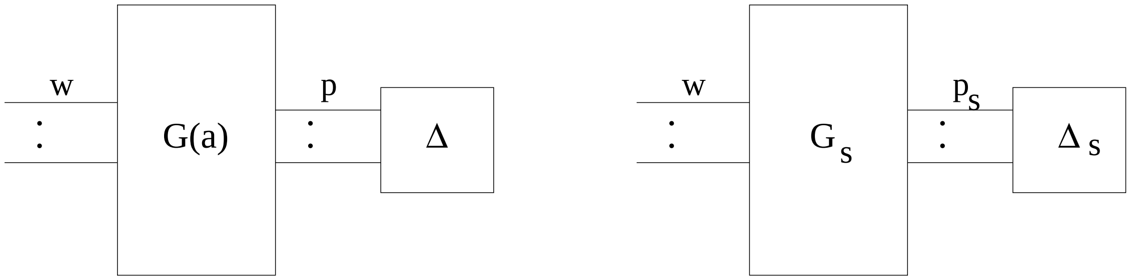

Figure 2: Model verification problem. The model on the left is the original model, which has the manifest behavior $\hat{\mathcal{B}}^m$. The model on the right is the simplified model, with the manifest behavior $\hat{\mathcal{B}}^m_s$. We want to check if $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}^m_s$.

Next, use the performance specification (4.9) to define the second behavior. By its own, this frequency domain inequality defines a behavior in the $(\hat{w}, \hat{z})$-space. However, we can easily lift it to the same space we use for $\hat{\mathcal{B}}^1$. Namely, we define

$$\hat{\mathcal{B}}^2 = \{(\hat{p}, \hat{q}, \hat{u}, \hat{y}, \hat{w}, \hat{z}) \in \hat{L}_2^n(j\mathbb{R}) \mid (4.9) \text{ is satisfied}\}. \tag{4.11}$$

Then the controller verification problem can be interpreted as the problem of testing whether or not $\hat{\mathcal{B}}^1 \subseteq \hat{\mathcal{B}}^2$. The machineries presented in the previous sections can be used to verify this inclusion. It is obvious that we can also generalize the problem to the case where there are more than one quadratic forms used in the uncertainty descriptions or performance specifications.

A related but harder problem is the controller synthesis problem: design a controller C such that the behavior of the controlled system $\hat{\mathcal{B}}^1$ is contained in $\hat{\mathcal{B}}^2$. With uncertainties present in the system, this problem in nonbehavioral settings has lead to the μ-synthesis problem, which is solved heuristically using the so-called D-K iterations. However, a solution to the synthesis problem for systems with uncertainties is still far from obvious, and the applicability of the methods in the previous sections to this problem still needs to be investigated.

4.2 Model Verification and Selection

Suppose that we have an LTI system with parametric and dynamic uncertainties as shown in Figure 2. This defines the "original" behavior

$$\hat{\mathcal{B}} = \bigcup_{a \in A} \{(\hat{p}, \hat{w}) \in \hat{L}_2^{n_p+n_w}(j\mathbb{R}) \;\mid\; G(j\omega, a) \begin{bmatrix} \hat{p}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix} = 0, \; \begin{bmatrix} \hat{p}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix}^* \Pi_i \begin{bmatrix} \hat{p}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix} \geq 0, \; i = 1, 2, ..., m\}, \tag{4.12}$$

with its corresponding manifest behavior

$$\hat{\mathcal{B}}^m = \{\hat{w} \in \hat{L}_2^{n_w}(j\mathbb{R}) \mid \exists \hat{p} \in \hat{L}_2^{n_p}(j\mathbb{R}) \text{ such that } (\hat{p}, \hat{w}) \in \hat{\mathcal{B}}\}. \tag{4.13}$$

Assume further that by some means, a "simplified" behavior

$$\hat{\mathcal{B}}_s = \{(\hat{p}_s, \hat{w}) \in \hat{L}_2^{n_{p_s}+n_w}(j\mathbb{R}) \mid G_s(j\omega)\begin{bmatrix}\hat{p}_s(j\omega)\\ \hat{w}(j\omega)\end{bmatrix} = 0,\ \begin{bmatrix}\hat{p}_s(j\omega)\\ \hat{w}(j\omega)\end{bmatrix}^* \Pi_i^s \begin{bmatrix}\hat{p}_s(j\omega)\\ \hat{w}(j\omega)\end{bmatrix} \geq 0,\ i = 1, 2, ..., m_s\}, \tag{4.14}$$

with its corresponding manifest behavior $\hat{\mathcal{B}}_s^m$ have been given to us. Simple description here is characterized by the use of low order differential equations, simple dynamic uncertainty description, no parametric uncertainty, etc. The question in the model verification problem can now be stated as follows: Is the second behavior a good representation of the first behavior, in the sense that $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}_s^m$?

The aforementioned question is of importance in control theory, because for the purpose of analysis and synthesis the original model corresponding to $\hat{\mathcal{B}}^m$ is often replaced by another model $\hat{\mathcal{B}}_s^m$ that is more tractable. Then one would very much like to have $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}_s^m$. Implications of this inclusion are quite obvious. For example, a controller that is synthesized for $\hat{\mathcal{B}}_s^m$ to meet some design specifications will also satisfy the same specifications for $\hat{\mathcal{B}}^m$, since $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}_s^m$.

When two behaviors $\hat{\mathcal{B}}_{s1}^m$ and $\hat{\mathcal{B}}_{s2}^m$ of the same complexity class are given, it is also natural to ask if one of them is more powerful than the other. This is the essence of the model selection problem. The first behavior is more powerful than the other if the inclusion $\hat{\mathcal{B}}_{s1}^m \subseteq \hat{\mathcal{B}}_{s2}^m$ is satisfied, in addition to $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}_{s1}^m$. Certainly a more powerful model is desirable, since its use will reduce the conservatism in system analysis or design.

The questions of model verification and selection as mentioned above can be addressed using the methods presented in this paper. However, as mentioned in Remark 2.2 and the paragraph preceding it, the inclusion can be verified provided the latent variables $\hat{p}_s$ can be eliminated from $\hat{\mathcal{B}}_s$ to obtain $\hat{\mathcal{B}}_s^m$. If that holds, we can lift $\hat{\mathcal{B}}_s^m$ to the $(\hat{p}, \hat{w})$-space and then check if $\hat{\mathcal{B}} \subseteq \hat{\mathcal{B}}_s^\ell$, where $\hat{\mathcal{B}}_s^\ell$ is the result of the lifting. This is similar to what we did in Subsection 4.1 (cf. (4.11)).

Related to these problems is the model reduction problem, which can be interpreted as the problem of finding a behavior with simple representation $\hat{\mathcal{B}}_s^m$ for a given behavior $\hat{\mathcal{B}}^m$, such that $\hat{\mathcal{B}}^m \subseteq \hat{\mathcal{B}}_s^m$. Unfortunately this is a harder problem, presumably in the same complexity class as the controller synthesis. Although many results for systems without and with uncertainties have been reported in the literature, the solution to this problem is still far from complete. Similar to the case of the controller synthesis problem, the applicability of the methods presented in this paper to the model reduction problem is still a subject of future research.

5 Conclusions

Issues related to behavior inclusion in the frequency domain have been addressed in this paper. Parametric dependence and frequency domain inequalities, in addition to frequency domain equalities, are used for defining behaviors. This enables us for example to describe behaviors of uncertain systems, and to define criteria for robust performance analysis.

It is further shown that verifying inclusion of behaviors in the frequency domain amounts to proving inclusion of basic semialgebraic sets. In turn, this is equivalent to the problem of proving that some basic semialgebraic sets are empty, for which a semidefinite programming relaxation method exists.

Two application examples, taken from the robust control and modelling domains, have been given to illustrate the relevance of the behavioral inclusion viewpoint. While the analysis questions are tractable using the machineries presented here, further research is still needed on synthesis related questions.

Acknowledgement

The work of the first author was supported by Caltech's Lee Center for Advanced Networking.

References

[1] G. E. Dullerud and F. Paganini. *A Course in Robust Control Theory: A Convex Approach*. Springer-Verlag NY, 2000.

[2] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. Preprint submitted to *Mathematical Programming*. Downloadable from `http://www.cds.caltech.edu/~pablo`, 2001.

[3] J. W. Polderman and J. C. Willems. *Introduction to Mathematical Systems Theory: A Behavioral Approach*. Springer-Verlag NY, 1997.

[4] S. Weiland and A. A. Stoorvogel. Rational representations of behaviors: Interconnectability and stabilizability. *Math. Control Signals Systems*, 10:125–164, 1997.

[5] S. Weiland, A. A. Stoorvogel, and B. de Jager. A behavioral approach to the H_∞ optimal control problem. *Systems Control Lett.*, 32:323–334, 1997.

[6] J. C. Willems. From time series to linear system. Part II: Exact modelling. *Automatica*, 22:675–694, 1986.

[7] J. C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Automatic Control*, 36:259–294, 1991.

[8] J. C. Willems and H. L. Trentelman. Synthesis of dissipative systems using quadratic differential forms: Part I. *IEEE Trans. Automatic Control*, 47:53–69, 2002.