# Sensitivity shaping with degree constraint by nonlinear least-squares optimization*

Ryozo Nagamune† and Anders Blomqvist‡

June 27, 2004

**Abstract**

This paper presents a new approach to shaping of the frequency response of the sensitivity function. In this approach, a desired frequency response is assumed to be specified at a finite number of frequency points. A sensitivity shaping problem is formulated as an approximation problem to the desired frequency response with a function in a class of sensitivity functions with a degree bound. The sensitivity shaping problem is reduced to a finite dimensional constrained nonlinear least-squares optimization problem. The reduction process involves the diffeomorphism from the set of all denominators for strictly positive-real functions with degree constraint to the set of Schur polynomials, firstly proven by Byrnes et al. To solve the optimization problem numerically, standard algorithms for an unconstrained version of nonlinear least-squares problems are modified to incorporate the constraint. Since the optimization problem is nonconvex, sensible selection of the initial point for the algorithms is crucial. Some rules of thumb for such selections are suggested. Based on the optimization problem and the algorithms to solve it, a controller design procedure is proposed. Numerical examples illustrate how these design parameters are tuned in an intuitive manner, as well as how the design proceeds in actual control problems.

**Key Words:** Sensitivity function, shaping, nonlinear least-squares optimization, degree bound, Schur polynomial

## 1 Introduction

It is well-known that the *sensitivity function*, denoted by $S$, is one of the essential factors in determining performances of feedback systems, such as robust stability, noise/disturbance attenuation, and tracking. It has been recognized since the classical control era that sensible control design can be accomplished by designing $S$ appropriately. Thus, it is significant to develop systematic design tools for $S$.

*This work was partially supported by grants from the Swedish Research Council (VR).

†Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA. Email: ryozo@me.berkeley.edu.

‡Division of Optimization and Systems Theory, Royal Institute of Technology, SE-100 44 Stockholm, Sweden. Email: andersb@math.kth.se.

Much effort has been made for such development, e.g., classical control methodologies such as PID-based control and lead-lag compensations [22], both open-loop [24] and closed-loop shaping techniques in $H^\infty$ control (e.g., [11]), an approach based on positive polynomials [21, 20], to name a few. However, these previous tools heavily require designers' engineering experience, knowledge and intuition in manual selection of design parameters such as controller parameters and weighting functions. Even for experienced designers, the manual selection involves trial and error, which is by no means an easy task.

In [6], a new paradigm is suggested for sensitivity shaping without weighting functions in an $H^\infty$ control framework, and it is further developed in [29, 26]. The paradigm is based on analytic interpolation theory with degree constraint initiated in [14, 15] and carried to completion in [9, 7, 6]. In this paradigm, design parameters are *spectral zeros* (or equivalently, *Schur polynomials* which will be used in this paper) and additional interpolation conditions. We have illustrated through numerical examples that the approach in [26] often generates controllers of lower degrees than conventional $H^\infty$ controller design does. (See also [5, 4, 2] for such examples.) However, only guidelines have been provided for the tuning of spectral zeros in [26], and it would be convenient to have a method for determining these parameters in a certain optimal sense. This is the motivation of this paper.

In this paper, for scalar systems, we shall propose a new method to design $S$ in the frequency domain. We will formulate a sensitivity shaping problem as an approximation problem, for a function in a class of $S$ with a bounded degree, relative to a desired frequency response given at a finite number of frequency points. The problem can be reduced to a finite dimensional constrained nonlinear least-squares (NLS) optimization problem. To solve the NLS problem numerically, we will use algorithms which are modifications of standard algorithms originally developed for unconstrained NLS optimization. Since the optimization problem is nonconvex, sensible selection of the initial point for the algorithms is crucial. Some rules of thumb for such selection are suggested. Based on the optimization problem and the algorithms to solve it, a controller design procedure is proposed, and tunable design parameters in this procedure are listed. Although trial-and-error process is necessary for choosing appropriate design parameters even in our approach, we believe that the way of selecting and tuning design parameters is more intuitive than that in previous approaches. This point will be illustrated through control design examples.

In addition to the advantage of intuitive design, another important advantage of our approach over the conventional $H^\infty$ methodology, including the LMI-based approach [23, 13], is as follows. To shape the frequency response, we will not rely on weighting functions which typically cause the increase of controller degrees. In fact, although we will introduce some "weights" which plays a similar role to weighting functions, the weights do *not* affect controller degrees. Also, the weights in our approach do not assume any rationality, while the weighting functions should be rational in most cases. The lack of rationality requirement increases the design flexibility.

The method in this paper may seem similar to the method in [19]. In fact, both methods solve certain optimization problems numerically to shape the frequency

response of $S$ using the information at discrete frequency points. However, there are two major differences. The first difference is the way to define the cost function. The approach in [19] uses the maximal weighted distance from what they call "center of a tube" as a cost function (see [19, Eqn. (4.11), page 39]), while our approach uses a weighted squares sum from a desired frequency response (see Eqn. (2) in Section 2). Our motivation for adopting the squares sum as a cost function is that least-squares optimization is well-studied (see [30, Chapter 13]), and that there exists efficient algorithms, such as the Gauss-Newton and the Levenberg-Marquardt methods, that we can utilize and easily modify for our special problem. The second difference is that we first bound the degree of $S$, and thus the degree of controllers, while the approach in [19] first obtain some optimized frequency response at discrete frequency points, and then approximate it with $S$ of a specified degree. If necessary, they use the model reduction technique to lower the controller degree. A disadvantage in the latter approach is that the approximation/reduction process may degrade the frequency shape (and thus system performance) in an unpredictable way.

The paper is organized as follows. In Section 2, we formulate a sensitivity shaping problem to be considered in this paper. This problem is reduced to a finite dimensional constrained nonlinear least-squares optimization problem in Section 3. In Section 4, we explain properties of the optimization problem, the detailed algorithms to be utilized, and some rules of thumb to select an initial point of the algorithms. Section 5 sums up the results in this paper as a controller design procedure. Using the procedure, we give numerical examples to show the usefulness of our approach, with comparisons to conventional $H^\infty$ controller design methods.

For ease of exposition, we deal with only scalar discrete-time systems in this paper. However, as will be shown in Section 5, our method is applicable even to continuous-time systems with bilinear transformations. Multivariable cases are under investigation, but one can expect that a similar idea can be applied to multivariable cases, using e.g. the theory in [2].

## 2 A sensitivity shaping problem

Consider the unity feedback system depicted in Figure 1. Here, $P$ is a given scalar real rational discrete-time plant and $C$ is a controller to be designed to fulfill both internal stability of the feedback system and some given performance specifications. In this paper, we consider only such specifications that can be expressed in terms of the *sensitivity function*

$$S(z) := \frac{1}{1 + P(z)C(z)}, \tag{1}$$

in the frequency domain. More precisely, we assume that, at a given finite number $N$ of frequencies $\boldsymbol{\theta} := \{\theta_k\}_{k=1}^N \subset [0, \pi]$, a "desired" frequency response $\boldsymbol{s} := \{s_k\}_{k=1}^N \subset \mathbb{C}$ of $S$ is given, and we try to find a "best-approximate" sensitivity function $S$ from a class of "allowable" sensitivity functions (see Figure 2[1]). Next, what we mean by "best-approximate" and "allowable" will be explained.

[1]The 3-D plot in Figure 2 can be interpreted as a combination of the gain plot and the phase plot in the Bode diagram.

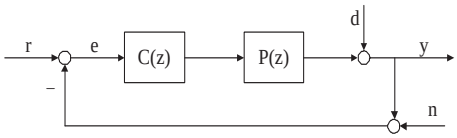


Figure 1: The unity feedback system.

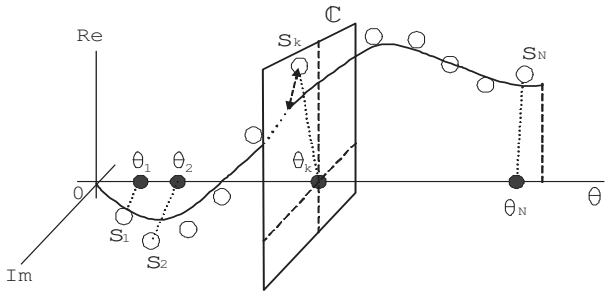


Figure 2: The frequency response of a "best-approximate" sensitivity function $S$ (solid curve) to data $s_k$ (circles) at frequencies $\theta_k$ (black dots on $\theta$-axis).

To clarify the meaning of "best-approximation," we need to introduce a discrepancy between the desired frequency response data $(\boldsymbol{\theta}, \boldsymbol{s})$ and a sensitivity function $S$. In this paper, we use the weighted squares sum[2]:

$$d_{\boldsymbol{w}}((\boldsymbol{\theta}, \boldsymbol{s}), S) := \frac{1}{2} \sum_{k=1}^{N} \frac{w_k}{|s_k|^2} \left| S(e^{i\theta_k}) - s_k \right|^2, \tag{2}$$

where the weights $\boldsymbol{w} := \{w_k\}_{k=1}^N$ are positive scalars to be chosen by the designer; if one wants a better approximation at the frequency $\theta_k$, one can choose a large $w_k$ relative to weights at other frequencies. In (2), the term $\left| S(e^{i\theta_k}) - s_k \right|$ is the distance of two complex numbers $S(e^{i\theta_k})$ and $s_k$ in the complex plane; see the dashed arrow in Figure 2. A "best-approximate" sensitivity function $S$ is the one which minimizes this discrepancy for given $(\boldsymbol{w}, \boldsymbol{\theta}, \boldsymbol{s})$.

In this paper, we call a sensitivity function $S$ "allowable" if it satisfies the following four conditions:

(C1) the internal stability condition,

(C2) $n_e$ interpolation conditions $S(\lambda_j) = \eta_j$, $j = 1, \ldots, n_e$, which are specified at points $\lambda_j \in \mathbb{C}$, not only outside the unit disc but also different from unstable poles and zeros of the plant,

(C3) the $H^\infty$ norm bound condition $\|S\|_\infty < \gamma$, where for a stable rational function $S$, $\|S\|_\infty := \max_{\theta \in [-\pi, \pi]} \left| S(e^{i\theta}) \right|$, and $\gamma$ is chosen to be large enough so that there exists an $S$ which satisfies (C1)–(C3), and

[2]Division by $|s_k|^2$ is for normalization. We assume $s_k \neq 0$.

(C4) rationality and a degree condition, i.e., $S$ must be real rational and $\deg S \leq n := n_p + n_z + n_e - 1$, where $n_p$ and $n_z$ are the number of unstable poles and zeros (including infinite zeros and counting multiplicities) of the plant $P$, respectively.

The motivations for these conditions are as follows. (C1) is a standard requirement for any practical feedback system. (C2)–(C4) are motivated by the work in [6, 26]. (C2) increases the flexibility of the shaping design. (See [26], where we call these conditions *additional interpolation constraints.*) We may not need this condition for achieving required performance, in which case, we just set $n_e = 0$. As for (C3), there are motivations from both control viewpoint and optimization viewpoint. From control viewpoint, the constraint (C3) is called the *gain-phase margin constraint* (see [19, p. 20]), and (C3) is important to avoid a large peak gain of $S$ for a large stability margin. From optimization viewpoint, (C3) is useful to avoid choosing an initial point far from the solution in nonconvex optimization that we need to solve; see Section 4. (C4) restricts a class to a degree constrained one, which eventually leads to a restriction on the controller degree, as stated in the following proposition.

**Proposition 2.1** *[26] If the plant $P$ is real rational and strictly proper, and the sensitivity function $S$ is real rational and satisfies (C1), (C2) and (C4), then the controller $C = (1 - S)/PS$ is real rational proper and its degree satisfies*

$$\deg C \leq \deg P - 1 + n_e. \tag{3}$$

With definitions of the discrepancy $d_{\boldsymbol{w}}$ in (2) and the class of allowable sensitivity functions

$$\mathcal{S} := \{S : \ S \text{ satisfies (C1)–(C4)}\}, \tag{4}$$

the *sensitivity shaping problem* to be considered in this paper is, for given weights $\boldsymbol{w}$ and data $(\boldsymbol{\theta}, \boldsymbol{s})$, to solve an optimization problem:

$$\inf_{S \in \mathcal{S}} d_{\boldsymbol{w}}((\boldsymbol{\theta}, \boldsymbol{s}), S). \tag{5}$$

The infimum may not be achieved by any $S$ in $\mathcal{S}$; this point will be discussed in Section 4. In the next section, we will reduce the problem (5) to a finite dimensional constrained nonlinear least-squares problem. by expressing the class $\mathcal{S}$ in terms of a finite dimensional parameter vector.

**Remark 2.2** The set $\mathcal{S}$ is actually the real rational solution set to the *Nevanlinna-Pick interpolation problem with degree constraint*, which was studied in, e.g., [6, 17, 16]. The degree bound in (C4) is chosen as $n$ to guarantee the nonemptyness of the set $\mathcal{S}$.

# 3 Finite dimensional constrained nonlinear least-squares problem

In this section, we will show that the sensitivity shaping problem (5) can be reduced to a finite dimensional constrained nonlinear least-squares (NLS) problem.

### 3.1 Reduction to an NLS problem

Suppose that $S$ is a feasible point of the optimization problem (5), i.e., $S \in \mathcal{S}$. Then, since $S$ satisfies (C4), it can be factored as

$$S(z) = \frac{b(z)}{a(z)}, \tag{6}$$

where $a(z) := \boldsymbol{z}^T\boldsymbol{\alpha}$, $b(z) := \boldsymbol{z}^T\boldsymbol{\beta}$, $\boldsymbol{\alpha} \in \mathbb{R}^{n+1}$, $\boldsymbol{\beta} \in \mathbb{R}^{n+1}$ and $\boldsymbol{z} := [z^n, \cdots, z, 1]^T$. In addition, since $S$ satisfies (C1) and (C2), $S$ needs to fulfill $n_p + n_z + n_e (= n+1)$ interpolation/derivative conditions at unstable poles and zeros (including infinite zeros) of the plant, as well as at points specified by (C2). Due to these $(n+1)$ conditions, we can derive a linear relation between $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ as

$$\boldsymbol{\beta} = K\boldsymbol{\alpha}, \tag{7}$$

for a uniquely determined real matrix $K$. See [28] for the detail of the construction of $K$. Besides, since $S$ satisfies (C3), $S$ must be stable and meet the norm condition $\|S\|_\infty < \gamma$. The stability condition can be stated that the denominator vector $\boldsymbol{\alpha}$ needs to be in the Schur stability region:

$$\mathfrak{S} := \left\{ \begin{array}{r} \boldsymbol{\alpha} := [\alpha_0, \cdots, \alpha_n]^T \in \mathbb{R}^{n+1} : \alpha_0 > 0 \\ \boldsymbol{z}^T\boldsymbol{\alpha} \neq 0, \forall z \in \mathbb{D}^c \end{array} \right\}, \tag{8}$$

with notation $\mathbb{D}^c := \{z \in \mathbb{C} : |z| \geq 1\}$. The norm condition can be expressed as

$$\gamma^2 \left|a(e^{i\theta})\right|^2 - \left|b(e^{i\theta})\right|^2 > 0, \quad \forall\theta,$$

which leads to spectral factorization

$$\gamma^2 a(z)a(z^{-1}) - b(z)b(z^{-1}) = \rho(z)\rho(z^{-1}), \tag{9}$$

for a unique[3] spectral factor $\rho(z) := \boldsymbol{z}^T\boldsymbol{\rho}$ with $\boldsymbol{\rho} \in \mathfrak{S}$.

So far, we have explained that, for each $S \in \mathcal{S}$, there corresponds to some $\boldsymbol{\alpha} \in \mathfrak{A}$, where $\mathfrak{A}$ is an open set in $\mathbb{R}^{n+1}$ defined by

$$\mathfrak{A} := \left\{ \boldsymbol{\alpha} \in \mathfrak{S} : \gamma^2 \left|\boldsymbol{e}(\theta)^T\boldsymbol{\alpha}\right|^2 - \left|\boldsymbol{e}(\theta)^T K\boldsymbol{\alpha}\right|^2 > 0, \ \forall\theta \right\},$$

with $\boldsymbol{e}(\theta) := [e^{in\theta}, e^{i(n-1)\theta}, \cdots, 1]^T$. The converse is trivial; for each $\boldsymbol{\alpha} \in \mathfrak{A}$, the function $S := (\boldsymbol{z}^T K\boldsymbol{\alpha})/(\boldsymbol{z}^T\boldsymbol{\alpha})$ is in $\mathcal{S}$. We have also explained that, for each $\boldsymbol{\alpha} \in \mathfrak{A}$, there corresponds to a unique $\boldsymbol{\rho} \in \mathfrak{S}$. Actually, a much stronger assertion holds for the map between $\mathfrak{A}$ and $\mathfrak{S}$, as stated in the following theorem taken from [9, 8].

**Theorem 3.1** *To each $\boldsymbol{\rho} \in \mathfrak{S}$, there exists a unique $\boldsymbol{\alpha} \in \mathfrak{A}$ such that $S(z) = b(z)/a(z)$ satisfies $\boldsymbol{\beta} = K\boldsymbol{\alpha}$ and (9). The map $\boldsymbol{h} : \mathfrak{S}$ to $\mathfrak{A}$ sending $\boldsymbol{\rho}$ to $\boldsymbol{\alpha}$ is a diffeomorphism.*

[3] Without the positivity condition $\alpha_0 > 0$ in (8), the spectral factor $\rho$ would be determined uniquely up to sign.

The proof of Theorem 3.1 is highly nontrivial. To each $\boldsymbol{\rho} \in \mathfrak{S}$, the existence of $\boldsymbol{\alpha} \in \mathfrak{A}$ in the theorem was proven by Georgiou in [14, 16, 15]. He also conjectured the uniqueness of such $\boldsymbol{\alpha}$. The conjecture was shown to be true by Byrnes et al. in [9] in the context of rational covariance extensions, and later in [17, 8] for Nevanlinna-Pick interpolation. It was also established in [9, 8] that the map $\boldsymbol{h}$ is a diffeomorphism, providing a complete parameterization of the set $\mathcal{S}$ in terms of $\boldsymbol{\rho} \in \mathfrak{S}$:

$$\mathcal{S} = \left\{ S(z) = \frac{\boldsymbol{z}^T K \boldsymbol{h}(\boldsymbol{\rho})}{\boldsymbol{z}^T \boldsymbol{h}(\boldsymbol{\rho})} : \boldsymbol{\rho} \in \mathfrak{S} \right\}.$$

Due to this parameterization of $\mathcal{S}$, we can reduce the sensitivity shaping problem (5) to the following finite dimensional constrained NLS problem:

$$\inf_{\boldsymbol{\rho} \in \mathfrak{S}} \frac{1}{2} \sum_{k=1}^{N} \frac{w_k}{|s_k|^2} \left| \frac{\boldsymbol{e}_k^T K \boldsymbol{h}(\boldsymbol{\rho})}{\boldsymbol{e}_k^T \boldsymbol{h}(\boldsymbol{\rho})} - s_k \right|^2, \tag{10}$$

where $\boldsymbol{e}_k := \boldsymbol{e}(\theta_k)$, $k = 1, \ldots, N$. The properties of the optimization problem (10) and algorithms to solve it will be explained in Section 4.

**Remark 3.2** The problem (5) can also be reduced to a finite dimensional constrained NLS problem with respect to $\boldsymbol{\alpha}$:

$$\inf_{\boldsymbol{\alpha} \in \mathfrak{A}} \frac{1}{2} \sum_{k=1}^{N} \frac{w_k}{|s_k|^2} \left| \frac{\boldsymbol{e}_k^T K \boldsymbol{\alpha}}{\boldsymbol{e}_k^T \boldsymbol{\alpha}} - s_k \right|^2. \tag{11}$$

However, we prefer solving (10) to solving (11). This is because, in many cases, we obtain a much smaller final cost by solving (10) than by solving (11). This point will be verified through an example in Section 5.

In Section 3.2, we will present an explicit form of the map $\boldsymbol{h}$ (that appears in (12)), as well as its Jacobian useful for numerical optimization in Section 4.

### 3.2 On the map $\boldsymbol{h}$ and its Jacobian

Here, we will introduce a map $\boldsymbol{h}$ from $\mathfrak{S}$ to $\mathfrak{A}$, which is actually a diffeomorphism, and thus, the map appeared in Theorem 3.1. Let us express a nonlinear map $\boldsymbol{h}$ from $\mathfrak{S}$ to $\mathfrak{A}$ as a composition of three maps:

$$\boldsymbol{h} := \boldsymbol{h}_3 \circ \boldsymbol{h}_2 \circ \boldsymbol{h}_1. \tag{12}$$

We will explain next what these three maps are.

First, the map $\boldsymbol{h}_1$ is defined in the domain $\mathfrak{S}$ as

$$\boldsymbol{h}_1(\boldsymbol{\rho}) := \frac{1}{2} T(\boldsymbol{\rho}) \boldsymbol{\rho}, \quad \boldsymbol{\rho} \in \mathfrak{S}. \tag{13}$$

It was shown in [9] that the map $\boldsymbol{h}_3$ is a diffeomorphism from $\mathfrak{S}$ to the range

$$\mathfrak{D} := \left\{ \begin{array}{l} \boldsymbol{d} := [d_0, \cdots, d_n]^T \in \mathbb{R}^{n+1} \\ \quad d_0 + \sum_{j=1}^{n} d_j (z^j + z^{-j}) > 0, \forall z \in \mathbb{T} \end{array} \right\}, \tag{14}$$

where $\mathbb{T} := \{z \in \mathbb{C} : |z| = 1\}$.

Next, the map $\boldsymbol{h}_2$ is defined in the domain $\mathfrak{D}$ as the inverse map of

$$\boldsymbol{g}_2(\hat{\boldsymbol{\alpha}}) := T(\hat{\boldsymbol{\alpha}})\hat{K}\hat{\boldsymbol{\alpha}}, \quad \hat{\boldsymbol{\alpha}} \in \hat{\mathfrak{A}}, \tag{15}$$

where, for a vector $\boldsymbol{v} := [v_0, \cdots, v_n]^T$, $T(\boldsymbol{v})$ is a Hankel + Toeplitz operator defined by

$$T(\boldsymbol{v}) := \begin{bmatrix} v_0 & \cdots & v_n \\ \vdots & \iddots & \\ v_n & & \end{bmatrix} + \begin{bmatrix} v_0 & \cdots & v_n \\ & \ddots & \vdots \\ & & v_n \end{bmatrix}. \tag{16}$$

The domain of $\boldsymbol{g}_2$ is an open set in $\mathbb{R}^{n+1}$:

$$\hat{\mathfrak{A}} := \left\{ \hat{\boldsymbol{\alpha}} \in \mathfrak{S} : \min_{\theta \in [-\pi, \pi]} \mathrm{Re} \left[ \frac{\hat{\boldsymbol{e}}(\theta)^T \hat{K} \hat{\boldsymbol{\alpha}}}{\hat{\boldsymbol{e}}(\theta)^T \hat{\boldsymbol{\alpha}}} \right] > 0 \right\}, \tag{17}$$

where $\hat{\boldsymbol{e}}(\theta) := \left[1, e^{i\theta}, \cdots, e^{in\theta}\right]^T$ and[4]

$$\hat{K} := (\gamma I - K)(\gamma I + K)^{-1}. \tag{18}$$

The set $\hat{\mathfrak{A}}$ is a set of denominator coefficient vectors for strictly positive real functions. Since the map $\boldsymbol{g}_2$ was proven to be a diffeomorphism in [9, 8], its inverse map $\boldsymbol{h}_2 := \boldsymbol{g}_2^{-1}$ is well-defined.

Finally, the linear map $\boldsymbol{h}_3$ is defined in the domain $\hat{\mathfrak{A}}$ by

$$\boldsymbol{h}_3(\hat{\boldsymbol{\alpha}}) := (\gamma I + K)^{-1}\hat{\boldsymbol{\alpha}}, \quad \hat{\boldsymbol{\alpha}} \in \hat{\mathfrak{A}}. \tag{19}$$

Now, we will state that the map $\boldsymbol{h}$ in (12) is actually a map appeared in Theorem 3.1, by analyzing the properties of the three maps $\boldsymbol{h}_k$, $k = 1, 2, 3$. Below, we use the following notation: for a vector-valued differentiable map $\boldsymbol{f} : \mathbb{R}^{n+1} \mapsto \mathbb{R}^{n+1}$,

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} := \left[ \frac{\partial \boldsymbol{f}}{\partial x_0}, \cdots, \frac{\partial \boldsymbol{f}}{\partial x_n} \right]. \tag{20}$$

**Proposition 3.3** *The maps $\boldsymbol{h}_k$, $k = 1, 2, 3$, are diffeomorphisms from $\mathfrak{S}$ to $\mathfrak{D}$, from $\mathfrak{D}$ to $\hat{\mathfrak{A}}$ and from $\hat{\mathfrak{A}}$ to $\mathfrak{A}$, respectively. Their derivatives are given by*

$$\begin{aligned} \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) &= T(\boldsymbol{\rho}), \\ \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{d}}(\boldsymbol{d}) &= \left[ T(\boldsymbol{h}_2(\boldsymbol{d}))\hat{K} + \sum_{j=0}^{n} [\boldsymbol{h}_2(\boldsymbol{d})]_j \, T(\hat{\boldsymbol{k}}_j) \right]^{-1}, \\ \frac{\partial \boldsymbol{h}_3}{\partial \hat{\boldsymbol{\alpha}}}(\hat{\boldsymbol{\alpha}}) &= (\gamma I + K)^{-1}, \end{aligned}$$

*where $[\boldsymbol{h}_2(\boldsymbol{d})]_j$ is the $(j+1)$-th element of the vector $\boldsymbol{h}_2(\boldsymbol{d})$, and $\hat{\boldsymbol{k}}_j$ is the $(j+1)$-th column of the matrix $\hat{K}$.*

[4] We remark that the matrix $\gamma I + K$ is invertible between Euclidean spaces in our problem setting; see [28].

***Proof.*** The derivatives are obtained via direct calculations from the definition of each map. The diffeomorphisms of $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ are the results in [9, 8]. Thus, we have only to prove that the map $\boldsymbol{h}_3$ is onto the set $\mathfrak{A}$, because if this is the case, the diffeomorphism of $\boldsymbol{h}_3$ follows from the linearity and invertibility of the map.

To prove that $\boldsymbol{h}_3$ is onto $\mathfrak{A}$, suppose that $\boldsymbol{\alpha}$ is in $\mathfrak{A}$, i.e.,

$$\boldsymbol{\alpha} \in \mathfrak{S}, \quad \gamma^2 \left|\boldsymbol{e}(\theta)^T \boldsymbol{\alpha}\right|^2 - \left|\boldsymbol{e}(\theta)^T K \boldsymbol{\alpha}\right|^2 > 0, \forall \theta. \tag{21}$$

We want to show that $\hat{\boldsymbol{\alpha}} := (\gamma I + K)\boldsymbol{\alpha}$ is in $\hat{\mathfrak{A}}$, i.e,

$$\hat{\boldsymbol{\alpha}} \in \mathfrak{S}, \quad \min_{\theta \in [-\pi, \pi]} \mathrm{Re} \left[ \frac{\hat{\boldsymbol{e}}(\theta)^T \hat{K} \hat{\boldsymbol{\alpha}}}{\hat{\boldsymbol{e}}(\theta)^T \hat{\boldsymbol{\alpha}}} \right] > 0. \tag{22}$$

If we define $S(z) := (\boldsymbol{z}^T K \boldsymbol{\alpha})/(\boldsymbol{z}^T \boldsymbol{\alpha})$, then $S$ is analytic in $\mathbb{D}^c$ and takes the absolute value less than $\gamma$ at each point on $\mathbb{T}$ due to (21). Using a bilinear transformation, define

$$F(z) := \frac{\gamma - S(z^{-1})}{\gamma + S(z^{-1})} = \frac{\hat{\boldsymbol{z}}^T (\gamma I - K)\boldsymbol{\alpha}}{\hat{\boldsymbol{z}}^T (\gamma I + K)\boldsymbol{\alpha}}, = \frac{\hat{\boldsymbol{z}}^T \hat{K} \hat{\boldsymbol{\alpha}}}{\hat{\boldsymbol{z}}^T \hat{\boldsymbol{\alpha}}}, \tag{23}$$

where $\hat{\boldsymbol{z}} := [1, z, \cdots, z^n]^T$. Then, $F$ is analytic for $|z| \leq 1$ and takes a positive real value at each point on $\mathbb{T}$. In addition, the sign of the first element in $\boldsymbol{\alpha}$ and that in $\hat{\boldsymbol{\alpha}}$ are the same for the following reason. Because the function $F$ is positive real, the first element of $\hat{\boldsymbol{\alpha}}$ and that of $\hat{K}\hat{\boldsymbol{\alpha}}$ must have the same sign. This means that the first element of $\hat{\boldsymbol{\alpha}}$ and that of $\hat{\boldsymbol{\alpha}} + \hat{K}\hat{\boldsymbol{\alpha}} = (\gamma I + K)\boldsymbol{\alpha} + (\gamma I - K)\boldsymbol{\alpha} = 2\gamma \boldsymbol{\alpha}$ have the same sign, and thus $\hat{\boldsymbol{\alpha}} \in \mathfrak{S}$. Therefore, we have established $\hat{\boldsymbol{\alpha}} \in \hat{\mathfrak{A}}$, and hence the surjectivity of $\boldsymbol{h}_3$. *Q.E.D.*

Due to Proposition 3.3, as well as the chain rule, we have arrived at the following assertion.

**Theorem 3.4** *The map $\boldsymbol{h}$ in (12) is a diffeomorphism from $\mathfrak{A}$ to $\mathfrak{S}$. Its derivative is given by*

$$\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) = \frac{\partial \boldsymbol{h}_3}{\partial \hat{\boldsymbol{\alpha}}}((\boldsymbol{h}_2 \circ \boldsymbol{h}_1)(\boldsymbol{\rho})) \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{d}}(\boldsymbol{h}_1(\boldsymbol{\rho})) \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}). \tag{24}$$

**Remark 3.5** One may think that it is beneficial to use $\boldsymbol{d}$, instead of $\boldsymbol{\rho}$, as optimization variables, since the set $\mathfrak{D}$ is convex. However, our numerical experience tells that $\boldsymbol{\rho}$-parameterization often gives better solutions than $\boldsymbol{d}$-parameterization.

# 4 Solving the nonlinear least-squares problems

In order to solve the sensitivity shaping problem formulated Section 2, we need a reliable and numerically robust algorithm to solve the optimization problem in (10). The precise meaning of "solving" will become clear in Section 4.1. The problem can be written

$$\inf_{\boldsymbol{\rho} \in \mathfrak{S}} \frac{1}{2} \boldsymbol{F}(\boldsymbol{\rho})^T \boldsymbol{F}(\boldsymbol{\rho}), \tag{25}$$

where $\boldsymbol{F} : \mathfrak{S} \mapsto \mathbb{R}^{2N}$ is the vector-valued residual map

$$\begin{aligned}\boldsymbol{F}(\boldsymbol{\rho}) := [&\mathrm{Re}\left\{f_1(\boldsymbol{\rho})\right\}, \cdots, \mathrm{Re}\left\{f_N(\boldsymbol{\rho})\right\}, \\ &\mathrm{Im}\left\{f_1(\boldsymbol{\rho})\right\}, \cdots, \mathrm{Im}\left\{f_N(\boldsymbol{\rho})\right\}]^T,\end{aligned}$$

$$f_k(\boldsymbol{\rho}) := \frac{\sqrt{w_k}}{|s_k|}\left(\frac{\boldsymbol{e}_k^T K \boldsymbol{h}(\boldsymbol{\rho})}{\boldsymbol{e}_k^T \boldsymbol{h}(\boldsymbol{\rho})} - s_k\right),\ k = 1, \ldots, N. \tag{26}$$

Next, we will discuss properties of the optimization problem (25), present modifications of two standard algorithms for unconstrained NLS problems dealing with the constraint $\boldsymbol{\rho} \in \mathfrak{S}$, and suggest possible initial solutions.

### 4.1 Properties of the optimization problem

Since the domain $\mathfrak{S}$ of the problem (25) is open, there is no guarantee that there exists a minimizer in $\mathfrak{S}$. In addition, since the cost functional in (25) is nonconvex and the domain $\mathfrak{S}$ in general is a nonconvex set, a global minimizer may not be unique, and there may even be several local minima. Therefore, by "solving" (25), we mean either finding a *local minimizer* in $\mathfrak{S}$ or an *approximation in* $\mathfrak{S}$ *of a local infimizer* within a certain tolerance.

A major advantage with the formulated nonlinear least-squares problem is the smoothness of the cost functional in (25). This smoothness is due to the continuous differentiability of the residual vector $\boldsymbol{F}$ with respect to $\boldsymbol{\rho}$; see Section 4.3 for derivative expressions. This enables local search algorithms based on derivative information, which will be proposed in Section 4.2. For derivative-based algorithms, nonconvexity means that it will not converge to a global minimizer unless algorithms are initialized properly. This makes the problem of finding good initial points important. Some guidelines to select proper initial points will be given in Section 4.4.

### 4.2 Two modified algorithms

The formulation as a nonlinear least-squares problem also has the advantage that the problem class is well-studied and that there are several efficient and numerically robust algorithms for solving the problem available; see e.g. [30]. Especially, two popular algorithms are the *Gauss-Newton* and the *Levenberg-Marquardt* methods, which were originally developed for unconstrained nonlinear least-squares problems. Here, we will modify these two algorithms in order to incorporate the constraint $\boldsymbol{\rho} \in \mathfrak{S}$. We will treat the constraint implicitly; more precisely, we will enforce a bound on the step length so that an updated point stays in $\mathfrak{S}$. As stopping criteria, we will either require the gradient to be close to zero or that the norm of the step is small. Detailed descriptions are shown in Algorithms 4.1 and 4.2, where the tolerances for the stopping criteria are denoted by $\varepsilon_1$ and $\varepsilon_2$, the Jacobian of $\boldsymbol{F}$ is denoted by $\nabla \boldsymbol{F}$, and $\|\cdot\|$ is the Euclidean norm. These two algorithms will be numerically compared through an example in Section 5.

**ALGORITHM 4.1** *Modified Gauss-Newton*

Set $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho}_0$ and $\boldsymbol{p}$ with $\|\boldsymbol{p}\| > \varepsilon_2$
Compute $\boldsymbol{F}$ and $\nabla \boldsymbol{F}$
**while** $\|\nabla \boldsymbol{F}\boldsymbol{F}\| > \varepsilon_1(1 + \boldsymbol{F}^T\boldsymbol{F})$ & $\|\boldsymbol{p}\| > \varepsilon_2$
  Set $H \leftarrow \nabla \boldsymbol{F} \nabla \boldsymbol{F}^T$
  Set $\boldsymbol{p} \leftarrow -(H + 10^{-6}\,\mathrm{trace}(H))^{-1}(\nabla \boldsymbol{F}\boldsymbol{F})$
  Set $\mu \leftarrow 1$
  **repeat**
    **if** $\boldsymbol{\rho} + \mu\boldsymbol{p} \notin \mathfrak{S}$ **then**
      Set $\mu \leftarrow \mu/2$
    **else**
      Compute $\hat{\boldsymbol{F}}$ and $\nabla \hat{\boldsymbol{F}}$ for $\boldsymbol{\rho} + \mu\boldsymbol{p}$
      **if** $\hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}} \leq \boldsymbol{F}^T\boldsymbol{F} + 0.1\mu\boldsymbol{p}^T\nabla \boldsymbol{F}\boldsymbol{F}$ **then**
        Break
      **else**
        Set $\mu \leftarrow \mu/2$
      **end if**
    **end if**
  **end repeat**
  Set $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho} + \mu\boldsymbol{p}$, $\boldsymbol{F} \leftarrow \hat{\boldsymbol{F}}$ and $\nabla \boldsymbol{F} \leftarrow \nabla \hat{\boldsymbol{F}}$
**end while**

### 4.3 Feasibility test and computation of $\boldsymbol{F}$ and $\nabla \boldsymbol{F}$

In the algorithms proposed above, we have two nontrivial steps: to check feasibility ($\boldsymbol{\rho} \in \mathfrak{S}$) and to compute $\boldsymbol{F}$ and $\nabla \boldsymbol{F}$. These steps will also constitute a substantial part of the computational effort.

To check whether $\boldsymbol{\rho} \in \mathfrak{S}$, we can, e.g., recursively compute the corresponding partial reflection coefficients and check that they are less or equal to one in modulus, since $\rho$ is a real polynomial, [32].

Computing the residual vector $\boldsymbol{F}$ for a given point $\boldsymbol{\rho} \in \mathfrak{S}$ involves the computation of $\boldsymbol{h}(\boldsymbol{\rho})$ as shown in (26). This computation can be done by the continuation method developed in [1], which however requires some computational effort. The Jacobian $\nabla \boldsymbol{F}$ is given by

$$\begin{aligned} \nabla \boldsymbol{F}(\boldsymbol{\rho}) := \Bigg[ & \mathrm{Re}\left\{\frac{\partial f_1}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})\right\}, \cdots, \mathrm{Re}\left\{\frac{\partial f_N}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})\right\}, \\ & \mathrm{Im}\left\{\frac{\partial f_1}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})\right\}, \cdots, \mathrm{Im}\left\{\frac{\partial f_N}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})\right\}\Bigg], \end{aligned} \tag{27}$$

where each column is calculated with

**ALGORITHM 4.2** *Modified Levenberg-Marquardt*

Set $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho}_0$, $\boldsymbol{p}$ with $\|\boldsymbol{p}\| > \varepsilon_2$, $\Delta \leftarrow 1$, $\mu \leftarrow 1/4$, and $\eta \leftarrow 3/4$
Compute $\boldsymbol{F}$ and $\nabla \boldsymbol{F}$
**while** $\|\nabla \boldsymbol{F} \boldsymbol{F}\| > \varepsilon_1(1 + \boldsymbol{F}^T\boldsymbol{F})$ & $\|\boldsymbol{p}\| > \varepsilon_2$
  Decide $\boldsymbol{p}$ by solving the trust-region subproblem $\min_{\|\boldsymbol{p}\| \le \Delta} \mathbb{I}(\boldsymbol{p}) := \boldsymbol{F}^T\boldsymbol{F} + \boldsymbol{p}^T\nabla \boldsymbol{F}\boldsymbol{F} + \boldsymbol{p}^T\nabla \boldsymbol{F}\nabla \boldsymbol{F}^T\boldsymbol{p}/2$,
  using the algorithm of [25]
  **if** $\boldsymbol{\rho} + \boldsymbol{p} \notin \mathfrak{S}$ **then**
    Set $\lambda \leftarrow 0$
  **else**
    Compute $\hat{\boldsymbol{F}}$ and $\nabla \hat{\boldsymbol{F}}$ for $\boldsymbol{\rho} + \boldsymbol{p}$
    Set $\lambda = (\boldsymbol{F}^T\boldsymbol{F} - \hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}})/(\boldsymbol{F}^T\boldsymbol{F} - \mathbb{I}(\boldsymbol{p}))$
  **end if**
  **if** $\lambda \le \mu$ **then**
    Set $\Delta \leftarrow \Delta/2$
  **else**
    Set $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho} + \boldsymbol{p}$, $\boldsymbol{F} \leftarrow \hat{\boldsymbol{F}}$, and $\nabla \boldsymbol{F} \leftarrow \nabla \hat{\boldsymbol{F}}$
    Set $\Delta \leftarrow 2\Delta$ if $\lambda \ge \eta$
  **end if**
**end while**

$$\frac{\partial f_k}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) = \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})\right)^T \times \frac{\sqrt{w_k}}{|s_k|} \frac{(\boldsymbol{e}_k^T\boldsymbol{h}(\boldsymbol{\rho}))K^T\boldsymbol{e}_k - (\boldsymbol{e}_k^TK\boldsymbol{h}(\boldsymbol{\rho}))\boldsymbol{e}_k}{(\boldsymbol{e}_k^T\boldsymbol{h}(\boldsymbol{\rho}))^2}. \tag{28}$$

and the term $(\partial \boldsymbol{h})/(\partial \boldsymbol{\rho})$ is computed by (24).

### 4.4 Determining a good initial point

The initialization of the algorithm is most important since the problem in general is nonconvex. If we have a controller design to be improved incrementally we can initialize with that solution. Otherwise we propose to use what we might call the *approximate peak solution*. This also serves as the default initial point in the MATLAB implementation [3].

The approximate peak solution is motivated by the tuning rules of [26]. The most effective tuning rule is to place a complex conjugate pair of roots of $\rho$ close to the unit circle at the frequency corresponding to a desired peak gain of the sensitivity function. Approximately knowing a desired peak location we place a pair of roots correspondingly and the rest in origin. Starting at the maximum entropy (ME) solution, we can use the continuation method of [1] to determine the approximate peak solution. The ME solutions can be computed using the formula [18, Eq. (6.2), p. 2915] for the positive real setting, with bilinear transformations.

# 5 Design procedure and examples

The flowchart of our controller design procedure is depicted in Figure 3. In the flowchart, "NLSsolver" is the nonlinear least-squares optimization solver which realizes the theory in Sections 3 and 4. The NLSsolver can be regarded as a *blackbox* whose inputs are a plant transfer function $P$ and design parameters, and whose output is a sensitivity function $S$, and thus, a controller $C = (1 - S)/PS$. As for inputs, the plant is given and fixed, whereas the design parameters are tunable for performance improvements. We have developed a user-friendly interface [3] that realizes this flowchart.

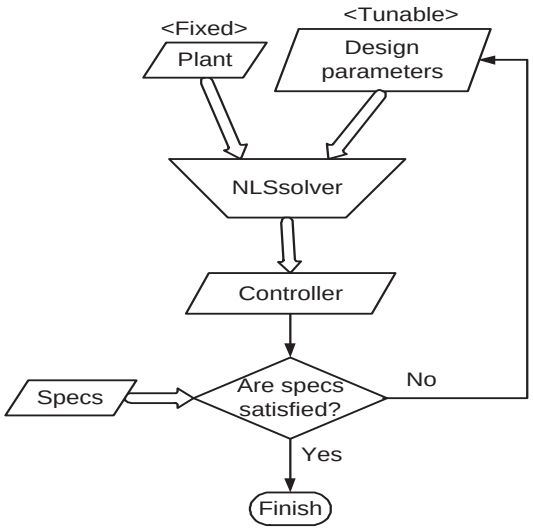


Figure 3: Flowchart of the design procedure

Our tunable design parameters are listed in Table 1. We briefly explain each design parameter.

- $\boldsymbol{\theta}$ : A vector of frequency points in $[0, \pi]$ (see Figure 2). One can take the points densely over some frequency range if one would like to emphasize the sensitivity approximation over the range, at the expense of increase of computational

| Design parameters | | in (25) |
|---|---|---|
| $\boldsymbol{\theta}$ | : discrete frequency points | $\boldsymbol{e}_k$ |
| $\boldsymbol{s}$ | : desired frequency response | $s_k$ |
| $\boldsymbol{w}$ | : weights | $w_k$ |
| $\gamma$ | : a uniform bound of $\lvert S \rvert$ | $\boldsymbol{h}$ |
| $\{(\lambda_j, \eta_j)\}_{j=1}^{n_e}$ | : additional interpolation points and values | $\boldsymbol{h}, K, \mathfrak{S}$ |

Table 1: Tunable design parameters: The right column shows which parameters in the optimization problem (25) are influenced by design parameters.

burden.

- $\boldsymbol{s}$ : A vector of desired frequency responses specified at frequencies in $\boldsymbol{\theta}$ (see Figure 2). This may be specified from a "desired" sensitivity function $S_d$ as $\boldsymbol{s} = \left\{S_d(e^{i\theta_k}),\ \theta_k \in \boldsymbol{\theta}\right\}$. Also, $\boldsymbol{s}$ can be modified during design iterations to improve desired performance (such as robust stability), which may result in actual performance improvement.
- $\boldsymbol{w}$ : A weight vector. One can choose a large $w_k$ if one wants a better sensitivity approximation around $\theta_k$.
- $\gamma$ : A uniform gain bound of $S$. The larger the $\gamma$ is, the larger the set $\mathcal{S}$ becomes. One should choose $\gamma$ larger than $\max_k |s_k|$. However, unduly large $\gamma$ will be undesirable because it becomes difficult to select a good initial point in optimization.
- $\{(\lambda_j, \eta_j)\}_{j=1}^{n_e}$ : Points and values for additional constraints. One can use a constraint $S(\lambda_j) = \eta_j$ with $\lambda_j \approx e^{i\theta_0}$ when one wants to enforce $S$ to have value $\eta_j$ around the frequency $\theta_0$ rad/sec. However, the introduction of additional constraints increases the controller degree bound; see (3) in Proposition 2.1.

Although there seems too many design parameters to select, we often fix some parameters and manipulate a few of them during design iterations.

Next, through a couple of examples from the control literature, we shall explain how to select and tune these design parameters to satisfy given design specifications. We will also compare the obtained controllers with ones designed in the literature. These problems assume the feedback structure depicted in Figure 1. To focus on the presentation of the selection and tuning strategies, we will skip the exposition of the physical meanings in each problem, and present it just as a mathematical problem. Readers interested in detailed problem settings are referred to each book.

**Remark 5.1** Recall that our problem is posed in discrete-time in Section 2; if a control problem is given in continuous-time, we will first transform the plant to discrete-time correspondence by means of bilinear transformations, input them into our NSLsolver to design a discrete-time controller, and then transform it back to a continuous-time controller. (This process is done automatically in the developed interface.)

## 5.1 Flexible beam control

Here, we will deal with a control problem in [11, Section 10 & 12] where a desired sensitivity function is naturally available from the specification.

### 5.1.1 Problem setting

The continuous-time plant $P$ is given as a transfer function:

$$P(s) = \frac{-6.4750s^2 + 4.0302s + 175.77}{s(5s^3 + 3.5682s^2 + 139.5021s + 0.0929)}. \tag{29}$$

Our goal in this problem is to design a strictly proper controller $C$ which satisfies, for a step reference $r$,

- the settling time is less than 8 seconds,
- the overshoot is less than 10 %, and
- the control input fulfills $|u(t)| \leq 0.5$ for all $t$.

**Remark 5.2** In our controller design, the specifications must be stated in the frequency domain. Time domain specifications will be translated into frequency domain ones in a reasonable manner, and after controller design based on the frequency domain specifications, we will check if the original time domain specifications are indeed satisfied.

In [11], the first two requirements in the time domain have been translated into a requirement in the frequency domain as a desired sensitivity function:

$$S_d(s) := \frac{s(s+1.2)}{s^2+1.2s+1}.$$

We also aim at designing a sensitivity function similar to $S_d$, with extra consideration of control input constraint.

**Remark 5.3** A controller for this problem was designed in [5, 4] by a manual tuning of our optimization parameter $\boldsymbol{\rho}$. The design procedure here is more systematic than the design [5, 4]. In fact, the manual tuning of $\boldsymbol{\rho}$ was quite heuristic for the achievement of the input constraint.

### 5.1.2 Initial selection of design parameters

Using $S_d$, we extract our desired frequency response at a finite number of frequencies. We take 100 discrete points in the frequency $\left[10^{-3}, 10^{3}\right]$ (rad/sec), equally distanced in the logarithmic scale, as

$$\boldsymbol{\omega} := \{\omega_k\}_{k=1}^{100}.$$

With these points, we set our desired frequency response $(\boldsymbol{\theta}, \boldsymbol{s})$ in the discrete-time setting as

$$\begin{aligned} \boldsymbol{\theta} &:= \left\{\theta_k : \ e^{i\theta_k} = \frac{1+i\omega_k}{1-i\omega_k}, \ \omega_k \in \boldsymbol{\omega}\right\}, \\ \boldsymbol{s} &:= \{s_k := S_d(i\omega_k), \ \omega_k \in \boldsymbol{\omega}\}. \end{aligned}$$

Since we have initially no information on the frequency emphasis, the weights are set as

$$\boldsymbol{w} := \{w_k := 1, \ k = 1, \ldots, 100\},$$

and the uniform upper bound of the sensitivity gain is chosen as

$$\gamma := 1.5.$$

We do not use any additional interpolation condition in this problem. From the gain plot of $S_d$, we would like to have a peak gain around 1 rad/sec. Therefore, we always set the initial point for optimization to a $\boldsymbol{\rho}$ in $\mathfrak{S}$ that has its roots at $\pm 0.95i$, which corresponds to an approximate peak solution having its peak close to 1 (rad/sec) in the continuous-time setting.

### 5.1.3 Controller design

With the initial selection of design parameters, NLSsolver outputs a controller and a sensitivity function as

$$C_0(s) := \frac{75.11s^3 + 53.6s^2 + 2095s + 1.395}{s^4 + 10.06s^3 + 449.1s^2 + 2735s + 3214}, \tag{30}$$

$$S_0(s) := \frac{s^4 + 5.156s^3 + 423.8s^2 + 654.8s}{s^4 + 5.156s^3 + 423.8s^2 + 557.6s + 537.9}. \tag{31}$$

Several frequency and time responses are plotted in Fig. 4. The uppermost figure

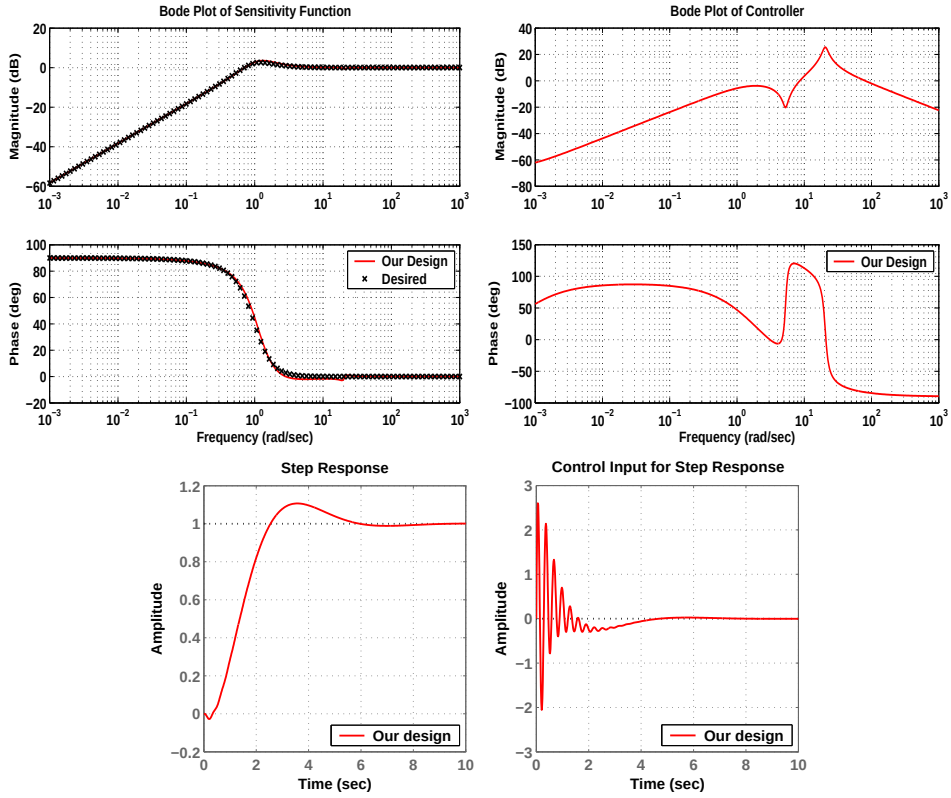


Figure 4: Bode plots and response signals by the initial design

shows the Bode plot of $S_0$ with the desired frequency response $(\boldsymbol{\theta}, \boldsymbol{s})$. As can be seen, NLSsolver indeed generates $S_0$ approximating $(\boldsymbol{\theta}, \boldsymbol{s})$.

Now, we check the original time domain specifications. The lower figures in Figure 4 show the step response and the input signal. Although the step response meets the specification, the input signal is too large to fulfill the specification $|u(t)| \leq 0.5$. Therefore, we need to update some of our design parameters, and redesign a controller.

To see the cause of large input signal, we draw the Bode plot of the controller $C_0$ in Fig. 4. From the figure, we see that there is a sharp gain peak around 20

rad/sec. In fact, this frequency coincides with the frequency of the input oscillation. Therefore, one natural way to suppress the input is to lower the gain peak of $C$.

Now, we update the design parameters. Since $C = (1-S)/PS$, we need to make $S$ close to one to decrease the gain of $C$. Desired frequency response $s_k$ is almost one around frequency 20 rad/sec, and thus, we increase the weight $w_k$ around the frequency to fit $S$ closer to $s_k$. (We do not change other design parameters in this example.) After some trial and error, we have chosen weights $\boldsymbol{w}$ as in Figure 5, that results in the following controller and sensitivity function:

$$C(s) = \frac{2.706s^3 + 1.931s^2 + 75.51s + 0.05028}{s^4 + 7.698s^3 + 33.59s^2 + 126.8s + 143}, \tag{32}$$

$$S(s) = \frac{s^4 + 2.789s^3 + 19.9s^2 + 29.13s}{s^4 + 2.789s^3 + 19.9s^2 + 25.62s + 19.38}. \tag{33}$$

The resulting Bode plots and response signals are shown in Figure 6, with response signals in [11]. The figures show that the sharp peak disappeared in the gain of $C$,

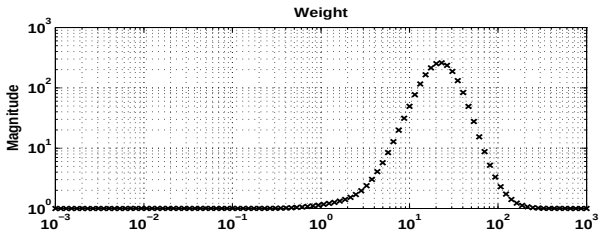


Figure 5: Weight $\boldsymbol{w}$.

which has been done at the price of degradation of sensitivity fitting, and that the original time domain specifications are indeed satisfied. Also, one can see that we have obtained a similar performance to that in [11]. We stress that the controller (32) is half the degree of the one obtained in [11].

**Remark 5.4** The role of weights $\boldsymbol{w}$ is similar to that of weighting functions $W$ in conventional $H^\infty$ control, in that they emphasize suppression of gain at some frequency regions. However, the weights here have two advantages over weighting functions. One is that $\boldsymbol{w}$ do not assume rationality, and rather arbitrary, while the weighting functions $W$ must be rational in most cases. This will increase the flexibility of the design. The other is that $\boldsymbol{w}$ do not increase $\deg C$; it just changes the cost function to be minimized. On the other hand, $W$ typically increases $\deg C$ by $\deg W$.

### 5.1.4 Comparisons between different algorithms

Finally, we compare final errors (cost functional values) and computational times to solve NLS problems, with modified Gauss-Newton (GN) and modified Levenberg-Marquardt (LM) algorithms proposed in Section 4. We sampled ten cases (i.e., ten different weights) during the design iterations in Section 5.1.3, and took mean values

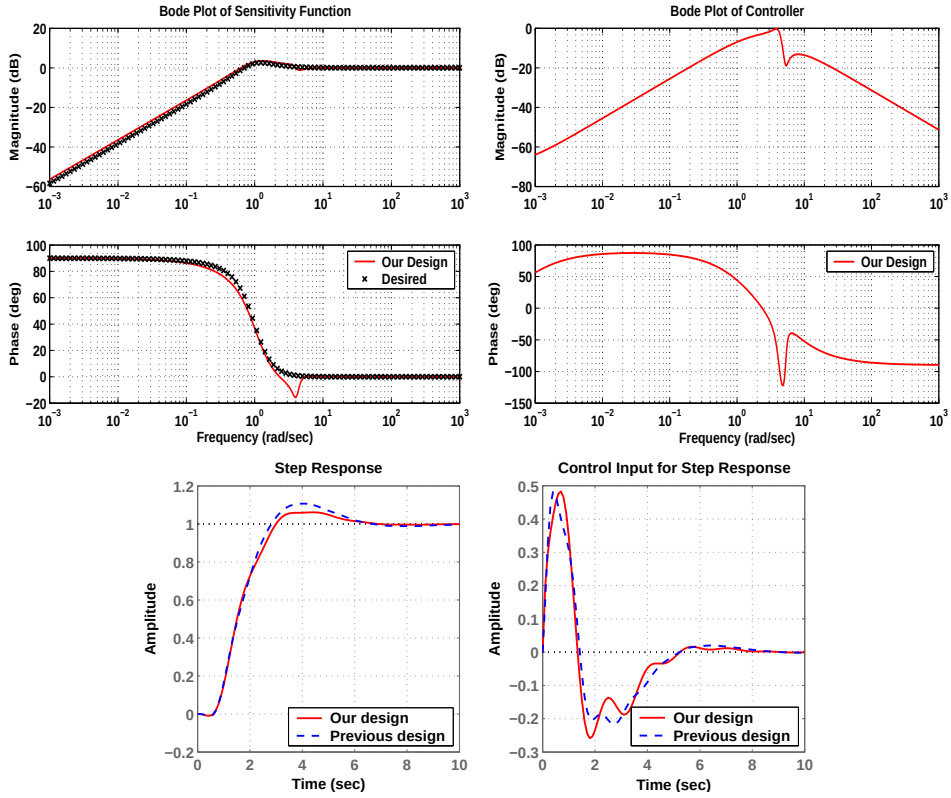


Figure 6: Bode plots and response signals by the new design

and standard deviations of the computational errors and the final errors, as shown in Table 2. We also made an exactly same examination for the problem (11).

From Table 2, we can see that the LM algorithm applied to the NLS problem (10) gives much more accurate solutions than other cases in this example. In fact, this is a general tendency that we have experienced through a number of other examples. The tendency urged us to use the formulation (10) rather than (11), and we use NLSsolver for (10) with the LM algorithm as a default setting.

| NLS | Alg. | Final error | | Comp. time |
|---|---|---|---|---|
| | | mean | standard dev. | mean |
| (10) | GN | 2.723 | 2.257 | 0.1302 |
| | LM | 0.043286 | 0.020431 | 1.7265 |
| (11) | GN | 2.1255 | 1.8231 | 16.9865 |
| | LM | 2.8793 | 2.3877 | 4.7548 |

Table 2: Final errors and computational times (sec).

## 5.2 Slide drive control

Here, we will deal with a slide drive control problem in the book [19, Chapter 6.2]. In this control problem, in contrast to the first example, a desired sensitivity function is not available at the outset. We will explain how to solve such problem with our approach.

### 5.2.1 Problem setting

Here, we will deal with a control problem of a slide drive in the book [19, Chapter 6.2]. The plant $P$ is given by

$$P(s) := \frac{2s^2 + 10s + 100}{s^4 + 7.01s^3 + 110.47s^2 + 452.6s + 521},$$

which is stable and minimum-phase. The performance specifications for the continuous-time sensitivity function are given as

$$\begin{cases} |S(i\omega)| < -20 \text{ dB}, & \omega \in [0, 0.1], \\ |S(i\omega)| < -10 \text{ dB}, & \omega \in [0.1, 1.0], \\ |S(i\omega)| < 6 \text{ dB}, & \omega \in [1.0, 5.0], \\ |1 - S(i\omega)| < -20 \text{ dB}, & \omega \in [5.0, 10.0], \\ |1 - S(i\omega)| < -40 \text{ dB}, & \omega \in [10.0, \infty]. \end{cases} \tag{34}$$

### 5.2.2 Initial selection of design parameters

First of all, since the plant is stable and minimum-phase, we can show that our allowable set $\mathcal{S}$ would be a singleton $\mathcal{S} = \{S : S \equiv 1\}$ without additional constraints $S(\lambda_j) = \eta_j$; see Proposition II.2 in [27]. The case of $S \equiv 1$ (i.e., $C \equiv 0$) is obviously unsatisfactory, and thus we need to introduce at least one additional constraint. Here, we will initially use two constraints as

$$S(\pm 0.01i) = 0.1 (= -20\text{dB}), \tag{35}$$

in the continuous-time setting to take into account the specification over low frequencies.

Next, we need to construct a desired frequency response from the specification (34). We take 50 points in $[0.01, 1]$ (rad/sec) and 50 points in $[5, 100]$ (rad/sec), equally distanced in the logarithmic scale, denoted by $\boldsymbol{\omega} := \{\omega_k\}_{k=1}^{100}$. With these points, we set our desired frequency response $(\boldsymbol{\theta}, \boldsymbol{s})$ in the discrete-time setting as

$$\boldsymbol{\theta} := \left\{ \theta_k : \ e^{i\theta_k} = \frac{1 + i\omega_k}{1 - i\omega_k}, \ \omega_k \in \boldsymbol{\omega} \right\}, \tag{36}$$

$$\boldsymbol{s} := \left\{ s_k := \begin{cases} 0.1, & k = 1, \ldots, 50 \\ 1, & k = 51, \ldots, 100 \end{cases} \right\}, \tag{37}$$

for the specifications over $[0, 1]$ (rad/sec) and $[5, \infty]$ (rad/sec). On the other hand, the specification over the intermediate frequencies $[1, 5]$ (rad/sec) is taken care of by setting the uniform upper bound to

$$\gamma := 2 \ (\approx 6\text{dB}). \tag{38}$$

The weights are initially set as

$$\boldsymbol{w} := \{w_k := 1,\ k = 1, \ldots, 100\}\,.$$

Figure 7 shows the initial selection of all design parameters.

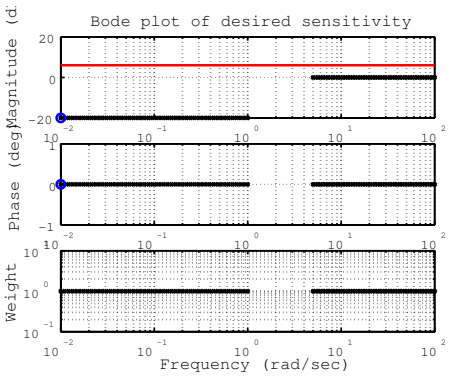


Figure 7: Initial selection of design parameters. The circles correspond to the condition (35) and the horizontal line in the uppermost figure is the uniform bound (38).

**Remark 5.5** Since we have no information to select desired phases, we just set phases to zero. Even though this selection may not be a best one, it will be one natural selection. After the first design, we will obtain an idea how the phase should look like; see the design below.

### 5.2.3 Controller design

Using the initial selections of design parameters in Figure 7, and by choosing the initial $\boldsymbol{\rho}$ whose roots locate at $0.99(1 \pm 3i)/(1 \mp 3i)$ (i.e., gain peak around 3 rad/sec in the continuous-time setting), the NLS solver[5] returns the controller and the sensitivity function as

$$\begin{aligned} C_0(s) &= \frac{3498s^5 + 10^4(4.037s^4 + 49.75s^3 + 333.4s^2 + 899.6s + 825.8)}{s^5 + 4049s^4 + 10^4(2.105s^3 + 20.96s^2 + 5.648s + 17.61}, \\ S_0(s) &= \frac{s^3 + 4044s^2 + 777.3s + 3523}{s^3 + 4044s^2 + 7773s + 3.522 \times 10^4}. \end{aligned}$$

Bode plots of the sensitivity function $S_0$ and complementary sensitivity function $T_0 := 1 - S_0$ are shown in Figure 8, in which we can see that some specifications in (34) are not satisfied.

Now, we will utilize $S_0$ to generate new desired frequency response data $(\boldsymbol{\theta}, \boldsymbol{s})$. The vector $\boldsymbol{\omega}$ is taken at 100 frequencies $\boldsymbol{\omega} := \{\omega_k\}_{k=1}^{100}$, equally distanced in the

[5]For this example, we used only the modified Levenberg-Marquardt algorithm.

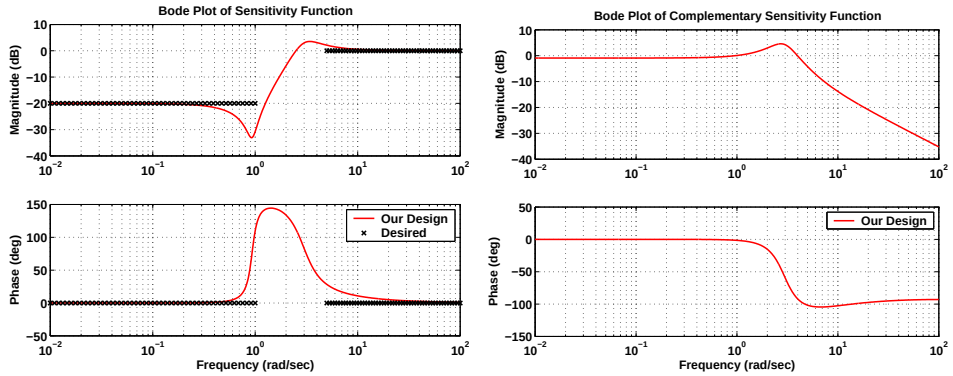


Figure 8: Bode plots of $S_0$ and $T_0 := 1 - S_0$

logarithmic scale over $[0.01, 100]$. Then, $\boldsymbol{\theta}$ is obtained by (36), and a vector $\boldsymbol{s}$ is given by

$$\boldsymbol{s} := \{s_k := S_0(i\omega_k),\ k = 1, \ldots, 100\}\,. \tag{39}$$

Our strategy here is to modify this $\boldsymbol{s}$, as well as (35), and to add other additional constraints if necessary, after every design iteration so that the specifications in (34) are fulfilled. More concretely, one design iteration consists of (i) gradually changing, toward the achievement of specifications, $\boldsymbol{s}$ and/or current additional conditions $S(\lambda_k) = \eta_k$, (ii) introducing new additional conditions if necessary, (iii) adopting an initial point $\boldsymbol{\rho}$ for NLSsolver as the minimizer of the previous design if the design is not bad, and (iv) designing a new controller and checking the performance in the Bode plot. After a number of design iterations, with four additional conditions (see the circles in Figure 9), we have obtained a controller and a sensitivity function as

$$\begin{aligned} C(s) &= \frac{\begin{array}{r}0.2238s^7 + 8.875s^6 + 82.58s^5 + 965.5s^4 \\ +4231s^3 + 7977s^2 + 8233s + 5493\end{array}}{\begin{array}{r}s^7 + 9.281s^6 + 85.02s^5 + 291.9s^4 \\ +741.3s^3 + 551.1s^2 + 594.1s + 96.21\end{array}}, \\ S(s) &= \frac{s^5 + 4.281s^4 + 13.61s^3 + 9.814s^2 + 11.69s + 1.924}{s^5 + 4.281s^4 + 14.06s^3 + 24.43s^2 + 24.97s + 23.01}. \end{aligned}$$

The corresponding Bode plots of $S$ and $T := 1 - S$ are shown in Figure 10, with the design result in [19, Chapter 6.2]. Although the complementary sensitivity slightly violates the requirements over high frequencies, we have obtained much lower gain in those frequencies than that designed in the book [19]. Note that the degree of controller is seven, comparable to the controller degree in [19, p. 80], which was eight.

Finally, we remark that there are many ways to tune our design parameters; for example, we can also manipulate weights to obtain a satisfactory result.

**Remark 5.6** At this point, it is quite heuristic to select $(\lambda_k, \eta_k)$ for additional constraints, even though we have some guidelines for the selections as was presented

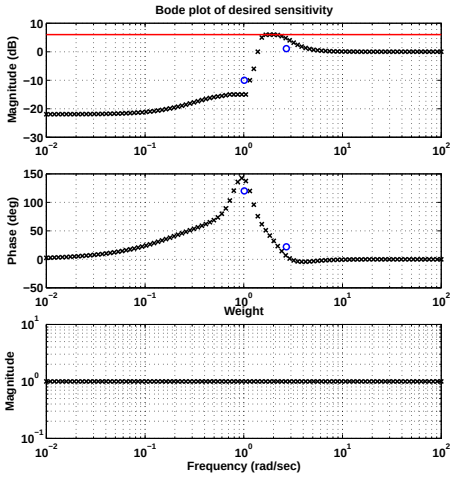


Figure 9: Design parameters in the final design. The horizontal line in the uppermost figure is the level $\gamma = 2$, and the circles correspond to additional conditions.

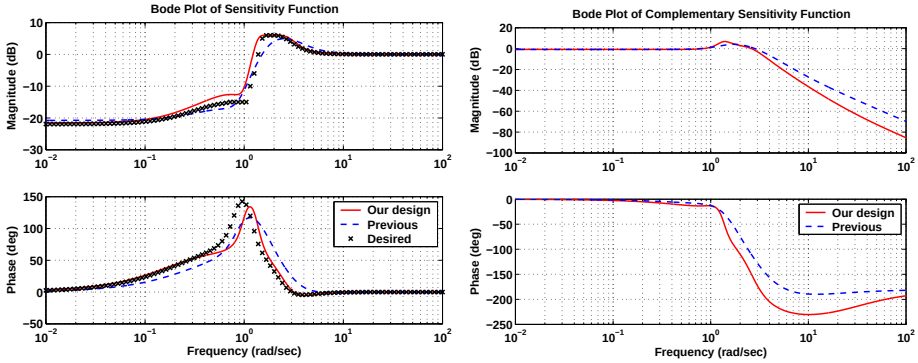


Figure 10: Bode plots of $S$ and $T := 1 - S$

in [26]. How to select these design parameters in a certain optimal sense is still an open question.

# 6 Conclusions

In this paper, we have proposed a new approach to design the sensitivity function in the frequency domain. We have formulated a sensitivity shaping problem, and reduced it to a finite dimensional constrained nonlinear least-squares optimization problem. To solve this problem, we have modified the Gauss-Newton and the Levenberg-Marquardt methods to incorporate the constraint. Numerical examples from the control literature have demonstrated the usefulness of the proposed method in designing relatively low degree controllers. We have developed a user-friendly software for the sensitivity shaping based on the developed theory. A multivariable

extension of the proposed sensitivity shaping method is currently under investigation.

# A Construction of the matrix $K$

In this appendix, we will derive the matrix $K$ in Equation 7. The derivation is based on interpolation conditions of a sensitivity function, which are caused by the internal stability condition, as well as by additional interpolation constraints. See also [27, Section 2].

Suppose that a real rational plant $P$ has relative degree $r$, unstable poles $p_j$ of multiplicity $\ell_j$, $j = 1, \ldots, N_p$, and unstable zeros $z_j$ of multiplicity $m_j$, $j = 1, \ldots, N_z$. Then, for the condition (C1) in Section 2 (i.e., internal stability condition), the sensitivity function must satisfy the following interpolation and derivative conditions (see e.g., [19, 12]):

$$\begin{array}{l} S(p_j) = S'(p_j) = \cdots = S^{(\ell_j - 1)}(p_j) = 0, \quad j = 1, \ldots, N_p, \\ S(z_j) = 1, \ S'(z_j) = \cdots = S^{(m_j - 1)}(z_j) = 0, \quad j = 1, \ldots, N_z, \\ S(\infty) = 1, \ \frac{d}{dz} S(z^{-1})\big|_{z=0} = \cdots = \frac{d^{r-1}}{dz^{r-1}} S(z^{-1})\Big|_{z=0} = 0. \end{array} \tag{40}$$

In addition, we assume that the interpolation conditions in (C2) are given as

$$S(\lambda_j) = \eta_j, \quad j = 1, \ldots, n_e. \tag{41}$$

We require that $\{\lambda_j\}_{j=1}^{n_e}$ is a distinct set, and

$$\{\lambda_j\}_{j=1}^{n_e} \cap \left[ \{p_j\}_{j=1}^{N_p} \cup \{z_j\}_{j=1}^{N_z} \right] = \emptyset. \tag{42}$$

Denoting the total number of the interpolation and derivative conditions in (40) and (41) by $n + 1$, our goal here is, by fixing the structure of the sensitivity function as

$$S(z) = \frac{\boldsymbol{z}^T \boldsymbol{\beta}}{\boldsymbol{z}^T \boldsymbol{\alpha}}, \quad \boldsymbol{\alpha} \in \mathbb{R}^{n+1}, \ \boldsymbol{\beta} \in \mathbb{R}^{n+1}, \ \boldsymbol{z} := \left[ z^n, z^{n-1}, \cdots, 1 \right]^T, \tag{43}$$

to derive a relation between the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ so that $S$ in (43) satisfies (40) and (41).

To express the conditions (40) and (41) in a simpler form, we introduce a function

$$f(z) := S(z^{-1}) = \frac{\hat{\boldsymbol{z}}^T \boldsymbol{\beta}}{\hat{\boldsymbol{z}}^T \boldsymbol{\alpha}}, \quad \hat{\boldsymbol{z}} := [1, z, \cdots, z^n]^T. \tag{44}$$

Then, using the formula in [4, Appendix] (or chain rules), the conditions (40) and (41) are translated into the conditions on $f$ as

$$\begin{array}{l} f(1/p_j) = f'(1/p_j) = \cdots = f^{(\ell_j - 1)}(1/p_j) = 0, \quad j = 1, \ldots, N_p, \\ f(1/z_j) = 1, \ f'(1/z_j) = \cdots = f^{(m_j - 1)}(1/z_j) = 0, \quad j = 1, \ldots, N_z, \\ f(0) = 1, \ f'(0) = \cdots = f^{(r-1)}(0) = 0, \\ f(1/\lambda_j) = \eta_j, \quad j = 1, \ldots, n_e. \end{array} \tag{45}$$

We will present, for more general conditions on $f$, the relation between $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in (44) (or equivalently, in (43)).

**Proposition A.1** *Suppose that a function $f$ has a structure in (44) and satisfies the interpolation and derivative conditions*

$$\frac{f^{(k)}(\zeta_j)}{k!} = \omega_{jk}, \quad k = 0, 1, \cdots, M_j, \ j = 0, 1, \ldots, \ell, \tag{46}$$

*where $\zeta_j \neq \zeta_k$ whenever $j \neq k$. Here, the degree $n$ in (43) is taken as the total number of conditions in (46) minus one:*

$$n := \sum_{j=0}^{\ell} M_j - 1. \tag{47}$$

*Then, the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ satisfies the relation*

$$\boldsymbol{\beta} = K\boldsymbol{\alpha}, \quad K := \Gamma^{-1} W \Gamma, \tag{48}$$

*where $\Gamma$ and $W$ are defined as follows: $\Gamma$ is the controllability matrix:*

$$\Gamma := [B, AB, \cdots, A^n B], \tag{49}$$

*for the pair $(A, B)$ defined by*

$$\begin{aligned} A &:= \begin{bmatrix} A_0 & & & \\ & A_1 & & \\ & & \ddots & \\ & & & A_\ell \end{bmatrix}, \quad \text{where } A_j := \begin{bmatrix} \zeta_j & & & \\ 1 & \zeta_j & & \\ & \ddots & \ddots & \\ & & 1 & \zeta_j \end{bmatrix} \in \mathbb{C}^{M_j \times M_j}, \\ B &:= \begin{bmatrix} B_0 \\ B_1 \\ \vdots \\ B_\ell \end{bmatrix}, \qquad\qquad \text{where } B_j := \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{M_j}, \end{aligned} \tag{50}$$

*and $W$ is a block diagonal matrix:*

$$W := \begin{bmatrix} W_0 & & & \\ & W_1 & & \\ & & \ddots & \\ & & & W_\ell \end{bmatrix}, \ \text{where } W_j := \begin{bmatrix} \omega_{j0} & & & \\ \omega_{j1} & \omega_{j0} & & \\ \vdots & \ddots & \ddots & \\ \omega_{jM_j} & \cdots & \omega_{j1} & \omega_{j0} \end{bmatrix}. \tag{51}$$

**Remark A.2** Note that this proposition assumes neither any analyticity of the interpolant $f$ nor the region that the interpolation points and values lie.

***Proof.*** Since $f$ is supposed to have a structure in (44), we have $\hat{\boldsymbol{z}}^T \boldsymbol{\beta} = f(z) \hat{\boldsymbol{z}}^T \boldsymbol{\alpha}$. By taking the first $(M_j - 1)$-derivatives of this equation and by evaluating them at $z = \zeta_j$, it is straightforward to obtain

$$\Gamma_j \boldsymbol{\beta} = W_j \Gamma_j \boldsymbol{\alpha}, \quad \Gamma_j := \left[B_j, A_j B_j, \cdots, A_j^n B_j\right], \tag{52}$$

where $W_j$ is defined in (51). From the structures of $A_j$ and $B_j$, it is easy to verify that the matrix $\Gamma_j$ is full row rank. By stacking the equation (52) for $j = 0, 1, \ldots, \ell$, we can have

$$\Gamma\boldsymbol{\beta} = W\Gamma\boldsymbol{\alpha}, \tag{53}$$

where $\Gamma$ and $W$ are defined in (49) and (51), respectively. Since the matrix $\Gamma_j$ is full row rank and $\{\zeta_j\}_{j=0}^{\ell}$ is assumed to be distinct, the matrix $\Gamma$ is nonsingular, and thus we have completed the proof. *Q.E.D.*

From this proposition, we have an immediate corollary.

**Corollary A.3** *Suppose that a sensitivity function $S$ satisfies the conditions (C1)–(C4) in Section 2. Let $K$ be a matrix defined in (48) corresponding to the conditions (C1) and (C2) (or equivalently, (40) and (41)). Then, the matrices $\gamma I + K$ and $\gamma I - K$ are invertible.*

***Proof.*** Due to the norm condition (C3) in Section 2, the absolute values of the diagonal elements of the matrix $W$ in (51) are less than $\gamma$. Noting that $W$ is triangular and that the eigenvalues of $K$ coincide with those of $W$, we conclude that the matrices $\gamma I + K$ and $\gamma I - K$ are invertible. *Q.E.D.*

## Acknowledgment

The authors would like to thank Prof. A. Lindquist, Prof. K. Svanberg, Prof. A. Forsgren and Prof. Ulf Jönsson at the Royal Institute of Technology for many invaluable discussions and comments.

## References

[1] A Blomqvist, G Fanizza, and R Nagamune. Computation of bounded N-P interpolants by solving nonlinear equations. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 4511–4516, Maui, Hawaii, December 2003.

[2] A. Blomqvist, A. Lindquist, and R. Nagamune. Matrix-valued Nevanlinna-Pick interpolation with complexity constraint: An optimization approach. *IEEE Trans. Automat. Control*, 48(12):2172–2190, December 2003.

[3] A. Blomqvist and R. Nagamune. Matlab interface "Sshaper". The MATLAB code is available at `http://www.math.kth.se/~andersb/software.html`.

[4] A. Blomqvist and R. Nagamune. Optimization-based computation of analytic interpolants of bounded complexity. submitted for publication.

[5] A. Blomqvist and R. Nagamune. An extension of a Nevanlinna-Pick interpolation solver to cases including derivative constraints. In *Proceeding of the 41st IEEE Conference on Decision and Control*, pages 2552–2557, Las Vegas, Nevada, December 2002.

[6] C. I. Byrnes, T. T. Georgiou, and A. Lindquist. A generalized entropy criterion for Nevanlinna-Pick interpolation with degree constraint. *IEEE Trans. Automat. Control*, 46(6):822–839, 2001.

[7] C. I. Byrnes, S. V. Gusev, and A. Lindquist. A convex optimization approach to the rational covariance extension problem. *SIAM J. Contr. and Optimiz.*, 37(1):211–229, 1998.

[8] C. I. Byrnes and A. Lindquist. On the duality between filtering and Nevanlinna-Pick interpolation. *SIAM J. Contr. and Optimiz.*, 39(3):757–775, 2000.

[9] C. I. Byrnes, A. Lindquist, S. V. Gusev, and A. S. Matveev. A Complete Parameterization of All Positive Rational Extensions of a Covariance Sequence. *IEEE Trans. Automat. Control*, 40(11):1841–1857, November 1995.

[10] C. I. Byrnes, A. Lindquist, and Y. Zhou. On the nonlinear dynamics of fast filtering algorithms. *SIAM J. Contr. and Optimiz.*, 32:744–789, 1994.

[11] J. C. Doyle, B. A. Francis, and A. R. Tannenbaum. *Feedback Control Theory.* MacMillan Publishing Company, 1992.

[12] J. S. Freudenberg and D. P. Loose. Right Half Plane Poles and Zeros and Design Tradeoffs in Feedback Systems. *IEEE Trans. Automat. Control*, 30(6):555–565, 1985.

[13] P. Gahinet and P. Apkarian. A linear matrix inequality approach to $H_\infty$ control. *Int. J. Robust and Nonlinear Control*, 4:421–448, 1994.

[14] T. T. Georgiou. *Partial Realization of Covariance Sequences.* PhD thesis, University of Florida, Gainesville, 1983.

[15] T. T. Georgiou. Realization of power spectra from partial covariance sequences. *IEEE Trans. Acoustics, Speech and Signal Processing*, 35:438–449, 1987.

[16] T. T. Georgiou. A topological approach to Nevanlinna-Pick interpolation. *SIAM J. Math. Anal.*, 18(5):1248–1260, 1987.

[17] T. T. Georgiou. The interpolation problem with a degree constraint. *IEEE Trans. Automat. Control*, 44(3):631–635, 1999.

[18] T. T. Georgiou and A. Lindquist. Kullback-Leibler approximation of spectral density functions. *IEEE Trans. Inform. Theory*, 49(11):2910–2917, 2003.

[19] J. W. Helton and O. Marino. *Classical Control Using $H^\infty$ Methods.* SIAM, 1998.

[20] D. Henrion. LMI optimization for fixed-order $H_\infty$ controller design. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 4646–4651, Maui, Hawaii USA, December 2003.

[21] D. Henrion, M. Šebek, and V. Kučera. Positive polynomials and robust stabilization with fixed-order controllers. *IEEE Trans. Automat. Control*, 48(7):1178–1186, July 2003.

[22] I. S. Horowitz. *Quantitative Feedback Design Theory*, volume 1. QFT publication, Boulder, Colorado, 1992.

[23] T. Iwasaki and R. E. Skelton. All Controllers for the General $H_\infty$ Control Problem: LMI Existence Conditions and State Space Formulas. *Automatica*, 30(8):1307–1317, 1994.

[24] D. McFarlane and K. Glover. A Loop Shaping Design Procedure Using $H_\infty$ Synthesis. *IEEE Trans. Automat. Control*, 37(6):759–769, June 1992.

[25] Jorge J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. Statist. Comput.*, 4(3):553–572, 1983.

[26] R. Nagamune. Closed-loop shaping based on the Nevanlinna-Pick interpolation with a degree bound. *IEEE Trans. Automat. Control*, 49(2):300–305, February 2004.

[27] R. Nagamune. A shaping limitation of rational sensitivity functions with a degree constraint. *IEEE Trans. Automat. Control*, 49(2):269–300, February 2004.

[28] R. Nagamune and A. Blomqvist. Sensitivity shaping with degree constraint by nonlinear least-squares optimization. Technical report, Optimization and Systems Theory, Royal Institute of Technology, 2004.

[29] R. Nagamune and A. Lindquist. Sensitivity shaping in feedback control and analytic interpolation theory. In J. L. Menaldi, E. Rofman, and A. Sulem, editors, *Optimal Control and Partial Differential Equations*. IOS Press, 2001.

[30] S. Nash and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, 1996.

[31] W. Rudin. *Principles of Mathematical Analysis, Third edition*. McGraw-Hill, New York, 1976.

[32] T. Söderström and P. Stoica. *System Identification*. Prentice Hall, 1989.