Exact and causal inversion of nonminimum-phase systems

Citation for published version (APA):

DOI:
10.1109/TMECH.2019.2945865

Document status and date:
Published: 01/12/2019

Document Version:
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 25. Jan. 2020
Exact and Causal Inversion of Nonminimum-Phase Systems: A Squaring-Down Approach for Overactuated Systems

Jurgen van Zundert, Fons Luijten, Tom Oomen, Senior Member, IEEE

Abstract—Nonminimum-phase zeros pose challenges for controller design, for example in inversion-based control approaches where inverting these zeros may result in “unstable” poles. The aim of this paper is to exploit the additional freedom in overactuated systems to eliminate these zeros, facilitating subsequent inversion. In particular, an approach for causal and exact inversion of systems with nonminimum-phase behavior is presented. In an either static or dynamic squaring-down step prior to inversion, the approach exploits the fact that non-square systems typically have no invariant zeros. The proposed approach is successfully demonstrated in experiments on an overactuated motion system. The method enables exact inversion for non-square systems without requiring preview or pre-actuation.

Index Terms—Causal feedforward, Nonminimum-phase behavior, Overactuation, Squaring down, Tracking control.

I. INTRODUCTION

System inversion is at the heart of achieving high performance in many control applications, including printing systems [1], atomic force microscopes [2], and wafer stages [3]. It is extensively used in, for example, inverse model feedforward [4] and iterative learning control (ILC) [5]. See also, for example, [6] for a recent overview and comparison of system inversion approaches.

Many systems have more actuators or sensors, for example flight control systems [7]–[9], dual-stage actuators [10], marine vessels [11], [12], wafer stages [13], ground vehicles [14]–[16], and many more [17]. For such systems, system inversion is often performed after the system is cast in a square system, see, for example, [18]. This can be done by, for example, choosing outputs such that they coincide with performance variables [19] or choosing inputs such that the system is decoupled [20]. This facilitates the design of decentralized feedback control [18], [21], as well as decentralized ILC [22].

One of the main challenges in system inversion is nonminimum-phase behavior, which is related to the invariant zeros of the system being “unstable”. Indeed, causal inversion of nonminimum-phase systems yields unbounded signals. To avoid unbounded signals, inversion techniques have been developed, see, for example, [6] for a recent overview. Traditionally, approximate inversion techniques such as ZPETC, ZMETC and NPZ-Ignore [23] are used, but such approaches inherently yield non-exact tracking. Stable inversion [6, Section 4.2] yields exact tracking, but for nonminimum-phase systems the approach requires preview, i.e., a priori knowledge of the reference trajectory, and pre-actuation, i.e., a non-zero input before the start of the trajectory. Furthermore, the approach is restricted to square systems. Optimization approaches such as norm-optimal feedforward [24] require preview, which is not always available. Synthesis approaches such as $H_2$-preview control and $H_{\infty}$-preview control [6, Section 4.4] are only optimal for trajectories with a specific frequency spectrum or address a control goal other than tracking error minimization, respectively.

Although there are many inversion techniques that yield bounded inputs for nonminimum-phase systems, they either require preview or pre-actuation. The aim of this paper is to develop a new approach for system inversion that reconsiders traditional choices for squaring down, i.e., aiming to decouple systems as is outlined above. In particular, the approach considers new criteria for squaring down in view of system inversion, exploiting the additional design freedom available in overactuated systems.

The main contribution of this paper is a causal inverse model feedforward solution for overactuated systems, also for systems with nonminimum-phase behavior. The approach exploits the design freedom at the input side to square down the non-square system to a square minimum-phase system, which enables exact causal inversion. A key aspect is that non-square systems generally do not have any invariant zeros. The following subcontributions are identified: (I) coordinate basis of non-square systems revealing key properties for squaring down; (II) a squaring-down approach with a static compensator; (III) a squaring-down approach with a dynamic compensator; (IV) systematic design framework for design of inverse model feedforward for overactuated systems; and (V) experimental validation of the design framework on an overactuated system with nonminimum-phase behavior. Initial results can be found in [25]. The present paper has several new contributions, including Contribution (I), application to systems with direct feedthrough, and Contribution (V). Related work on overactuated systems addressing different aspects includes [26], [27], see also [17].

The outline of the paper is as follows. In Section II, the control problem is formulated. In Section III, the main idea of the approach is presented together with preliminary results. The static squaring-down approach is presented in Section IV,
with extension to dynamic squaring down in Section V. An overview of the complete approach is presented in Section VI. In Section VII, the experimental validation on a motion system is presented. Conclusions are presented in Section VIII.

II. PROBLEM FORMULATION

In this section, the challenges associated to inversion of (non-)square systems are illustrated through several examples, which leads to the problem considered in this paper. In addition, the experimental motion system that is used for validation of the results in Section VII is presented.

A. Motivation for inversion of (non-)square systems

Consider the inversion diagram in Fig. 1, where discrete-time system $H$ is assumed to be stable and has the minimal realization

$$
H : \begin{align*}
x[k + 1] &= Ax[k] + Bu[k], \\
y[k] &= Cx[k] + Du[k],
\end{align*}
$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$. Without loss of generality, it is assumed that $\text{rank}([B]) = \bar{m}$ and $\text{rank}([C \ D]) = \bar{p}$. The system $H$ is stable if and only if $|\lambda_i(A)| < 1$, for all $i = 1, 2, \ldots, n$, where $\lambda_i$ denotes the $i$th eigenvalue. The following definition is adopted.

Definition 1 (Invariant zeros [28], [29]): The invariant zeros of $H$ in (1) are given by

$$
z = \{z_i \in \mathbb{C} \mid \text{rank} \{RSM(z_i)\} < \max_{z \in \mathbb{C}} \text{rank} \{RSM(z)\}\},
$$

with

$$
RSM(z) = \begin{bmatrix} zI_n - A & -B \\ C & D \end{bmatrix}.
$$

The system $H$ in (1) is minimum phase if and only if all invariant zeros, see Definition 1, satisfy $|z_i| < 1$, for all $i$. System $H$ is unstable (resp. nonminimum phase) if it is not stable (resp. minimum phase). If $|z_i| = 1$ for some $i$, techniques as in [30] can be used. It is assumed that $H$ is right invertible as defined by Definition 2.

Definition 2 (Invertibility): System $H$ in (1) is

- left invertible if and only if $r_m = \bar{m}$,
- right invertible if and only if $r_m = \bar{p}$,
- invertible if and only if $r_m = \bar{m} = \bar{p}$,

with $r_m = \max_{z \in \mathbb{C}} \text{rank} \{D + C(zI_n - A)^{-1}B\}$.

Given Fig. 1 and $H$, the objective is to let output $y$ track the reference trajectory $r$, while input $u$ remains bounded. The following examples illustrate that the inversion of $H$ is nontrivial.

Example 1: Let $H$ be scalar and given by

$$
H = \frac{(z-0.6)(z+2)}{(z-0.1)(z+0.8)(z-0.4)},
$$

then

$$
F = H^{-1} = \frac{(z-0.1)(z+0.8)(z-0.4)}{(z-0.6)(z+2)}
$$

is unique, but unstable (pole at $z = -2$) as a consequence of $H$ being nonminimum phase. Hence, causal filtering with $F$ leads to unbounded signals.

Example 1 shows the challenges associated with nonminimum-phase behavior. The following example illustrates the key concepts of this paper.

Example 2: Let $H$ be non-square and given by

$$
H = \frac{1}{(z-0.1)(z+0.8)(z-0.4)} \begin{bmatrix} (z-0.6)(z+2) & (z-5) \end{bmatrix},
$$

then $F$ such that $HF = 1$ is non-unique. The inverses

$$
F_1 = \begin{bmatrix} (z-0.1)(z+0.8)(z-0.4) \\ 0 \end{bmatrix},
$$

$$
F_2 = \begin{bmatrix} 0 \\ (z-5) \end{bmatrix}
$$

may lead to unbounded responses since they are unstable. In contrast, the inverse

$$
F = \begin{bmatrix} 1 & -0.3 \end{bmatrix} \begin{bmatrix} (z-0.1)(z+0.8)(z-0.4) \\ (z+0.8)(z+0.9) \end{bmatrix},
$$

allows for bounded causal solutions since $F$ is stable.

The main point of Example 2 is that for systems with nonminimum-phase behavior, selecting either one of the inputs may yield unstable inverses, see (6) and (7). Interestingly, a smart combination of inputs as in (8) yields a stable inverse. This is the main idea of this paper and exploits the fact that in general non-square systems are minimum phase, regardless of the properties of the individual transfer functions. However, the solution is not always as straightforward as in Example 2, as shown by the following example.

Example 3: Let $H$ be non-square and given by

$$
H = \frac{1}{(z-0.1)(z+0.8)(z-0.4)} \begin{bmatrix} (z-0.6)(z+2) & (z-5)(z+0.9) \end{bmatrix},
$$

then $F$ such that $HF = 1$ is non-unique. The only difference with Example 2 is one additional minimum-phase zero at the second input. For (9), the design of a stable $F$ such that $HF = 1$ is not as straightforward as in (8) in Example 2.

The examples show that inversion of $H$ is nontrivial if $H$ is nonminimum phase or non-square. Example 2 shows that additional freedom in the inputs can be exploited to create a stable system that yields exact inversion. However, such a design is not straightforward as illustrated by Example 3. In the next sections, a systematic design framework for inversion of overactuated systems is presented. A design for Example 3 is presented later on.

Remark 3: In this paper, discrete-time systems are considered since this allows for a digital controller implementation. Results for continuous-time systems follow along the same lines.

B. Case study: Experimental motion system

In this section, the experimental overactuated motion system is presented that is used as a case study to evaluate the proposed inversion approach. The experimental results are presented in Section VII.
The motion system consists of a thin, suspended beam and is shown in Fig. 2. For the experiments, the actuators \( u_1, u_2 \) and encoder \( y \) are used. The system has a stroke of 1 mm. The reference trajectory \( r \) is a forward and backward movement over 0.7 mm as shown in Fig. 3.

The measured frequency response function of the system is shown in Fig. 4 together with the identified 14th order parametric model \( H \) in (1) obtained using the procedure in [31]. The zeros and poles of \( H \) are listed in Table I, together with the zeros of the individual transfers of \( H \). Both transfer functions \( u_1 \rightarrow y \) and \( u_2 \rightarrow y \) are nonminimum phase and hence only using one input would require pre-actuation to ensure bounded signals, see also Example 2. In contrast, the non-square system \( H \) is minimum phase and hence exact inversion with bounded inputs and without pre-actuation is possible.

III. CAUSAL AND STABLE INVERSION FOR NON-SQUARE SYSTEMS

In this section, the main idea of the proposed approach is presented together with preliminary results.

A. Main idea: Squaring down in view of system inversion

Squaring down is a standard step in control applications. Typical considerations include decoupling. In this paper, a systematic design approach is presented that exploits the additional freedom of overactuated systems \( H \) in Fig. 1 in view of system inversion, i.e., with \( m > \bar{p} \), to obtain stable \( F \) such that \( HF = I \) and hence \( y = r \). A key aspect is that non-square systems generally have no invariant zeros. Hence, the invariant zeros of the squared-down system are determined by the squaring-down approach and can be affected.

The tracking control application of Fig. 1 with more inputs than outputs (\( \bar{m} > \bar{p} \)) is shown in Fig. 5(a), with input \( u \in \mathbb{R}^\bar{m} \) divided into \( u_1 \in \mathbb{R}^\bar{p} \) and \( u_2 \in \mathbb{R}^{m-p} \). The design approach consists of two steps as shown in Fig. 5(b). First, a precompensator \( \hat{K}_{\text{pre}} \) is designed such that

\[
\hat{\Sigma}_{\text{sq}} = H\hat{K}_{\text{pre}} \tag{10}
\]

is square with dimensions \( \bar{p} \times \bar{p} \). Second, \( F \) is selected as

\[
F = \hat{K}_{\text{pre}}\hat{\Sigma}_{\text{sq}}^{-1} \tag{11}
\]

such that perfect tracking is obtained since

\[
y = HF r = H\hat{K}_{\text{pre}}\hat{\Sigma}_{\text{sq}}^{-1} r = H\hat{K}_{\text{pre}}^{-1}(H\hat{K}_{\text{pre}})^{-1} r = r. \tag{12}
\]

Note that the second step is straightforward once the first step is completed.

The proposed approach is also used in Example 2 in Section II where squaring down is performed by the static precompensator \( \hat{K}_{\text{pre}} = [ -0.3 ] \). The precompensator yields a minimum-phase, square system as desired. It is shown later on...
that there does not exist a static precompensator for the system in Example 3 such that the square system is minimum phase. However, there does always exist a dynamic compensator such that the square system is minimum phase. A dynamic compensator design for Example 3 is presented later on.

The precompensator design $K_{\text{pre}}$ introduces additional invariant zeros in the square system $\hat{\Sigma}_{sq}$ in (10). The interest is in stable precompensator designs $\hat{K}_{\text{pre}}$ that yield minimum-phase, square systems $\hat{\Sigma}_{sq}$ as they result in stable $F$ and thus bounded $u$ when using the causal solution. If the compensator is stable, but the square system is nonminimum phase, then inversion techniques are required to compute bounded outputs of $\hat{\Sigma}_{sq}^{-1}$, since the invariant nonminimum-phase zeros become unstable poles in $\hat{\Sigma}_{sq}^{-1}$. Inversion techniques, such as stable inversion, are not preferred since they are noncausal and require pre-actuation and preview.

The critical step in the proposed approach is the design of the compensator $\hat{K}_{\text{pre}}$ in (10) which is presented in the subsequent sections based on the coordinate basis presented in the next section.

### B. Coordinate basis for squaring down

In this section, a coordinate basis is presented to facilitate the design of the precompensator $\hat{K}_{\text{pre}}$ in (10). The basis is specifically developed for use in the squaring-down approach presented in Sections IV and V, in contrast to the more general special coordinate basis (s.c.b.) [32], [33]. The results in this section constitute Contribution (I).

The coordinate basis is developed for left-invertible systems. The right-invertible system $H \doteq (A, B, C, D)$ in Fig. 5(a) is transformed to a left-invertible system by considering its dual $\hat{\Sigma} = H_d$ given by

$$
\hat{\Sigma} : \begin{bmatrix} \hat{x}[k+1] \\ \hat{y}[k] \end{bmatrix} = \begin{bmatrix} \hat{A} & \hat{B} \\ \hat{C} & \hat{D} \end{bmatrix} \begin{bmatrix} \hat{x}[k] \\ \hat{u}[k] \end{bmatrix},
$$

with $(\hat{A}, \hat{B}, \hat{C}, \hat{D}) = (A^T, C^T, B^T, D^T)$ and $\hat{x} \in \mathbb{R}^n$, $\hat{y} \in \mathbb{R}^p$, $\hat{u} \in \mathbb{R}^m$, where $p = m$, $m = \bar{p}$. Note that since $H$ is right invertible, $\hat{\Sigma} = H_d$ is left invertible, see also Definition 2.

In the related work [32], [33] the special coordinate basis (s.c.b.) is used. Existing algorithms to obtain this special structure, for example in [34, Chapter 12], are experienced to be numerically inaccurate. Instead, another structure is used which only enforces those features that are required for the squaring-down approach and thereby results in numerically more accurate results. The coordinate basis that is used is given by Theorem 4. See Appendix A for the construction of the basis.

**Theorem 4 (Coordinate basis):** Let $\Sigma = \Gamma^{-1}_o \hat{\Sigma}_i$, with $\hat{\Sigma}$ in (13) and

$$
\hat{y} = \Gamma_o \begin{bmatrix} y_f \\ y_s \end{bmatrix}, \quad \hat{u} = \Gamma_i u,
$$

with non-singular $\Gamma_o \in \mathbb{R}^p$, $\Gamma_i \in \mathbb{R}^m$ and $y_f = \begin{bmatrix} y_{0f} \\ y_{sf} \end{bmatrix}$, $y_{0f} \in \mathbb{R}^{m_0}$, $y_{sf} \in \mathbb{R}^{m-m_0}$, $y_s \in \mathbb{R}^{p-m}$, and $u = \begin{bmatrix} u_0 \\ u_s \end{bmatrix}$, $u_0 \in \mathbb{R}^{m_0}$, $u_s \in \mathbb{R}^{m-m_0}$, where $m_0 = \text{rank}(D)$. Then, $\Sigma$ satisfies the coordinate basis

$$
x_a[k+1] = A_{aa} x_a[k] + A_{af} y_f[k],
$$

$$
x_b[k+1] = A_{bb} x_b[k] + A_{bf} y_f[k],
$$

$$
x_f[k+1] = A_{fa} x_a[k] + A_{fb} x_b[k] + B_f u_x[k] + L_f y_f[k],
$$

$$
y_{0f}[k] = C_{0f} x_a[k] + C_{0f} x_b[k] + C_{0f} x_f[k] + u_0[k],
$$

$$
y_{sf}[k] = x_f[k],
$$

$$
y_s[k] = C_{sb} x_b[k].
$$

The coordinate basis in Theorem 4 reveals several features that are used in the squaring-down approach in subsequent sections. For example, the invariant zeros, see Definition 1, of $\Sigma$ in Theorem 4 are given by $\lambda(A_{aa})$. In the next section, the coordinate basis in Theorem 4 is exploited in the squaring-down approach to obtain the compensator $\hat{K}_{\text{pre}}$ in (10).

### IV. CAUSAL FEEDFORWARD THROUGH STATIC SQUARING DOWN

In this section, the static squaring-down approach is presented, which constitutes Contribution (II). The approach exploits properties of the coordinate basis presented in Section III, and in particular Theorem 4. First, the static compensator design is presented. Second, the static compensator design is applied to the examples in Section II. The extension to a dynamic compensator design is presented in Section V. An overview of the complete feedforward design approach for $H$ in Fig. 5(a) is presented in Section VI.

#### A. Static compensator

The concept of squaring down a left-invertible system $\Sigma$ in (15) is illustrated in Fig. 6. The outputs $y_f \in \mathbb{R}^m$, $y_s \in \mathbb{R}^{p-m}$ are combined into a new output $\hat{y} \in \mathbb{R}^m$ through a postcompensator as $\hat{y} = K_{\text{post}} \begin{bmatrix} y_f \\ y_s \end{bmatrix}$, such that $\Sigma_{sq} = K_{\text{post}} \Sigma$ is square with dimensions $m \times m$. The main idea is to design $K_{\text{post}}$ such that $\Sigma_{sq}$ is invertible and nonminimum phase to enable direct inversion.

The precompensator $K_{\text{pre}}$ for the right-invertible dual system $\Sigma_d$ is obtained as the dual of the postcompensator of $\Sigma$, i.e., $K_{\text{pre}} = K_{\text{post}, d}$. The precompensator $K_{\text{pre}}$ for system $H$ in Fig. 5(b) is given by

$$
\hat{K}_{\text{pre}} = \Gamma_o^{-\top} K_{\text{pre}} = \Gamma_o^{-\top} K_{\text{post}, d}.
$$
The design of static compensator design $K_{\text{post}}$ in Fig. 6 is given by Theorem 5 and illustrated in Fig. 7(a).

**Theorem 5 (Static compensator):** Given $H$ in (1) and $\Sigma$ in (15), the static compensator is given by

$$K_{\text{post}} = \begin{bmatrix} I_m & L \end{bmatrix},$$

(17)

i.e., $\tilde{y} = y_f + Ly_s$, with $L \in \mathbb{R}^{m \times (p - m)}$.

**Proof:** The proof follows along similar lines as for continuous-time systems based on the s.c.b. in [35, III.A].

Properties of the square system $\Sigma_{sq}$ for the static compensator in Theorem 5 are provided by Theorem 6, where $n_a, n_b$ are the dimensions of $x_a, x_b$ in (15), respectively.

**Theorem 6 (Properties $\Sigma_{sq}$ static compensator):** Given $H$ in (1) and $\Sigma$ in (15), the square system $\Sigma_{sq}$ in Fig. 6 with $K_{\text{post}}$ the static compensator of Theorem 5 has the following properties.

- Invertible;
- $n_a + n_b$, invariant zeros: $\lambda(A_{aa}) + \lambda(A_{bb} - A_{bf}LC_{sb})$;
- $n$ poles: $\lambda(\tilde{A})$.

**Proof:** Substitution of $y_f = \tilde{y} - Ly_s$ in (15) and obtaining the coordinate basis in Theorem 4 for the new system shows that $\tilde{x}_a = \begin{bmatrix} x_a \\ x_b \end{bmatrix}$ with $A_{aa} = \begin{bmatrix} A_{aa} & -A_{af}LC_{sb} \\ 0 & A_{bb} - A_{bf}LC_{sb} \end{bmatrix}$ from which the results follow directly.

Theorem 6 shows that the static compensator introduces new invariant zeros in addition to the invariant zeros $\lambda(A_{aa})$ of $\Sigma$. When inverting the square system $\Sigma_{sq}$, these zeros become poles. Hence, these zeros are preferred to be minimum phase to avoid the use of inversion techniques which require pre-actuation. The zeros $\lambda(A_{aa})$ are fixed, but since $\Sigma$ is non-square, there are typically no (nonminimum-phase) zeros. Theorem 7 shows how the additional zeros can possibly be placed through static output feedback.

**Theorem 7 (Invariant zero placement static compensator):**

The invariant zeros introduced by the static compensator in Theorem 5 can possibly be placed by solving the static output feedback problem for the triplet $(A_{bb}, A_{bf}, C_{sb})$.

**Proof:** By Theorem 6, the additional invariant zeros are given by $\lambda(A_{bb} - A_{bf}LC_{sb})$ and thus affected by $L$.

These zeros are also the poles of the state-space system $(A_{bb}, A_{bf}, C_{sb})$ with static output feedback gain $-L$.

Importantly, the static output feedback problem in Theorem 7 is not always solvable [36]. This is also shown through examples in the next section.

### B. Application of the static compensator

In this section, the static compensator design of Theorem 5 is applied to Example 2 and Example 3 in Section II.

**Example 2 (continued):** The dual left-invertible system, see also Section III-B, of $H$ in (5) is given by

$$H_d = \frac{1}{(z - 0.6)(z + 2)(z - 1)} \left( \frac{z - 0.6}{z - 1} \right),$$

(18)

with $m = 1, p = 2$. System $\Sigma = H_d$ in the coordinate basis of Theorem 4 is given by

$$x_b[k+1] = \begin{bmatrix} 0.5935 & 0.1569 \\ 0.1069 & -1.9935 \end{bmatrix} x_b[k] + \begin{bmatrix} -1.9347 \\ 3.5010 \end{bmatrix} y_f[k],$$

$$x_f[k+1] = \begin{bmatrix} 0.0580 \\ -0.6477 \end{bmatrix} x_b[k] + 1.1y_f[k] + u_x[k],$$

$$y_f[k] = x_f[k],$$

$$y_s[k] = \begin{bmatrix} 0.9512 \\ 0.8113 \end{bmatrix} x_b[k],$$

(19a) \hspace{1cm} (19b) \hspace{1cm} (19c) \hspace{1cm} (19d)

with $\Gamma_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. The coordinate basis confirms $\Sigma$ has no invariant zeros since $x_a$ is non-existing. Selecting $L = -0.3$ in Theorem 5 yields $K_{\text{post}} = \begin{bmatrix} 1 \\ -0.3 \end{bmatrix}$ and $\lambda(A_{bb} - A_{bf}LC_{sb}) = \{-0.5, -0.6\}$. Hence, by (16), $K_{\text{pre}} = \begin{bmatrix} 1 \\ -0.3 \end{bmatrix}$ which yields controller $F$ in (8).

**Example 3 (continued):** The dual left-invertible system $H_d$, see Section III-B, of $H$ in (9) in the coordinate basis of Theorem 4 is given by

$$x_b[k+1] = \begin{bmatrix} 0.5137 \\ 0.1257 \end{bmatrix} x_b[k] + \begin{bmatrix} 1.7257 \\ -1.9137 \end{bmatrix} y_f[k],$$

$$x_f[k+1] = \begin{bmatrix} 0.0445 \\ -0.3942 \end{bmatrix} x_b[k] + 1.1y_f[k] + u_x[k],$$

$$y_f[k] = x_f[k],$$

$$y_s[k] = \begin{bmatrix} 1.1556 \\ 0.2525 \end{bmatrix} x_b[k],$$

(20a) \hspace{1cm} (20b) \hspace{1cm} (20c) \hspace{1cm} (20d)

with $\Gamma_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. It can be verified that there does not exist an $L \in \mathbb{R}$ such that $|\lambda(A_{bb} - A_{bf}LC_{sb})| < 1$. Hence, for this system, there does not exist a static compensator that yields a square system which is minimum phase.

Theorem 7 shows that the problem of designing a postcompensator reduces to solving a static output feedback problem. However, the static output feedback problem does not always have a solution, as also illustrated by Example 3 above. Therefore, in the next section a dynamic compensator is constructed for which the additional invariant zeros can always be placed arbitrarily.
V. CAUSAL FEEDFORWARD: EXTENSION TO DYNAMIC SQUARING DOWN

In the previous section, a static compensator design is presented that yields a square system. However, it is also shown that such a design may not suffice to obtain a minimum-phase square system, as is also illustrated through Example 3. In this section, the compensator design is extended to a dynamic compensator which enables to always obtain a minimum-phase square system. This section constitutes Contribution (III).

The minimum-phase square system is obtained by using an observer to reconstruct the full state $x_b$, followed by state feedback on this observed state $\hat{x}_b$. The resulting dynamic compensator is presented in Theorem 8 and illustrated in Fig. 7(b). Note that the observer poles $\lambda(N)$ can be placed arbitrarily since $(A_{bb}, A_{bf})$ is controllable as $\Sigma$ is a minimal realization. For minimal order observers, see, for example, [37, Section 2.3].

Theorem 8 (Dynamic compensator): Given Fig. 6 with $\Sigma$ in (15), the dynamic compensator $K_{post}$ is given by

$$K_{post} = [I_m \ 0] + J\Sigma_{obs}, \quad (21)$$

i.e., $\tilde{y}[k] = y_f[k] + J\tilde{x}_b[k], \hat{x}_b[k] = \Sigma_{obs} \begin{bmatrix} y_f[k] \\ y_s[k] \end{bmatrix}$, where $\Sigma_{obs}$ denotes a minimal order observer with state matrix $N$ for the matrix triplet $(A_{bb}, A_{bf}, C_{sb})$.

Proof: The proof follows along similar lines as for continuous-time systems based on the s.c.b. in [35, III.A].

Properties of the square system $\Sigma_{sq}$ for the dynamic compensator of Theorem 8 are provided by Theorem 9.

Theorem 9 (Properties $\Sigma_{sq}$ dynamic compensator): Given $H$ in (1) and $\Sigma$ in (15), the minimal realization of the square system $\Sigma_{sq}$ in Fig. 6 with $K_{post}$ the dynamic compensator of Theorem 8 has the following properties.

- Invertible;
- $n_a + n_b$ invariant zeros: $\lambda(A_{aa}) + \lambda(A_{bb} - A_{bf}J)$;
- $n$ poles: $\lambda(A)$.

Proof: The proof follows along similar lines as for Theorem 6.

The additional invariant zeros $\lambda(A_{bb} - A_{bf}J)$ are affected by $J$ and can be arbitrarily placed through static state feedback of the pair $(A_{bb}, A_{bf})$. This requires that $(A_{bb}, A_{bf})$ is controllable which is satisfied since $\Sigma$ is a minimal realization. Hence, if $H$ is minimum phase, which typically holds, $\Sigma_{sq}$ can be made minimum phase enabling the use of direct inversion without pre-actuation.

Next, the dynamic compensator design is applied to Example 3 in Section IV-B.

Example 3 (continued): The subsystem $(A_{bb}, A_{bf}, C_{sb})$ in (20) has two states and one output. Let the desired invariant zeros be $\lambda(N) = 0.7$ and $\lambda(A_{bb} - A_{bf}J) = \{-0.5, -0.6\}$, then pole placement on the pair $(A_{bb}, A_{bf})$ yields $J = [-0.2188, -0.2977]$. The dynamic postcompensator, see Theorem 8, is given by

$$K_{post} = \frac{1}{(z + 0.6)} \begin{bmatrix} 0.291(z - 0.531) \end{bmatrix}, \quad (22)$$

which, by (11), yields

$$F = \begin{bmatrix} (z-0.4)(z+0.8)(z-0.1) \\ (z+0.6)(z+0.3)(z-0.7) \end{bmatrix} \begin{bmatrix} 0.709(z + 1.064) \\ 0.291(z - 0.531) \end{bmatrix}, \quad (23)$$

which is stable as desired. It can be verified that $HF = 1$ and hence perfect tracking is obtained.

In summary, the dynamic postcompensator design in Theorem 8 can always create a stable, minimum-phase, square system, if $H$ in (1) is stable and has no nonminimum-phase invariant zeros, which typically holds for non-square systems. Next, the complete design framework is summarized, including both static and dynamic compensators.

VI. APPLICATION IN TRACKING CONTROL

In this section, the design framework is summarized which constitutes Contribution (IV).

The systematic design framework for design of $F$ in Fig. 5 given a right-invertible system $H$ is shown in Fig. 8. The result follows directly from combining the results of the previous sections. There are two main design types: static and dynamic squaring down. Properties of both are given by Lemma 10 and Lemma 11, respectively.

Lemma 10 (Properties controller with static compensator): Given $H$ in (1) and $\Sigma$ in (15), the minimal realization of $F$ in Fig. 8 based on a static compensator has:

- $n$ invariant zeros: $\lambda(A)$,
- $n_a + n_b$ poles: $\lambda(A_{aa}) + \lambda(A_{bb} - A_{bf}LC_{sb})$.

Proof: The results follow from Theorem 6 and Fig. 8.

Lemma 11 (Properties controller with dynamic compensator): Given $H$ in (1) and $\Sigma$ in (15), the minimal realization of $F$ in Fig. 8 based on a dynamic compensator has:
---

- $n$ invariant zeros: $\lambda(A)$,
- $n_a+2n_b-p+m$ poles: $\lambda(A_{aa})+\lambda(N)+\lambda(A_{bb}-A_{bf}J)$.

**Proof:** The results follow from Theorem 9 and Fig. 8. 

The locations of the invariant zeros of $\Sigma_{sq}$ directly influence the dynamics and the resulting input signals $u_1, u_2$ in Fig. 5. To avoid the use of pre-actuation, these invariant zeros can be made minimum phase.

In the next section, the controller design framework presented in Fig. 8 is validated through experiments.

**VII. EXPERIMENTAL VALIDATION ON AN OVERACTUATED SYSTEM**

In this section, the squaring-down approach is validated in experiments on the overactuated system presented in Section II-B by comparing it to traditional inversion approaches. The results demonstrate the high tracking performance of the squaring-down approach, without requiring pre-actuation or preview, and constitute Contribution (V).

**A. Control design: Squaring down and norm optimal**

The static and dynamic squaring-down solutions are compared with the commonly used (noncausal) norm-optimal feedforward solution and the causal norm-optimal feedforward solution. Table II provides an overview of the controllers and experimental results. In this section, the controller designs are presented. The experimental results are presented in Section VII-B.

The inversion approach in Fig. 8 is used to construct stable controllers to avoid the use of pre-actuation. Note that the $x_a$-dynamics in Theorem 4 are non-existing since $H$ is minimum phase, see Section II-B and in particular Table I. A static compensator design, see Theorem 5, that satisfies this requirement is feasible. The poles of the static squaring-down controller $F_{stat}$ are visualized in Fig. 9 and the Bode plot of $F_{stat}$ is shown in Fig. 12. The input signals $u_1, u_2$ that are generated by the controller are shown in Fig. 10. It turns out that, due to the limited design freedom of a static compensator in combination with the requirement of no pre-actuation, only oscillatory input signals can be generated with a static compensator. Moreover, there is a large difference in the magnitude of $u_1$ and $u_2$, which might be undesired in view of actuator constraints.

To avoid oscillatory input signals and better balance $u_1$ and $u_2$, a dynamic squaring-down controller $F_{dyn}$ is designed. The oscillatory behavior in the inputs generated by $F_{stat}$ is caused by low damping around 30.25 Hz since the corresponding poles of $F_{stat}$ are close to the unit circle, see Fig. 9. Using the additional design freedom of a dynamic compensator compared to a static compensator, the poles of $F_{dyn}$ are placed farther from the unit circle as shown in Fig. 11. The Bode plot of $F_{dyn}$ is shown in Fig. 12. Similar to controller $F_{stat}$, controller $F_{dyn}$ is stable and hence no pre-actuation is required. The input signals that are generated by the controller are shown in Fig. 10 and, as desired, show less oscillatory behavior than those generated by $F_{stat}$.

The squaring-down approaches are compared with norm-optimal feedforward. Norm-optimal feedforward [6, Section 4.3] is directly applicable to non-square systems. However,
the standard causal solution requires preview, i.e., a priori knowledge of trajectory $r$. The weights of the performance criterion, i.e., $Q$ for the error $e = r - y$ and $R$ for the input $\|u_2\|$, see also [6, Section 4.3], are selected to minimize the error without restricting the input signals, see Table II. The input signals are shown in Fig. 10. Importantly, the standard norm-optimal controller requires preview of the entire reference trajectory $r$ as well as pre-action, whereas this is not the case for the causal norm-optimal controller and the static and dynamic squaring-down controllers.

B. Experimental results

In simulation, all controllers achieve perfect tracking. However, in experiments the tracking is non-perfect due to model mismatches as shown by the tracking error signals $e = r - y$ for the different controllers in Fig. 13. Due to pre-action and model mismatches, there is a non-zero error with $F_{\text{norm}}$ for $t < 0$ in Fig. 13(a), i.e., before any movement is required. Since this error is significant, the causal alternative $F_{\text{norm,c}}$ achieves similar performance. Controller $F_{\text{stat}}$ shows oscillatory behavior as a consequence of the oscillatory input in Fig. 10 and model mismatches. The smallest error is achieved with the dynamic squaring-down controller $F_{\text{dyn}}$, see also Table II.

The controller designs and experimental results are summarized in Table II. The results show that the dynamic squaring-down controller outperforms the static squaring-down and the norm-optimal controllers. Importantly, both the static and dynamic squaring-down controllers do not require preview, i.e., they do not require a priori knowledge of the reference trajectory, in contrast to the standard norm-optimal controller.

---

**TABLE II**

Experimental results and settings for the different controller designs. $F_{\text{norm}}$, $F_{\text{norm,c}}$ are the norm-optimal controllers with and without preview, respectively, and serve as benchmark to compare the squaring-down controllers $F_{\text{stat}}$, $F_{\text{dyn}}$, which both do not require preview. The dynamic squaring-down controller $F_{\text{dyn}}$ achieves superior performance in terms of both $\|e\|_{\infty}$ and $\|e\|_2$.

<table>
<thead>
<tr>
<th>Label</th>
<th>Type</th>
<th>Reference</th>
<th>Parameters</th>
<th>Preview</th>
<th>$|u_1|_{\infty}$ [A]</th>
<th>$|u_2|_{\infty}$ [A]</th>
<th>$|e|_{\infty}$ [mm]</th>
<th>$|e|_2$ [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{\text{norm}}$</td>
<td>Norm-optimal noncausal</td>
<td>[6, Section 4.3]</td>
<td>$Q = 1; R = \begin{bmatrix} 1 &amp; 0 \ 0 &amp; 1 \end{bmatrix}$</td>
<td>No</td>
<td>0.213</td>
<td>0.126</td>
<td>1.886</td>
<td>0.075</td>
</tr>
<tr>
<td>$F_{\text{norm,c}}$</td>
<td>Norm-optimal causal</td>
<td>[6, Section 4.3]</td>
<td>$Q = 1; R = \begin{bmatrix} 0 &amp; 1 \ 0 &amp; 0 \end{bmatrix}$</td>
<td>No</td>
<td>0.282</td>
<td>0.785</td>
<td>2.025</td>
<td>0.080</td>
</tr>
<tr>
<td>$F_{\text{stat}}$</td>
<td>Static squaring down</td>
<td>Theorem 5</td>
<td>Fig. 9</td>
<td>No</td>
<td>0.308</td>
<td>0.231</td>
<td>1.774</td>
<td>0.064</td>
</tr>
<tr>
<td>$F_{\text{dyn}}$</td>
<td>Dynamic squaring down</td>
<td>Theorem 8</td>
<td>Fig. 11</td>
<td>No</td>
<td>0.201</td>
<td>0.126</td>
<td>1.999</td>
<td>0.075</td>
</tr>
</tbody>
</table>

---

**Fig. 12.** Bode plot of the static squaring-down controller $F_{\text{stat}}$ (---) and the dynamic squaring-down controller $F_{\text{dyn}}$ (---) from input $r$ to outputs $u_1, u_2$.

**Fig. 13.** Error $e = r - y$ in the time and frequency domain for controllers $F_{\text{norm}}$ (---), $F_{\text{norm,c}}$ (---), $F_{\text{stat}}$ (---), and $F_{\text{dyn}}$ (---). Due to model mismatches the tracking is non-perfect. The dynamic squaring-down controller $F_{\text{dyn}}$ achieves the highest performance.

---

VIII. CONCLUSIONS

In this paper, system inversion is investigated and in particular from the perspective of squaring down. This leads to new insights that enable causal and exact feedforward for overactuated systems. An inversion approach is presented that
does not require preview or pre-actuation for nonminimum-phase systems. The approach exploits the additional design freedom in overactuated systems and the fact that non-square systems generally have no invariant zeros. Experimental results demonstrate superior tracking performance without requiring preview of pre-actuation.

The approach provides a systematic method to exploit overactuation to the benefit of control. Besides inverse model feedforward, the approach is also of interest to feedback control design by creating a square system with favorable properties, such as nonminimum-phase behavior.

Future research focuses on taking model uncertainty into account and using the available design freedom in overactuated systems to create robustness.

**APPENDIX**

**CONSTRUCTION OF COORDINATE BASIS**

The coordinate basis in Theorem 4 is obtained through the following steps.

**Step 1:** Separate direct feedthrough

Let \( \tilde{D} = USV^\top \) be the singular value decomposition of \( \tilde{D} \), with \( U \in \mathbb{R}^{p \times p}, V \in \mathbb{R}^{m \times m}, \) and \( S = [S_{m_0} \ 0 \ 0] \in \mathbb{R}^{p \times m} \), where \( S_{m_0} \in \mathbb{R}^{m_0 \times m_0} \) with \( m_0 = \text{rank}(\tilde{D}) \) is a diagonal matrix with the singular values on the diagonal. Then \( \Gamma_1^{-1} \Sigma \Gamma_1 \), with \( \Sigma \) in (13), \( \Gamma_{1,1} = S_{m_0}^{-1} \) admits the representation

\[
x_1[k + 1] = A_1 x_1[k] + B_{x,1} u_{x,1}[k] + B_{0,1} y_{o,1}[k],
\]

\[
y_{o,1}[k] = C_{o,1} x_1[k] + u_{o,1}[k],
\]

\[
y_{x,1}[k] = C_{x,1} x_1[k],
\]

\[
y_{s,1}[k] = C_{s,1} x_1[k],
\]

with \( A_1 = \tilde{A} - B_{0,1} C_{0,1} \).

**Step 2:** Separate \( x_f \)

To separate \( x_f \), find invertible \( \Gamma_{s,2} \) such that \( \Gamma_{s,2}^{-1} B_{x,1} = \begin{bmatrix} 0 & 0 \\ \vdots & \ddots \\ 0 & \end{bmatrix} \) and \( C_{x,f,1} \Gamma_{s,2} = \begin{bmatrix} \times & \times & \times \\ \vdots & \ddots & \vdots \\ \times & \times & \end{bmatrix} \), where \( \times \) are arbitrary elements. Such transformations can be found by considering the last row/column pairs in \( C_{x,f,1} \) and \( B_{x,1} \), and working back as follows.

1) Let the column/row pair be given by

\[
\begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix} \in \mathbb{R}^n, \quad [c_1 \ c_2 \ c_3 \ldots] \in \mathbb{R}^n.
\]

2) Find matrices \( P = Q^{-1} \in \mathbb{R}^{2 \times 2} \), such that \( P \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 0 \\ b_2 \end{bmatrix} \) and \( \begin{bmatrix} c_1 & c_2 \end{bmatrix} Q = \begin{bmatrix} 0 & \tilde{c}_2 \end{bmatrix} \). If \( P, Q \) exist, there exist unitary \( P, Q \) or \( P, Q \) such that \( |\text{det}(P)| = |\text{det}(Q)| = 1 \). If \( P, Q \) do not exist, a random (unitary) transformation may be applied and the procedure can be restarted.

3) Repeat the previous step for the pair \( \tilde{b}_2, \tilde{c}_2 \) and \( c_3, \ldots \), and so on.

The process is repeated until all pairs and rows/columns are processed and the desired structure is obtained. The concatenation of all \( P, Q \) yields \( \Gamma_{s,2} \).

**Step 3:** Separate \( x_a \)

The invariant zeros \( z_{i}, i = 1, 2, \ldots, n_a \) are given by Definition 1. Let \( A_2 = \Gamma_{s,2}^{-1} A_1 \Gamma_{s,2} \). To separate \( x_a \), an eigenvalue decomposition can be used to obtain invertible \( \Gamma_{s,3} \) such that

\[
\Gamma_{s,3}^{-1} A_2 \Gamma_{s,3} = \begin{bmatrix} z_1 & \ldots & z_{n_a} & \times & \times & \times \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & \times & \times & \times \\ \end{bmatrix},
\]

where the structure related to \( x_f \) remains unaltered.

**Step 4:** Make \( y_s \) independent of \( x_f \) and \( y_f = x_f \)

Let the output matrix after step 3 be given by

\[
C_3 = \begin{bmatrix} \times & \times & \times & \times & \times & \times \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \times & \times & \times & \times & \times & \times \\ \end{bmatrix}
\]

Then \( \Gamma_{o,4} = \begin{bmatrix} I_{m_0} & 0 \\ 0 & C_{x,f,3} & 0 \\ 0 & C_{s,d} & I_{p-m} \end{bmatrix} \) yields

\[
\Gamma_{o,4}^{-1} C_3 = \begin{bmatrix} \times & \times & \times & \times & \times & \times \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \times & \times & \times & \times & \times & \times \\ \end{bmatrix}
\]

which makes \( y_s \) independent of \( x_f \) and \( y_f = x_f \).

This concludes the derivation of the coordinate basis in Theorem 4.

**ACKNOWLEDGMENT**

Robin de Rozario is gratefully acknowledged for his contributions to the experimental system.

**REFERENCES**


Jurgen van Zundert received the M.Sc. degree (with great appreciation) and Ph.D. degree in Mechanical Engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands in 2014 and 2018, respectively. He currently works as a mechatronic system engineer at DEMCON, Best, The Netherlands. His research interests are in feedforward motion control, multi-rate control, and iterative learning control.

Fons Luijten received the M.Sc. degree (cum laude) in Mechanical Engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands in 2017. He is currently a Technologist Dynamics and Control at MI-Partners in Eindhoven working on the development of various mechatronic systems in the high-tech industry.
Tom Oomen received the M.Sc. degree (cum laude) and Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands. He held visiting positions at KTH, Stockholm, Sweden, and at The University of Newcastle, Australia. Presently, he is associate professor with the Department of Mechanical Engineering at the Eindhoven University of Technology. He is a recipient of the Corus Young Talent Graduation Award, the 2015 IEEE Transactions on Control Systems Technology Outstanding Paper Award, and the 2017 IFAC Mechatronics Best Paper Award. He is Associate Editor of the IEEE Control Systems Letters (L-CSS) and IFAC Mechatronics.

His research interests are in the field of system identification, robust control, and learning control, with applications in mechatronic systems.