Parameter-Dependent Feedforward Strategies for Motion Systems

T.A.H.Bloemers, I.Proimadis, Y.Kasemsinsup and R.Tóth

***Abstract*— In high performance motion systems, the ever increasing demand for high accelerations and tracking accuracy necessitates the explicit incorporation of the parameter-dependent dynamics in the control design. For feedforward control, this can be achieved via the application of the Linear Parameter-Varying (LPV) framework or by including the measured parameters into the nonlinear control framework. Through simulation studies it is shown that addressing position-dependent dynamics with an LPV or nonlinear feedforward design increases the performance of motion systems compared to traditional linear time-invariant and nonlinear methods. Furthermore, similarities and advantages among the feedforward methods are investigated.**

I. INTRODUCTION

Feedback and feedforward control plays an important role in position control of motion systems. In general, feedback is used to stabilize the system, reject disturbances and to account for model uncertainties or unmodeled dynamics, whereas feedforward is used to realize a desired nominal behavior such that the output response of the nominal plant model follows a desired and known reference trajectory. In case the disturbances are known or measurable, one can utilize this knowledge to compensate for undesired behavior through feedforward actions as well.

In high performance motion systems, position-dependent dynamics become non-negligible towards the increasing demands on performance. This often comes from an intrinsic property in multi-degree of freedom motion systems, e.g., through the variations of actuator and/or sensor positioning. An example can be given by wafer stage control in the lithographic industry [1]. In contrast to position-dependent dynamics, control hereof is commonly treated by utilizing the Linear Time-Invariant (LTI) framework. Strategies such as acceleration, jerk or snap feedforward (see, e.g., [1]–[4]) or more advanced feedforward strategies as in [5] are typically used to compensate for undesired plant dynamics. In this context, a comprehensive survey of LTI feedforward methods can be found in [6].

The Linear Parameter-Varying (LPV) framework is an attractive alternative to cope with position-dependent dynamics. LPV systems are characterized by a linear input-output (IO) mapping, while this mapping depends on the so-called scheduling variables. These scheduling variables have to be available (measurable) in real-time and are used to capture the position-dependent behavior (in the context of motion applications). LPV control strategies benefit from the availability of the scheduling variable to increase the control performance. In the past, research was aimed towards norm-based synthesis approaches for the LPV feedforward control problem (see, e.g., [7], [8]). These approaches rely on a weighted optimization problem where one can specify e.g., the frequency region of interest. Although [9] studies inversion of time-varying characteristics of motion systems due to flexible behavior, it turns out that position-dependent effects can be found much alike when embedded in an LPV structure. The same holds true from a nonlinear perspective as shown in [10].

This paper investigates model-based analytic methodologies to incorporate position-dependent behavior in the feedforward control design. These techniques are extensions of well-studied LTI feedforward methodologies which utilize concepts of nonlinear control. Furthermore, we also employ the LPV framework to address the position-dependent behavior and compare it to LTI and nonlinear techniques. The methods presented here are considered for motion systems subject to position-dependent or time-varying nonlinear behavior under the assumption that the position-dependency is a tractable signal in real-time. The benefit of incorporating position-dependent behavior into the feedforward controller is shown through two simulation studies, from which it is further demonstrated that LPV and parameter-dependent nonlinear techniques provide similar results.

The paper is organized in the following manner: in Section II, the considered class of rigid-body motion systems is introduced. Next in Section III, the feedforward methodologies are discussed: namely, A) nonlinear feedforward, B) the incorporation of the measured parameter into the nonlinear feedforward and C) LPV feedforward through inversion. This is followed by simulation studies on (i) a high-accuracy position-dependent motion system and (ii) a DC motor with mass imbalance in Section IV. Here, the developed nonlinear and LPV feedforward strategies are compared with a conventional LTI acceleration feedforward. Finally in Section V conclusions are given.

II. PROBLEM FORMULATION

Consider the following dynamic model describing a position-dependent rigid-body motion system denoted Σ_{rb}:

$$\mathcal{M}(q(t))\ddot{q}(t) + \mathcal{C}(q(t),\dot{q}(t))\dot{q}(t) + \mathcal{K}(q(t)) = W(t) \quad (1)$$

where $t \in \mathbb{R}_0^+$, $q(t) \in \mathbb{R}^{n_q}$ are the generalized coordinates of the system, $\mathcal{M}(q) \in \mathbb{R}^{n_q \times n_q}$ is the mass-matrix which is positive definite in the range of operation, $\mathcal{C}(q,\dot{q}) \in \mathbb{R}^{n_q \times n_q}$ is the Coriolis matrix and $\mathcal{K} \in \mathbb{R}^{n_q \times n_q}$ contains the stiffness terms. $W \in \mathbb{R}^{n_w}$ represents the generalized forces and

T.A.H.Bloemers, I.Proimadis, Y.Kasemsinsup and R.Tóth are with the Control System group of Electrical Engineering Department, TU Eindhoven, 5612 AJ Eindhoven The Netherlands. Email: t.a.h.bloemers@student.tue.nl

torques acting on the system.

The LPV framework stems from the observation that (1) represents a linear structure that varies with q and $\dot{q}$ in terms of the variation of $\mathcal{M}(q(t))$, $\mathcal{C}(q(t), \dot{q}(t))$ and $\mathcal{K}(q(t))$. Hence using measurements of q and $\dot{q}$, which are often available in a motion system, we can capture the behavior of (1) in terms of a linear model with varying parameters. The linear structure of the resulting representation allows to generalize powerful results of LTI control to address the dynamics of (1). An LPV representation is obtained through direct embedding of the nonlinear model (1) in a descriptor state-space LPV representation

$$E(p(t))\dot{x}(t) = A(p(t))x(t) + B(p(t))u(t) \tag{2a}$$
$$y(t) = C(p(t))x(t) \tag{2b}$$

where $x(t) := \begin{bmatrix} q^\top & \dot{q}^\top \end{bmatrix}^\top \in \mathbb{R}^{n_x}$ is the state variable, $u(t) := W(t) \in \mathbb{R}^{n_u}$ are the control inputs, $y(t) \in \mathbb{R}^{n_y}$ are the outputs and $p(t) := \begin{bmatrix} p_1(t) & \cdots & p_{n_p}(t) \end{bmatrix}^\top \in \mathcal{P} \subset \mathbb{R}^{n_p}$. A descriptor form is used to avoid inversion of the mass-matrix $\mathcal{M}(q)$ and therefore retain affine dependency of the system matrices in p. Note that (2) describes a linear system with respect to the IO partition (u, y). If $p(t)$ is constant i.e., $p(t) \equiv \bar{p}$, $\forall t \in \mathbb{R}$, then (2) becomes an LTI representation, which is often referred to as the frozen dynamical aspect of (2) for the constant $\bar{p}$. If $p(t)$ varies with time, the model becomes a time-varying representation. The LPV representation is said to be affine if

$$A(p(t)) = A_0 + \sum_{i=1}^{n_p} p_i(t) A_i, \tag{3}$$

and similar decomposition holds for the matrices $E(p(t))$, $B(p(t))$ and $C(p(t))$. For an LPV embedding (see, [11]) of (1), the resulting system matrices are:

$$E(p(t)) = \begin{bmatrix} I_{n_q \times n_q} & 0_{n_q \times n_q} \\ 0_{n_q \times n_q} & \mathcal{M}(q) \end{bmatrix} \quad B(p(t)) = \begin{bmatrix} 0_{n_q \times n_u} \\ I_{n_q \times n_u} \end{bmatrix}$$
$$A(p(t)) = \begin{bmatrix} 0_{n_q \times n_q} & I_{n_q \times n_q} \\ -\mathcal{K}(q) & -\mathcal{C}(q, \dot{q}) \end{bmatrix}$$

and $C(p(t))$ is a parameter-dependent matrix depending on the measurement system. The system representation (2) is a global system description as it represents all possible behaviors of (1). Furthermore, it is rendered affine by appropriately choosing the scheduling variables.

III. FEEDFORWARD STRATEGIES

This section focusses on the design aspects of the feedforward methods that are taken into consideration, namely A) nonlinear feedforward, B) the incorporation of the measured parameter into the nonlinear feedforward and C) LPV feedforward through inversion. Hereon, we will use p to distinguish the measured signal (also the scheduling variable in the LPV case) which is available in the feedforward.

A. Nonlinear feedforward

In this section, the feedforward input is computed based upon a known reference trajectory and the nonlinear model as shown in Figure 1 (see, e.g., [10]). Given the nonlinear

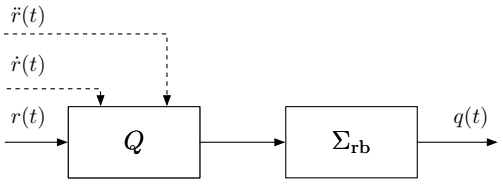

Fig. 1. Nonlinear feedforward block diagram where Q is the mapping from $r(t)$, $\dot{r}(t)$ and $\ddot{r}(t)$ to $W(t)$, Σ_{rb} the rigid body motion system (1), $r(t)$ the reference signal, $q(t)$ the output and $p(t)$ the scheduling variable.

system represented by (1) and a desired trajectory $r(t)$ on the generalized coordinate vector $q(t)$, the inputs $W(t)$ to steer the system over the desired trajectory satisfy

$$W(t) = \mathcal{M}(r(t))\ddot{r}(t) + \mathcal{C}(r(t), \dot{r}(t))\dot{r}(t) + \mathcal{K}(r(t)). \tag{4}$$

The system achieves perfect tracking of the reference trajectory under the following conditions:

- The trajectories r, $\dot{r}$ and $\ddot{r}$ exist and are known.
- The model is perfectly accurate and if $q(0) = r(0)$ and $\dot{q}(0) = \dot{r}(0)$.

However, if the system deviates slightly from the desired trajectory, the prediction of the system's trajectory based upon the reference may lead to a deviation of the desired reference trajectory once disturbances are present. A refinement of this feedforward controller would be to include the (measured) position and velocity of the system, i.e., replacing $r(t)$ and $\dot{r}(t)$ with $q(t)$ and $\dot{q}(t)$ respectively such that

$$W(t) = \mathcal{M}(q(t))\ddot{r}(t) + \mathcal{C}(q(t), \dot{q}(t))\dot{r}(t) + \mathcal{K}(q(t)). \tag{5}$$

B. LPV inversion

In this section, the objective is to find a feedforward system Σ_{ff} such that (1) tracks a desired reference signal $r(t)$ as depicted in Figure 2. To achieve this, we aim at

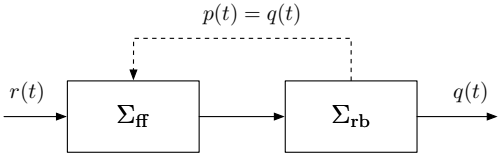

Fig. 2. LPV feedforward block diagram where Σ_{ff} is the LPV inversion based feedforward controller, Σ_{rb} the rigid body motion system (1), $r(t)$ the reference signal, $q(t)$ the output and $p(t)$ the scheduling variable.

finding the feedforward input $u_{\mathrm{ff}}(t)$ through inversion of the LPV embedding (2), e.g., find $u_{\mathrm{ff}}(t)$ such that

$$E(p(t))\dot{x}_{\mathrm{ref}}(t) = A(p(t))x_{\mathrm{ref}}(t) + B(p(t))u_{\mathrm{ff}}(t) \tag{6a}$$
$$y_{\mathrm{ref}}(t) = C(p(t))x_{\mathrm{ref}}(t) \tag{6b}$$

where $x_{\mathrm{ref}}(t)$ and $y_{\mathrm{ref}}(t)$ denote the desired compatible state and output trajectory over time respectively and $p(t) = q(t)$ captures the position-dependency in the scheduling variable. In obtaining an exact inverse of the plant, consider the concept of relative degree, as introduced in [12], by which the explicit influence of the input $u(t)$ is observed through the n-fold derivative of the output $y(t)$. The concept presented in [9] is used to recover dynamical relations between the desired trajectory y_{ref}, state trajectory x_{ref} and feedforward input

u_{ff} and their time derivatives in the LPV descriptor state-space setting for the case where the system matrices $E(p(t))$, $A(p(t))$, $B(p(t))$ and $C(p(t))$ have affine dependency on the scheduling variables.

Define the n-th order time derivative of y as

$$y^{[n]} := \frac{\mathrm{d}^n y}{\mathrm{dt}^n}$$

Given the state and output equations in (6), the n-th order derivative of the output y_{ref} satisfies:

$$y_{\mathrm{ref}}^{[n]} = \sum_{k=0}^{n} \binom{n}{k} C^{[n-k]}(p(t)) x_{\mathrm{ref}}^{[k]} \tag{7}$$

where

$$x_{\mathrm{ref}}^{[n]} = \sum_{k=0}^{n-1} \binom{n-1}{k} E_{\mathrm{inv}}^{[n-k-1]}(p(t)) z^{[k]}$$

$$z^{[n]} = \sum_{k=0}^{n} \binom{n}{k} A^{[n-k]}(p(t)) x_{\mathrm{ref}}^{[k]}(t) + B^{[n-k]}(p(t)) u_{\mathrm{ff}}^{[k]}(t)$$

and $\binom{i}{k} = \frac{i!}{k!(i-k)!}$ denotes the binomial coefficient. Let $E_{\mathrm{inv}}(p(t)) := E^{-1}(p(t))$ denote the inverse of $E(p(t))$. The first derivative of the inverse of a matrix $E(p(t))$ is given as [13]:

$$E_{\mathrm{inv}}^{[1]}(p(t)) = -E^{-1}(p(t)) E^{[1]}(p(t)) E^{-1}(p(t))$$

and higher order derivatives of $E^{-1}(p(t))$ can be recursively computed. The system matrices are assumed to have affine dependency on the scheduling variables, therefore the n-th order derivatives are given as:

$$E^{[n]}(p(t)) = \sum_{i=1}^{n_p} E_i p_i^{[n]}(t).$$

Collecting the terms in (7) dependent on the state x_{ref} and the input vector ξ_{ref} results in a general representation:

$$\begin{aligned} y_{\mathrm{ref}}^{[n]} =& \breve{E}_n(p(t), p^{[1]}(t), \ldots, p^{[n]}(t)) x_{\mathrm{ref}} + \\ & \breve{F}_n(p(t), p^{[1]}(t), \ldots, p^{[n]}(t)) \xi_{\mathrm{ff}}, \end{aligned} \tag{8}$$

with

$$\xi_{\mathrm{ff}} := \begin{bmatrix} u_{\mathrm{ff}}^{\top} & u_{\mathrm{ff}}^{\top[1]} & \ldots & u_{\mathrm{ff}}^{\top[n-1]} \end{bmatrix}^{\top}.$$

For compactness, we will abbreviate $\breve{E}_n(p(t), p^{[1]}(t), \ldots, p^{[n]}(t))$ and $\breve{F}_n(p(t), p^{[1]}(t), \ldots, p^{[n]}(t))$ as $\breve{E}_n(\breve{p}(t))$ and $\breve{F}_n(\breve{p}(t))$ respectively. From (8) it can be derived that:

$$\begin{aligned} u_{\mathrm{ff}} &= \breve{C} \xi_{\mathrm{ff}} \\ &= \breve{C} \breve{F}_n^{\dagger}(\breve{p}(t)) \left(y_{\mathrm{ref}}^{[n]} - \breve{E}_n(\breve{p}(t)) x_{\mathrm{ref}} \right), \end{aligned} \tag{9}$$

where $\breve{C} \in \mathbb{R}^{n_u \times n \cdot n_u}$ is a matrix that selects the feedforward input u_{ff}:

$$\breve{C} = \begin{bmatrix} I_{(n_u \times n_u)} & 0_{(n_u \times (n-1) n_u)} \end{bmatrix}.$$

Here, $\breve{F}_n^{\dagger}(\breve{p}(t))$ denotes the point-wise pseudo inverse dependent on the scheduling variable $p(t)$ and its derivatives. If the exact or right inverse of $\breve{F}_n(\breve{p}(t))$ exists for all scheduling variables $\bar{p} \in \mathcal{P}$, then it is possible to find the ideal u_{ff} that drives the system exactly according to $y_{\mathrm{ref}}^{[n]}$. Utilizing the system description (6) and (9), the LPV system inverse Σ_{ff} is given as:

$$E_{\mathrm{ff}}(p(t)) \dot{x}_{\mathrm{ref}}(t) = A_{\mathrm{ff}}(p(t)) x_{\mathrm{ref}}(t) + B_{\mathrm{ff}}(p(t)) y_{\mathrm{ref}}^{[n]}(t) \tag{10a}$$

$$u_{\mathrm{ff}}(t) = C_{\mathrm{ff}}(p(t)) x_{\mathrm{ref}}(t) + D_{\mathrm{ff}}(p(t)) y_{\mathrm{ref}}^{[n]}(t) \tag{10b}$$

where

$$\begin{aligned} E_{\mathrm{ff}}(p(t)) &= E(p(t)) \\ A_{\mathrm{ff}}(p(t)) &= A(p(t)) - B(p(t)) \breve{C} \breve{F}_n^{\dagger}(\breve{p}(t)) \breve{E}_n(\breve{p}(t)) \\ B_{\mathrm{ff}}(p(t)) &= B(p(t)) \breve{C} \breve{F}_n^{\dagger}(\breve{p}(t)) \\ C_{\mathrm{ff}}(p(t)) &= -\breve{C} \breve{F}_n^{\dagger}(\breve{p}(t)) \breve{E}_n(\breve{p}(t)) \\ D_{\mathrm{ff}}(p(t)) &= \breve{C} \breve{F}_n^{\dagger}(\breve{p}(t)). \end{aligned}$$

The system description (10) represents the inverse system of (6) that takes $y_{\mathrm{ref}}^{[n]}(t)$ as input and generates the feedforward signal $u_{\mathrm{ff}}(t)$ as output along the corresponding reference state trajectory $x_{\mathrm{ref}}(t)$. This method holds under the following conditions

- the relative degree n is known and it is the same for all values $\bar{p} \in \mathcal{P}$,
- $y_{\mathrm{ref}}^{[n]} \in L_2$ is known a priori with $\int_{t_0}^{\infty} y_{\mathrm{ref}}^{[n]} \mathrm{d}t = 0$, resulting in a bounded feedforward input,
- $\hat{F}_n^{\dagger}(\breve{p}(t))$ is computable in the sense of least-norm (right) or exact inverse for all values $\bar{p} \in \mathcal{P}$ and their time derivatives $\bar{p}^{[i]} \in \mathcal{P}^{[i]}$ with $i = 1, \ldots, n$,
- $\bar{p}$ and the time derivatives $\bar{p}^{[i]}$ are available in real time.
- $E(\bar{p})$ is invertible for all values $\bar{p} \in \mathcal{P}$.

IV. Simulation study

This section first introduces a high performance rigid-body motion system subject to position-dependent dynamic behavior. As a second simulation example, a DC motor with a mass imbalance is introduced. A simulation study is performed on both examples to assess the performance of the feedforward strategies in comparison to acceleration feedforward. Finally, the results are extensively discussed.

The feedforward controllers will be tested in a closed-loop environment, for both example systems, according to Figure 3 to assess the performance under stable closed-loop conditions. The feedback controllers have the following PID structure consisting of an integral action, a derivative action and a lowpass filter

$$C(s) = K \frac{(s + \omega_{\mathrm{bw}} r_{\omega_{\mathrm{i}}})(s + \omega_{\mathrm{bw}} r_{\omega_{\mathrm{d}}})}{s(s^2 + 2\beta_{\mathrm{lp}} \omega_{\mathrm{bw}} r_{\omega_{\mathrm{lp}}} s + \omega_{\mathrm{bw}}^2 r_{\omega_{\mathrm{lp}}}^2)}, \tag{11}$$

where for either of the considered examples corresponding values of parameters will be given. The feedforward strategies will be referred to as

1) LTI acceleration feedforward,
2) nonlinear feedforward based on (4),
3) parameter-dependent nonlinear feedforward (5),
4) LPV inversion feedforward of the system embedding (2) as in III-B.

An condition mismatch, e.g., $q(0) \neq r(0)$ between the initial conditions and the reference, are simulated to mimic an inaccurate position measurement at the start of the trajectory. The ℓ_2 and ℓ_∞-norms of the sampled error data will be inspected to assess the performance.

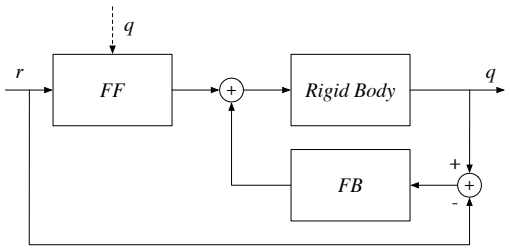

Fig. 3. Block diagram for the closed-loop simulation. Here the FF-block represents the feedforward controller, the FB-block represents the feedback controller, r is the reference signal and q is the system output.

A. Example system 1: Magnetically levitated planar actuator

Magnetically levitated planar actuators are high-performance positioning devices. A schematic diagram of the translator is shown in Figure 4, where (x_m, y_m, z_m) denotes the global reference frame and (x_t, y_t, z_t) denotes the reference frame of the translator. Let the position of

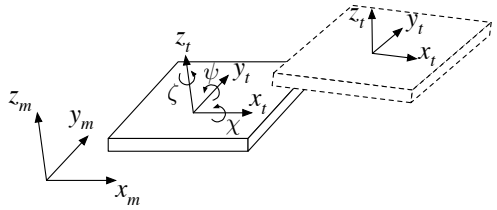

Fig. 4. Schematic of the magnetically levitated planar actuator. The triplet (x_m, y_m, z_m) denotes the global reference frame and (x_t, y_t, z_t) denotes the translators reference frame. Let the dashed plate denote another position of the translator in the global reference frame.

the translator in the global reference frame be given by the generalized coordinates $q \in \mathbb{R}^{n_q}$ defined as

$$q := \begin{bmatrix} x_s & y_s & z_s & \chi_s & \psi_s & \zeta_s \end{bmatrix}^\top, \tag{12}$$

where (x_s, y_s, z_s) denote the translation of the Center of Mass (CoM) of the translator with respect to the global reference frame (x_m, y_m, z_m) and the angles $(\chi_s, \psi_s, \zeta_s)$ denote the rotation of the center of mass around the axes of the global reference frame, defined under the Pitch-Yaw-Roll representation [10]. In this representation, the three angles, as in Figure 4, denote the successive rotation around the three inertial frames: first a rotation χ around the x-axis (yaw), then a rotation ψ around the y-axis (pitch) and finally a rotation ζ around the z-axis (roll). Furthermore, the wrench vector is defined as

$$W := \begin{bmatrix} F_x & F_y & F_z & \tau_\chi & \tau_\psi & \tau_\zeta \end{bmatrix}^\top, \tag{13}$$

and contains the generalized forces and torques acting on the plate at the center of mass expressed in the global reference frame (x_m, y_m, z_m). The output of the system is equal to the states e.g., $y(t) = q(t)$. The solution to the Euler-Lagrange equations lead to equations of motion of the form (1). The matrices $\mathcal{M}(q)$, $\mathcal{C}(q, \dot{q})$ and $\mathcal{K}(q)$ are given in Appendix I.

Throughout this example, we will consider the model in the forms (1) and (2). In the LPV case (2) the scheduling variables $p(t)$ are chosen as the Christoffel symbols $\Gamma_{ijk}\dot{q}_k$ (see Appendix I), thus being a function of the measured variables i.e., $p(t) = f(q(t), \dot{q}(t))$. One can also apply the LPV inversion method on the local aspects of a model, which for motion systems are often obtained through identification or linearization techniques around a set of operating conditions (see, e.g., [11], [14]), however due to lack of space this is not explained in this paper, but, instead, can be found in [15].

The following parameters for the feedback controller (11) are used: $\omega_{\text{bw}} = 32\pi$, $r_{\omega_{\text{i}}} = \frac{1}{5}$, $r_{\omega_{\text{d}}} = \frac{1}{5}$, $r_{\omega_{\text{lp}}} = 4$, $\beta_{\text{lp}} = 1$ and $K = \text{diag}(3.579, 3.579, 3.579, 0.044, 0.044, 0.087) \cdot 10^7$. An initial condition mismatch of $\chi(0) = 5$ μrad and no mismatch for the other coordinates is used.

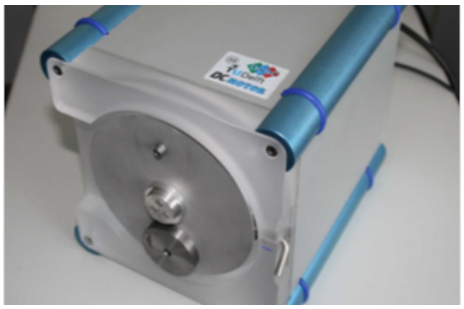

Fig. 5. The DC motor with mass imbalance.

B. Example system 2: DC motor with mass imbalance

A DC motor with mass imbalance depicted in Figure 5, which was introduced in [16], is used as a second example system. The DC motor dynamics become position-varying if an additional mass is mounted off-center to the rotation disc. A mathematical description of the system is given in form (1), where $q(t)$ is the rotation angle of the disc in [rad] as well as the measured output $y(t) = q(t)$ and $W(t)$ is the input voltage in [V]. The Mass, Coriolis and Stiffness matrices are

$$\mathcal{M}(q(t)) = \frac{RJ}{K_m(Rb + K^2)} \qquad \mathcal{C}(q(t), \dot{q}(t)) = K_m$$
$$\mathcal{K}(q(t)) = \frac{-RMgl}{K_m(Rb + K^2)} \sin(q(t)) \qquad K_m = \frac{K}{Rb + K^2}$$

and the parameters can be found in Table I.

TABLE I
PARAMETERS OF THE DC MOTOR.

Parameters	Value
Motor torque constant	$K = 0.0536 Nm/A$
Motor resistance	$R = 9.50\Omega$
Motor impedance	$L = 0.84 \cdot 10^{-3} H$
Disc inertia	$J = 2.2 \cdot 10^{-4} Nm^2$
Viscous friction	$b = 6.6 \cdot 10^{-5} Nms/rad$
Additional mass	$M = 0.07 kg$
Mass - center disc distance	$l = 0.042 m$

One can embed (1) in a global LPV embedding (2) by choosing the scheduling variable as $p(t) = \frac{\sin(q(t))}{q(t)}$.

The following parameters for the feedback controller (11) are used: $\omega_{\text{bw}} = 10\pi$, $r_{\omega_{\text{i}}} = \frac{1}{5}$, $r_{\omega_{\text{d}}} = \frac{1}{3}$, $r_{\omega_{\text{lp}}} = 6$, $\beta_{\text{lp}} = 0.8$ and $K = 8.0657 \cdot 10^7$. An initial condition mismatch of $q(0) = \frac{\pi}{12}$ rad is used.

C. Simulation results and discussion

This section provides simulation results of the feedforward controllers, given in Section III, and are compared to an acceleration feedforward, as described in [2]. Improvement of feedforward control which explicitly incorporates position dependent behavior is highlighted.

For example system 1, the reference trajectory with corresponding error signal is given in Figure 6. The ℓ_2 and ℓ_∞-norms of the sampled error data are given in Table II. Note that only the results for the angles (χ, ψ, ζ) are shown as the effects due to parameter-dependent behavior only affects the angular rotations. The translations (x, y, z) are therefore perfectly tracked using all the feedforward methods and do not contain any relevant information for this example.

For example system 2, the reference trajectory and the error trajectories can be found in Figure 7. The ℓ_2 and ℓ_∞-norms of the sampled error data are given in Table III.

A number of conclusions can be drawn regarding both simulation examples. Firstly, The acceleration feedforward does not take into account position dependent behavior. It is outperformed by all other feedforward methods, but will be used as a baseline for comparing the other methods.

Secondly, the nonlinear feedforward tries to cancel the dynamics of the system based on the information from the reference signal. As long as the system does not deviate from the reference trajectory, this method achieves good results. However, it is not very robust with respect to disturbances or model uncertainties. For example, an initial condition mismatch or disturbance has a direct effect on this method. The further away the system is from the reference trajectory, the worse the tracking performance becomes.

Thirdly, the parameter-dependent nonlinear feedforward is an improvement over the nonlinear feedforward. The inclusion of the measured position gives it the ability to slightly correct for disturbances, e.g., when the system deviates from the desired trajectory.

Fourthly, the LPV inversion based feedforward describes the inverse dynamics of the nonlinear system in an LPV setting. When disturbances or initial condition mismatches are present, it can be seen that it is more robust than the nonlinear feedforward as it takes the position-dependency of the disturbed system into account by utilising the available scheduling parameters instead of the prediction from the reference signal.

Finally, an important observation to make is that the parameter-dependent nonlinear feedforward and the LPV inversion feedforward methods produce similar results. Both have accurate tracking capabilities and give similar improvements under disturbances compared to the other feedforward methods. A possible drawback of the LPV inversion based feedforward method is the computational cost. This method first has to solve the system matrices (10) to get the inverse system, after which (10) has to be solved over time. This has to happen in real-time and adds significantly more computation time compared to the parameter-dependent nonlinear feedforward, for which the input can be computed by simple matrix multiplications. The LPV inversion method

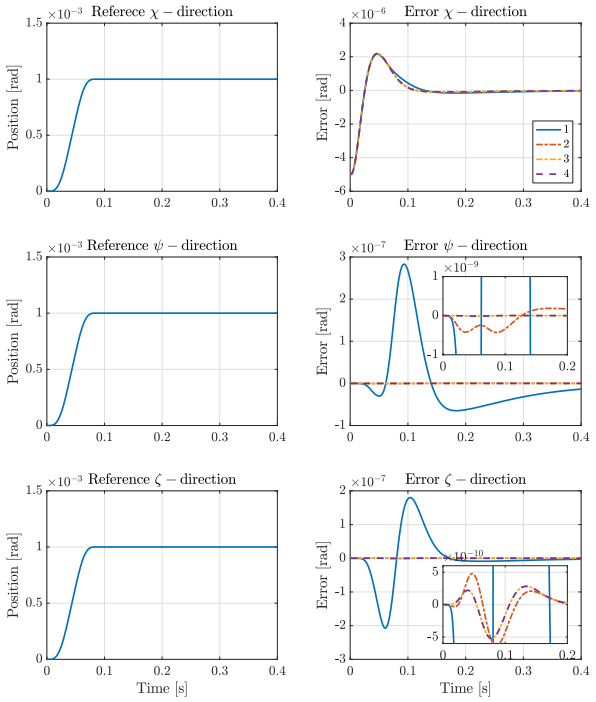

Fig. 6. Simulation results for on example system 1. The reference trajectories are shown on the left hand side. The resulting error plots for the four feedforward controllers are shown on the right hand side.

TABLE II

EXAMPLE SYSTEM 1 CLOSED-LOOP SIMULATION DATA. ℓ_2 AND ℓ_∞-NORMS OF THE SAMPLED ERROR SIGNAL WITH RESPECT TO THE χ, ψ AND ζ ANGLES, USING A SAMPLING TIME $T_s = 65$ μSEC.

FF	ℓ_2-norm $\times 10^{-4}$			ℓ_∞-norm $\times 10^{-5}$		
	χ	ψ	ζ	χ	ψ	ζ
1)	0.7591	0.0811	0.0486	0.5000	0.0290	0.0219
2)	0.7429	0.0002	0.0001	0.5000	0.0001	0.0000
3)	0.7429	0.0000	0.0001	0.5000	0.0000	0.0000
4)	0.7429	0.0000	0.0001	0.5000	0.0000	0.0000

requires the availability of the derivatives of the scheduling variables which can be obtained either through estimation or by placing extra sensors if these are not directly available. Furthermore, the inclusion of the scheduling parameters in the feedforward controller creates an additional feedback loop which remains to be investigated.

TABLE III

EXAMPLE SYSTEM 2 CLOSED-LOOP SIMULATION DATA. ℓ_2 AND ℓ_∞-NORMS OF THE SAMPLED ERROR SIGNAL WITH RESPECT TO THE ANGLE q, USING A SAMPLING TIME $T_s = 1$ mSEC.

FF	1)	2)	3)	4)
ℓ_2-norm	5.5005	1.4407	1.4146	1.4146
ℓ_∞-norm	0.2790	0.2618	0.2618	0.2618

V. CONCLUSIONS

In this paper several methods were presented that incorporate parameter-dependent dynamic behavior in the design of feedforward controllers by utilizing the nonlinear control

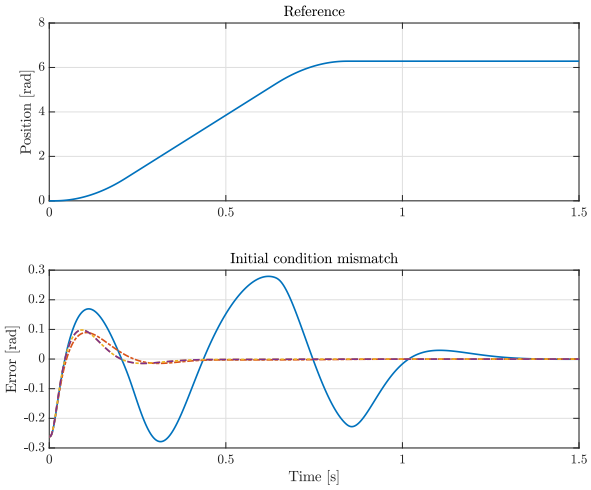

Fig. 7. Simulation results for example system 2. The top figure displays the reference trajectory. The bottom figure displays the error signal for the four feedforward controllers.

represenation to incorporate parameter-dependency as well as the Linear Parameter-Varying (LPV) framework. The performance of the feedforward methodologies is assessed through simulation studies on (i) a high-performance magnetically levitated planar actuator and (ii) a DC motor with mass imbalance. The feedforward methods investigated in this paper are shown to be able to cope with position-dependent dynamics and lead to an increase in performance compared to LTI or nonlinear techniques investigated in the simulation studies. The best and, most interestingly, similar results are obtained by incorporating the measured parameter-variations in the nonlinear feedforward as well as by the inversion of the LPV embedding. Due to computational benefits the parameter-dependent nonlinear feedforward is preferred over the LPV inverse feedforward. Future work contains further investigation and possibly a proof of the equivalence of the parameter-dependent nonlinear feedforward and the LPV inversion feedforward. Also the effect of noise on the scheduling parameters will be investigated. Moreover, experimental validation will provide additional insight on the results.

APPENDIX I

Given the dynamics in summation form:

$$\sum_{j=1}^{n} \mathcal{M}_{ij}(q)\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} \Gamma_{ijk}\dot{q}_j\dot{q}_k + \sum_{j=1}^{n} K_{ij}(q) = W_i,$$

for $i = 1, \cdots, n$ and $q, W \in \mathbb{R}^n$, where the mass-matrix $\mathcal{M}(q)$ is given as

$$\mathcal{M}(q) = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_\chi & 0 & \alpha_2 \\ 0 & 0 & 0 & 0 & I_\psi \cos^2(\chi) + I_\zeta \sin^2(\chi) & \alpha_1 \\ 0 & 0 & 0 & \alpha_2 & \alpha_1 & \alpha_3 \end{bmatrix}$$

with

$$\alpha_1 = \sin(2\chi)\cos(\psi)\frac{I_\psi - I_\zeta}{2}, \qquad \alpha_2 = -I_\chi \sin(\psi),$$
$$\alpha_3 = \cos^2(\psi)\left(I_\zeta \cos^2(\chi) + I_\psi \sin^2(\chi)\right) + I_\chi \sin^2(\psi)$$

and

$$m = 10.137, \quad I_\chi = 0.125, \quad I_\psi = 0.125, \quad I_\zeta = 0.248.$$

Next, define the components of the Coriolis matrix $\mathcal{C}(q, \dot{q}) \in \mathbb{R}^{n\times n}$ as:

$$\mathcal{C}_{ij}(q, \dot{q}) = \sum_{k=1}^{n} \Gamma_{ijk}\dot{q}_k, \quad \text{for } i, j = 1, \ldots, n$$

where Γ_{ijk} are called the Christoffel symbols corresponding to the inertia matrix $\mathcal{M}(q)$ and are chosen as:

$$\Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial \mathcal{M}_{ij}(q)}{\partial q_k} + \frac{\partial \mathcal{M}_{ik}(q)}{\partial q_j} - \frac{\partial \mathcal{M}_{kj}(q)}{\partial q_i}\right).$$

The stiffness matrix $\mathcal{K} = 0$. An LPV embedding (2) is obtained by choosing the scheduling variables as $p(t) = \Gamma_{ijk}\dot{q}_k$ and additionally the sin and cos terms in $\mathcal{M}(q)$.

REFERENCES

[1] H. Butler, "Position control in lithographic equipment," *IEEE Control Systems Magazine*, vol. 31, no. 5, pp. 28–47, 2011.

[2] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechanical motion systems," *Control Engineering Practice*, vol. 13, no. 2, pp. 145–157, 2005.

[3] M. Boerlage, R. Tousain, and M. Steinbuch, "Jerk derivative feedforward control for motion systems," in *Proceedings of the 2004 American Control Conference*, 2004, pp. 4843–4848.

[4] M. Boerlage, "MIMO jerk derivative feedforward for motion systems," in *Proceedings of the 2006 American Control Conference*, 2006, pp. 3892–3897.

[5] M. Boerlage, M. Steinbuch, P. Lambrechts, and M. van de Wal, "Model-based feedforward for motion systems," in *Proc. IEEE Conf. Control Appl. 2*, 2003, pp. 1158–1163.

[6] G.M. Clayton, S. Tien, K.K. Leang, Q. Zou, and S. Devasia, "A review of feedforward control approaches in nanopositioning for high-speed SPM," *Journal of Dynamic Systems, Measurement and Control*, vol. 131, no. 6, p. 061101, 2009.

[7] E. Prempain and I. Postlethwaite, "A Feedforward Control Synthesis Approach for LPV Systems," in *2008 American Control Conference*, 2008, pp. 3589–3594.

[8] J. Theis, H. Pfifer, A. Knoblack, F. Saupe, and H. Werner, "Linear Parameter-Varying Feedforward Control: A missile Autopilot Design," in *Guidance, Navigation and Control Conference*. AIAA, 2015.

[9] Y. Kasemsinsup, M. Heertjes, H. Butler, and S. Weiland, "Exact plant inversion of flexible motion systems with a time-varying state-to-output map," in *2016 European Control Conference*, 2016.

[10] R.M. Murray, Z. Li, and S.S Sastry, "A mathematical introduction to robotic manipulation," 1994.

[11] R. Tóth, *Modeling and identification of linear parameter-varying systems*. Springer-Verlag, 2010, vol. 403.

[12] A. Isidori, *Nonlinear Control Systems: An introduction*. Springer-Verlag Berlin Heidelberg, 1989.

[13] K.B. Petersen and M.S. Pedersen, "The matrix cookbook," 2012.

[14] R. Tóth, M. van de Wal, P.S.C. Heuberger, and P.M.J. Van den Hof, "LPV Identification of High Performance Positioning Devices," in *2011 American Control Conference*, 2011, pp. 151–158.

[15] T.A.H. Bloemers, I. Proimadis, Y. Kasemsinsup, and R. Tóth, "Feedforward Control of Magnetically Levitated Planar Actuators," *ArXiv e-prints*, 2018.

[16] B. Kulcsár, J. Dong, J.W. van Wingerden, and M. Verhaegen, "LPV subspace identification of a DC motor with unbalanced disc," in *Proceedings of the 15th IFAC Symposium on System Identification*, 2009.