

Keval S. Ramani

G. G. Brown Laboratory,
Department of Mechanical Engineering,
University of Michigan,
2350 Hayward,
Ann Arbor, MI 48109
e-mail: ksramani@umich.edu

Molong Duan

G. G. Brown Laboratory,
Department of Mechanical Engineering,
University of Michigan,
2350 Hayward,
Ann Arbor, MI 48109
e-mail: molong@umich.edu

Chinedum E. Okwudire¹

Mem. ASME
G. G. Brown Laboratory,
Department of Mechanical Engineering,
University of Michigan,
2350 Hayward,
Ann Arbor, MI 48109
e-mail: okwudire@umich.edu

A. Galip Ulsoy

Fellow ASME
G. G. Brown Laboratory,
Department of Mechanical Engineering,
University of Michigan,
2350 Hayward,
Ann Arbor, MI 48109
e-mail: ulsoy@umich.edu

Tracking Control of Linear Time-Invariant Nonminimum Phase Systems Using Filtered Basis Functions

An approach for minimizing tracking errors in linear time-invariant (LTI) single-input single-output (SISO) discrete-time systems with nonminimum phase (NMP) zeros using filtered basis functions (FBF) is studied. In the FBF method, the control input to the system is expressed as a linear combination of basis functions. The basis functions are forward filtered using the dynamics of the NMP system, and their coefficients are selected to minimize the error in tracking a given desired trajectory. Unlike comparable methods in the literature, the FBF method is shown to be effective in tracking any desired trajectory, irrespective of the location of NMP zeros in the z -plane. The stability of the method and boundedness of the control input and system output are discussed. The control designer is free to choose any suitable set of basis functions that satisfy the criteria discussed in this paper. However, two rudimentary basis functions, one in time domain and the other in frequency domain, are specifically highlighted. The effectiveness of the FBF method is illustrated and analyzed in comparison with the truncated series (TS) approximation method. [DOI: 10.1115/1.4034367]

1 Introduction

There are two general classes of motion control tasks; they are: point-to-point control and tracking control. Point-to-point control focuses primarily on the beginning and the end states of the motion, with little or no concern about the motion states in between. However, in tracking control, the output of the controlled system is expected to follow the desired trajectory as closely as possible at every time instant. Tracking control is critical in several applications, including manufacturing, robotics, and aeronautics. The focus of this paper is on tracking control of LTI SISO discrete-time systems, representing open-loop or feedback-controlled plants.

Perfect tracking control (PTC) can be achieved, in theory, by model inversion (i.e., pole-zero cancellation) [1]. PTC results in zero gain and phase errors between the desired and output trajectories, if the system model is accurate and has a stable inverse. However, when applied to systems with NMP zeros, PTC gives rise to highly oscillatory or unstable control trajectories which are unacceptable. NMP zeros are prevalent in practice. For example, they occur in systems with fast sampling rates [2], as well as in systems with noncollocated placement of sensors and actuators [3]. Hence, a lot of research has been done on developing methods for tracking control of systems with NMP zeros. Available approaches can be classified into two categories: approximate

model inversion [1,4–13] and direct model inversion with bounded control trajectories [14–23].

The most-straightforward way to implement approximate inversion is to cancel all the poles and cancelable zeros while ignoring the NMP and poorly damped zeros (hereinafter referred to as uncancelable zeros). This technique is called stable pole-zero canceling or NMP zeros ignore (NPZ-ignore) [8,9], and it results in a controlled system that exhibits magnitude and phase errors between its desired and output trajectories. Zero magnitude error tracking control (ZMETC) [8,10,12] focuses on canceling magnitude errors across all the frequencies at the expense of phase errors, whereas zero-phase error tracking control (ZPETC) [1] focuses on canceling phase errors across all the frequencies at the expense of magnitude errors. Depending on the system and the performance specifications, NPZ-ignore, ZMETC, and ZPETC may not yield satisfactory tracking performance due to the approximations involved. Various methods have been suggested to improve the tracking performance of approximate inversion methods [5–7,11,13]. The most notable is extended bandwidth zero-phase error tracking control (EBZPETC) [5], which is based on the power series expansion of ZPETC. The tracking performance of EBZPETC gets progressively better compared to ZPETC as more terms are added to the power series, provided that the series is convergent [7]. The problem is that the series is not convergent for NMP zeros on the right half z -plane, in which case the performance of EBZPETC becomes worse compared to that of ZPETC [7]. As an improvement to EBZPETC, the truncated series (TS) approximation method [4] uses the power series of the inverse of the uncancelable dynamics to improve the tracking accuracy of the system. The TS method always results in a convergent series. However, it fails completely (i.e., is undefined) when the uncancelable zeros lie on the unit circle. Moreover, as the

¹Corresponding author.

Contributed by the Dynamic Systems Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received October 3, 2015; final manuscript received July 22, 2016; published online September 8, 2016. Assoc. Editor: Zongxuan Sun.

uncancelable zeros approach the unit circle, the number of terms required to achieve a desired level of tracking accuracy increases drastically.

Direct inversion techniques [14–17] can achieve perfect tracking with infinite preview (i.e., knowledge of future desired trajectories) and pre-actuation (i.e., actuation applied before a time interval without affecting the output during the time interval). Methods with finite preview and pre-actuation have been developed, to achieve tracking with reasonable accuracy [18–20,22]. Similar to the TS method, the pre-actuation needed for direct inversion techniques [16,18–20,22] tends to increase as the uncancelable zero approaches the unit circle in the z -plane and is infinite for zeros on the unit circle (i.e., for nonhyperbolic systems) [21]. Methods have been proposed to solve this problem [21,23–25]. However, such methods are applicable to only a restrictive set of systems [24] or desired trajectories [21,25]; alternatively, stable inversion is achieved by perturbation of the unstable inverse system, at the cost of tracking accuracy [23].

One therefore observes a major problem with the discussed methods: they are not effective in tracking arbitrary desired trajectories regardless of the location of NMP zeros in the z -plane.

In Ref. [26], the first three authors proposed FBFs method for tracking control of NMP systems and showed through comparative simulations that the FBF method outperforms popular methods like NPZ-ignore, ZPETC, and ZMETC in tracking examples. The FBF method assumes that the desired trajectory to be tracked is fully known, and that the control trajectory can be expressed as a linear combination of basis functions having unknown coefficients. The basis functions are forward filtered using the modeled dynamics of the system and the coefficients selected to minimize the errors in tracking a given desired trajectory. A similar idea was presented by Frueh and Phan [27], in the context of iterative learning control, as the inverse linear quadratic learning (or inverse LQL) solution. Moreover, Lunenberg et al. [28] presented a tracking control method based on inverting the convolution matrix of the system to be tracked; this method [28] can be viewed as a special case of FBF that uses a set of basis functions that completely span the desired trajectory space. Jetto et al. [25] employed another special case of FBF, based on spline segments as basis functions, in determining the transient portion of a tracking control signal for tracking NMP systems. However, as noted earlier, their method is applicable to only a restrictive class of desired trajectories [25]; furthermore, it requires the use of feedback control to be effective.

The goal of this paper is to provide a thorough theoretical treatment of the FBF approach in the context of tracking control of SISO discrete-time systems with NMP zeros. Specifically, the original contributions of this paper in relation to the FBF method are in: (i) providing conditions for existence of a solution; (ii) providing conditions for bounded input bounded output (BIBO) stability; (iii) analyzing effects of the number and type of basis functions; and (iv) demonstrating its effectiveness in tracking arbitrary desired trajectories, with or without feedback, regardless of the location of NMP zeros in the z -plane.

Section 2 presents the FBF method and answers fundamental questions related to the existence of a solution, stability, and the effect of the number of basis functions on performance; it also discusses the limiting case when the number of basis functions is equal to the number of discrete points in the desired trajectory, resulting in perfect tracking. Section 3 introduces two rudimentary basis functions—one in the time domain and the other in the frequency domain. They are then employed in Sec. 4 to illustrate the theoretical concepts discussed in Sec. 2, using simulation-based examples, followed by conclusions and future work in Sec. 5.

2 Problem Statement and Overview of Filtered Basis Functions Approach

Consider the LTI SISO discrete-time system (with transfer function $G(z)$) shown in Fig. 1, controlled by a tracking controller

$C(z)$. The system $G(z)$ could represent the model of an open-loop plant or that of a closed-loop controlled system. Given a desired trajectory, $y_d(k)$, where $0 \leq k \leq E$, $k \in \mathbb{Z}$, and $E + 1$ is the number of discrete points in the trajectory, the objective of tracking control is to design $C(z)$, such that its output (i.e., the control trajectory) $u(k)$, after passing through $G(z)$, results in an output trajectory $y(k)$ that is sufficiently close to $y_d(k)$. Ideally, $C(z) = G(z)^{-1}$ should be selected, such that the overall dynamics $L(z) = C(z)G(z)$, from $y_d(k)$ to $y(k)$, is unity over the entire range of frequencies, resulting in perfect tracking. However, $C(z) = G(z)^{-1}$ is unrealizable if $G(z)$ contains uncancelable zeros. Let

$$G(z) = \frac{B(z)}{A(z)} = \frac{B_s(z)B_u(z)}{A(z)} \quad (1)$$

be the transfer function representation of the system, where $A(z)$, $B(z)$, $B_s(z)$, and $B_u(z)$ are the polynomials in z with constant coefficients. The polynomial $B_s(z)$ contains the stable and sufficiently damped zeros of $G(z)$, while $B_u(z)$ contains its uncancelable zeros which prevent $C(z) = G(z)^{-1}$ from being realized. The state-space representation (minimal realization) of the system is given by

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d u(k) \\ y(k) &= \mathbf{C}_d \mathbf{x}(k) + D_d u(k) \end{aligned} \quad (2)$$

where $\mathbf{x}(k)$ is the state vector of length n_o (equal to the order of the system), while $\mathbf{A}_d$, $\mathbf{B}_d$, $\mathbf{C}_d$, and D_d are the system's state, input, output, and feedforward matrices, respectively. (Note that D_d is a scalar because the system is SISO.) The state-space representation of the system can be converted to its transfer function representation using the relationship

$$G(z) = \mathbf{C}_d(z\mathbf{I}_{n_o} - \mathbf{A}_d)^{-1}\mathbf{B}_d + D_d \quad (3)$$

where $\mathbf{I}_{n_o}$ is the identity matrix of order n_o . It is assumed in this paper that the entire desired trajectory, $y_d(k)$, is known a priori, which is reasonable in many practical tracking control applications, like those in manufacturing, robotics, and aeronautics.

2.1 Filtered Basis Functions (FBF) Approach. The FBF method approximates the control trajectory as

$$u(k) = \sum_{i=0}^n \gamma_i \varphi_i(k) \quad (4)$$

where $\{\varphi_i(k)\}$, $i = 0, 1, \dots, n$, is a set of $n + 1$ linearly independent basis functions, and $\{\gamma_i\}$ is a set of coefficients, $\gamma_i \in \mathbb{R}$. Based on Eq. (4), the output trajectory can be expressed as

$$y(k) = Gu(k) = \sum_{i=0}^n \gamma_i \tilde{\varphi}_i(k) \quad (5)$$

where

$$\tilde{\varphi}_i(k) = G\varphi_i(k) \quad (6)$$

represents the filtered basis functions, i.e., forward filtered by system G (or an accurate model of it). Let us define U and Y , the input and output vector spaces (with an inner product defined as the vector dot product), respectively, as

$$\begin{aligned} U &= \text{span}\{\varphi_i(k)\} \\ Y &= \text{span}\{\tilde{\varphi}_i(k)\} \end{aligned} \quad (7)$$

The vector spaces U and Y are subspaces of $\mathbb{R}^{E+1}$ because any trajectory defined over $E + 1$ discrete-time points belongs to

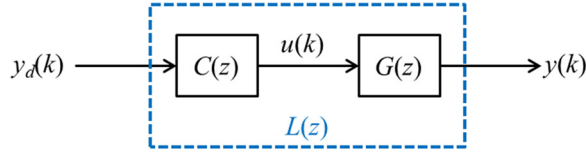


Fig. 1 Block diagram for tracking control

$\mathbb{R}^{E+1}$. Note that the input vector space U is $n+1$ dimensional but the output vector space Y may not be $n+1$ dimensional because the filtered basis functions $\tilde{\varphi}_i(k)$ are not necessarily linearly independent.

For perfect tracking (i.e., $y(k) = y_d(k)$ for all k), the coefficients $\{\gamma_i\}$ should satisfy the relation

$$\underbrace{\begin{bmatrix} \tilde{\varphi}_0(0) & \tilde{\varphi}_1(0) & \dots & \tilde{\varphi}_n(0) \\ \tilde{\varphi}_0(1) & \tilde{\varphi}_1(1) & \dots & \tilde{\varphi}_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\varphi}_0(E) & \tilde{\varphi}_1(E) & \dots & \tilde{\varphi}_n(E) \end{bmatrix}}_{\tilde{\Phi}} \underbrace{\begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_n \end{bmatrix}}_{\gamma} = \underbrace{\begin{bmatrix} y_d(0) \\ y_d(1) \\ \vdots \\ y_d(E) \end{bmatrix}}_{y_d} \quad (8)$$

An exact solution for Eq. (8) exists if and only if the rank of $\tilde{\Phi}$ is equal to the rank of $[\tilde{\Phi} y_d]$, i.e., if $y_d \in Y$. However, for $n < E$, since Eq. (8) is overdetermined [29], generally $y_d \notin Y$, and the solution to Eq. (8) is approximated as [29]

$$\gamma = (\tilde{\Phi}^T \tilde{\Phi})^{-1} \tilde{\Phi}^T y_d \quad (9)$$

The approximation is equivalent to the minimization of $\|\mathbf{e}\|_2^2$, where $\mathbf{e} = [e(0) \ e(1) \ \dots \ e(E)]^T$ is the tracking error vector, based on the definition $e(k) = y_d(k) - y(k)$. Note that the solution to Eq. (9) exists and is unique, if and only if the rank of $\tilde{\Phi}$ is $n+1$, i.e., if and only if the filtered basis functions are linearly independent. Section 2.2 discusses conditions for linear independence of filtered basis functions. Figure 2 provides a pictorial representation of the FBF method. The coefficients γ depend on the given desired trajectory y_d and the filtered basis functions $\tilde{\Phi}$ and hence, on the system $G(z)$ and the basis functions Φ selected by the user.

The relationship between output vector $\mathbf{y} = [y(0) \ y(1) \ \dots \ y(E)]^T$ and desired vector y_d can be expressed as

$$\mathbf{y} = \underbrace{\tilde{\Phi} (\tilde{\Phi}^T \tilde{\Phi})^{-1} \tilde{\Phi}^T}_{\mathbf{L}} y_d \quad (10)$$

The matrix $\mathbf{L}$ is a projection matrix from the desired trajectory to the output trajectory [30], functionally equivalent to $L(z) = C(z)G(z)$ in Fig. 1 (i.e., $\mathbf{L}$ is the convolution matrix or lifted system representation [28] of $L(z)$). It is a square matrix of order $E+1$, symmetric and idempotent (i.e., $\mathbf{L}^2 = \mathbf{L}$), with rank and trace both equal to $n+1$, i.e., the number of basis functions. Similarly, the relationship between the control vector $\mathbf{u} = [u(0) \ u(1) \ \dots \ u(E)]^T$ and the desired vector y_d can be expressed as

$$\mathbf{u} = \underbrace{\Phi (\Phi^T \Phi)^{-1} \Phi^T}_{\mathbf{C}} y_d \quad (11)$$

where

$$\Phi = \begin{bmatrix} \varphi_0(0) & \varphi_1(0) & \dots & \varphi_n(0) \\ \varphi_0(1) & \varphi_1(1) & \dots & \varphi_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_0(E) & \varphi_1(E) & \dots & \varphi_n(E) \end{bmatrix} \quad (12)$$

The matrix $\mathbf{C}$ is a projection matrix from the desired trajectory to the control trajectory, i.e., the convolution matrix of $C(z)$ defined in Fig. 1.

Every digital filter can be expressed as a matrix over a finite time duration (for more details, refer to Appendix A). Hence, each row k of matrices $\mathbf{L}$ and $\mathbf{C}$ represents a finite impulse response (FIR) filter which acts on y_d to generate the corresponding $y(k)$ and $u(k)$, respectively. Note that $\mathbf{L}$ and $\mathbf{C}$ are not necessarily Toeplitz matrices [31] (i.e., matrices having constant entries along their diagonals).

Remark 1. The fact that $\mathbf{L}$ and $\mathbf{C}$ for the FBF method are non-Toeplitz implies that $L(z)$ and $C(z)$ are both equivalent to time-varying FIR filters (for more details, see Appendix A). This constitutes a fundamental difference between the FBF method and techniques like NPZ-ignore, ZPETC, ZMETC, EBZPETC, and TS, whose $\mathbf{L}$ and $\mathbf{C}$ matrices are always Toeplitz because their corresponding $L(z)$ and $C(z)$ are LTI filters (for more details, see Appendix A). It is also interesting to note that the $L(z)$ (or $\mathbf{L}$) of NPZ-ignore, ZPETC, ZMETC, EBZPETC, TS, etc., depends solely on $B_u(z)$, whereas for the FBF method, the elements of $\mathbf{L}$ depend on the entire system dynamics, $G(z)$, as well as the selected basis functions, $\{\varphi_i(k)\}$.

2.2 Linear Independence of Filtered Basis Functions.

A unique solution to Eq. (9) exists if $(\tilde{\Phi}^T \tilde{\Phi})^{-1}$ exists, i.e., if $\tilde{\Phi}^T \tilde{\Phi}$ is invertible. The matrix $\tilde{\Phi}^T \tilde{\Phi}$ is invertible provided $\tilde{\Phi}$ has linearly independent columns, meaning that the rank of the matrix $\tilde{\Phi}$ and the dimension of the vector space Y are both equal to $n+1$ [32]. The linear dependence of filtered basis functions $\{\tilde{\varphi}_i(k)\}$ implies the existence of nonzero $\{\eta_i\}$, such that

$$\begin{aligned} y(k) &= \sum_{i=0}^n \eta_i \tilde{\varphi}_i(k) = 0 \\ u(k) &= \sum_{i=0}^n \eta_i \varphi_i(k) \neq 0 \end{aligned} \quad (13)$$

given that $\{\varphi_i(k)\}$ are linearly independent by definition. Hence, $\{\tilde{\varphi}_i(k)\}$ are linearly dependent if there exists a nonzero $u(k)$, such that

$$y(k) = Gu(k) = 0 \quad (14)$$

implying that a nonzero $u(k)$ belongs to the null space of the system.

For further analysis, the state-space representation of the system given by Eq. (2) is used, because it accounts for nonzero initial conditions. The output $y(k)$ is composed of a zero input response (ZIR) and a zero state response (ZSR) [29], i.e.,

$$y(k) = \underbrace{\mathbf{C}_d \mathbf{A}_d^k \mathbf{x}(0)}_{\text{Zero input response (ZIR)}} + \underbrace{\mathbf{C}_d \sum_{q=0}^{k-1} \mathbf{A}_d^{k-q-1} \mathbf{B}_d u(q) + D_d u(k)}_{\text{Zero state response (ZSR)}} \quad (15)$$

Consequently, the linear dependence of the filtered basis functions requires the determination of $u(k) \neq 0$ and $\mathbf{x}(0)$, such that $y(k) = 0$ for $0 \leq k \leq E$, i.e.,

$$0 = \mathbf{C}_d \mathbf{A}_d^k \mathbf{x}(0) + \mathbf{C}_d \sum_{q=0}^{k-1} \mathbf{A}_d^{k-q-1} \mathbf{B}_d u(q) + D_d u(k) \quad (16)$$

The system of equations given by Eq. (16) represents an underdetermined system [29] (with $E+1$ equations and $E+1+n_o$ unknowns) with infinitely many solutions. The solutions to Eq. (16) form a vector space (denoted as $\hat{U}$) which is n_o dimensional.

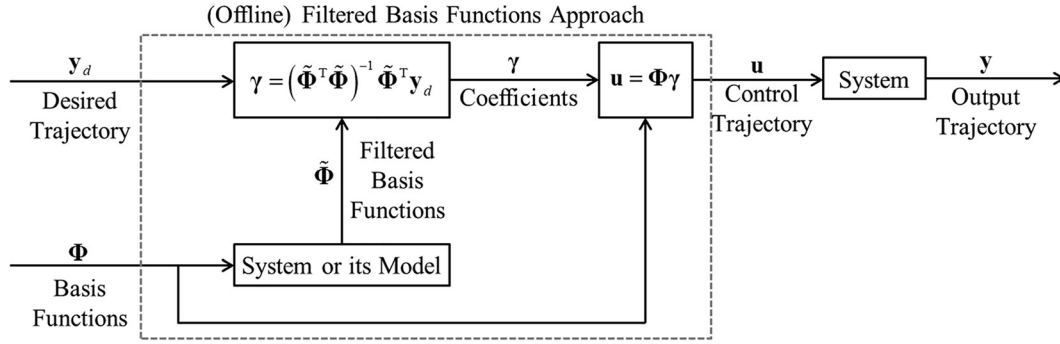


Fig. 2 Flowchart of filtered basis functions approach for tracking NMP systems. The controller can be implemented offline since the desired trajectory is assumed to be entirely known.

If $\mathbf{x}_0(0), \mathbf{x}_1(0), \dots, \mathbf{x}_n(0)$ represent the initial states used to forward filter basis functions $\varphi_0(k), \varphi_1(k), \dots, \varphi_n(k)$, respectively, then the filtered basis functions are linearly dependent if there exists non-zero $\{\eta_i\}$ and $[\hat{\mathbf{u}}^T \quad \hat{\mathbf{x}}^T(0)]^T \in \hat{\mathbf{U}}$, such that

$$\begin{bmatrix} \Phi \\ \mathbf{X} \end{bmatrix} \boldsymbol{\eta} = \begin{bmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{x}}(0) \end{bmatrix} \quad (17)$$

where

$$\begin{aligned} \mathbf{X} &= [\mathbf{x}_0(0) \quad \mathbf{x}_1(0) \quad \dots \quad \mathbf{x}_n(0)] \\ \boldsymbol{\eta} &= [\eta_0 \quad \eta_1 \quad \dots \quad \eta_n]^T \\ \hat{\mathbf{u}} &= [\hat{u}(0) \quad \hat{u}(1) \quad \dots \quad \hat{u}(E)]^T \end{aligned} \quad (18)$$

Therefore, the filtered basis functions $\{\tilde{\varphi}_i(k)\}$ are linearly dependent if and only if the basis functions $\{\varphi_i(k)\}$ and filter initial states $\{\mathbf{x}_i(0)\}$ satisfy the condition

$$\text{rank} \left(\begin{bmatrix} \Phi \\ \mathbf{X} \end{bmatrix} \right) = \text{rank} \left(\begin{bmatrix} \Phi & \hat{\mathbf{u}} \\ \mathbf{X} & \hat{\mathbf{x}}(0) \end{bmatrix} \right) \quad (19)$$

Note that $\hat{\mathbf{u}}$ and $\hat{\mathbf{x}}(0)$ are dependent on the dynamics of a given system, whereas Φ and $\mathbf{X}$ can be freely selected by the user.

Remark 2. The rank condition given by Eq. (19) is very hard to satisfy, meaning that linear dependence of filtered basis functions is highly improbable. Nonetheless, should Eq. (19) be satisfied by a given Φ and $\mathbf{X}$, the control designer can easily modify either Φ or $\mathbf{X}$ to establish linear independence of the filtered basis functions, as is demonstrated using an example in Sec. 4.1.

2.3 Stability. As noted in Remark 1, the FBF method leads to a linear time-varying (LTV) system, whereas methods like ZPETC result in LTI systems which are inherently designed to be stable. Consequently, it is of interest in this section to examine the stability of the LTV system resulting from the FBF method using a BIBO stability approach. Let the bound on the desired trajectory, $y_d(k)$, be given by

$$\begin{aligned} |y_d(k)| &\leq B_{y_d} \\ 0 < B_{y_d} < \infty \end{aligned} \quad (20)$$

Let us denote the elements of matrix $\mathbf{L}$ as $l(j,k)$ ($0 \leq j \leq E$, $0 \leq k \leq E$) and hence any row j of Eq. (10) can be expressed as

$$y(j) = [l(j,0) \quad l(j,1) \quad \dots \quad l(j,E)] \begin{bmatrix} y_d(0) \\ y_d(1) \\ \vdots \\ y_d(E) \end{bmatrix} \quad (21)$$

From Eqs. (20) and (21), one can conclude that

$$|y(j)| \leq \sum_{k=0}^E |l(j,k)| B_{y_d} \quad (22)$$

Hence, $y(k)$ is bounded by

$$\begin{aligned} |y(k)| &\leq \max_{0 \leq j \leq E} \left(\sum_{k=0}^E |l(j,k)| B_{y_d} \right) \\ &\leq \max_{0 \leq j \leq E} \left(\sum_{k=0}^E |l(j,k)| \right) B_{y_d} \\ &\leq \|\mathbf{L}\|_{\infty} B_{y_d} \end{aligned} \quad (23)$$

where $\|\mathbf{L}\|_{\infty}$ represents the infinity norm of matrix $\mathbf{L}$. Similarly, $u(k)$ is bounded by

$$|u(k)| \leq \|\mathbf{C}\|_{\infty} B_{y_d} \quad (24)$$

The implication is that a bounded desired trajectory, $y_d(k)$, results in output, $y(k)$, and control trajectories, $u(k)$, which are bounded by the infinity norms of $\mathbf{L}$ and $\mathbf{C}$, respectively. The bounds on the trajectories depend on matrices $\mathbf{L}$ and $\mathbf{C}$, and hence on the system and the selected basis functions. This implies that, for BIBO stability, the infinity norms of $\mathbf{L}$ and $\mathbf{C}$ should be bounded. The projection matrices $\mathbf{L}$ and $\mathbf{C}$ defined by Eqs. (10) and (11), respectively, are quite cumbersome to analyze due to the presence of $(\tilde{\Phi}^T \tilde{\Phi})^{-1}$. As discussed in Ref. [27], the analysis can be simplified by use of decoupled filtered basis functions $\{\tilde{\psi}_i\}$ in place of filtered basis functions $\{\tilde{\phi}_i\}$, where $\tilde{\phi}_i$ is the i th column of $\tilde{\Phi}$. For more details on decoupled filtered basis functions, see Appendix B. Based on the discussion in Appendix B (Eq. (B11)), the matrix $\mathbf{L}$ can be expressed in terms of decoupled filtered basis functions, as [33]

$$\mathbf{L} = \sum_{i=0}^n \tilde{\psi}_i \tilde{\psi}_i^T = \sum_{i=0}^n \mathbf{L}_i \quad (25)$$

where $\mathbf{L}_i = \tilde{\psi}_i \tilde{\psi}_i^T$. Based on Eq. (25)

$$\|\mathbf{L}\|_{\infty} \leq \sum_{i=0}^n \|\mathbf{L}_i\|_{\infty} \quad (26)$$

and $\|\mathbf{L}_i\|_{\infty}$ is given by

$$\|\mathbf{L}_i\|_{\infty} = \|\tilde{\psi}_i\|_{\infty} \|\tilde{\psi}_i\|_1 \quad (27)$$

The implication of Eqs. (26) and (27) is that

$$\|\mathbf{L}\|_\infty \leq \sum_{i=0}^n \|\tilde{\Psi}_i\|_\infty \|\tilde{\Psi}_i\|_1 \quad (28)$$

Similarly, for matrix $\mathbf{C}$ (based on Appendix B, Eq. (B11))

$$\|\mathbf{C}\|_\infty \leq \sum_{i=0}^n \|\Psi_i\|_\infty \|\tilde{\Psi}_i\|_1 \quad (29)$$

For BIBO stability, $\|\mathbf{L}\|_\infty < \infty$ and $\|\mathbf{C}\|_\infty < \infty$. For a finite number of basis functions, this implies that

$$\begin{aligned} \|\Psi_i\|_\infty &< \infty \\ \|\tilde{\Psi}_i\|_1 &< \infty \end{aligned} \quad (30)$$

The decoupled filtered basis functions $\tilde{\psi}_i(\bar{k})$, $-\infty < \bar{k} < \infty$, $\bar{k} \in \mathbb{Z}$, can be expressed as a convolution of corresponding basis function $\psi_i(\bar{k})$ and system $G(z)$ (with a time-domain representation $g(\bar{k})$) as follows:

$$\tilde{\psi}_i(\bar{k}) = \sum_{\bar{j}=-\infty}^{\infty} \psi_i(\bar{j}) g(\bar{k} - \bar{j}) \quad (31)$$

Based on Eq. (31)

$$\|\tilde{\Psi}_i\|_1 \leq \sum_{\bar{j}=-\infty}^{\infty} |\psi_i(\bar{j})| \sum_{\bar{k}=-\infty}^{\infty} |g(\bar{k} - \bar{j})| \quad (32)$$

The BIBO stability of $G(z)$ implies

$$\sum_{\bar{k}=-\infty}^{\infty} |g(\bar{k})| < \infty \quad (33)$$

The implication of Eqs. (32) and (33) is that $\|\Psi_i\|_1$ (and hence $\|\Phi_i\|_1$, where Φ_i is the i th column of Φ .) should be bounded for BIBO stability, i.e., the basis functions should be absolutely summable.

Remark 3. The results of this section demonstrate that the LTV system resulting from the FBF method is BIBO stable for any finite number of absolutely summable basis functions; basis functions with finite length (i.e., for which $E < \infty$) will always satisfy these criteria.

2.4 Effect of Number of Basis Functions on Tracking Performance. For an orthonormal basis $\{\psi_i(k)\}$, if $\mathbf{y}_d \in Y$, then according to Parseval's identity [34,35] (based on Appendix B, Eq. (B9))

$$\|\mathbf{y}_d\|_2^2 = \mathbf{y}_d \cdot \mathbf{y}_d = \sum_{i=0}^n \left| \mathbf{y}_d \cdot \tilde{\Psi}_i \right|^2 = \sum_{i=0}^n \gamma_{Di}^2 \quad (34)$$

Similarly, if $\mathbf{y}_d \notin Y$, Bessel's inequality [34,35] (based on Appendix B, Eq. (B9)) gives the condition

$$\|\mathbf{y}_d\|_2^2 \geq \sum_{i=0}^n \left| \mathbf{y}_d \cdot \tilde{\Psi}_i \right|^2 = \sum_{i=0}^n \gamma_{Di}^2 \quad (35)$$

The implication is that tracking accuracy generally improves as more basis functions are added, because the inequality condition of Eq. (35) approaches the equality condition of Eq. (34). Assuming that a set of basis functions for which the addition of new basis functions does not alter existing basis functions are selected, then the coefficient of any new basis function (after decoupling) is independent of previously computed coefficients. Consequently,

$$\begin{aligned} \sum_{i=0}^n \gamma_{Di}^2 &\leq \sum_{i=0}^n \gamma_{Di}^2 + \gamma_{D(n+1)}^2 \\ &\leq \sum_{i=0}^{n+1} \gamma_{Di}^2 \end{aligned} \quad (36)$$

indicating that tracking accuracy improves monotonically as more basis functions are added. Frueh and Phan [27] have illustrated this monotonic improvement in accuracy using a simulation example. However, if the set of basis functions is such that new basis functions alter existing basis functions, then the improvement in tracking accuracy may not be monotonic. Note that as n increases, the dimension of the output vector space Y increases and the overdetermined system of equations described by Eq. (8) tends toward a determined system of equations. Similarly, as n increases, the overdetermined system of equations given by Eq. (17) tends toward a determined system of equations, thus increasing the probability of linear dependence of filtered basis functions, as is demonstrated using an example in Sec. 4. It is instructive to examine the limiting situation when $n = E$. Following the procedure outlined in Appendix A, if the convolution matrix (i.e., lifted system representation [28]) of $G(z)$ is denoted as $\mathbf{G}$ (an order $E + 1$ square matrix), then

$$\mathbf{L} = \mathbf{G}\mathbf{C} \quad (37)$$

For $n = E$ (see Appendix B, Eq. (B2))

$$\tilde{\Psi}^T \tilde{\Psi} = \mathbf{I}_{E+1} \quad (38)$$

and $\tilde{\Psi}$ is an order $E + 1$ square matrix. This implies that

$$\tilde{\Psi}^{-1} = \tilde{\Psi}^T \quad (39)$$

Hence, based on Eq. (39) (see Appendix B, Eq. (B11))

$$\mathbf{L} = \mathbf{I}_{E+1} \quad (40)$$

Therefore, from Eqs. (37) and (40)

$$\mathbf{C} = \mathbf{G}^{-1} \quad (41)$$

For a causal LTI system, $\mathbf{G}$ is a lower triangular Toeplitz matrix [31], if the initial state is zero. For a biproper system, $\mathbf{G}$ is an invertible matrix. Therefore, when $n = E$, the matrices $\mathbf{L}$ and $\mathbf{C}$ represent time-invariant FIR filters. The condition $\mathbf{L} = \mathbf{I}_{E+1}$ is equivalent to $L(z) = 1$. For a strictly proper system (i.e., number of poles exceeds the number of zeros by r , $r > 0$), the filtered basis functions are linearly dependent. In this case, the tracking error can be defined as $e(k) = y_d(k) - y(k+r)$ [36] (i.e., r zeros can be added at $z=0$ to make the system biproper), such that the matrix $\mathbf{G}$ becomes invertible [28]. For specific applications, modifications to the system might not be possible nor desirable; in such cases, the user can reduce n slightly to achieve near-perfect tracking (given that perfect tracking is rarely required in practice).

Remark 4. Equation (40) implies that, irrespective of the basis functions employed, the FBF method with $n = E$ theoretically results in perfect tracking, provided $E < \infty$. Perfect tracking is not theoretically achievable by comparable methods like EBZ-PETC and TS because they require an infinite number of terms.

Remark 5. It is well known that if $G^{-1}(z)$ contains NMP zeros, it is not BIBO stable, but, as demonstrated in Sec. 2.3, the FBF method is BIBO stable for $E < \infty$. This is because the FBF equivalent of $C(z)$, i.e., $\mathbf{C} = \mathbf{G}^{-1}$, represents the finite series expansion of $G^{-1}(z)$, which is bounded because of the finite time duration considered.

Note, however, that $\mathbf{G}$ has been shown to become ill-conditioned as $E \rightarrow \infty$, meaning that, even though bounded, its inverse could yield undesirably large control signals [28,37]. A number of approximations have been proposed to improve the condition number of $\mathbf{G}$ at the expense of tracking accuracy [28,38,39]. However, the same goal can be achieved using the FBF method by selecting $n < E$.

3 Basis Functions

There is a wide range of basis functions available for use with the FBF method, e.g., pulses [40], Laguerre functions [41],

wavelets [42], Legendre polynomials [43], discrete cosine transform (DCT) [36], nonuniform rational B-spline (NURBS) [26], etc. The previous work of the authors [26] has illustrated the method using NURBS basis functions. This section introduces two rudimentary basis functions, which are used to elucidate the importance of the choice of basis functions on the performance of the FBF method.

3.1 Pulse Basis Functions. Pulses in time domain are the most rudimentary type of piecewise constant basis functions [40].

$$\varphi_i(k) = \begin{cases} 1 & k \in \left[i \frac{E}{n+1}, (i+1) \frac{E}{n+1} \right), 0 \leq i < n \text{ \& } k \in \left[i \frac{E}{n+1}, (i+1) \frac{E}{n+1} \right], i = n \\ 0 & \text{otherwise} \end{cases} \quad (43)$$

The BPF expressed in Eq. (43) seeks to divide the time interval from 0 to E among $n+1$ basis functions in a uniform manner. However, some pulses may be lengthier than others if $E+1$ is not an integer multiple of $n+1$.

3.2 Discrete Cosine Transform (DCT) Basis Functions. The DCT is a frequency-based transform that is widely used in signal processing applications [44–47]. DCT expresses the signal $u(k)$ as a linear combination of real-valued cosines as follows [36,47]:

$$u(k) = \sum_{i=0}^E \gamma_i \beta_i \cos\left(\frac{\pi(2k+1)i}{2(E+1)}\right) \quad (44)$$

where

$$\beta_i = \begin{cases} \frac{1}{\sqrt{E+1}} & i = 0 \\ \sqrt{\frac{2}{E+1}} & i > 0 \end{cases} \quad (45)$$

The factor β_i is used to preserve the energy of the signal during transformation (i.e., DCT is a unitary transform) [46,47]. The DCT definition given by Eq. (44) results in energy compaction. For most signals, the DCT has its coefficients concentrated at lower frequencies when compared, for example, to the discrete Fourier transform [36,47]. Hence, the first $n+1$ terms in Eq. (44)

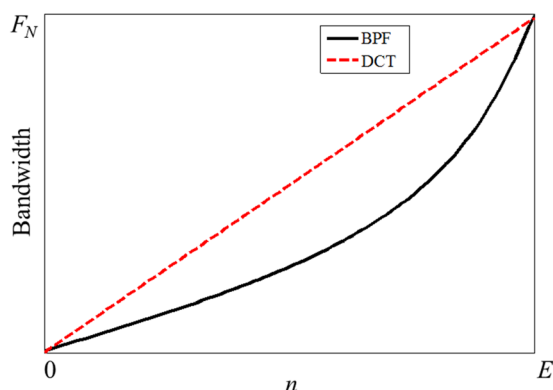


Fig. 3 Variation of 3 dB bandwidth with number of basis functions for BPF and DCT

There are many varieties of time-domain pulses but the most basic is the unit pulse function (UPF) [40]

$$\varphi_i(k) = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases} \quad (42)$$

UPF is restrictive, as $u(k)$ is confined to $\mathbb{R}^{n+1}$. Hence, in this paper, the block pulse function (BPF), which can represent $u(k)$ in $\mathbb{R}^{E+1}$, is adopted; it is defined as [40]

can be used to approximate $u(k)$ with little loss of information [36,46,47], as follows:

$$u(k) = \sum_{i=0}^n \gamma_i \beta_i \cos\left(\frac{\pi(2k+1)i}{2(E+1)}\right) \quad (46)$$

Note the equivalence between Eqs. (4) and (46). Based on this equivalence, the DCT basis functions $\{\varphi_i(k)\}$ are defined as

$$\varphi_i(k) = \beta_i \cos\left(\frac{\pi(2k+1)i}{2(E+1)}\right) \quad (47)$$

Figure 3 compares the 3 dB bandwidths [48] of BPF and DCT basis functions. The 3 dB bandwidths are computed based on Fourier transforms of BPF and DCT basis functions [47]. For both BPF and DCT, the 3 dB bandwidth increases with increase in n . At $n=0$ and $n=E$, both basis functions have the same 3 dB bandwidth, i.e., 0 and F_N (Nyquist frequency), respectively, whereas for $0 < n < E$, DCT has a higher 3 dB bandwidth compared to BPF, for the same n .

Remark 6. Note from Eq. (43) that with BPF, the addition of a new basis function alters existing basis functions because the time interval $[0, E]$ has to be redivided to accommodate the new basis function. However, one observes from Eq. (47) that DCT basis functions are not altered by the addition of new basis functions. As a result, BPF does not satisfy the monotonic convergence property of Eq. (36), while DCT does, as is demonstrated in Sec. 4.2.

Remark 7. Observe from Fig. 3 that except at $n=0$ and $n=E$, BPF requires more basis functions to achieve the same 3 dB bandwidth as DCT. The implication is that, while both basis functions achieve perfect tracking when $n=E$, DCT results in more accurate (approximate) tracking than BPF when $n \in (0, E)$ (for a general desired trajectory, $y_d(k)$, which is not spanned by the filtered BPF basis functions)—as is demonstrated using examples in Sec. 4.

4 Illustrative Examples

4.1 Tracking Comparison. This section illustrates the FBF method using the first-order NMP system studied by Butterworth et al. [8], it is defined as

Table 1 State-space representation

	A_d	B_d	C_d	D_d
$a=2$	0.5	1.0	0.75	−0.50
$a=1.001$	0.5	16.0	15.66	−500.00
$a=−1$	0.5	0.5	0.75	0.25

$$G(z) = K \frac{z-a}{z-p}; \quad K = \frac{1-p}{1-a} \quad (48)$$

where K , a , and p are the gain, zero, and pole of the system, respectively. Note that the gain K is defined such that the direct current (DC) gain of the system is unity. The previous work of the authors [26] compared the FBF method with NPZ-ignore, ZMETC, and ZPETC for the system defined by Eq. (48).

As discussed in Sec. 1, among the more advanced techniques for approximate inversion of NMP systems, the TS method is the most versatile with regards to its applicability irrespective of the location of the NMP zeros in the z -plane; therefore, it is selected for comparison with the FBF method. Using the TS method, the controller $C(z)$ for $G(z)$ defined by Eq. (48) can be expressed as

$$C(z) = \frac{1}{K} (z-p) \left(- \sum_{q=1}^{n_1} \frac{z^{q-1}}{a^q} \right) \frac{1}{1-a^{-n_1}}, \quad |a| > 1 \quad (49)$$

The first two terms in Eq. (49) cancel the stable dynamics (in a manner similar to NPZ-ignore, ZPETC, and ZMETC); the next approximates the inverse of the uncancelable dynamics using a truncated series (where n_1 is the number of terms in the series), whereas the last term ensures unity DC gain for the overall dynamics, $L(z)$. As $n_1 \rightarrow \infty$, $C(z) \rightarrow G^{-1}(z)$, thus achieving perfect tracking. However, in practice n_1 cannot reach infinity, hence a finite series approximation is used, resulting in the overall dynamics $L(z)$, given by

$$L(z) = \frac{1 - \frac{z^{n_1}}{a^{n_1}}}{1 - a^{-n_1}} \quad (50)$$

As such, the gain of $L(z)$ lies within the band

$$\frac{1 \pm a^{-n_1}}{1 - a^{-n_1}} \quad (51)$$

whereas its phase is bounded by

$$\pm \sin^{-1}(a^{-n_1}) \quad (52)$$

The implication is that, as $|a|$ decreases, a higher value of n_1 is required to achieve the same level of tracking accuracy. Note also from Eq. (49) that the TS method cannot be applied to zeros on the unit circle (i.e., $|a| = 1$).

To demonstrate the versatility of the FBF method with regard to handling uncancelable zeros anywhere on the z -plane, three values of the uncancelable zero, a , are examined: $a = 2$ (i.e., zero far from the unit circle), $a = 1.001$ (i.e., zero close to the unit circle), and $a = -1$ (i.e., zero on the unit circle). The three systems have

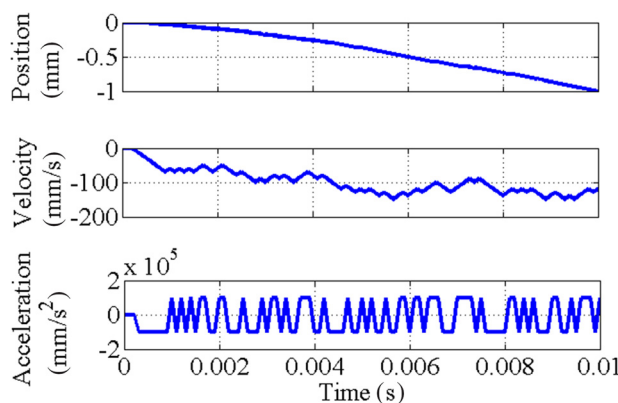


Fig. 4 Desired trajectory ($y_d(k)$) and its first and second derivatives

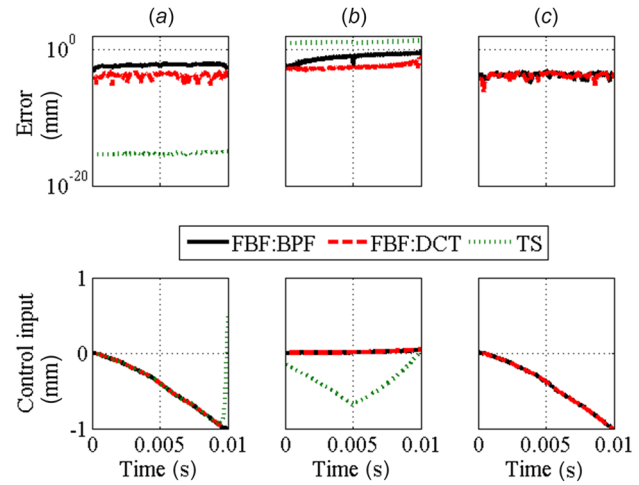


Fig. 5 Comparison of tracking error for FBF (using BPF and DCT) and TS, $n = n_1 = 50$: (a) $a = 2$, (b) $a = 1.001$, and (c) $a = -1$

Table 2 Comparison of root-mean-square (RMS) tracking errors in millimeter for FBF (BPF and DCT) and TS, $n = n_1 = 50$

	BPF	DCT	TS
$a = 2$	5.20×10^{-3}	2.49×10^{-4}	5.67×10^{-16}
$a = 1.001$	1.72×10^{-1}	1.12×10^{-2}	1.22×10^1
$a = -1$	2.71×10^{-4}	2.24×10^{-4}	N/A

the same pole $p = 0.5$ and sampling frequency 10 kHz. Table 1 provides the state-space parameters for each case; note that A_d , B_d , and C_d are scalars because the system is first-order. For the systems under consideration, $\hat{u}(k)$ and $\hat{x}(0)$ can be determined from Eq. (16) as

$$\begin{aligned} \hat{u}(k) &= a^k \hat{u}(0) \\ C_d \hat{x}(0) &= -D_d \hat{u}(0) \end{aligned} \quad (53)$$

Note that $\hat{u}(k)$ is an exponential signal. For comparative analysis between the two types of basis functions, the initial states $\mathbf{X} = 0_{n+1}^T$ are selected. Accordingly, with $n = 50$, the DCT and BPF basis functions both give

$$\text{rank} \left(\begin{bmatrix} \Phi \\ \mathbf{X} \end{bmatrix} \right) = 51 \neq \text{rank} \left(\begin{bmatrix} \Phi & \hat{\mathbf{u}} \\ \mathbf{X} & \hat{\mathbf{x}}(0) \end{bmatrix} \right) = 52 \quad (54)$$

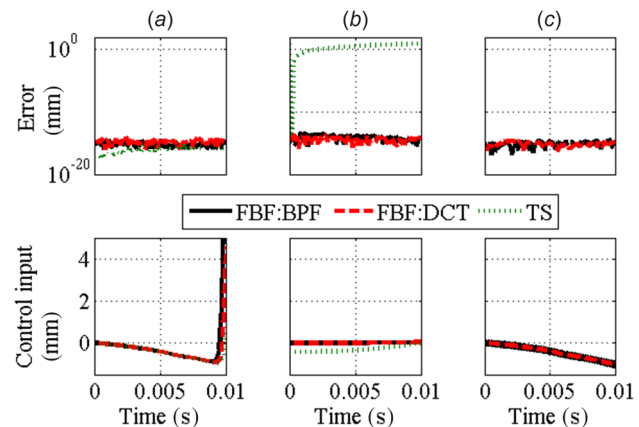


Fig. 6 Comparison of tracking error for FBF (BPF and DCT) and TS, $n = n_1 = 100$: (a) $a = 2$, (b) $a = 1.001$, and (c) $a = -1$

Table 3 Comparison of RMS tracking errors in millimeter for FBF (BPF and DCT) and TS, $n = n_1 = 100$

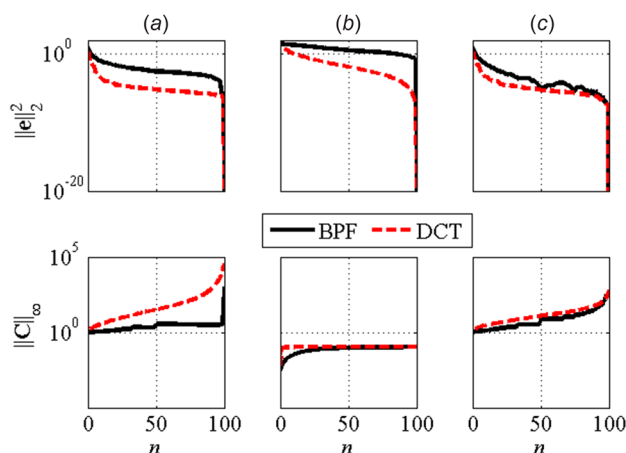
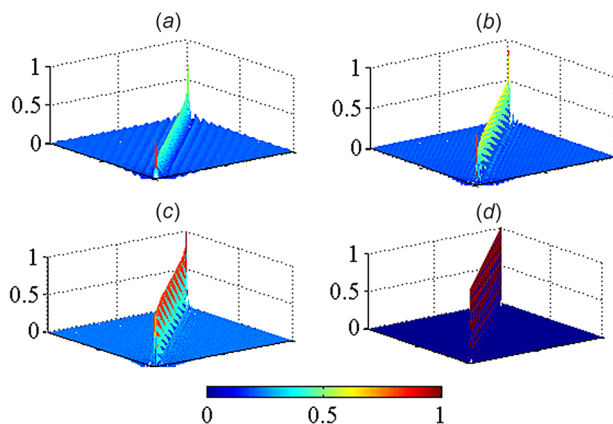
	BPF	DCT	TS
$a = 2$	1.01×10^{-15}	2.37×10^{-15}	1.13×10^{-16}
$a = 1.001$	1.31×10^{-16}	8.02×10^{-15}	4.99×10^0
$a = -1$	1.14×10^{-15}	1.04×10^{-15}	N/A

implying linearly independent filtered basis functions for the three systems.

Figure 4 shows the position, velocity, and acceleration profiles of a desired trajectory used for tracking error comparisons. The acceleration profile is generated using a pseudorandom binary sequence [49] (with $\text{band}=[0 \ 1]$ and $\text{limits}=\pm 10^5 \text{ mm/s}^2$) in MATLAB[®]. The velocity and position profiles are obtained by integration and double integration of the acceleration profile, respectively. The sampling frequency is 10 kHz, and the total duration is 0.01 s (i.e., $E=100$). Figure 5 compares the tracking errors and control input signals of FBF (using BPF and DCT as basis functions) and TS for the case where $n = n_1 = 50$ (i.e., $E/2$). Table 2 shows their RMS tracking errors. With $a = 2$, TS shows much better tracking performance than the two cases of FBF. However, with $a = 1.001$, the tracking performance of TS becomes worse than both FBF cases because it requires more terms to accurately track systems with zeros close to the unit circle. When $a = -1$, the TS method fails but the FBF method works very well, indicating its versatility. Note that in all the cases, FBF using DCT is slightly better than FBF using BPF. While it is possible to select the desired trajectory, $y_d(k)$, such that BPF outperforms DCT (for example, by defining $y_d(k)$ as a linear combination of the filtered BPF basis functions), the frequency characteristics of DCT (discussed in Sec. 3) make it in general better than BPF in tracking an arbitrary desired trajectory. The implication is that a judicious choice of basis functions could help in improving the tracking performance of the FBF method.

Another, perhaps more straightforward, way of improving the tracking performance of FBF is to increase n . Let us consider the limiting case where $n = E = 100$ for tracking the desired trajectory given by Fig. 4. The corresponding limit for TS is $n_1 \rightarrow \infty$, which is unrealizable; therefore, $n_1 = 100$ is selected. As discussed in Sec. 2.4, for $n = E$, the initial states are crucial. For example, for $a = 2$ and $n = 100$, the rank test for $\mathbf{X} = \mathbf{0}_{n+1}^T$ gives

$$\text{rank}\left(\begin{bmatrix} \Phi \\ \mathbf{X} \end{bmatrix}\right) = \text{rank}\left(\begin{bmatrix} \Phi & \hat{\mathbf{u}} \\ \mathbf{X} & \hat{\mathbf{x}}(0) \end{bmatrix}\right) = 101 \quad (55)$$

**Fig. 7 Effect of n on error and control matrix norms: (a) $a = 2$, (b) $a = 1.001$, and (c) $a = -1$** **Fig. 8 Effect of n on $\mathbf{L}$ for DCT basis functions: (a) $n = 25$, (b) $n = 50$, (c) $n = 75$, and (d) $n = 100$**

implying linear dependence. However, changing the initial states to $\mathbf{x}_i(0) = 10^{-3}$ results in

$$\text{rank}\left(\begin{bmatrix} \Phi \\ \mathbf{X} \end{bmatrix}\right) = 101 \neq \text{rank}\left(\begin{bmatrix} \Phi & \hat{\mathbf{u}} \\ \mathbf{X} & \hat{\mathbf{x}}(0) \end{bmatrix}\right) = 102 \quad (56)$$

i.e., linearly independent filtered basis functions. Figure 6 compares the tracking errors and control inputs for FBF (using BPF and DCT) with TS for the three systems. Table 3 compares their RMS tracking errors.

Remark 8. Note that in all the three cases, FBF (irrespective of the basis functions used) achieves perfect tracking (to the extent that is numerically possible). However, achieving perfect tracking comes at the cost of higher or more oscillatory control signals, which may be problematic. Therefore, it makes sense to select the value of n that provides the needed level of tracking accuracy rather than opting for perfect tracking, which is often not necessary in practice.

4.2 Effect of Number of Basis Functions on Tracking Accuracy. To corroborate the discussion in Secs. 2.4 and 3, regarding the convergence of the FBF method as n increases, Fig. 7 shows the effect of n on $\|\mathbf{e}\|_2^2$ in tracking the desired trajectory of Fig. 4 for the three systems under consideration. As n increases, $\|\mathbf{e}\|_2^2$ decreases and at $n = E = 100$, BPF and DCT achieve perfect tracking with $\|\mathbf{e}\|_2^2 \approx 0$. The decrease in $\|\mathbf{e}\|_2^2$ for DCT is monotonic, while that for BPF (specifically in Fig. 7(c)) is not necessarily monotonic, as noted in Remark 6. Note that for all the three cases, DCT results in lower errors compared to BPF, which follows the general observation made in Remark 7.

As discussed in Sec. 2.3, the bound on control trajectory $u(k)$ is proportional to $\|\mathbf{C}\|_\infty$. Figure 7 shows the variation of $\|\mathbf{C}\|_\infty$ with n . Note that as n increases, the bounds on the control trajectories increase, indicating the control effort penalty associated with improving tracking accuracy. Figure 8 shows the effect of n on elements of $\mathbf{L}$, using DCT basis functions. As n increases, the contribution of diagonal terms in matrix $\mathbf{L}$ increases. At $n = E = 100$, $\mathbf{L} = \mathbf{I}_{101}$ results in perfect tracking. A similar pattern can be observed for BPF.

5 Conclusions and Future Work

This paper has presented a theoretical study of the FBF approach in the context of tracking control of discrete-time LTI SISO NMP systems. In the FBF method, it is assumed that the desired trajectory to be tracked is entirely known. Accordingly, the control input is expressed as a linear combination of linearly independent basis functions which are forward filtered using the

system model, and their coefficients are selected to minimize the norm of the tracking error. It is shown that the FBF solution exists and is unique if the FBF are linearly independent. A rank test is established to ascertain linear independence of FBFs, based on the unfiltered basis functions, filter initial states, and system parameters. It is shown that the linear independence of FBFs can be easily achieved by properly selecting the basis functions and filter initial states.

The analyses in this paper show that the FBF method generally results in a linear time-varying system, indicating its fundamental difference from methods like ZPETC which always result in LTI systems. The concept of decoupled filtered basis functions is used to study the stability of the FBF method and the effect of the number of basis functions on its tracking accuracy. It is shown that any set of finite duration bounded basis functions results in a BIBO stable implementation of the FBF method. Moreover, its tracking accuracy improves (monotonically in some cases) as the number of basis functions increases, converging to perfect tracking when the number of basis functions equals the number of discrete points in the desired trajectory.

Two rudimentary basis functions are introduced and used in simulations to evaluate the tracking performance of the FBF method in comparison with the truncated series (TS) method. The tracking performance of the FBF method is shown to depend on a judicious selection of basis functions, as well as the number of basis functions. Most notably, however, the FBF method is shown to be effective for tracking any given trajectory irrespective of the location of the uncancelable zeros in the z -plane, while other methods do not demonstrate such levels of flexibility.

Ongoing research focuses on experimental implementation of the FBF method [50] and robustness [51] of the FBF method to errors in the system model. Future work will seek to relax the assumption that the entire desired trajectory is known a priori.

Acknowledgment

This work was partially funded by the National Science Foundation's CAREER Award No. 1350202 and the second author's Rackham Centennial Award. Both awards are gratefully acknowledged.

Appendix A: Matrix Representation of a Digital Filter

Consider filter p , input signal u , and output signal y defined as

$$\begin{aligned} p &= \{p_{-2} \quad p_{-1} \quad p_0 \quad p_1 \quad p_2\} \\ u &= \{u_0 \quad u_1 \quad u_2\} \\ y &= \{y_0 \quad y_1 \quad y_2\} \end{aligned} \quad (\text{A1})$$

Signals y and u and filter p are related by the convolution operator as follows:

$$y = u * p \quad (\text{A2})$$

From Eqs. (A1) and (A2)

$$\begin{aligned} y_0 &= p_0 u_0 + p_{-1} u_1 + p_{-2} u_2 \\ y_1 &= p_1 u_0 + p_0 u_1 + p_{-1} u_2 \\ y_2 &= p_2 u_0 + p_1 u_1 + p_0 u_2 \end{aligned} \quad (\text{A3})$$

This can be expressed in matrix form as

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} p_0 & p_{-1} & p_{-2} \\ p_1 & p_0 & p_{-1} \\ p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} \quad (\text{A4})$$

Note that the main diagonal element (p_0) represents the influence of the current input on the current output; the first upper diagonal element (p_{-1}) represents the influence of the succeeding input on the current output and the second upper diagonal element (p_{-2}) represents the influence of the second succeeding input on the current output. Similarly, the first (p_1) and second lower (p_2) diagonal elements represent the influence of first and second preceding inputs on the current output, respectively. Hence, the z -transform of p obtained from Eq. (A4) is given by

$$p_2 z^{-2} + p_1 z^{-1} + p_0 z^0 + p_{-1} z^1 + p_{-2} z^2 \quad (\text{A5})$$

which is in accordance with the time-domain definition given by Eq. (A1). In this manner, any digital filter can be expressed as a matrix and vice versa. Also, note that an LTI filter has a Toeplitz matrix representation (i.e., diagonal-constant matrix).

Appendix B: Decoupled Filtered Basis Functions

The approximate solution to Eq. (8) is obtained by projecting the desired trajectory, y_d , onto the vector space, Y . As is evident from Eq. (9), the coefficients corresponding to any basis function depend on all the filtered basis functions, which implies that the filtered basis functions are coupled. Hence, Eqs. (9)–(11) require the inversion of an order $n+1$ matrix, $\tilde{\Phi}^T \tilde{\Phi}$, which is cumbersome numerically and analytically, especially as n increases. The analyses and numerical efficiency of the FBF method are therefore facilitated by decoupling the filtered basis functions.

Let us denote the decoupled filtered basis functions as $\{\tilde{\psi}_i(k)\}$ and the corresponding unfiltered basis functions as $\{\psi_i(k)\}$ such that

$$\tilde{\psi}_i(k) = G \psi_i(k) \quad (\text{B1})$$

and

$$\tilde{\Psi}^T \tilde{\Psi} = \mathbf{I}_{n+1} \quad (\text{B2})$$

where

$$\begin{aligned} \tilde{\Psi} &= \begin{bmatrix} \tilde{\psi}_0 & \tilde{\psi}_1 & \dots & \tilde{\psi}_n \end{bmatrix} \\ &= \begin{bmatrix} \tilde{\psi}_0(0) & \tilde{\psi}_1(0) & \dots & \tilde{\psi}_n(0) \\ \tilde{\psi}_0(1) & \tilde{\psi}_1(1) & \dots & \tilde{\psi}_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\psi}_0(E) & \tilde{\psi}_1(E) & \dots & \tilde{\psi}_n(E) \end{bmatrix} \end{aligned} \quad (\text{B3})$$

The coupled and decoupled filtered basis functions are related by the transformation matrix α as follows:

$$\begin{aligned} \tilde{\Phi} &= \tilde{\Psi} \alpha \\ \Phi &= \Psi \alpha \end{aligned} \quad (\text{B4})$$

where

$$\begin{aligned} \Psi &= \begin{bmatrix} \psi_0 & \psi_1 & \dots & \psi_n \end{bmatrix} \\ &= \begin{bmatrix} \psi_0(0) & \psi_1(0) & \dots & \psi_n(0) \\ \psi_0(1) & \psi_1(1) & \dots & \psi_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_0(E) & \psi_1(E) & \dots & \psi_n(E) \end{bmatrix} \end{aligned} \quad (\text{B5})$$

Based on Eqs. (B2) and (B4), the matrix $\tilde{\Phi}^T \tilde{\Phi}$ can be expressed as

$$\tilde{\Phi}^T \tilde{\Phi} = \alpha^T \tilde{\Psi}^T \tilde{\Psi} \alpha = \alpha^T \alpha \quad (\text{B6})$$

Without loss of generality, let us assume that α is an upper triangular matrix that can be obtained from Cholesky decomposition of $\tilde{\Phi}^T \tilde{\Phi}$ [31]. Accordingly, Eq. (8) can be written as

$$\tilde{\Psi} \gamma_D = y_d \quad (B7)$$

where $\gamma_D = [\gamma_{D0} \ \gamma_{D1} \ \dots \ \gamma_{Dn}]^T$ denotes the coefficients of the decoupled filtered basis functions. Note that although the coefficients change, the control and output trajectories as well as the matrices L and C are invariant under decoupling, because the input (U) and output (Y) vector spaces are not changed by transformations [29]. The approximate solution to Eq. (B7) is

$$\gamma_D = \tilde{\Psi}^T y_d \quad (B8)$$

and the coefficient corresponding to the i th basis function is given by

$$\gamma_{Di} = \tilde{\psi}_i \cdot y_d \quad (B9)$$

The implication is that any coefficient computed using a given decoupled filtered basis function does not depend on other decoupled filtered basis functions. Using the decoupled filtered basis functions, the control trajectory is given by

$$u = \Psi \gamma_D = \Phi \alpha^{-1} \gamma_D \quad (B10)$$

while the matrices L and C can be expressed as

$$\begin{aligned} L &= \tilde{\Psi} \tilde{\Psi}^T = \sum_{i=0}^n \tilde{\psi}_i \tilde{\psi}_i^T \\ C &= \Psi \tilde{\Psi}^T = \sum_{i=0}^n \psi_i \tilde{\psi}_i^T \end{aligned} \quad (B11)$$

The elements of L ($l(j,k)$) and C ($c(j,k)$) are given by $\sum_{i=0}^n \tilde{\psi}_i(j) \tilde{\psi}_i(k)$ and $\sum_{i=0}^n \psi_i(j) \tilde{\psi}_i(k)$, respectively. This clearly shows that the diagonal elements of the two convolution matrices are necessarily not constant and hence, they are non-Toeplitz.

Note that decoupling has replaced the inversion of $\tilde{\Phi}^T \tilde{\Phi}$ with the inversion of α . The conditions for the existence of α^{-1} are the same as those for the existence of $(\tilde{\Phi}^T \tilde{\Phi})^{-1}$, as discussed in Sec. 2.2. Moreover, the upper triangular form of α enables Eq. (B8) and the decoupled equivalents of Eqs. (10) and (11) to be computed recursively [27], thereby eliminating the need for the inversion of an order $n+1$ matrix.

Decoupling also facilitates analysis, as is demonstrated in Sec. 2.3 for analysis of stability of the FBF method and in Sec. 2.4 to study the effect of number of basis functions on tracking performance.

References

- [1] Tomizuka, M., 1987, "Zero Phase Error Tracking Algorithm for Digital Control," *ASME J. Dyn. Syst. Meas. Control*, **109**(1), pp. 65–68.
- [2] Astrom, K., and Wittenmark, B., 1984, *Computer Controlled Systems: Theory and Design*, Prentice Hall, Upper Saddle River, NJ.
- [3] Miu, D. K., 1993, *Mechatronics: Electromechanics and Contromechanics*, Springer, Heidelberg, Germany.
- [4] Gross, E., and Tomizuka, M., 1994, "Experimental Flexible Beam Tip Tracking Control With a Truncated Series Approximation to Uncancelable Inverse Dynamics," *IEEE Trans. Control Syst. Technol.*, **2**(4), pp. 382–391.
- [5] Torfs, D., De Schutter, J., and Swevers, J., 1992, "Extended Bandwidth Zero Phase Error Tracking Control of Nonminimal Phase Systems," *ASME J. Dyn. Syst. Meas. Control*, **114**(3), pp. 347–351.
- [6] Yamada, M., Funahashi, Y., and Fujiwara, S., 1997, "Zero Phase Error Tracking System With Arbitrarily Specified Gain Characteristics," *ASME J. Dyn. Syst. Meas. Control*, **119**(2), p. 260.
- [7] Yamada, M., Funahashi, Y., and Riadh, Z., 1999, "Generalized Optimal Zero Phase Error Tracking Controller Design," *ASME J. Dyn. Syst. Meas. Control*, **121**(2), p. 165.
- [8] Butterworth, J. A., Pao, L. Y., and Abramovitch, D. Y., 2012, "Analysis and Comparison of Three Discrete-Time Feedforward Model-Inverse Control Techniques for Nonminimum-Phase Systems," *Mechatronics*, **22**(5), pp. 577–587.
- [9] Fujimoto, H., Hori, Y., and Kawamura, A., 2001, "Perfect Tracking Control Based on Multirate Feedforward Control With Generalized Sampling Periods," *IEEE Trans. Ind. Electron.*, **48**(3), pp. 636–644.
- [10] Wen, J. T., and Potsaid, B., 2004, "An Experimental Study of a High Performance Motion Control System," American Control Conference, pp. 5158–5163.
- [11] Weck, M., and Ye, G., 1990, "Sharp Corner Tracking Using the IKF Control Strategy," *CIRP Ann. Technol.*, **39**(1), pp. 437–441.
- [12] Rigney, B. P., Pao, L. Y., and Lawrence, D. A., 2009, "Nonminimum Phase Dynamic Inversion for Settle Time Applications," *IEEE Trans. Control Syst. Technol.*, **17**(5), pp. 989–1005.
- [13] Haack, B., and Tomizuka, M., 1991, "The Effect of Adding Zeros to Feedforward Controllers," *ASME J. Dyn. Syst. Meas. Control*, **113**(1), pp. 6–10.
- [14] Hunt, L., Meyer, G., and Su, R., 1996, "Noncausal Inverses for Linear Systems," *IEEE Trans. Autom. Control*, **41**(4), pp. 608–611.
- [15] Devasia, S., Chen, D., and Paden, B., 1996, "Nonlinear Inversion-Based Output Tracking," *IEEE Trans. Autom. Control*, **41**(7), pp. 930–942.
- [16] Marro, G., Prattichizzo, D., and Zattoni, E., 2002, "Convolution Profiles for Right Inversion of Multivariable Non-Minimum Phase Discrete-Time Systems," *Automatica*, **38**(10), pp. 1695–1703.
- [17] Kwon, D.-S., and Book, W. J., 1994, "A Time-Domain Inverse Dynamic Tracking Control of a Single-Link Flexible Manipulator," *ASME J. Dyn. Syst. Meas. Control*, **116**(2), p. 193.
- [18] Zou, Q., and Devasia, S., 1999, "Preview-Based Stable-Inversion for Output Tracking of Linear Systems," *ASME J. Dyn. Syst. Meas. Control*, **121**(4), pp. 625–630.
- [19] Marconi, L., Marro, G., and Melchiorri, C., 2001, "A Solution Technique for Almost Perfect Tracking of Non-Minimum-Phase, Discrete-Time Linear Systems," *Int. J. Control*, **74**(5), pp. 496–506.
- [20] Piazza, A., and Visioli, A., 2005, "Using Stable Input-Output Inversion for Minimum-Time Feedforward Constrained Regulation of Scalar Systems," *Automatica*, **41**(2), pp. 305–313.
- [21] Jetto, L., Orsini, V., and Romagnoli, R., 2014, "Accurate Output Tracking for Nonminimum Phase Nonhyperbolic and Near Nonhyperbolic Systems," *Eur. J. Control*, **20**(6), pp. 292–300.
- [22] Devasia, S., 2011, "Nonlinear Minimum-Time Control With Pre- and Post-Actuation," *Automatica*, **47**(7), pp. 1379–1387.
- [23] Devasia, S., 1997, "Output Tracking With Nonhyperbolic and Near Nonhyperbolic Internal Dynamics: Helicopter Hover Control," *J. Guid. Control Dyn.*, **20**(3), pp. 573–580.
- [24] Wang, H., Kim, K., and Zou, Q., 2013, "B-Spline-Decomposition-Based Output Tracking With Preview for Nonminimum-Phase Linear Systems," *Automatica*, **49**(5), pp. 1295–1303.
- [25] Jetto, L., Orsini, V., and Romagnoli, R., 2015, "Spline Based Pseudo-Inversion of Sampled Data Non-Minimum Phase Systems for an Almost Exact Output Tracking," *Asian J. Control*, **17**(5), pp. 1866–1879.
- [26] Duan, M., Ramani, K. S., and Okwudire, C. E., 2015, "Tracking Control of Non-Minimum Phase Systems Using Filtered Basis Functions: A NURBS-Based Approach," *ASME Paper No. DSCC2015-9859*.
- [27] Frueh, J. A., and Phan, M. Q., 2000, "Linear Quadratic Optimal Learning Control (LQL)," *Int. J. Control*, **73**(10), pp. 832–839.
- [28] Lunenburg, J., Bosgra, O., and Oomen, T., 2009, "Inversion-Based Feedforward Design for Beyond Rigid Body Systems: A Literature Survey," DCT Report No. 2009.105, Eindhoven University of Technology, Eindhoven, The Netherlands.
- [29] Bay, J. S., 1999, *Fundamentals of Linear State Space Systems*, McGraw-Hill Science, Engineering & Mathematics, New York.
- [30] Hoaglin, D. C., and Welsch, R. E., 1978, "The Hat Matrix in Regression and ANOVA," *Am. Stat.*, **32**(1), pp. 17–22.
- [31] Bernstein, D. S., 2009, *Matrix Mathematics: Theory, Facts, and Formulas*, Princeton University Press, Princeton, NJ.
- [32] Strang, G., 1993, "The Fundamental Theorem of Linear Algebra," *Am. Math. Mon.*, **100**(9), pp. 848–855.
- [33] Laub, A. J., 2005, *Matrix Analysis for Scientists and Engineers*, SIAM, Philadelphia, PA.
- [34] Zygmund, A., 1968, *Trigonometric Series*, Cambridge University Press, New York.
- [35] Johnson, L., and Riess, R., 1982, *Numerical Analysis*, Addison-Wesley, Reading, MA.
- [36] Ye, Y., and Wang, D., 2005, "DCT Basis Function Learning Control," *IEEE/ASME Trans. Mechatronics*, **10**(4), pp. 449–454.
- [37] Chu, B., and Owens, D., 2013, "Singular Value Distribution of Non-Minimum Phase Systems With Application to Iterative Learning Control," 52nd IEEE Conference on Decision and Control, IEEE, pp. 6700–6705.
- [38] Ronde, M., and van den Bulk, J., 2013, "Feedforward for Flexible Systems With Time-Varying Performance Locations," 2013 American Control Conference, pp. 6033–6038.
- [39] Ronde, M., van de Molengraft, R., and Steinbuch, M., 2012, "Model-Based Feedforward for Inferential Motion Systems, With Application to a Prototype Lightweight Motion System," American Control Conference (ACC), **2012**, pp. 5324–5329.
- [40] Deb, A., Sarkar, G., and Sen, S. K., 1994, "Block Pulse Functions, the Most Fundamental of all Piecewise Constant Basis Functions," *Int. J. Syst. Sci.*, **25**(2), pp. 351–363.

- [41] Hamamoto, K., and Sugie, T., 2001, "An Iterative Learning Control Algorithm Within Prescribed Input-Output Subspace," *Automatica*, **37**(11), pp. 1803–1809.
- [42] Gopinath, S., Kar, I., and Bhatt, R., 2009, "Wavelet Series Based Learning Controller Design for Kinematic Path-Tracking Control of Mobile Robot," International Conference on Advances in Computing, Communication and Control, pp. 129–135.
- [43] Phan, M., and Frueh, J., 1996, "Learning Control for Trajectory Tracking Using Basis Functions," 35th IEEE Conference on Decision and Control, pp. 2490–2492.
- [44] Ahmed, N., Natarajan, T., and Rao, K. R., 1974, "Discrete Cosine Transform," *IEEE Trans. Comput.*, **100**(1), pp. 90–93.
- [45] Rao, K., and Yip, P., 1990, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press Professional – Elsevier, Amsterdam, The Netherlands.
- [46] Jain, A. K., 1989, *Fundamentals of Digital Image Processing*, Prentice-Hall, Upper Saddle River, NJ.
- [47] Oppenheim, A. V., and Schaffer, R. W., 1999, *Discrete-Time Signal Processing*, Prentice Hall, Upper Saddle River, NJ.
- [48] Couch, L. W., 2001, *Digital and Analog Communication Systems*, Prentice-Hall, Upper Saddle River, NJ.
- [49] Bardell, P. H., McAnney, W. H., and Savir, J., 1987, *Built-In Test for VLSI: Pseudorandom Techniques*, Wiley-Interscience, New York.
- [50] Okwudire, C., Ramani, K., and Duan, M., 2016, "A Trajectory Optimization Method for Improved Tracking of Motion Commands Using CNC Machines That Experience Unwanted Vibration," *CIRP Ann. Manuf. Technol.*, **65**(1), pp. 373–376.
- [51] Ramani, K. S., and Okwudire, C. E., 2016, "Regularized Filtered Basis Functions Approach for Accurate Tracking of Discrete-Time Linear Time Invariant Systems With Bounded Random Uncertainties," ASME Paper No. DSCC2016-9885.