

BOOSTING SPEED AND ACCURACY IN PRECISION MOTION CONTROL:

A CASE FOR A CONSISTENT AND VERSATILE FEEDFORWARD TRACKING APPROACH

Feedforward (FF) control uses *a priori* knowledge about a given system and its disturbances to influence the system's behavior in a pre-defined way. However, unlike feedback (FB) control, it does not adjust the control signal (or manipulated variable) in response to how the system actually reacts. In other words, FF control is proactive while FB control is reactive.

Model-inversion-based FF (MIBFF) control is a specific type of FF control where the control variable is determined by inverting a model of the system to be controlled. It is very popular for its use in output tracking control problems, where the goal is to force the system's output to follow a desired trajectory. Output tracking is critical in several different applications, e.g., in manufacturing, robotics, automotive, aerospace, and is typically achieved using a combination of FF and FB control. As motivating examples, consider MIBFF in ultra-precise wafer scanners used in photolithography (see **Sidebar S1**) and in desktop 3D printing (see **Sidebar S2**). As these motivating examples illustrate, non-minimum phase (NMP) zeros often arise in practical engineering applications due to non-collocated sensors and actuators, fast sampling, etc. MIBFF for

**BY KEVAL S. RAMANI
MOLONG DUAN
DEOKKYUN YOON
CHINEDUM E. OKWUDIRE
A. GALIP ULSOY**

DEPARTMENT OF
MECHANICAL ENGINEERING,
UNIVERSITY OF MICHIGAN,
ANN ARBOR

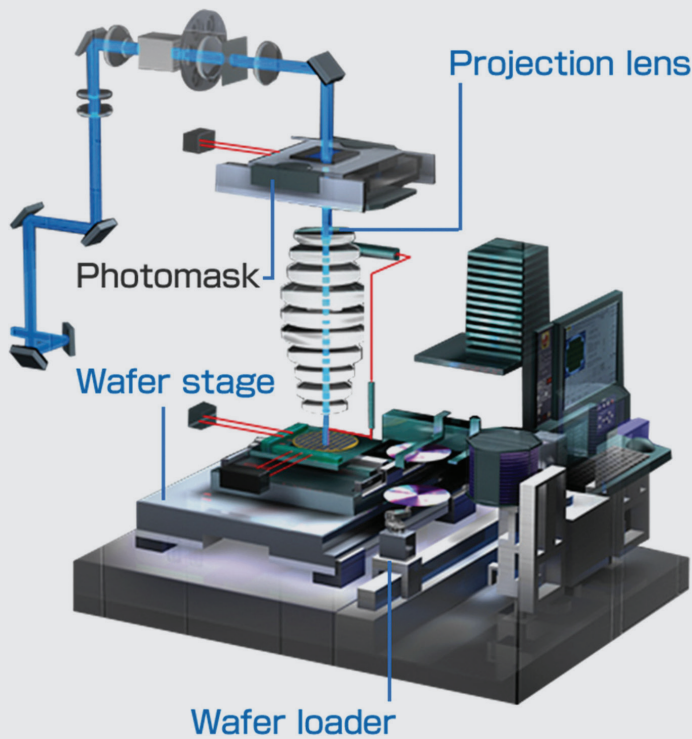


FIGURE 1 Schematic of a wafer scanner; it is critical to integrated circuit manufacturing and relies >99% on FF control.

S1: BEYOND RIGID BODY FF CONTROL OF WAFER SCANNERS FOR PHOTOLITHOGRAPHY

A wafer scanner (see **Figure 1**) is a device used to deliver ultra-precise motion needed for the photolithography process in integrated circuit manufacturing. Wafer scanners must achieve nanometer-level servo errors in tracking motion trajectories with velocities of up to 1 m/s and 10 g acceleration [1]. Over 99% of the control effort required for generating the desired motion is contributed by FF control [2], because wafer scanners are specially designed to minimize uncertainties and unknown disturbances. In a typical wafer scanner, MIBFF is achieved by inverting a rigid body model of a moving stage of mass m (i.e., control input, $u = ma = m\ddot{y}_d$). However, as \ddot{y}_d (desired acceleration) increases, there is a push to reduce m so that the required u is not excessive. An unintended result of reducing m is that the stiffness of the stage reduces, thus invalidating the rigid body assumption used for MIBFF. This has given rise to so-called *beyond rigid body* (BRB) FF control, which considers the flexible modes of the stage in MIBFF control of wafer scanners [1][2]. A major concern that arises with the inclusion of flexibilities in MIBFF control is the presence of NMP zeros, which are almost certain to occur when sensors and actuators are non-collocated as in wafer scanners. Moreover, model uncertainties increase as high-frequency vibration modes are introduced into MIBFF. These issues are further complicated by the fact that the structural dynamics vary from location to location on the stage, hence a position (and time) varying model is required for MIBFF control of wafer scanners.

systems with NMP zeros is arguably the most important (and certainly the most researched) issue related to MIBFF.

METHODS FOR FF CONTROL OF SYSTEMS WITH NMP ZEROS

The FF output tracking problem is illustrated in block diagram form in **Figure 3**. Consider the simple case of a linear time invariant (LTI) single-input single-output (SISO) discrete-time system (with transfer function $G(q)$), controlled by a tracking controller $C(q)$, where q is the forward shift operator. The system $G(q)$ 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 $0 \leq k \leq M$, $k \in \mathbb{Z}$ and $M+1$ is the number of discrete points in the trajectory, the objective of tracking control is to design $C(q)$ such that its output (i.e., the control trajectory) $u(k)$, after passing through $G(q)$, results in an output trajectory $y(k)$ that is sufficiently close to $y_d(k)$.

Ideally, $C(q) = G(q)^{-1}$ should be selected such that the overall transfer function $L(q) = C(q)G(q)$, from $y_d(k)$ to $y(k)$, is unity over the entire range of frequencies, resulting in perfect tracking. However, $C(q) = G(q)^{-1}$ yields unbounded or highly-oscillatory control signal, $u(k)$, if $G(q)$ contains NMP zeros.

A lot of research has been done on developing methods for tracking control of systems with NMP zeros [3]-[5]. The simplest methods are NMP zero ignore (NPZ-ignore), zero phase error tracking controller (ZPETC), and zero magnitude error controller (ZMETC) [4]. NPZ-ignore cancels all poles and cancellable zeros while ignoring the NMP and poorly-damped zeros; it results in a controlled system that exhibits magnitude and phase errors between its desired and output trajectories. ZMETC focuses on cancelling magnitude errors across all frequencies at the expense of phase errors, whereas zero phase error tracking control (ZPETC) focuses on cancelling phase errors across all frequencies at the expense of magnitude errors.

However, depending on the system and the performance specifications, NPZ-ignore, ZMETC and ZPETC may not yield

satisfactory tracking accuracy due to the approximations involved [4]. To improve tracking accuracy, advanced methods have been developed, e.g., extended bandwidth ZPETC, truncated series (TS), direct inversion with bounded reference trajectories, approximate frequency domain inversion, H_∞ matching, B-spline-based tracking with preview using iterative learning control, spline filtering with feedback, causal/anti-causal dynamics decomposition, etc. (as summarized in excellent review papers like [3]-[5]). A major shortcoming of most advanced methods is that they are not versatile in terms of the systems and/or the desired trajectories to which they are applicable (e.g., several of the methods cannot be applied to nonhyperbolic systems, i.e., systems with zeros on the unit circle) – see [7] for a more detailed discussion of this matter. Moreover, their tracking accuracy varies significantly depending on NMP zero location (in the complex plane) [7][8].

The filtered basis function (FBF) approach has been proposed by the authors as a means for achieving consistent feedforward tracking accuracy regardless of the location of the system NMP zeros [6][7]. The FBF approach traces its origins to the work of Frueh and Phan on Inverse Linear Quadratic Learning in the context of Iterative Learning Control [9], and also bears some similarities to less-versatile techniques proposed by Lunenberg et al. [1] and Jetto et al. [10]. *Without any system inversion*, the FBF approach approximates $u(k)$ in **Figure 3** as:

$$u(k) = \sum_{i=0}^n \gamma_i \varphi_i(k) \quad (1)$$

where $\{\varphi_i(k)\}$ 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. (1), the control input and output trajectory can be expressed as

$$y(k) = Gu(k) = \sum_{i=0}^n \gamma_i \tilde{\varphi}_i(k), \quad (2)$$

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

In Eq. (2), $\tilde{\varphi}_i$ represents the filtered basis functions, i.e., *forward filtered* by the system G (or its accurate model). For perfect tracking (i.e., $y(k) = y_d(k)$ for all k), the coefficients $\{\gamma_i\}$ should satisfy:

$$\underbrace{\begin{bmatrix} \tilde{\varphi}_0(0) & \tilde{\varphi}_1(0) & \dots & \tilde{\varphi}_n(0) \\ \tilde{\varphi}_0(1) & \tilde{\varphi}_1(1) & \dots & \tilde{\varphi}_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\varphi}_0(M) & \tilde{\varphi}_1(M) & \dots & \tilde{\varphi}_n(M) \end{bmatrix}}_{\Phi} \underbrace{\begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_n \end{bmatrix}}_{\gamma} = \underbrace{\begin{bmatrix} y_d(0) \\ y_d(1) \\ \vdots \\ y_d(M) \end{bmatrix}}_{y_d} \quad (3)$$

S2: BEYOND RIGID BODY FF CONTROL OF DESKTOP 3D PRINTERS

There is a push to make desktop 3D printers as widely available as desktop office printers, in a bid to democratize manufacturing. To achieve this goal, desktop 3D printers must be low-cost, lightweight, and have reasonable print speeds and print quality (see **Figure 2a**). To keep costs low, desktop 3D printers are typically controlled 100% in open loop (i.e., FF) using stepper motors. However, given their lightweight structures, they have structural flexibilities, which cause them to vibrate excessively due to the motion of their print head and/or platform. Excessive vibration leads to poor surface quality or, even worse, causes the stepper motors to skip counts leading to scrapped parts (see **Figure 2b**). These issues are currently dealt with by reducing the speed of the printers, which significantly slows down the printing process (already notorious for its low throughput). Without unduly sacrificing throughput, the vibration-induced problems of desktop 3D printers could be resolved using FB control, but this would require servo motors, sensors, and high-performance drives to run active vibration control algorithms, all adding significant cost to the 3D printers. Moreover, to be effective, such sensors would have to be placed close to the print head where they can properly observe the unwanted vibration; this may not be practical. A more practical alternative, which can be implemented via stepper motors without extra hardware investments, is MIBFF control. However, much like wafer scanners, NMP zeros, and position-varying dynamics, are also issues in BRB FF control of 3D printers that must be addressed.

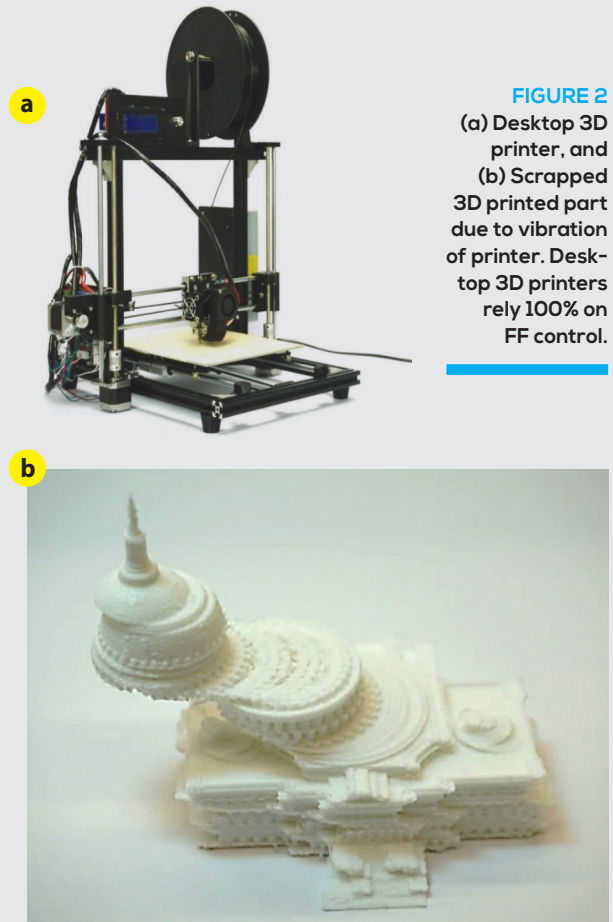


FIGURE 2
(a) Desktop 3D printer, and (b) Scrapped 3D printed part due to vibration of printer. Desktop 3D printers rely 100% on FF control.

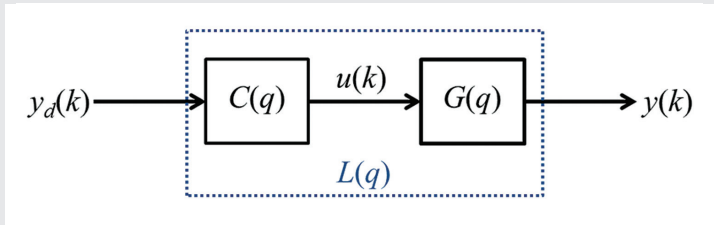


FIGURE 3 Block diagram for tracking control.

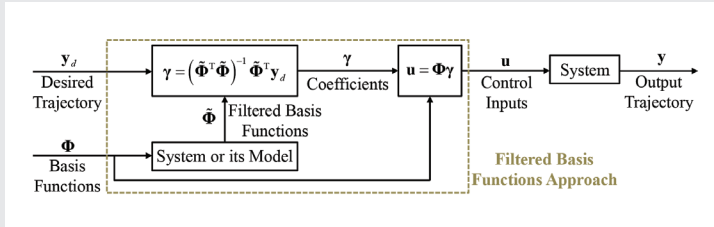


FIGURE 4 Flowchart of FBF approach for FF tracking of NMP systems.

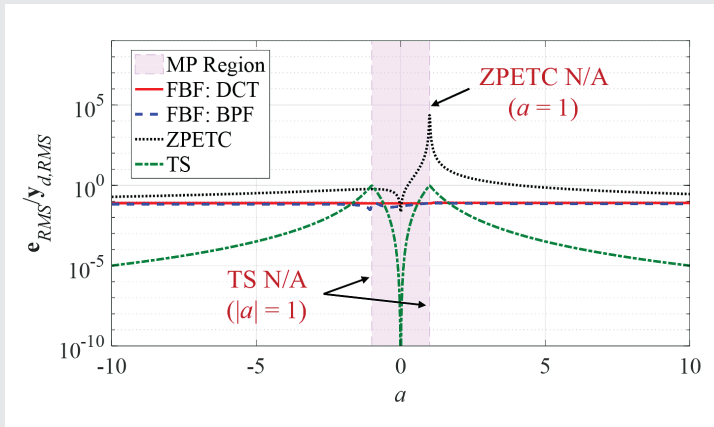


FIGURE 5 Normalized tracking error over different zero locations for FBF-BPF, FBF-DCT, ZPETC and TS. Notice that both cases of FBF maintain consistent tracking accuracy irrespective of zero location, while the accuracies of ZPETC and TS vary significantly. Note also that FBF is applicable to all zero locations, while ZPETC and TS are not applicable at $a=1$ and $|a|=1$, respectively.

An exact solution for Eq. (3) exists if and only if the rank of $\tilde{\Phi}$ is equal to the rank of $[\tilde{\Phi} | \mathbf{y}_d]$, i.e., if \mathbf{y}_d belongs to the span of the filtered basis functions $\{\tilde{\varphi}_i\}$. However, for the general situation where $n < M$, Eq. (3) is overdetermined, and the solution to Eq. (3) is approximated as [6][7]:

$$\mathbf{y} = (\tilde{\Phi}^T \tilde{\Phi})^{-1} \tilde{\Phi}^T \mathbf{y}_d \quad (4)$$

This approximation is the least square solution minimizing $\|\mathbf{e}\|_2^2$, 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. (4) exists and is unique, if and only if the filtered basis functions $\{\tilde{\varphi}_i\}$ are linearly-independent (i.e., if $\tilde{\Phi}$ is of full rank $n+1$). It is shown in [7] that this linear-independence condition can be readily achieved, given the flexibility in selecting the basis functions and their initial conditions in filtering.

Figure 4 depicts the FBF method's generation of control input $\mathbf{u} = [u(0) u(1) \dots u(M)]^T$ from \mathbf{y}_d and BFs (Φ), as well as its application to system $G(q)$ to output $\mathbf{y} = [y(0) y(1) \dots y(M)]^T$. Following the tracking control block diagram of Figure 3, the overall dynamics L and tracking controller C have functionally-equivalent counterparts in the FBF framework, given by the matrices \mathbf{L}_{FBF} and \mathbf{C}_{FBF} :

$$\mathbf{y} = \underbrace{\tilde{\Phi}(\tilde{\Phi}^T \tilde{\Phi})^{-1} \tilde{\Phi}^T}_{\mathbf{L}_{FBF}} \mathbf{y}_d, \quad \mathbf{u} = \underbrace{\Phi(\Phi^T \Phi)^{-1} \Phi^T}_{\mathbf{C}_{FBF}} \mathbf{y}_d \quad (5)$$

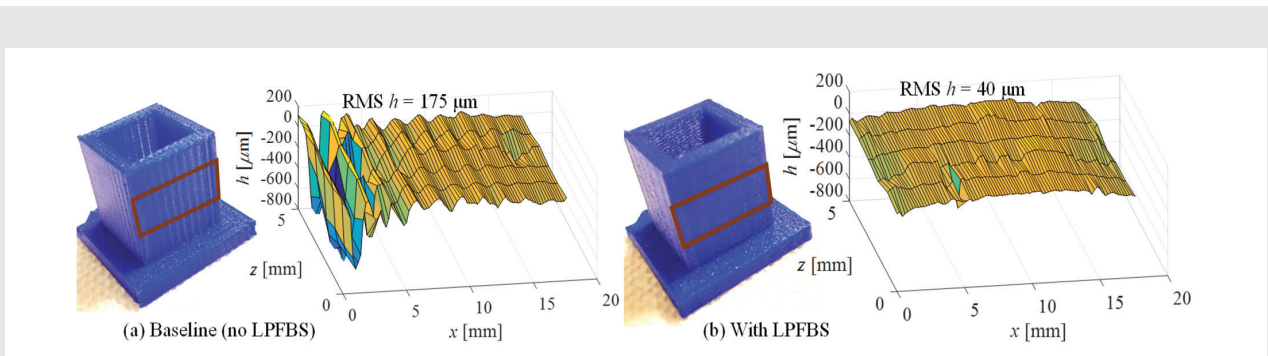


FIGURE 6 Comparison of photographs and measured surface profiles (h) of the highlighted surfaces of blocks printed using (a) baseline approach (no vibration compensation) and (b) LPFBS method. LPFBS stands for limited-preview filtered B-splines. It is a low-computational-cost implementation of the FBF method using B-splines as basis functions.

Note that the $L(q)$ and $C(q)$ of any linear MIBFF method can be expressed as matrices \mathbf{L} and \mathbf{C} , respectively, via the so-called lifted system representation [7]. It is interesting to note that \mathbf{L}_{FBF} and \mathbf{C}_{FBF} are not generally Toeplitz matrices [7] (i.e., matrices having constant entries along their diagonals). *This constitutes a fundamental difference* between the FBF method and most other MIBFF methods, whose \mathbf{L} and \mathbf{C} matrices are *always* Toeplitz. The implication is that while most other methods result in LTI controllers, *the FBF method generally yields an LTV controller*. It is this fundamental difference that enables the improvements over other MIBFF methods, in terms of consistent performance regardless of NMP zero location, that is evident from the comparisons in the next section. Note also that, though discussed in the context of LTI SISO systems, *the FBF approach is applicable to any discrete-time linear system – including multi-input, multi-output and linear time-varying systems* (see, e.g., [11])

COMPARISON OF FBF TO ZPETC AND TS

This section evaluates the FBF method in comparison with the ZPETC and TS methods using the first-order NMP system studied by Butterworth et al. [8]; it is defined as:

$$G(q) = \frac{q-a}{q-p} \quad (6)$$

where a and $p = 0.5$ are the zero and pole of the system, respectively. TS is selected for the comparison because, among the advanced techniques for approximate inversion, it is very versatile with regards to its applicability irrespective of the location of the NMP zeros in the z -plane; ZPETC is considered because of its popularity. There is a wide range of basis functions available for use with the FBF method; e.g., Laguerre functions, wavelets, B-splines, etc. In the simulation studies, two basic basis functions – the block pulse function (BPF) and the discrete cosine transform (DCT) – are adopted [7]. The desired trajectory (y_d) used for the simulations is a zero-mean white noise signal with unity variance, $M = 1000$

ABOUT THE AUTHORS



Keval S. Ramani is a Ph.D. candidate in Mechanical Engineering at University of Michigan since 2015. He received his M.S. from University of Michigan

in 2015 and B.E. from BITS Pilani, Goa Campus, India, in 2012. From 2012 to 2013, he worked with Ingersoll Rand, based in Bengaluru, India, as an Engineering Trainee. His research interests are in controls, dynamics, vibration and optimization. He is a co-author of the best student paper award-winning paper at the 2015 Dynamic Systems and Controls Conference.

Molong Duan is a Ph.D. candidate in Mechanical Engineering at the University of Michigan, Ann Arbor. He received his B.S. degree from Peking University, Beijing, China, and his M.S.E. degree from the University of Michigan, in 2012 and 2013, respectively. His research focuses on over-actuated systems, sustainable manufacturing, and trajectory generation. He was awarded the Rackham Centennial Fellowship from the University of Michigan in 2013, the best poster award at the 2014 International Forum on Sustainable Manufacturing, and the best student paper award at the 2015 Dynamic Systems and Controls Conference.



Deokkyun Yoon received his B.S. and M.S. degrees in 2009 and 2010 and is working toward the Ph.D. degree, all in mechanical engineering from the University of Michigan. Prior to returning to continue working toward the Ph.D. degree, he was with Korea Institute of Machinery and Materials as a research staff member from 2010 to 2013. His research interests include mechatronics system design for improved precision, throughput, and power efficiency, applied to advanced manufacturing processes. He received the best poster award at the 2014 International Forum on Sustainable Manufacturing, and was awarded the inaugural 3M scholarship from the American Society for Precision Engineering.














Chinedum Okwudire received his Ph.D. degree in Mechanical Engineering from the University of British Columbia in 2009 and joined the Mechanical Engineering faculty at the University of Michigan in 2011. Prior to joining Michigan, he was the mechatronic systems optimization team leader at DMG Mori USA based in Davis, CA. His expertise lies in smart and sustainable automation, where he leverages the fundamental



engineering disciplines of machine design, structural dynamics, and control theory to tackle challenging problems in precision, throughput, and energy efficiency faced by the manufacturing and vehicle automation industries. He has received the International Symposium on Flexible Automation Young Investigator Award; the Society of Manufacturing Engineers Outstanding Young Manufacturing Engineer Award; the SAE International Ralph Teeter Educational Award; the National Science Foundation CAREER award; and a number of best paper awards.

A. Galip Ulsoy is C.D. Mote, Jr. Distinguished University Professor Emeritus of Mechanical Engineering and W.C. Ford Professor Emeritus of Manufacturing at University of Michigan, Ann Arbor. He received the Ph.D. from University of California at Berkeley (1979), M.S. from Cornell University (1975), and B.S. from Swarthmore College (1973). His research interests are in dynamics and control of mechanical systems. He has received numerous awards, including the 2008 Rufus T. Oldenburger Medal and the 2013 Charles Russ Richards Award from ASME, is a member of the National Academy of Engineering and is a Fellow of ASME, SME, IEEE and IFAC.



Acceleration Limit	1 m/s ²	3 m/s ²	5 m/s ²	7 m/s ²	10 m/s ²	30 m/s ²
Printing Time	3:59 h	2:41 h	2:21h	2:12 h	2:06 h	1:50 h
Baseline (No LPFBS)						N/A
LPFBS						

← 47% (1:53 h) time saving using conservative accelerations for both Baseline and LPFBS →
 ← > 31% (0:51 h) time saving using aggressive accelerations for both Baseline and LPFBS (It is unclear at which acceleration LPFBS fails) →

TABLE 1 3D printed models of US Capitol using different acceleration limits (which influences total printing time). The LPFBS method eliminates the vibration-induced distortions observed in the baseline case at higher acceleration limits; the feedrate is 60 mm/s for all cases. LPFBS stands for limited-preview filtered B-splines. It is a low-computational-cost implementation of the FBF method using B-splines as basis functions.

and sampling time, $T_s = 0.1$ ms.

Figure 5 compares the normalized tracking error norm ($\mathbf{e}_{RMS}/\mathbf{y}_{d,RMS}$) of the methods under consideration for $a \in [-10, 10]$. For the two cases of FBF, $n = 990$ is selected, while for TS, $n_1 = 5$ is chosen, where n_1 is the number of terms in the truncated series. Notice that, unlike ZPETC and TS, both cases of FBF exhibit very consistent tracking performance for all values of a investigated – including at $|a| = 1$ where TS (and most other methods) fail because the system has a zero on the unit circle. Note that, at any given value of a , the tracking accuracy of FBF can be matched to that of TS via the selection of n . Hence, the relative accuracy of the FBF approach is not of particular interest.

Through this simulation, we have demonstrated some of the interesting observations made in [7] with regard to the FBF method's effectiveness and consistency in tracking arbitrary desired trajectories, with or without feedback, regardless of the location of NMP zeros in the z -plane. Analytical justifications for the FBF method's consistent tracking performance relative other techniques are presented in [12]. In the next section we present experimental results obtained for a low-cost desktop

3D printer with stepper motors.

EXPERIMENTAL VALIDATION: VIBRATION COMPENSATION OF A DESKTOP 3D PRINTER

The FBF method has been validated in experiments reported in [13] using the commercially-available HICTOP Prusa i3 3D printer shown in **Figure 2a**. It is utilized for FF compensation of tracking errors induced by vibrations of the printer as a result of motions of its build platform and nozzle in the x and y directions, respectively. The axis frequency response functions (FRFs) are measured by applying swept sine acceleration signals (with amplitudes ranging from 1.5 m/s² to 3.0 m/s²) to the printer's stepper motors and measuring the relative acceleration of the build platform and print head using accelerometers. The curve fit models are generated using MATLAB®'s *invfreqs* function applied to the 3.0 m/s² data, in order to better compensate for vibrations induced by more aggressive motions. The resulting discrete-time transfer function for the x -axis is

$$G(q) = \frac{0.026q^4 - 0.048q^3 - 0.003q^2 + 0.048q - 0.023}{q^5 - 4.792q^4 + 9.274q^3 - 9.060q^2 + 4.466q - 0.889} \quad (7)$$

with sampling time, $T_s = 1$ ms. Note that $G(q)$ has a zero at $q = -1$; most MIBFF control methods cannot handle such systems.

The FBF method is implemented with B-splines as the basis functions. The local property of B-splines is exploited to implement the FBF method with limited preview (i.e., in small batches) to drastically reduce computational costs [13]. **Figure 6** compares the roughness (h) of the highlighted surfaces of two identical blocks, both printed at 60 mm/s feedrate and 7 m/s² maximum acceleration. Notice that the RMS value of h for the block printed with FBF is 77% smaller than that of the baseline case without FBF. **Table 1** shows scale models of the US Capitol printed at 60 mm/s feedrate with various acceleration limits. The FBF method allows much higher accelerations (i.e., by a factor of ten), thereby shortening print time significantly

(i.e., by a factor of two) without sacrificing print quality. Full details of the limited-preview implementation of the FBF approach and the experimental investigation can be found in [13].

CONCLUDING REMARKS

FF controllers play an important role in many engineering applications, but have not received as much research attention as FB controllers. A major issue in FF controller design arises when system models with uncancellable (e.g., NMP) zeros are inverted. We have presented the filtered basis functions (FBF) approach for MIBFF tracking control and compared it to well-known methods such as ZPETC and TS. The proposed FBF method uses basis functions to represent the control signal and determines the coefficients of those basis functions to minimize the error between the desired and actual trajectories in a least-square sense. The FBF design can be applied to any linear system and yields a LTV controller even for a LTI system.

For a first-order LTI plant with a NMP zero, the simulation results show that the FBF approach, unlike ZPETC and TS, can be used and yields consistent tracking performance regardless of NMP zero location. We have also validated the FBF approach experimentally for tracking control of a desktop 3D printer with stepper motors. The results show that the



PHOTO COURTESY OF EVAN DOUGHERTY, MICHIGAN ENGINEERING

FIGURE 7 Molong Duan and Deokkyun Yoon, both mechanical engineering PhD students, and Chinedum Okwudire, associate professor in mechanical engineering, examine the effectiveness of the limited-preview filtered B-spline algorithm on a desktop 3D printer.

print time can be cut in half using an FBF FF tracking controller, while other methods may not be applicable due to an NMP zero on the unit circle.

The main conclusions are:

- The proposed FBF approach, which yields a LTV controller, is applicable to any linear system, and yields consistent tracking accuracy irrespective of NMP zero location.
- FBF stands out in situations where other methods struggle or fail (e.g., zeros on or near the unit circle).

We are considering various extensions of the current research to more general LTV tracking controllers, to multi-input multi-output systems, nonlinear systems, and to combined FF and FB control. ■

REFERENCES

1. Lunenburg, J., Bosgra, O., & Oomen, I. T. (2009). Inversion-based feedforward design for beyond rigid body systems: a literature survey. DCT Report, Eindhoven University of Technology.
2. Bruijnen, D., & van Dijk, N. (2012, June). Combined input shaping and feedforward control for flexible motion systems. In *2012 American Control Conference (ACC)*. IEEE, pp. 2473-2478.
3. Clayton, G. M., Tien, S., Leang, K. K., Zou, Q., & Devasia, S. (2009). A review of feedforward control approaches in nanopositioning for high-speed SPM. *Journal of dynamic systems, measurement, and control*, 131(6), 061101.
4. Rigney, B. P., Pao, L. Y., & Lawrence, D. A. (2009). Nonminimum phase dynamic inversion for settle time applications. *IEEE Transactions on Control Systems Technology*, 17(5), pp. 989-1005.
5. van Zundert, J., & Oomen, T. (2018). On inversion-based approaches for feedforward and ILC. *Mechatronics*, 50, pp. 282-291.
6. Duan, M., Ramani, K. S., & Okwudire, C. E. (2015, October). Tracking control of non-minimum phase systems using filtered basis functions: a NURBS-based approach. In *ASME 2015 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers, pp. V001T03A006-V001T03A006.
7. Ramani, K. S., Duan, M., Okwudire, C. E., & Ulsoy, A. G. (2017). Tracking control of linear time-invariant nonminimum phase systems using filtered basis functions. *Journal of Dynamic Systems, Measurement, and Control*, 139(1), 011001.
8. Butterworth, J. A., Pao, L. Y., & Abramovitch, D. Y. (2008, June). The effect of nonminimum-phase zero locations on the performance of feedforward model-inverse control techniques in discrete-time systems. In *2008 American control conference*, IEEE, pp. 2696-2702.
9. Frueh, J. A., & Phan, M. Q. (2000). Linear quadratic optimal learning control (LQL). *International Journal of Control*, 73(10), pp.832-839.
10. Jetto, L., Orsini, V., & Romagnoli, R. (2015). Spline based pseudo-inversion of sampled data non-minimum phase systems for an almost exact output tracking. *Asian Journal of Control*, 17(5), pp.1866-1879.
11. Kasemsinsup, Y., Romagnoli, R., Heertjes, M., Weiland, S., & Butler, H. (2017). Reference-tracking feedforward control design for linear dynamical systems through signal decomposition. In *American Control Conference (ACC)*, 2017, IEEE, pp. 2387-2392.
12. Ramani, K. S., Duan, M., Okwudire, C. E., & Ulsoy, A. G. (2018, July). A lifted domain-based metric for performance evaluation of LTI and LTV discrete-time tracking controllers. In *2018 International Symposium on Flexible Automation (ISFA 2018)*, July 15-19, 2018, Kanazawa, Japan.
13. Duan, M., Yoon, D. and Okwudire, C.E. (2017). A limited-preview filtered B-spline approach to tracking control—With application to vibration-induced error compensation of a 3D printer. *Mechatronics*.