Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Architectures for Tracking Control in Atomic Force Microscopes ⋆

Jeffrey A. Butterworth * **Lucy Y. Pao** *
Daniel Y. Abramovitch **

* *University of Colorado at Boulder, Boulder, CO 80309 USA (Tel: 303-492-4083; e-mail: butterwo@colorado.edu, pao@colorado.edu)*
** *Agilent Laboratories Nanotechnology Group, 5301 Stevens Creek Blvd., M/S: 4U-SB, Santa Clara, CA 95051 USA (e-mail: danny@agilent.com)*

Abstract: We evaluate the performance of two control architectures applied to atomic force microscopes (AFM). Feedback-only control is a natural solution and has been applied widely. Expanding on that, combining feedback controllers with plant-injection feedforward filters has been shown to greatly improve tracking performance in AFMs. Alternatively, performance can also be improved by the use of a closed-loop-injection feedforward filter applied to the reference input before it enters the feedback loop. In this paper, we compare the plant-injection architecture with the closed-loop-injection architecture when used in controlling AFMs. In particular, we find that even in the presence of plant uncertainty, the closed-loop-injection architecture yields better tracking performance of a raster scan.

Keywords: microsystems: nano-technologies; atomic force microscopes; feedforward control

1. INTRODUCTION

An atomic force microscope (AFM) can provide images with resolution at the atomic scale (10^{-10} m). In terms of resolution, cost, imaging environments allowable, and ease of sample preparation, AFMs have advantages over other micro- and nanoscale imaging instruments such as tunneling electron microscopes (TEMs), scanning electron microscopes (SEMs), and optical imaging devices. However, the quality and speed of AFM images depend upon the overall dynamics of the AFM system. The behavior of AFMs varies considerably across AFM tips as well as changes in samples and environmental changes. Currently, the variability causes commercial AFMs to not behave like reliable instruments, and this slows down and frustrates AFM users. Since the time required to attain a quality AFM image is typically on the order of several minutes or more, substantial motivation exists to reduce the imaging time in AFMs. Faster imaging is required to capture and explore the dynamics of biological samples (El Feninat et al. [2001], Shao et al. [1996]) and improved speed is also necessary for nanofabrication to be economically viable.

In this paper, we discuss two combined feedforward/feedback control architectures for AFMs, the *plant-injection* architecture and the *closed-loop-injection* architecture. This is an expansion of our previous tutorial paper (Pao et al. [2007]) which limited its feedforward discussion to only the plant-injection architecture. Further background on AFMs can be found in Abramovitch et al. [2007] and the references therein.

A schematic diagram of a *scanning-sample-design* AFM is shown in Fig. 1. The AFM operation process is initiated by gradually reducing the distance between the AFM probe and the sample (by using a piezo actuator) until a prespecified probe-sample interaction is achieved, i.e., the AFM cantilever deflection error reaches a specified setpoint value. The AFM cantilever deflection error (which depends on the tip-sample interaction) can be measured using an optical sensor as shown in Fig. 1. The x-direction motion is the fast scan direction, while motion in the y direction is much slower. The z-direction motion control

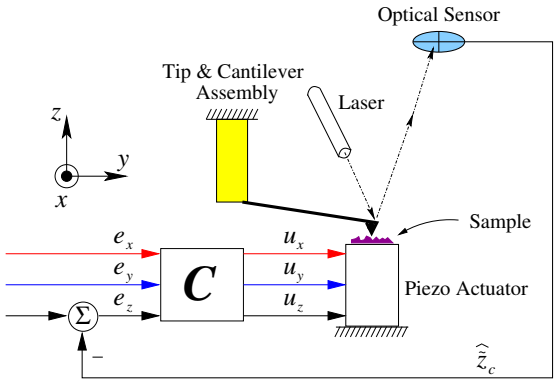


Fig. 1. Schematic diagram of a *scanning-sample-design* AFM. Here, the piezoscanner enables positioning of the sample both parallel (along the x and y axes) and perpendicular (along the z axis) to the AFM cantilever tip.

⋆ This work was supported in part by Agilent Technologies, Inc. and the US National Science Foundation (NSF Grant CMMI-0700877). The authors thank Brian P. Rigney for useful discussions on control architectures and model-inverse control methods.

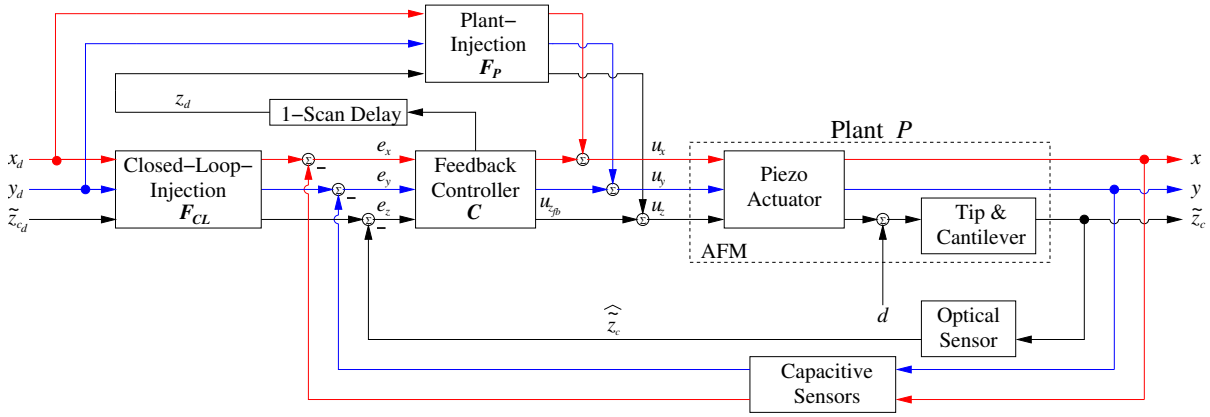


Fig. 2. A complete multi-input multi-output (MIMO) AFM block diagram consisting of a feedback compensator C, a plant-injection feedforward controller F_P, and a closed-loop-injection feedforward controller F_{CL}. When using the closed-loop-injection architecture, F_P is set to zero. In contrast, when using the plant-injection architecture, F_{CL} is set to the identity (or a delay in the case of model-inverse control when the relative degree of the plant is greater than zero).

is carried out depending upon the amount of cantilever deflection, which depends upon the topology of the sample.

In Fig. 2, the dynamics of the AFM piezo actuator and tip and cantilever are represented in the AFM block. The piezo actuators in the x, y, and z directions move the piezo scanner in these directions. d is the surface of the sample being imaged, and since it is yet unknown, d acts as a disturbance to the AFM. The sample surface causes the cantilever to deflect, the deflection error $\tilde{z}_c$ is measured by an optical sensor, and the result is fed back to the controller C.

There are several variations in AFM designs in terms of whether the tip and cantilever assembly are actuated and/or the sample stage is actuated. In one typical scanning-sample-design AFM (Fig. 1), the sample is moved below a stationary tip, and the x, y, and z actuation are done by a single piezo tube actuator (Binning and Smith [1986]). In a scanning-tip-design AFM, the sample is stationary while the tip is moved in x, y, and z. In these two designs, the x and y directions are *not always* measured and as a result are often only controlled in a feedforward manner. The optical sensor provides a measurement for the z direction such that feedback control can be applied. In a third AFM design, the x-y motion is driven by a stage that moves the sample while the z motion is driven by a separate actuator moving the cantilever up and down. Often this design uses capacitive sensors or a linearly variable differential transformer (LVDT) to measure the x-y motion allowing feedback control for all three directions (Salapaka et al. [2002], Sebastian and Salapaka [2005]). While this third design will be the focus of this paper, all the control techniques discussed here can be applied to each AFM design and all directions of each.

Several AFM imaging modes exist. In *constant-force mode*, the control goal is to regulate the AFM cantilever deflection error $\tilde{z}_c$ at a constant value (the setpoint value $\tilde{z}_{c_d}$, which is often zero). Large variations in the AFM cantilever deflection error $\tilde{z}_c$ can cause sample or AFM-probe damage. Variations in the setpoint value of the cantilever deflection error $\tilde{z}_{c_d}$ may be required, however, to manipulate or modify a sample, e.g., to indent a sample during nanofabrication. This paper generally assumes constant-force mode, as opposed to dynamic or AC mode operation of AFMs (Abramovitch et al. [2007], Kodera et al. [2005, 2006], Sebastian et al. [2007]). Regardless, the control methods discussed here are applicable to both constant-force and AC modes.

In this paper, we compare two combined feedforward-feedback control architectures. Naturally, feedback-only control has been an obvious control solution to the task of improving AFM performance, and for years proportional-integral-derivative control has been the industry standard for AFM feedback control. Referencing Fig. 2, we note that when using feedback-only control, F_P is set to zero and F_{CL} is set to the identity. Additionally, several research groups have investigated improving upon the performance of feedback-only control by combining it with a feedforward filter F_P. For the sake of discussion, we have named this technique the *plant-injection* architecture and we will also often refer to it as the FFPI (feedforward plant-injection) architecture. When using the plant-injection architecture, F_{CL} is set depending on the type of controller used for F_P. Most often F_{CL} is the identity in the FFPI architecture, but occasionally it is a delay function when F_P is a model-inverse controller. The details of this delay will be discussed further in Section 4. Some examples of applying the plant-injection architecture to AFMs in the literature include Croft and Devasia [1999], Croft et al. [2001], Schitter et al. [2004a,b], and Tien et al. [2005].

In addition to discussing the plant-injection architecture, this paper also employs the *closed-loop-injection* architecture (also referred to as FFCLI in this paper) for AFM control. While this control architecture has been used in other mechatronic systems (Gross et al. [1994], Haack and Tomizuka [1991], Potsaid and Wen [2004], Potsaid et al. [2007], Rigney et al. [2006a,b], Tomizuka [1987], Torfs et al. [1992, 1998]), to the best of the authors' knowledge, it has only very recently been applied specifically to AFMs. Previous versions of an FFCLI-like architecture in AFM control have included Bhikkaji et al. [2007], Leang and Devasia [2007], Li and Bechhoefer [2007], and Schitter et al. [2006]. Both Bhikkaji et al. [2007] and Schitter et al.

[2006] use FFCLI in an input-shaping manner. Leang and Devasia [2007] discusses FFCLI in an abstract way with limited details, while Li and Bechhoefer [2007] discusses this architecture in a manner more consistent with our discussion, but is limited in the discussion of the design of feedforward controllers. Here we extend the use of F_{CL} beyond the above to more general control techniques and compare it to using F_P exclusively. When using FFCLI, F_P in Fig. 2 is set to zero and the feedforward filter F_{CL} acts on the reference signal (the desired scan trajectory in the case of x and y motions) ahead of the closed-loop system.

This paper is organized as follows. In Section 2, we review some basic ideas on AFM control. We then discuss some feedback-only controller design methods in Section 3. Some control methods for the plant-injection architecture (that have been previously applied to AFMs) are presented in Section 4, and methods for the closed-loop-injection architecture are investigated in Section 5. We compare the methods for the various control architectures in Section 6. In Subsection 6.5, we present results of these comparisons as applied to the x-axis control loop, and show that in simulation the closed-loop-injection outperforms the plant-injection architecture. A model of a nPoint (`www.npoint.com`) NPXY100A stage is used, where the model is extracted based on measurements of an actual stage. We then draw conclusions and discuss future work for further improving the control of AFMs in Section 7.

2. A GENERAL DISCUSSION ON AFM CONTROL

Fig. 2 shows a multi-input multi-output (MIMO) arrangement of the AFM. Assuming a linear, time-invariant (LTI) plant model, as has often been done in the literature, the AFM plant is MIMO of the form

$$P(z) = \begin{bmatrix} P_{xx}(z) & P_{xy}(z) & P_{xz}(z) \\ P_{yx}(z) & P_{yy}(z) & P_{yz}(z) \\ P_{zx}(z) & P_{zy}(z) & P_{zz}(z) \end{bmatrix} \quad (1)$$

where $P_{xx}(z)$ represents the transfer function from the x-axis control input to the x position, $P_{xy}(z)$ is the transfer function from the y-axis control input to the x position, $P_{xz}(z)$ is the transfer function from the z-axis control input to the x position, and so forth. Due to coupling effects, $P(z)$ is a full matrix. P_{xy}, P_{yx}, P_{yz}, and P_{zy} are generally relatively small compared with the other entries (Sebastian and Salapaka [2005], Tien et al. [2005]). The cross-coupling between the x and z directions, however, can be significant (Tien et al. [2005]).

The severity of the cross-coupling in AFMs depends upon whether a tube actuator or a separate x-y scanner is used. With a tube actuator, there is coupling due to the structure with piezos on the outside for the x and y motion and a piezo on the inside for the z motion (Abramovitch et al. [2007]). When there is a separate x-y scanner, the coupling from the lateral motion into the z axis is less pronounced. Furthermore, modern external x-y scanners are designed to specifically decouple the motion of their fast and slow axes by having the x scanner (fast direction) mounted within a frame that is moved in y (slow direction) by the y scanner (Ando et al. [2001], Schitter et al. [2006], Sebastian and Salapaka [2005]). Our AFM uses a separate x-y scanner, so making an assumption that the coupling effects are minimal is not without merit, and as a result (and for the purposes of the comparative nature of this paper) we focus on a single-input single-output (SISO) arrangement of the control system with a specific look at the x direction of a scanner. Although the following discussion focuses on the x direction of the AFM, the extension to the y direction comes naturally. The connection to the z direction, however, is not immediately obvious due to the unknown surface being imaged. By using data from the previous scan line via a *1-Scan Delay* (shown in Fig. 2), and assuming the sample topography does not change dramatically from one scan line to the next, improved z direction control can be achieved (Abramovitch et al. [2007]). How the FFCLI architecture can be applied to improve z direction control is an area of future work.

3. AFM FEEDBACK-ONLY CONTROL

In the case of feedback-only control, F_P equals zero and F_{CL} in Fig. 2 is the identity. It should be clear that the two control architectures reduce to the same structure when using feedback-only control. Two major types of feedback-only control applied to AFMs are Proportional-Integral-Derivative (PID) control and H_∞ control and they are discussed in some detail in the following subsections.

3.1 Proportional-Integral-Derivative Control

Commercial AFMs are typically controlled with basic feedback-only Proportional (P), Proportional-Integral (PI), Proportional-Integral-Derivative (PID), Proportional-Double-Integral (PII), or Proportional-Double-Integral-Derivative (PIID) compensators (El Rifai and Youcef-Toumi [2003], Salapaka et al. [2002], Schitter et al. [2001], Sebastian and Salapaka [2005], Stemmer et al. [2005]). A SISO continuous-time compensator transfer function for a PIID feedback controller is

$$C(s) = K_p + \frac{K_i}{s} + \frac{K_{ii}}{s^2} + K_d s.$$

For a P, PI, PII, or PID controller, one or more of the K_d, K_i, or K_{ii} gains are set to zero. PIID controllers are typically specified in continuous time, s, and are usually implemented in discrete time, z, typically using an integrator equivalent (Åström and Hägglund [2005]). However, analog implementations are used in some high bandwidth experiments (Ando et al. [2001], Schitter et al. [2006]). The K_p, K_i, K_{ii}, and K_d gains must be tuned carefully to achieve high-bandwidth and good regulation of the cantilever deflection error in the z direction or tracking performance in the x and y directions. Users of commercial AFMs know all too well that the tuning of the PIID gains is a tedious process, and several control systems researchers have recently shown significant improvements in the speed and quality of AFM images using more advanced controllers discussed in the following subsection and in Section 4.

3.2 H_∞ Feedback Control

H_∞ feedback control has been applied to AFMs in work by Salapaka et al. [2002, 2005], Schitter et al. [2001], Sebastian and Salapaka [2005], and others. Just as with any

other system, the H_∞ feedback controller C is designed to minimize the H_∞ norm of

$$T_{fb} = \left\| \begin{array}{c} W_e S \\ W_u C S \\ W T \end{array} \right\|_\infty . \tag{2}$$

In (2), S and T are the sensitivity and complimentary sensitivity functions, respectively,

$$S = (1 + \hat{P}C)^{-1}, \tag{3}$$
$$T = \hat{P}C(1 + \hat{P}C)^{-1}. \tag{4}$$

Here $\hat{P}$ is the plant model used for control design, which due to modeling errors is likely not exactly equal to the actual plant P. We will use the "hat" notation ($\wedge$) throughout this paper to indicate the model or portions of the model. W_e, W_u, and W in (2) are weighting functions that act on the error signal e, the plant input u, and the output y of the closed-loop system, respectively. The choice of the weights has a large effect on disturbance rejection, robustness to modeling errors, and overall performance (Skogestad and Postlethwaite [2005]). In general, W_e has low-pass characteristics as it penalizes the sensitivity function and is critical to low-frequency tracking performance. W_u is often a constant to avoid saturating the actuators, or in the presence of uncertainty near a resonance it can be designed to weight that resonance to avoid exciting the plant near this uncertainty. Since it acts on the complimentary sensitivity function, W has high-pass qualities to improve disturbance rejection and robustness to high-frequency modeling errors (Schitter et al. [2003], Sebastian and Salapaka [2005]).

4. PLANT-INJECTION ARCHITECTURE

Recent work in AFMs has shown increases in AFM performance when feedback controllers (such as those discussed in Section 3) are combined with a feedforward controller in the plant-injection architecture (Croft and Devasia [1999], Croft et al. [2001], Schitter et al. [2004a,b], Zou and Devasia [2004]). In general, this architecture ensures stability through the feedback controller C while the feedforward controller F_P increases tracking performance, disturbance rejection, and robustness to model uncertainties. Model-inverse and H_∞ are two major types of feedforward control for AFMs that have appeared in the literature and are highlighted in the following subsections. Rieber et al. [2005] and Stemmer et al. [2005] have also applied ℓ_1 optimal control to AFMs. Due to the fact that there are no readily available software packages that enable ℓ_1 controllers to be solved conveniently, we focus this discussion on model-inverse and H_∞ controllers.

For our system, we simplify Fig. 2 into the SISO version in Fig. 3. Specifically, Fig. 3 is representative of the block diagram used for simulations, hence the inclusion of a saturation block (which can be ignored for this portion of the discussion, but will be addressed again in Section 6.5).

Similar to work in Rigney et al. [2006a,b, 2008], we examine the transfer function from the desired input x_d to the output x of the FFPI system of Fig. 3. Ignoring the

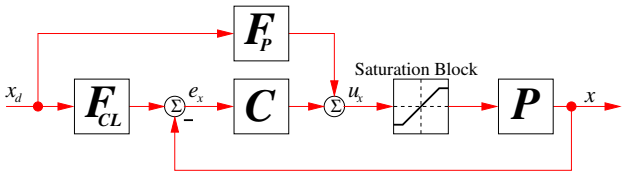


Fig. 3. The SISO x-direction block diagram used for simulations. The ±10 volts saturation block is representative of limitations of our actual system.

saturation block, and assuming F_{CL} is unity, the transfer function is

$$\left.\frac{X(z)}{X_d(z)}\right|_{FFPI} = \frac{P(z)F_P(z) + P(z)C(z)}{1 + P(z)C(z)}. \tag{5}$$

Here it should be clear that if the feedforward filter $F_P(z) = 0$, then (5) reduces to the common expression for the dynamics of a feedback-only closed-loop system,

$$H_{CL}(z) = \frac{P(z)C(z)}{1 + P(z)C(z)}. \tag{6}$$

When $F_P(z)$ is equal to the inverse of $P(z)$, then (5) becomes the identity and we are able to track any desired input perfectly. Of course, this assumes a perfect inverse of $P(z)$ exists and is practically implementable, and we will discuss this further in Subsection 4.1.

For convenience we drop the argument z, and define the plant dynamics P, the feedback controller C, and the feedforward controller F_P as a ratio of polynomials as in

$$P = \frac{B}{A}, \quad C = \frac{C_N}{C_D}, \quad \text{and} \quad F_P = \frac{F_{PN}}{F_{PD}}. \tag{7}$$

Here, the subscripts N and D indicate numerator or denominator polynomials. A and B represent polynomials defining the plant poles and zeros, respectively. Using the definitions in (7) we can further reduce (6) and (5) to

$$H_{CL} = \frac{BC_N}{AC_D + BC_N} \quad \text{and} \tag{8}$$
$$\left.\frac{X(z)}{X_d(z)}\right|_{FFPI} = \frac{BF_{PN}C_D + BC_NF_{PD}}{F_{PD}(AC_D + BC_N)}. \tag{9}$$

The value of (9) will become clear when we review a model-inverse control technique in Subsection 4.1 and discuss the closed-loop-injection architecture in detail in Section 5.

4.1 FFPI: Feedforward Model-Inverse Control

Devasia and others have studied model-inverse methods in the plant-injection architecture for AFMs (Croft and Devasia [1999], Croft et al. [2001], Tien et al. [2005], Wu and Zou [2006], Zou and Devasia [2004]). A SISO simplification of their work reduces the MIMO C to a PID or other feedback controller, and F_P of Fig. 3 is *approximately* equal to $\hat{P}^{-1}$. Ideally, F_P would be *exactly* equal to $\hat{P}^{-1}$ (or even better, P^{-1}), but often the existence of nonminimum-phase zeros in the plant force a *stable* approximate inverse to be used in place of the exact inverse. It is important to note that typically the plant and model of piezo scanners have nonminimum-phase zeros due to the flexible structure and non-collocated sensing and actuation.

Many control systems researchers, working in a myriad of application areas, have developed and used stable approximate inversions in order to implement model-inverse based control on nonminium-phase systems; this work has been applied to both the plant-injection and closed-loop-injection architectures. Some examples include Potsaid and Wen [2004], Rigney et al. [2006a,b], Tomizuka [1987], and Wen and Potsaid [2004]. Croft et al. [2001] have shown that the model-inverse based plant-injection architecture is effective in AFMs at compensating for loss of precision due to hysteresis during long range applications, due to creep effects during positioning over extended periods of time, and due to induced vibrations during high-speed positioning.

Various stable approximate model-inversion techniques exist, including Tomizuka's popular *zero-phase-error tracking controller* (ZPETC) (Tomizuka [1987]). A cousin of the ZPETC is the comparatively named *zero-magnitude-error tracking controller* (ZMETC) that has appeared in Rigney et al. [2008], Pao et al. [2007], Potsaid and Wen [2004], Wen and Potsaid [2004], and Butterworth et al. [2008]. Yet another approximation method is to use a noncausal series expansion (Gross and Tomizuka [1994], Gross et al. [1994], Rigney et al. [2006b]). Using a zeroth-order series expansion is effectively the same as choosing to ignore the nonminimum-phase zeros (while accounting for the proper DC gain); approximating the inverse of a system in this way offers a more simplistic approach, but may not be as accurate (Haack and Tomizuka [1991], Rigney et al. [2008]). In contrast, some researchers have chosen to use the exact unstable inverse and maintain stability of the system by pre-loading initial conditions or using noncasual plant inputs (Devasia et al. [1996], Hunt et al. [1996], Zou and Devasia [2004]).

Ultimately, the proper choice of a stable approximate model-inversion technique depends on the system on which it will be applied and how the controller will be implemented. Generally, ZPETC and ZMETC offer a simple and *usually* effective method for model-inverse control (Rigney et al. [2008]). Butterworth et al. [2008] compares ZMETC and ZPETC for different nonminimum phase characteristics in the plant model. When compared to ZPETC, ZMETC provides better performance results when applied to our model of the AFM x-y stage, and as such it will be the focus of the discussion of the model-inversion technique for the duration of this paper. Later in Subsection 6.4, we will discuss the importance of magnitude tracking (rather than phase) for obtaining high quality AFM images. As a result, the use of the *zero-magnitude-error* tracking controller is natural to our overall goals.

To design a ZMETC stable approximate model inverse of a nonminimum-phase plant, write the model of the plant dynamics as in (10), partitioning the polynomial $\hat{B}_s$ containing the stable (invertible) zeros from the polynomial $\hat{B}_u$ containing the unstable (noninvertible) zeros:

$$\hat{P}(z) = \frac{\hat{B}(z)}{\hat{A}(z)} = \frac{\hat{B}_s(z)\hat{B}_u(z)}{\hat{A}(z)}. \tag{10}$$

The polynomial $\hat{A}(z)$ contains all the poles of the model of the plant. $\hat{B}_u$ has the form

$$\hat{B}_u(z) = b_{un}z^n + b_{u(n-1)}z^{n-1} + \cdots + b_{u0} \tag{11}$$

where n is the number of nonminimum-phase zeros. The ZMETC technique then yields a stable approximation of the inverse of the plant

$$\tilde{P}^{-1}(z) = \frac{\hat{A}(z)}{\hat{B}_s(z)\hat{B}_u^*(z)}, \tag{12}$$

where the $\sim$ indicates an *approximate* inverse (as compared to a $\wedge$ indicating the *model* of the plant), and $\hat{B}_u^*(z)$ is

$$\hat{B}_u^*(z) = b_{u0}z^n + b_{u1}z^{n-1} + \cdots + b_{un}. \tag{13}$$

Note that the difference between (11) and (13) is the "flipping" of the coefficients. It is this action which reflects the unstable $\hat{B}_u(z)$ about $|z| = 1$ into the stable $\hat{B}_u^*(z)$. Setting F_P equal to $\tilde{P}^{-1}(z)$ and F_{CL} equal to unity or a delay block (discussed further below) in Fig. 3 would constitute a model-inverse based control using the plant-injection architecture.

If the relative degree r of $P(z)$ is greater than zero, the resulting $\tilde{P}^{-1}(z)$ will be noncausal. Additional delay equal to r will have to be incorporated into $\tilde{P}^{-1}(z)$ in order for it to be implementable in a causal way:

$$F_P = \tilde{P}^{-1}(z) = \frac{z^{-r}\hat{A}(z)}{\hat{B}_s(z)\hat{B}_u^*(z)}. \tag{14}$$

If $r > 0$, then the feedforward block F_{CL} should be defined not as unity, but rather as a delay block equal to z^{-r}. Setting $F_{CL} = z^{-r}$ changes the transfer function in (5) to

$$\left.\frac{X(z)}{X_d(z)}\right|_{FFPI} = \frac{P(z)F_P(z) + z^{-r}P(z)C(z)}{1 + P(z)C(z)}. \tag{15}$$

If (a) there exist no nonminimum-phase zeros in the plant, (b) the model exactly matches the plant ($\hat{P}(z) = P(z)$), and (c) $r = 0$, then the output of the system exactly tracks the desired input x_d and the feedback loop is not excited. If the first two conditions (a) and (b) hold and $r > 0$, then the output of the system exactly tracks the desired input x_d with a delay equal to r.

Returning to the discussion of FFPI, if we assume a nonminimum-phase, perfectly known plant ($\hat{P}(z) = P(z)$) with relative degree r, we can apply the above discussion on ZMETC and write (8) and (15) as

$$H_{CL} = \frac{B_sB_uC_N}{AC_D + B_sB_uC_N} \quad \text{and} \tag{16}$$

$$\begin{aligned}\left.\frac{X(z)}{X_d(z)}\right|_{FFPI} &= \frac{B_uAC_D + B_sB_uC_NB_u^*}{z^rB_u^*(AC_D + B_sB_uC_N)} \\ &= \frac{AC_D + B_sC_NB_u^*}{z^rB_u^*B_sC_N}H_{CL}.\end{aligned} \tag{17}$$

It should be clear that if all plant zeros are minimum phase, then (17) reduces to a delay block indicating perfectly delayed tracking.

4.2 FFPI: Feedforward H_∞ Control

H_∞ control has been applied to AFMs in the plant-injection architecture (Schitter et al. [2004a,b]). While H_∞ feedforward controllers are intended to be used in conjunction with some form of feedback control to ensure stability, the H_∞ feedforward controller can be designed independently of the feedback controller. For the plant-injection architecture, the filter F_P is designed such that the H_∞ norm of

$$T_{ff_P} = \left\| \begin{array}{c} W_{u_P} F_P \\ W_P - W_P \hat{P} F_P \end{array} \right\| \tag{18}$$

is minimized. Again, W_{u_P} and W_P are weighting functions. W_{u_P} acts on the signal to the plant. W_P weights the tracking error signal represented by the difference between the expected plant output and the reference input. Similar to the feedback weights, the selection of these weights is critical to the performance of the controller and its ability to handle uncertainty in the model. Generally, W_P has low-pass qualities and is key to the low-frequency tracking performance of the reference input. Similarly to W_u in the feedback case, the weight W_{u_P} can be designed to limit actuator saturation and/or exciting an uncertain resonance of the model (Schitter et al. [2004a,b]). Having the ability to design for actuator saturation is a major strength of H_∞ feedforward controllers when compared to ZMETC feedforward controllers which do not have such a feature. We will see the effect of this in Subsection 6.5. Although H_∞ feedforward control is not necessarily a model-inverse based technique, with careful selection of the weighting functions the result of minimizing the H_∞ norm of (18) can result in a filter that has properties that are very indicative of an approximate model inverse.

5. CLOSED-LOOP-INJECTION ARCHITECTURE

Until very recently, the closed-loop-injection architecture has not made an appearance in the AFM control literature. When compared to plant-injection, the closed-loop-injection's superior ability to perform under the presence of uncertainties in a disk drive application (Rigney et al. [2006a]) provides motivation for application to AFM control. The same feedforward techniques overviewed in Section 4 are briefly discussed specifically for the closed-loop-injection architecture in the following subsections.

In this architecture, we set F_P in Fig. 3 equal to zero and design F_{CL} accordingly. Ignoring the saturation block, we can condense a FFCLI version of Fig. 3 into a transfer function from the desired input x_d to the output x:

$$\left.\frac{X(z)}{X_d(z)}\right|_{FFCLI} = H_{CL} F_{CL}. \tag{19}$$

If we make the assumption that a stable inverse of H_{CL} exists, we can see that setting F_{CL} equal to that stable inverse would result in perfect tracking.

Further, the feedforward filter F_{CL} functions ahead of the loop that is stabilized by the feedback controller C. In this arrangement, we expect the feedback controller C to be continually excited. This is in contrast to the plant-injection architecture where the feedback loop is not activated unless modeling errors or deficiencies in the approximate model inverse force the feedback controller to be excited.

5.1 FFCLI: Feedforward Model-Inverse Control

The creation of a ZMETC model-inverse controller for the closed-loop-injection architecture follows very closely to the ZMETC procedure described in Subsection 4.1 for plant-injection architectures. The major difference is the use of ZMETC to create a stable approximate model inverse of the closed-loop dynamics ($H_{CL}(z)$) and setting F_{CL} in Fig. 3 equal to that approximate inverse $\tilde{H}_{CL}^{-1}(z)$. Following the technique described in Subsection 4.1, and making the assumption that the feedback controller C is stable, minimum phase, and exactly proper, we obtain an approximate stable inverse of $H_{CL}(z)$:

$$\begin{aligned} F_{CL} = \tilde{H}_{CL}^{-1} &= \frac{z^{-r}\hat{A}_{CL}}{\hat{B}_{sCL}\hat{B}^*_{uCL}}, \\ &= \frac{z^{-r}(\hat{A}C_D + \hat{B}_s\hat{B}_u C_N)}{(\hat{B}_s C_N)(\hat{B}^*_u)} \end{aligned} \tag{20}$$

where (16) defines A_{CL} and B_{sCL}. For minimum phase C, B^*_{uCL} is equal to the same B^*_u from Subsection 4.1. Because C is assumed to be exactly proper, the relative degree r remains the same as before as well.

With C stable, minimum phase, and exactly proper, and further assuming the nonminimum-phase plant is known with certainty, we can use (16) and (20) to rewrite (19):

$$\begin{aligned} \left.\frac{X(z)}{X_d(z)}\right|_{FFCLI} &= \frac{B_s B_u C_N}{AC_D + B_s B_u C_N} \frac{z^{-r}(AC_D + B_s B_u C_N)}{(B_s C_N)(B^*_u)}, \\ &= \frac{z^{-r} B_u}{B^*_u}. \end{aligned} \tag{21}$$

This result shows one of the advantages of FFCLI over FFPI: the FFCLI transfer function from x_d to x reduces to only the dynamics of the plant's nonminimum phase zeros, their approximate inverse, and possibly some delay. This is in contrast to (17) of the plant-injection architecture which is considerably more complex. Like the FFPI architecture, we can see that when $H_{CL}(z)$ is minimum-phase and the model $\hat{P}(z)$ is known with certainty, (21) becomes unity or a delay block. In this case, we can expect x to perfectly track x_d (perhaps with some delay).

5.2 FFCLI: Feedforward H_∞ Control

The H_∞ feedforward controller design for closed-loop-injection follows closely to that for the plant-injection-architecture. F_{CL} is designed to minimize the H_∞ norm of

$$T_{ff_{CL}} = \left\| \begin{array}{c} W_{u_{CL}} F_{CL} \\ W_{CL} - W_{CL}\hat{H}_{CL} F_{CL} \end{array} \right\|. \tag{22}$$

The weighting function $W_{u_{CL}}$ acts on the reference input to the closed-loop system. W_{CL} penalizes the difference between the closed-loop output and the reference input.

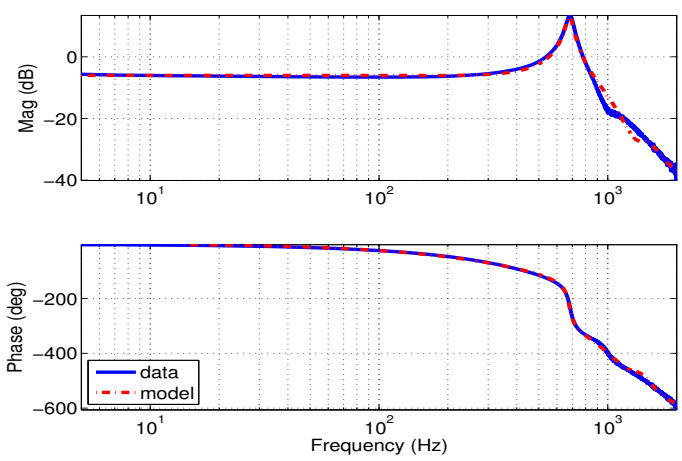


Fig. 4. Frequency response functions characterizing the plant for motion in the x direction. The plant model has been extracted based upon the measurement data shown of an actual x-y stage.

As with all H_∞ controllers, the choice of these weights is critical to the performance of the controller. The general design choice of $W_{u_{CL}}$ and W_{CL} follows closely to the design of W_{u_P} and W_P described in Subsection 4.2.

6. COMPARISON OF ARCHITECTURES

This section discusses a comparison of the plant-injection and closed-loop-injection architectures. A model for the x direction of our nPoint NPXY100A x-y piezoscanner stage will be used for all control comparisons. Using frequency response methods to match actual measured data from our nPoint x-y stage (see Fig. 4), we obtained a 7th-order discrete-time model

$$\hat{P}_{xx}(z) = \frac{-0.0014(z-0.0061)(z-1.7824)}{(z-0.8884)(z-0.8572\pm j0.4032)} \times \frac{(z-1.1264\pm j0.4627)(z-0.8762\pm j0.3766)}{(z-0.8717\pm j0.2742)(z-0.9716\pm j0.2022)}. \quad (23)$$

The nPoint stage has a signal conditioner that converts the high-voltage piezo signals into a ±10 volt range to represent the stage's position. Both x and y directions of the stage have a $\pm 50\mu$m range corresponding to the ±10 volts. This signal conditioner was considered part of the stage when the system identification was performed, and as a result $\hat{P}_{xx}(z)$ in (23) and every controller described in this paper are defined on the volt scale rather than the μm scale. Translating between volts and microns is done via the linear ratio of the two.

The model's three nonminimum-phase zeros will challenge the performance of all control designs and more specifically limit the effectiveness of the model-inverse based methods. The relative degree $r = 1$, and the sample rate for this model and all associated controllers is 20.833 kHz.

6.1 Companion Simulations

In an effort to closely compare these two architectures, we have selected four key simulation types for each architecture. Each simulation under one architecture has a corresponding *companion* simulation in the contrasting architecture. In the interest of comparison, the companion simulations share not just the same classification of feedback control, but more importantly, they share the *exact same* feedback controller C. Each feedforward controller (F_P or F_{CL}) has been designed to perform specifically for its architecture and therefore varies throughout the simulations. Table 1 provides a summary of the simulations with a focus on the relationships between certain pairs. We will take a close look at a total of 8 simulations.

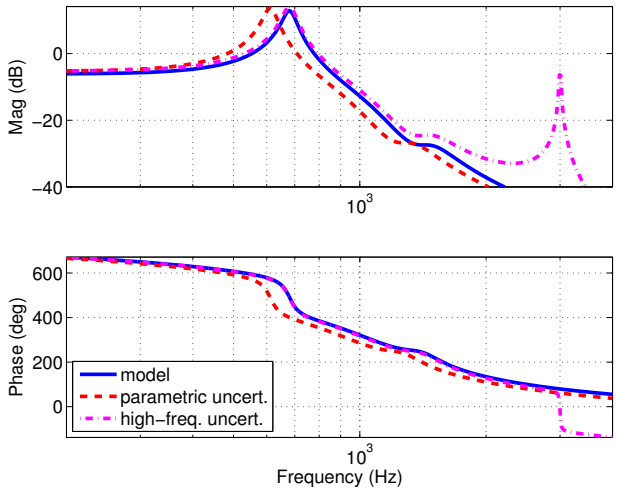


Fig. 5. The three variations of the true plant used for the simulations: 1) perfectly known plant (solid blue), 2) parametric structured uncertainty (dashed red), and 3) unstructured high-frequency uncertainty (dash-dot magenta).

6.2 Plant Uncertainty

Each of the eight simulations were run for three different variations of the plant P. In the first case, we assume the plant is known perfectly, $\hat{P}(z) = P(z)$. We also studied two cases of plant uncertainty: parametric structured uncertainty and unstructured high-frequency uncertainty. The parametric structured uncertainty was designed to represent a low-frequency modeling error in which all of the parameters in the numerator and denominator of (23) were shifted by 10%. Further, the DC gain was adjusted to be 10% greater. The unstructured high-frequency uncertainty represents a high frequency modeling error in the form of an additional unmodelled resonance at 3kHz. As with the parametric structured uncertainty, the DC gain was again adjusted to be 10% greater than the plant model. Fig. 5 shows the frequency responses of all three "true" plants considered.

Table 1. Simulations and their Relationships.

Sim. #	Relationship	F_P	F_{CL}	C
1 2	Companion Sims.	H_∞ Zero	Unity H_∞	PID
3 4	Companion Sims.	H_∞ Zero	Unity H_∞	H_∞
5 6	Companion Sims.	ZMETC Zero	z^{-r} ZMETC	PID
7 8	Companion Sims.	ZMETC Zero	z^{-r} ZMETC	H_∞

Table 2. Feedback Controller Designs.

Control Type	Transfer Function	Zeros	Poles
C: PID	$-0.05 + \frac{955}{s} + 2.5 \times 10^{-5} s$	$1000 \pm j6099$	0
C: H_∞	$\frac{0.8154z^6-2.923z^5+3.208z^4+0.4087z^3-3.421z^2+2.514z-0.6023}{z^6-3.66z^5+4.539z^4-1.134z^3-2.155z^2+1.864z-0.454}$	$-1.000,\ 0.8884,$ $0.8717 \pm j0.2742,$ $0.9764 \pm j0.2062$	$-0.7586,\ 1.000,$ $0.8913 \pm j0.2609,$ $0.8178 \pm j0.1585$

Table 3. Feedforward Controller Designs.

Sim. #	Cntrl. Type	Transfer Function	Zeros	Poles
1 & 3	F_P: H_∞	$\frac{42.67z^8-237.2z^7+517.4z^6-488.9z^5+4.428z^4+437.4z^3\cdots}{z^8-3.768z^7+4.372z^6+0.6148z^5-5.265z^4+3.769z^3\cdots}$ $\frac{\cdots-423.7z^2+177z-28.97}{\cdots-0.2352z^2-0.694z+0.2079}$	$0.8884,\ -0.9496,$ $0.9716 \pm j0.2022,$ $0.9664 \pm j0.2118,$ $0.8717 \pm j0.2742$	$-0.9936,\ -0.4346,$ $0.7524 \pm j0.1027,$ $0.8779 \pm j0.2862,$ $0.9678 \pm j0.2061$
2	F_{CL}: H_∞	$\frac{173.4z^9-1129z^8+3006z^7-3890z^6+1623z^5+2219z^4\cdots}{z^9-4.386z^8+6.267z^7-0.2441z^6-8.392z^5+7.979z^4\cdots}$ $\frac{\cdots-3857z^3+2616z^2-886.6z+124}{\cdots-0.667z^3-3.049z^2+1.835z-0.3416}$	$-0.9967,$ $0.8700 \pm j0.2724,$ $0.9664 \pm j0.2118,$ $0.9676 \pm j0.1994,$ $0.9504 \pm j0.0140$	$-0.9959,$ $0.7023,\ -0.7538,$ $0.9683 \pm j0.2012,$ $0.8666 \pm j0.2697,$ $0.8821 \pm j0.1613$
4	F_{CL}: H_∞	$\frac{0.6304z^8-3.382z^7+7.007z^6-5.905z^5-1.226z^4+6.905z^3\cdots}{z^8-7.002z^7+21.6z^6-38.3z^5+42.72z^4-30.68z^3\cdots}$ $\frac{\cdots-6.039z^2+2.382z-0.3719}{\cdots+13.85z^2-3.591z+0.4097}$	$0.8491,\ -1.000,$ $0.9163 \pm j0.0863,$ $0.9701 \pm j0.2037,$ $0.8716 \pm j0.2742$	$0.7917 \pm j0.0601,$ $0.8587 \pm j0.1893,$ $0.8764 \pm j0.2819,$ $0.9744 \pm j0.2061$
5 & 7	F_P: ZMETC	$\frac{719.2z^7-4523z^6+1.239e4z^5-1.914e4z^4+1.802e4z^3\cdots}{2.643z^7-10.15z^6+16.14z^5-13.17z^4+5.503z^3\cdots}$ $\frac{\cdots-1.034e4z^2+3346z-471.5}{\cdots-0.9427z^2+0.005554z}$	$0.8884,$ $0.9716 \pm j0.2022,$ $0.8717 \pm j0.2742,$ $0.8572 \pm j0.4032$	$0,$ $0.0061,\ 0.5610,$ $0.8762 \pm j0.3766,$ $0.7596 \pm j0.3120$
6	F_{CL}: ZMETC	$\frac{522.4z^9-3809z^8+1.229e4z^7-2.291e4z^6+2.701e4z^5\cdots}{z^9-5.743z^8+14.41z^7-20.41z^6+17.63z^5\cdots}$ $\frac{\cdots-2.062e4z^4+9955z^3-2779z^2+343.4z-0.005554}{\cdots-9.263z^4+2.746z^3-0.3578z^2+0.002085z}$	$1.617e-5,$ $0.9504 \pm j0.0132,$ $0.9676 \pm j0.1995,$ $0.8700 \pm j0.2723,$ $0.8570 \pm j0.4031$	$0,$ $0.0061,\ 0.5610,$ $0.9520 \pm j0.2928,$ $0.8762 \pm j0.3766,$ $0.7596 \pm j0.3120$
8	F_{CL}: ZMETC	$\frac{333.7z^{10}-2442z^9+7724z^8-1.344e4z^7+1.306e4z^6-5002z^5\cdots}{z^{10}-4.792z^9+8.806z^8-6.131z^7-2.835z^6+8.508z^5\cdots}$ $\frac{\cdots-3544z^4+5939z^3-3565z^2+1071z-133.9}{\cdots-6.613z^4+2.412z^3-0.3572z^2+0.002093z}$	$-0.7587,\ 0.8491,$ $0.8560 \pm j0.4045,$ $0.8716 \pm j0.2742,$ $0.9701 \pm j0.2037,$ $0.9163 \pm j0.0863$	$-1.000,\ 0,$ $0.0061,\ 0.5610,$ $0.9764 \pm j0.2062,$ $0.8762 \pm j0.3766,$ $0.7596 \pm j0.3120$

6.3 The Controllers

Following the feedback and feedforward controller design methods outlined in Sections 3, 4, and 5, we designed 2 feedback controllers and 6 feedforward controllers. All controllers are summarized in Tables 2 and 3. In some cases, the controller order was reduced via the removal of near pole/zero cancellations. The value for the proportional parameter of the PID feedback controller in Table 2 is negative due to a restriction in the nPoint software. The same feedforward controller was used for both H_∞ and ZMETC feedforward plant-injection simulations (#1 & #3 and #5 & #7). This is because regardless of the feedback controller C, the feedforward control objectives remain the same. A brief description on the particular designs of the H_∞ weighting functions W_e, W_u, W, W_{u_P}, W_P, $W_{u_{CL}}$, and W_{CL} is provided below.

H_∞ Feedback Weight Designs: The H_∞ feedback control weighting functions W_e, W_u, and W were designed following the guidelines described in Subsection 3.2. Specifically, the feedback weights took the form of

$$W_e = \frac{4z - 1.278}{100(z-1)},$$

$$W_u = \frac{1.3779(z-0.9441)^2}{z^2 - 1.953z + 0.9959}, \text{ and}$$

$$W = \frac{0.6z - 0.4858}{0.1903z}.$$

Here, W_e and W have low and high-pass characteristics, respectively. Meanwhile, W_u is designed to both limit actuator saturation at low frequencies and to weight the resonance of the plant.

Table 4. Feedforward Weight Designs.

Sim. #	W_i	W_{u_i}
1 & 3	$W_P = \frac{0.0667z-0.014}{z-0.9947}$	$W_{u_P} = \frac{9.82e-3(z-0.766)(z+1)}{z^2-1.933z+0.9788}$
2	$W_{CL} = \frac{0.05z+0.101}{z-0.9998}$	$W_{u_{CL}} = \frac{9.82e-3(z-0.766)(z+1)}{z^2-1.933z+0.9788}$
4	$W_{CL} = \frac{0.05z+0.101}{z-0.9998}$	$W_{u_{CL}} = 0.1$

H_∞ Feedforward Weight Designs: Table 4 summarizes the H_∞ feedforward weighting functions (W_{u_P}, W_P, $W_{u_{CL}}$, and W_{CL}) used in our simulations. For both H_∞ feedforward plant-injection simulations (#1 and #3), the choice of the weights was exactly the same. Again, this is because regardless of the feedback controller C, the control objectives remain the same in this FFPI architecture and the weights need not change as the feedback controller changes. In simulations #1, #2, and #3, the weights W_{u_P} and $W_{u_{CL}}$ were designed for robustness against potential uncertainty near the resonance of the plant. W_{u_P} for simulations #1 and #3 was chosen to be the same as $W_{u_{CL}}$ for simulation #2 as both controllers were working to address the same possible uncertainty in the plant. W_{CL} in simulations #2 and #4 are also the same as the choice provided good tracking results for both. In simulation #4, only limiting actuator saturation was considered as weighting the resonance provided no added performance even under the presence of uncertainty.

6.4 Performance Metrics

When discussing the performance of the tracking of a raster scan in AFMs, it is important to recall the overall goal of AFMs: to create a quality image in a timely manner. But this goal requires a definition of a "quality image" when referring to an x-y raster scan. Focusing on the x direction, an ideal controller would cause the system output $x(t)$ to track the desired raster pattern $x_d(t)$ flawlessly. This suggests that the two performance metrics in (24) and (25) might be informative in defining the performance of a controller. Here we define two metrics over one period T of the raster scan after time t_{ss} after which all transients (from initial conditions for example) have died out:

$$J_e = \int_{t_{ss}}^{t_{ss}+T} \big(x_d(t) - x(t)\big)^2 dt \tag{24}$$

$$J_m = \max_{t\in[t_{ss},t_{ss}+T)} \big(x_d(t) - x(t)\big)^2. \tag{25}$$

While these metrics quantify tracking error to some extent, a variation of them may be more effective for determining the best controllers *for imaging*. This is because phase lag in the raster scan used for AFM imaging is not nearly as critical as consistently tracking the magnitude. Ultimately, this means that perfectly delayed tracking is better than imperfect timely tracking if we know the delay well. As a result, we introduce two variations on the metrics in (24) and (25). First, let us define the integer variable k^* as

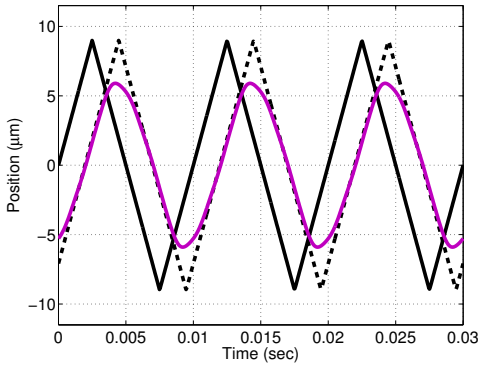


Fig. 6. Example of simulation results using PID feedback-only. The 100Hz input $x_d(t)$ is displayed in solid black, $x_d(t-k^*T_s)$ used for the performance metrics $J_{e\tau}$ and $J_{m\tau}$ is shown in dashed black, and the system output $x(t)$ is in solid purple.

$$k^* = \arg\min_k \int_{t_{ss}}^{t_{ss}+T} \big(x_d(t - kT_s) - x(t)\big)^2 dt \tag{26}$$

where k is also an integer and is defined on $[0, \frac{T}{T_s}]$ where $T_s = 48\mu$sec is the controller sample period. k^* is defined to be an integer as it represents actual implementation in a digital controller. Specifically, k^* reflects the phase lag in the system or in other words, the discrete delay. We can then use this information to define two new metrics that disregard the phase lag in the system and emphasizes magnitude tracking:

$$J_{e\tau} = \int_{t_{ss}}^{t_{ss}+T} \big(x_d(t - k^*T_s) - x(t)\big)^2 dt \tag{27}$$

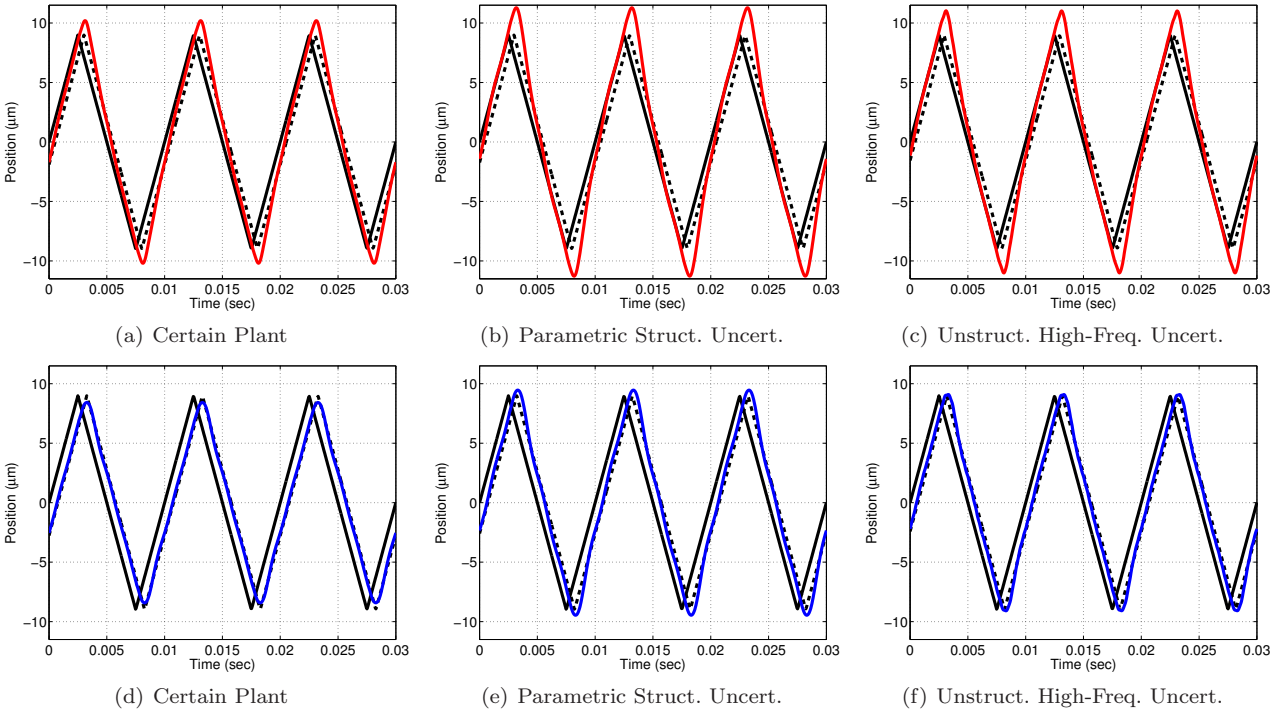
$$J_{m\tau} = \max_{t\in[t_{ss},t_{ss}+T)} \big(x_d(t - k^*T_s) - x(t)\big)^2 \tag{28}$$

We will use (27) and (28) extensively when comparing the simulations of plant-injection and closed-loop-injection architectures.

For further clarification, we provide Fig. 6 which is an example simulation of a 100Hz raster scan input into a PID feedback-only control loop. We have chosen a simple PID feedback-only simulation as an example for its clear delineation of each line. Here we see the actual 100Hz raster scan input $x_d(t)$ in solid black, and the shifted 100Hz input $x_d(t-k^*T_s)$ in dashed black. In this case, $k^* = 39$ or $k^*T_s = 1.9$msec. From Fig. 6, it is clear that the PID feedback-only controller lacks the bandwidth to be able to track the 100Hz raster scan. Images created while using this raster scan would be highly distorted at the edges.

6.5 Simulation Results

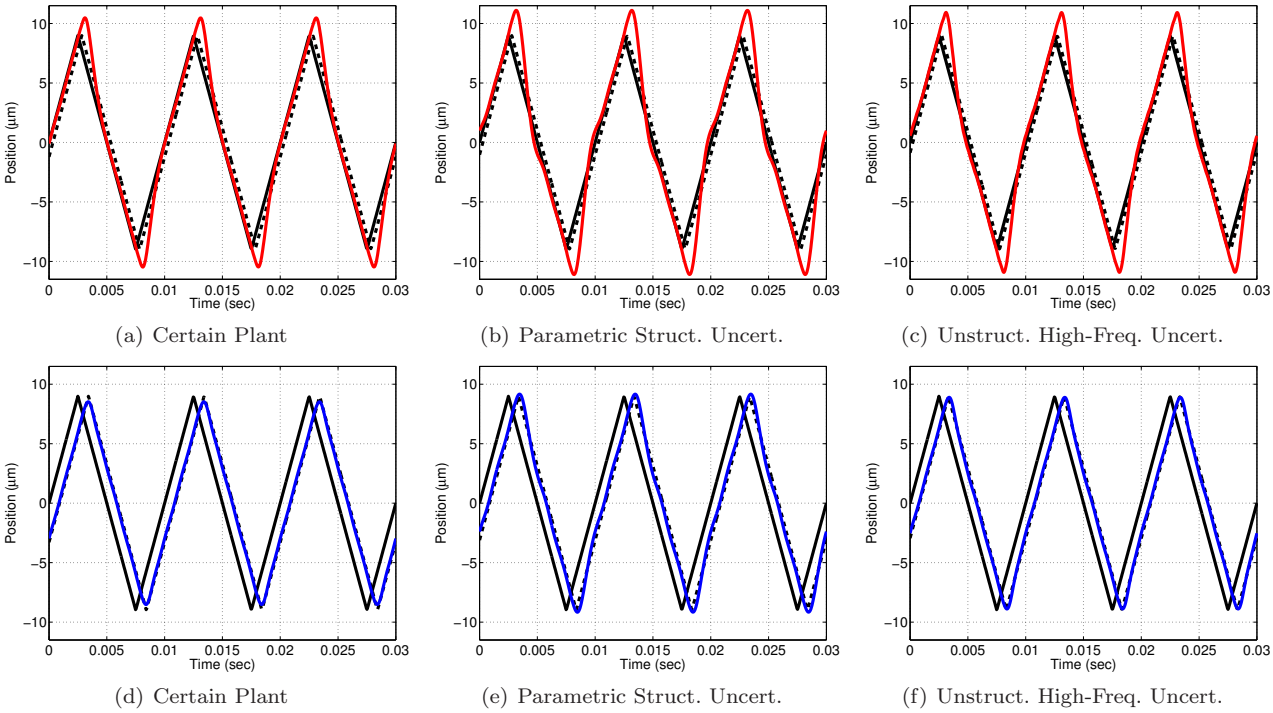
This section summarizes the results of each of the simulations listed in Table 1. All simulations were run on a block diagram similar to Fig. 3 with a 100Hz raster scan as the input on the SISO x direction model of our nPoint x-y stage. The saturation block in Fig. 3 plays a critical role in ensuring that the results are an accurate representation of a real-world implementation. Further, we include



(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

(d) Certain Plant (e) Parametric Struct. Uncert. (f) Unstruct. High-Freq. Uncert.

Fig. 7. Results for simulations #1 & 2: PID feedback and H_∞ feedforward. $x_d(t)$ is displayed in solid black and $x_d(t - k^*T_s)$ used for the performance metrics $J_{e\tau}$ and $J_{m\tau}$ is shown in dashed black. FFPI results are displayed in solid red in the top row while the corresponding FFCLI results are shown in solid blue in the bottom row.

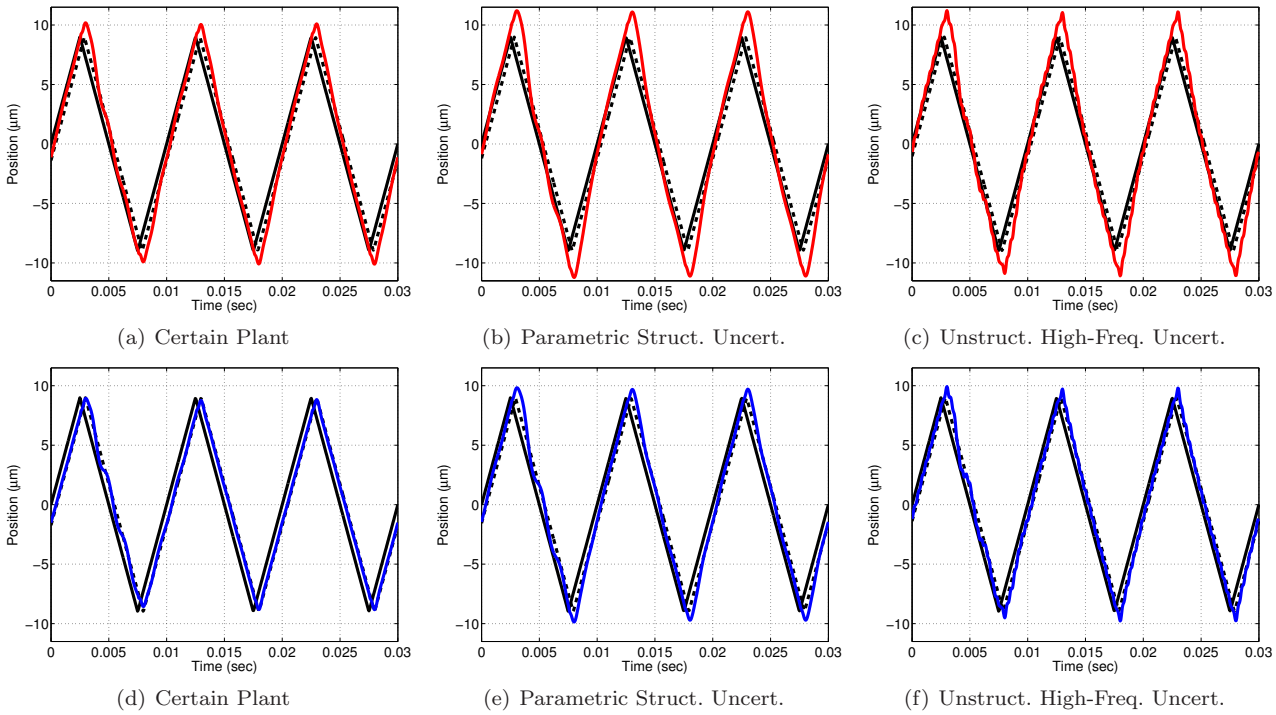


(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

(d) Certain Plant (e) Parametric Struct. Uncert. (f) Unstruct. High-Freq. Uncert.

Fig. 8. Results for simulations #3 & 4: H_∞ feedback and H_∞ feedforward. $x_d(t)$ is displayed in solid black and $x_d(t - k^*T_s)$ used for the performance metrics $J_{e\tau}$ and $J_{m\tau}$ is shown in dashed black. FFPI results are displayed in solid red in the top row while the corresponding FFCLI results are shown in solid blue in the bottom row.

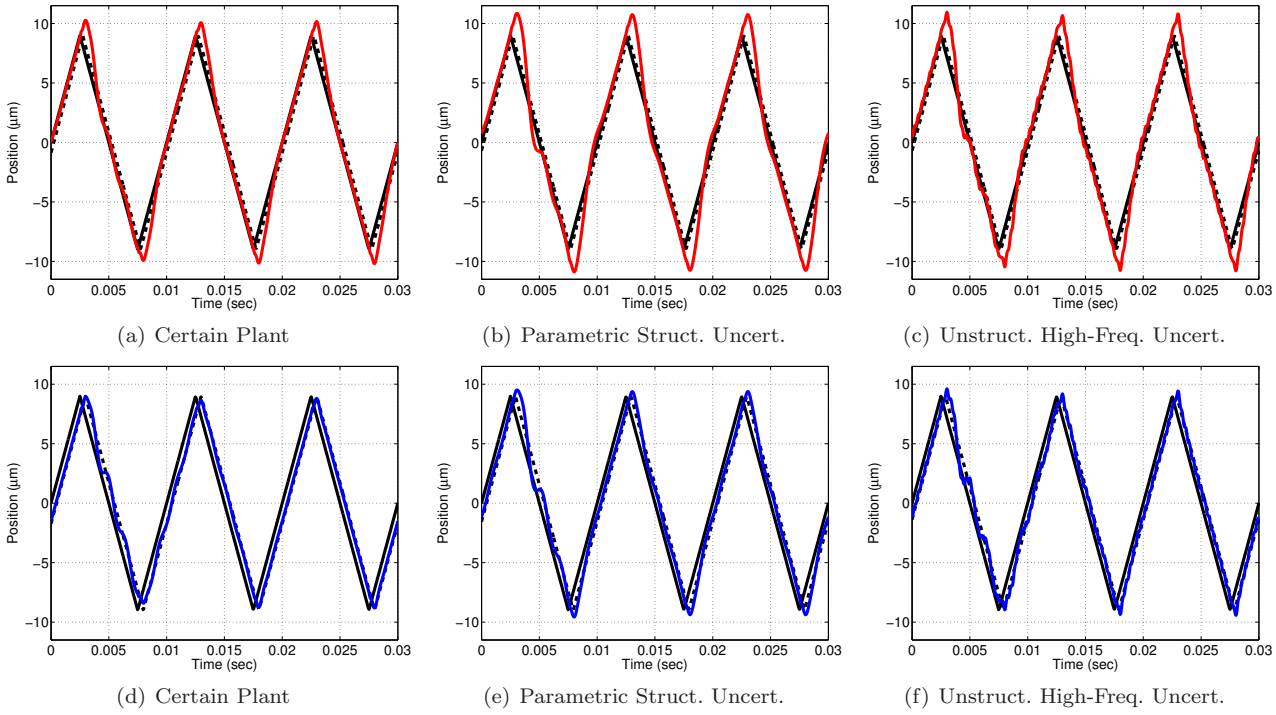
it in our simulations because it helps us more closely compare H_∞ control designs with ZMETC designs. The key difference being that when designing H_∞ controllers, adjustments can be made for saturation (see the discussion on H_∞ weighting functions in Subsection 6.3). In contrast, designers of ZMETC controllers have no direct tools to limit actuator saturation other than altering the filter which would reduce tracking performance. As a result, the saturation block critically affects the performance of the



(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

(d) Certain Plant (e) Parametric Struct. Uncert. (f) Unstruct. High-Freq. Uncert.

Fig. 9. Results for simulations #5 & 6: PID feedback and ZMETC feedforward. $x_d(t)$ is displayed in solid black and $x_d(t - k^*T_s)$ used for the performance metrics $J_{e\tau}$ and $J_{m\tau}$ is shown in dashed black. FFPI results are displayed in solid red in the top row while the corresponding FFCLI results are shown in solid blue in the bottom row.



(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

(d) Certain Plant (e) Parametric Struct. Uncert. (f) Unstruct. High-Freq. Uncert.

Fig. 10. Results for simulations #7 & 8: H_∞ feedback and ZMETC feedforward. $x_d(t)$ is displayed in solid black and $x_d(t - k^*T_s)$ used for the performance metrics $J_{e\tau}$ and $J_{m\tau}$ is shown in dashed black. FFPI results are displayed in solid red in the top row while the corresponding FFCLI results are shown in solid blue in the bottom row.

ZMETC controllers. We will discuss this further below for each applicable simulation.

We have provided plots of the simulation results (Figs. 7, 8, 9, and 10) and plots featuring the error between the shifted 100Hz input $x_d(t - k^*T_s)$ and the system output $x(t)$ (Figs. 11, 12, 13, and 14). Specifically, *error* is defined as

$$error = x_d(t - k^*T_s) - x(t), \tag{29}$$

which appears in the right-hand side of both (27) and (28). The intention is to compare the performance of companion simulations in one plot, so each contains an error plot for a FFPI simulation and its companion FFCLI simulation. The corresponding k^* for each shifted input is displayed in the legend box. Due to the shift associated with each individual simulation, the error for simulation #2 at a given time is not perfectly comparable to the error for simulation #1 at that same given time. For this reason, the reader should focus on comparing the general magnitude of the error curves over time rather than the error at a specific time.

In Figs. 7 and 11, three versions of each set of companion simulations are displayed from left to right, each with a varying degree of plant uncertainty (as discussed in Subsection 6.2). The leftmost subfigures of Figs. 7 and 11 are for simulations with a perfectly known plant. The middle and rightmost subfigures of Figs. 7 and 11 are of simulations when there exists parametric structured uncertainty and unstructured high-frequency uncertainty in the plant. Figs. 8, 9, 10, 12, 13, and 14 present the rest of the simulation results in a similar way for the other sets of companion simulations.

Figs. 7 and 11 compare simulations #1 and #2 that use PID feedback and H_∞ feedforward control. It is clear in Fig. 7 that in all cases, the closed-loop-injection architecture tracks the 100Hz raster pattern better than the plant-injection architecture. The large peaks in the tracking error of the plant-injection technique of Fig. 11 correspond to the overshoot at the turn-around points of the raster scan of the FFPI system in Fig. 7. An overshoot of this magnitude will compromise the quality of imaging for the plant-injection architecture. In contrast, the closed-loop-injection tracking error experiences limited overshoot and for this reason will provide a more consistent scan for imaging. Even in the presence of uncertainty, the closed-loop-injection system performs better. The parametric uncertainty causes slight ringing in both architectures, but it is less severe in closed-loop-injection.

Figs. 8 and 12 compare H_∞ feedforward controllers when using H_∞ feedback control in the FFPI and FFCLI architectures. Again, we see overshoot at the turn-around points in the plant-injection architecture, which becomes more extreme as we add plant uncertainty. The closed-loop-injection architecture tracks the raster scan more closely in all cases of this set of companion simulations.

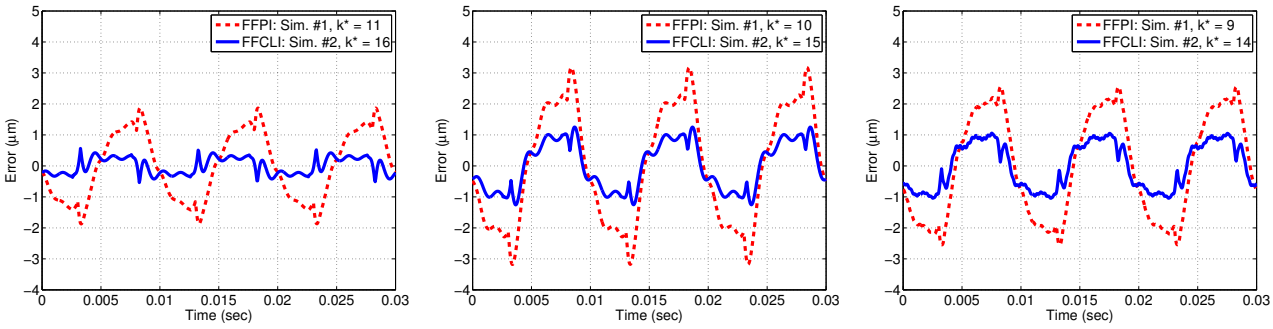
Figs. 9 and 13 compare ZMETC feedforward control when using a PID feedback controller. When the plant is perfectly known in Fig. 13(a), we again see peaks in the error tracking indicative of overshoot at the turn-around points in the FFPI architecture in Fig. 9(a). The FFPI's errors in tracking amplifies with the inclusion of parametric uncertainty and severe ringing appears in the presence of high-frequency uncertainty. Parametric uncertainty results in some overshoot in the FFCLI plot (Fig. 13(b)), but it is no more than that of the overshoot for the certain plant in the FFPI architecture. In Figs. 9(c),and 9(f), we see that both architectures struggle with the unstructured high-frequency uncertainty as severe ringing appears in both. It is clear that in both architectures, neither would produce a decent image with these controllers in the presence of such high-frequency uncertainty.

Looking closely at all three subfigures in Fig. 13, one will notice the lack of repeatability of the error plots over the three periods of interest. This is the result of occasional actuator saturation from the high energy of the ZMETC controllers (for both architectures). This effect only appears every third period of a sustained raster scan; due to our choice of axis scaling it appears in the first period of our plots and can be seen in *all* simulations that include ZMETC feedforward control (Figs. 9, 10, 13, and 14). This phenomenon completely disappears when the saturation block is removed from the simulations. For our simulations, this effect makes it difficult to define the k^*. In order to define the metrics, we chose to measure them over the last period prior to the period exhibiting a slight increase in tracking error due to saturation. Clearly this will limit the quality of any AFM images obtained with these feedforward controllers, and it suggests that ZMETC may not be a wise choice for control of this plant due to its actuator limits.

Figs. 10 and 14 compare the ZMETC feedforward control and H_∞ feedback controller of simulations #7 and #8. One should note the presence of the previously discussed ringing due to saturation limits in the first period displayed in all subfigures of Fig. 10. Ignoring this ringing, we see that the overshoot characteristic of the FFPI architecture simulation plots continue, and the degradation of tracking increases as plant uncertainty is included. In contrast, in Fig. 10(d), the FFCLI architecture tracks the magnitude of the scan well when the plant is perfectly known. Similar to Fig. 13(c), we note that when including unstructured high-frequency uncertainty, severe ringing appears for both architectures. Again, utilizing these controllers, neither architecture would produce a decent AFM image in the presence of such high-frequency uncertainty.

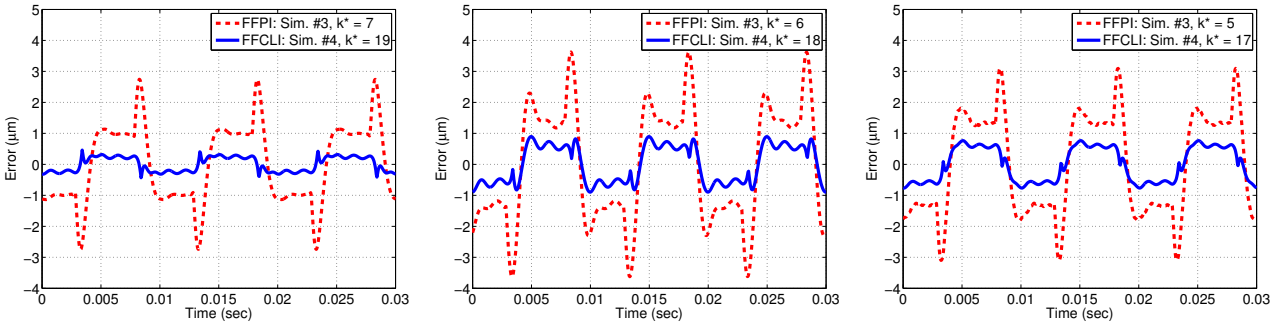
One should note that in the top row of the simulation results in Figs. 7, 8, 9, and 10, the FFPI architecture tends to force the trajectory back toward the original desired scan $x_d(t)$ rather than the shifted scan $x_d(t - k^*T_s)$. In particular, the ZMETC FFPI simulations converge to a delayed scan $x_d(t - rT_s)$ where z^{-r} is the delay implemented in the F_{CL} block. In contrast, the delay associated with the H_∞ FFPI simulations is not as well defined. This action is likely responsible for much of the extreme overshoot in the plant-injection architecture, and is not present in the FFCLI simulations. Further, this characteristic of the FFPI architecture suggests that varying from the typical choice of the identity or z^{-r} for F_{CL} may be beneficial. In particular, additional simulations including discrete delay near the order of k^* in F_{CL} for both the H_∞ and ZMETC F_P controllers showed improved FFPI performance. However, the exact choice of the length of delay was determined empirically in these additional simulations, and currently there are no clear theoretical design paths for the choice of this F_{CL} delay time. Further, since F_{CL} should not necessarily be limited to only a delay, this suggests future research in concurrently designing F_P and F_{CL}.

Fig. 15 displays the metrics $J_{e\tau}$ and $J_{m\tau}$ in separate subfigures for each and every simulation. The blue symbols (◯, +, and □: representing results of the FFCLI archi-



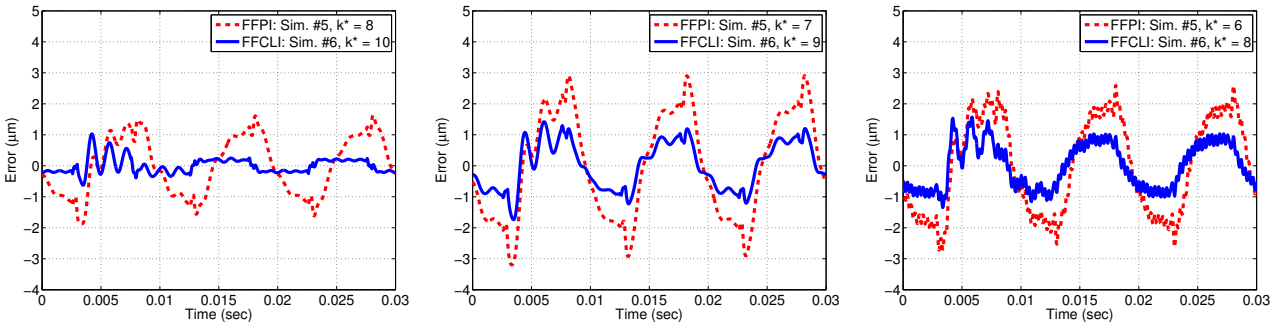
(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

Fig. 11. Tracking errors for companion simulations #1 & 2: PID feedback and H_∞ feedforward.



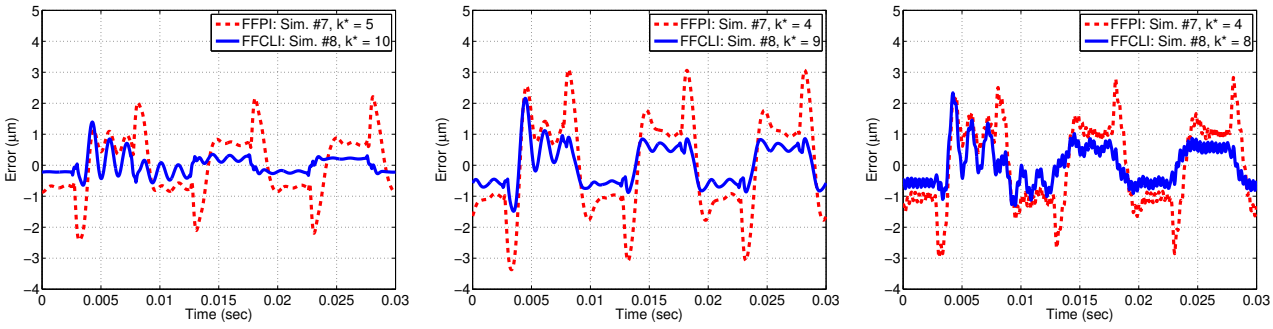
(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

Fig. 12. Tracking errors for companion simulations #3 & 4: H_∞ feedback and H_∞ feedforward.



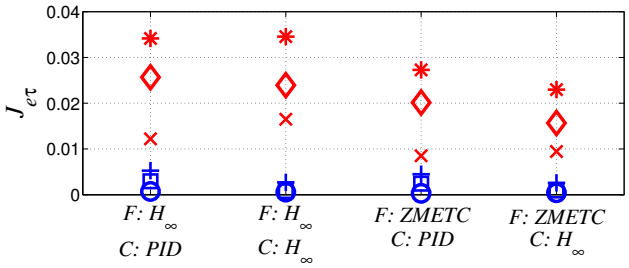
(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

Fig. 13. Tracking errors for companion simulations #5 & 6: PID feedback and ZMETC feedforward.

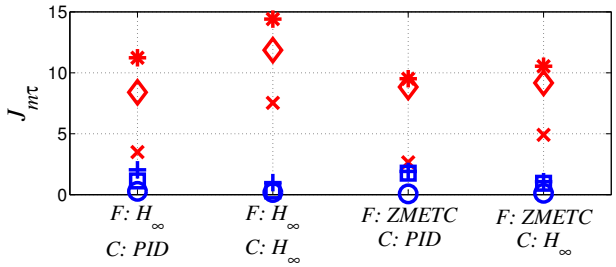


(a) Certain Plant (b) Parametric Struct. Uncert. (c) Unstruct. High-Freq. Uncert.

Fig. 14. Tracking errors for companion simulations #7 & 8: H_∞ feedback and ZMETC feedforward.



(a) $J_{e\tau}$



(b) $J_{m\tau}$

Fig. 15. Comparison of the performance metrics ($J_{e\tau}$ and $J_{m\tau}$) between the FFPI and FFCLI architectures for a 100Hz raster scan. The blue symbols represent FFCLI results and the red symbols represent FFPI results. Specifically, ◯: FFCLI perfectly known plant, ×: FFPI perfectly known plant, +: FFCLI parametric uncertainty, ∗: FFPI parametric uncertainty, □: FFCLI high-frequency uncertainty, ◇: FFPI high-frequency uncertainty.

tecture) consistently appear below the red symbols (×, ∗, and ◇: representing results of the FFPI architecture). At a glance, this tells us that when considering the metrics $J_{e\tau}$ and $J_{m\tau}$, the closed-loop-injection outperforms the plant-injection architecture in all simulations. In further testing of parametric uncertainty, we varied the parameters by ±15% and the FFCLI architecture continued to outperform the FFPI architecture in each case.

7. CONCLUSIONS AND FUTURE WORK

Many researchers have applied advanced control techniques in the form of the plant-injection architecture to AFMs. Extending upon that, we have investigated the closed-loop-injection architecture for AFM control. The closed-loop-injection architecture has been successfully applied to other mechatronics devices and its application to AFMs is natural. We have shown that even in the presence of model uncertainties, the closed-loop-injection architecture will lead to more precise tracking of a raster scan than the plant-injection architecture.

In general, FFCLI has several advantages over the FFPI architecture. Specifically in Subsection 5.1, we show that when using model-inverse control, the dynamics of the overall FFCLI SISO transfer function from the desired output (reference x_d) to the actual output (x) can often reduce to a delay block and filter of order much lower than that of the original plant. Further, when working perfectly, the FFPI architecture will not excite the closed-loop dynamics. But when F_P is not a perfect inverse of the true plant, the closed-loop dynamics can be excited in such a way that limit the performance of the overall system. Also, the physical implementation (in a digital-signal processor, for example) of the plant-inverse architecture for model inverse control can become complicated by the programming of a possible delay block in the F_{CL} position. Finally, in many systems, system identification can only be done (or is best done) in closed-loop which indicates the closed-loop-injection architecture as a natural choice.

There is one key disadvantage to the FFCLI system. Specifically, the FFCLI filter F_{CL} can become a rather high-order filter in order to capture all the necessary inverse dynamics of the closed-loop system. This can be a problem for implementation, not only due to the high order, but also due to the numeric sensitivity that may result with such a high order.

Advanced control of AFMs is a growing area and there is much more work to be done and many areas to be explored including:

- All of the above simulations will be implemented in hardware to further our understanding of the advantages and disadvantages of both the plant-injection and closed-loop-injection architectures when applied to all directions of the AFM.
- No truly MIMO algorithms have been developed and applied for controlling AFMs. While cross-coupling has been noted in AFMs, the effect of the cross-coupling terms has not been explored extensively in the AFM control literature. MIMO controllers for the full plant model in (1) should be developed and evaluated to determine the performance gains achievable.
- For both combined feedforward/feedback control approaches shown in Fig. 2, a more thorough investigation of how best to jointly design the feedforward and feedback compensators warrants further study.

In summary, the AFM is already recognized as an important tool for imaging nanoscale structures, and it is becoming a driving technology in nanomanipulation and nanoassembly. There are many areas for further investigation that can improve the control of AFMs significantly to enable an even wider range of applications throughout various disciplines.

REFERENCES

D. Y. Abramovitch, S. B. Andersson, L. Y. Pao, and G. Schitter. A tutorial on the mechanisms, dynamics, and control of atomic force microscopes. In *Proc. Amer. Ctrl. Conf.*, pages 3488–3502, July 2007.

T. Ando, N.Kodera, E. Takai, D. Maruyama, K. Saito, and A. Toda. A high-speed atomic force microscope for studying biological macromolecules. *Proc. Nat. Acad. of Sci.*, 98(22):12468–12472, Oct. 2001.

K. J. Åström and T. Hägglund. *Advanced PID Control*. ISA Press, 2005.

B. Bhikkaji, M. Ratnam, A. Fleming, and S. O. R. Moheimani. High-performance control of piezoelectric tube scanners. *IEEE Trans. Ctrl. Sys. Tech.*, 15(5):853–866, Sept. 2007.

G. Binning and D. P. E. Smith. Single-tube three dimensional scanner for scanning tunneling microscopy. *Rev. Sci. Instr.*, 57 (8):1688–1689, March 1986.

J. A. Butterworth, L. Y. Pao, and D. Y. Abramovitch. The effect of nonminimum-phase zero locations on the performance

of feedforward model-inverse control techniques in discrete-time systems. In *Proc. Amer. Ctrl. Conf.*, June 2008.

D. Croft and S. Devasia. Vibration compensation for high speed scanning tunneling microscopy. *Rev. Sci. Instr.*, 70(12):4600–4605, Dec. 1999.

D. Croft, G. Shedd, and S. Devasia. Creep, hysteresis, and vibration compensation for piezoactuators: Atomic force microscopy application. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 123:35–43, March 2001.

S. Devasia, D. Chen, and B. Paden. Nonlinear inversion-based output tracking. *IEEE Trans. Auto. Ctrl.*, 41(7):930–942, July 1996.

F. El Feninat, T. H. Ellis, E. Sacher, and I. Stangel. A tapping mode AFM study of collapse and denaturation in dentinal collagen. *Dental Materials*, 17:248–288, 2001.

O. M. El Rifai and K. Youcef-Toumi. Design and control of atomic force microscopes. In *Proc. Amer. Ctrl. Conf.*, pages 3714–3719, June 2003.

E. Gross and M. Tomizuka. Experimental beam tip tracking control with a truncated series approximation to uncancelable dynamics. *IEEE Trans. Ctrl. Sys. Tech.*, 2(4):382–391, Dec. 1994.

E. Gross, M. Tomizuka, and W. Messner. Cancellation of discrete time unstable zeros by feedforward control. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 116:33–38, March 1994.

B. Haack and M. Tomizuka. The effect of adding zeros to feedforward controllers. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 113(4):6–10, March 1991.

L. R. Hunt, G. Meyer, and R. Su. Noncausal inverses for linear systems. *IEEE Trans. Auto. Ctrl.*, 41(4):608–611, April 1996.

N. Kodera, H. Yamashita, and T. Ando. Active damping of the scanner for high-speed atomic force microscopy. *Rev. Sci. Instr.*, 76:0537081–5, May 2005.

N. Kodera, H. Yamashita, and T. Ando. Dynamic proportional-integral-differential controller for high-speed atomic force microscopy. *Rev. Sci. Instr.*, 77:0837041–7, Aug. 2006.

K. K. Leang and S. Devasia. Feedback-linearized inverse feedforward for creep, hysteresis, and vibration compensation in AFM piezoactuators. *IEEE Trans. Ctrl. Sys. Tech.*, 15(5):927–935, Sept. 2007.

Y. Li and J. Bechhoefer. Feedforward control of a closed-loop piezoelectric translation stage for atomic force microscope. *Rev. Sci. Instr.*, 78:013702–1–8, Jan. 2007.

L. Y. Pao, J. A. Butterworth, and D. Y. Abramovitch. Combined feedforward/feedback control of atomic force microscopes. In *Proc. Amer. Ctrl. Conf.*, pages 3509–3515, July 2007.

B. Potsaid and J. T. Wen. High performance motion tracking control. In *Proc. IEEE Int. Conf. Ctrl. Apps.*, pages 718–723, Sept. 2004.

B. Potsaid, J. T. Wen, M. Unrath, D. Watt, and M. Alpay. High performance motion tracking control for electronic manufacturing. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 129:767–776, Nov. 2007.

J. M. Rieber, G. Schitter, A. Stemmer, and F. Allgöwer. Experimental application of ℓ_1-optimal control in atomic force microscopy. In *Proc. IFAC World Cong.*, pages 718–723, July 2005.

B. P. Rigney, L. Y. Pao, and D. A. Lawrence. Model inversion architectures for settle time applications with uncertainty. In *Proc. IEEE Conf. Dec. & Ctrl.*, pages 6518–6524, Dec. 2006a.

B. P. Rigney, L. Y. Pao, and D. A. Lawrence. Settle-time performance comparisons of approximate inversion techniques for LTI nonminimum phase systems. In *Proc. Amer. Ctrl. Conf.*, pages 600–605, June 2006b.

B. P. Rigney, L. Y. Pao, and D. A. Lawrence. Nonminimum phase dynamic inversion for settle time applications. *Provisionally accepted for publication in: IEEE Trans. Ctrl. Sys. Tech.*, 2008. Pre-print available here: http://ece.colorado.edu/∼pao/journals.html.

S. M. Salapaka, A. Sebastian, J. P. Cleveland, and M. V. Salapaka. High bandwidth nano-positioner: A robust control approach. *Rev. Sci. Instr.*, 15:3232–3241, Sept. 2002.

S. M. Salapaka, T. De, and A. Sebastian. A robust control based solution to the sample-profile estimation problem in fast atomic force microscopy. *Int. J. Robust & Nonlin. Ctrl.*, 15:821–837, 2005.

G. Schitter, P. Menold, H. F. Knapp, F. R. Allgöwer, and A. Stemmer. High performance feedback for fast scanning atomic force microscopes. *IEEE Tran. Ctrl. Sys. Tech.*, 72(8):3320–3327, Aug. 2001.

G. Schitter, A. Stemmer, and F. Allgöwer. Robust 2DOF-control of a piezoelectric tube scanner for high speed atomic force microscopy. In *Proc. Amer. Ctrl. Conf.*, pages 3720–3725, June 2003.

G. Schitter, R. W. Stark, and A. Stemmer. Identification and open-loop tracking control of a piezoelectric tube scanner for high-speed scanning-probe microscopy. *IEEE Trans. Ctrl. Sys. Tech.*, 12(3): 449–454, May 2004a.

G. Schitter, R. W. Stark, and A. Stemmer. Fast contact-mode atomic force microscopy on biological specimen by model-based control. *Ultramicroscopy*, 100:253–257, 2004b.

G. Schitter, G. E. Fantner, P. J. Thurner, J. Adams, and P. K. Hansma. Design and characterization of a novel scanner for high-speed atomic force microscopy. In *IFAC Symp. on Mechatronic Sys.*, pages 819–824, Sept. 2006.

A. Sebastian and S. M. Salapaka. Design methodologies for robust nano-positioning. *IEEE Trans. Ctrl. Sys. Tech.*, 13:868–876, Nov. 2005.

A. Sebastian, A. Gannepalli, and M. V. Salapaka. A review of the systems approach to the analysis of dynamic-mode atomic force microscopy. *IEEE Trans. Ctrl. Sys. Tech.*, 15(5):952–959, Sept. 2007.

Z. Shao, J. Mou, D. M. Czajkowsky, J. Yang, and J.-Y. Yuan. Biological atomic force microscopy: What is achieved and what is needed. *Adv. Physics*, 45(1):1–86, 1996.

S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley and Sons, Ltd., 2^{nd} edition, 2005.

A. Stemmer, G. Schitter, J. M. Rieber, and F. Allgöwer. Control strategies towards faster quantitative imaging in atomic force microscopy. *Euro. J. Ctrl.*, 11(4–5):384–395, 2005.

S. Tien, Q. Zou, and S. Devasia. Iterative control of dynamics-coupling-caused errors in piezoscanners during high-speed AFM operation. *IEEE Trans. Ctrl. Sys. Tech.*, 13(6):921–931, Nov. 2005.

M. Tomizuka. Zero phase error tracking algorithm for digital control. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 109:65–68, March 1987.

D. E. Torfs, J. De Schutter, and J. Swevers. Extended bandwidth zero phase error tracking control of nonminimal phase systems. *ASME J. Dyn. Sys., Meas., & Ctrl.*, 114:347–351, Sept. 1992.

D. E. Torfs, R. Vuerinckx, J. Swevers, and J. Schoukens. Comparison of two feedforward design methods aiming at accurate trajectory tracking of the end point of a flexible robot arm. *IEEE Trans. Ctrl. Sys. Tech.*, 6(1):2–14, Jan. 1998.

J. T. Wen and B. Potsaid. An experimental study of a high performance motion control system. In *Proc. Amer. Ctrl. Conf.*, pages 5158–5163, June 2004.

Y. Wu and Q. Zou. Iterative control approach to compensate for the hysteresis and the vibrational dynamics effects of piezo actuators. In *Proc. Amer. Ctrl. Conf.*, pages 424–429, June 2006.

Q. Zou and S. Devasia. Preview-based optimal inversion for output tracking: Application to scanning tunneling microscopy. *IEEE Trans. Ctrl. Sys. Tech.*, 12(3):375–386, May 2004.