Control Engineering: Maximizing Performance of the Fourth-Order Motion Setup

Tobias Wejbora [redacted], [redacted] [redacted], Group 67

I. Introduction

From semiconductor lithography to high speed printing, the demand for faster and more accurate motion control is rapidly increasing. This report highlights these challenges by outlining the steps needed in order to maximize performance of a Fourth-Order Motion Setup, showcased in Figure 1. The primary objective is to develop and implement a control system to achieve the fastest possible scanning motion over a 120-radian stroke, while simultaneously maintaining the lowest tracking error possible on the non-collocated side of the motor.

The experiment is limited by multiple constraints. These include a total allowable turn-around stroke of just 2.5 radians, a maximum sampling rate of 4 kHz, and a robustness requirement that the sensitivity modulus margin, $\|S(j\omega)\|$, must always remain below 6 dB. The performance of the final system is then measured based on three distinct regions, each with specific motion time and error requirements:

- **Region I (Minimal Performance):**
  - Motion Time (T_r): 10 s $< T_r \leq$ 12 s
  - RMS Tracking Error: < 6 mrad
  - Peak Tracking Error: < 12 mrad
- **Region II (Improved Performance):**
  - Motion Time (T_r): 10 s $< T_r \leq$ 11 s
  - RMS Tracking Error: < 3 mrad
  - Peak Tracking Error: < 6.3 mrad
- **Region III (Faster Performance):**
  - Motion Time (T_r): $T_r <$ 10 s
  - RMS Tracking Error: < 3 mrad
  - Peak Tracking Error: < 6.3 mrad

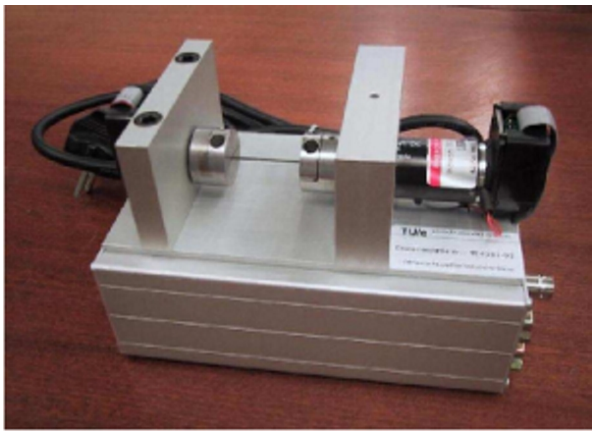

Fig. 1. PATO Setup

II. System Identification

Before a controller can be designed, system identification must be performed to obtain a mathematical model of the plant's dynamics. The setup can be modeled as a two-mass spring-damper system, which captures the dynamic interaction between the motor (actuator) and the end-effector (load).

A. Theoretical Model

In this setup, the two-mass model results in two different transfer functions depending on where the motion is measured.

For the **collocated case**, where the actuator and encoder are on the motor side, the transfer function, denoted as $H_c(s)$, relates the motor position $X_a(s)$ to the applied motor force $F_a(s)$. The model is given in Equation (1):

$$H_c(s) = \frac{m_e s^2 + ds + k}{m_e m_a s^4 + (m_e + m_a) d s^3 + (m_e + m_a) k s^2} \quad (1)$$

In the **non-collocated configuration**, the position is measured at the end-effector instead. The corresponding transfer function, $H_{nc}(s)$, is shown in Equation (2):

$$H_{nc}(s) = \frac{ds + k}{m_e m_a s^4 + (m_e + m_a) d s^3 + (m_e + m_a) k s^2} \quad (2)$$

Here, m_a represents the mass of the actuator and m_e represents the mass of the end-effector, while k and d are the stiffness and damping coefficients of the flexible transmission connecting them [1].

B. Experimental Identification Methodology

To identify the frequency response of the plant in simulation, a Frequency Response Function (FRF) was estimated using a broadband disturbance. A noise signal was injected into the system, which excites a wide range of frequencies at once. This makes it possible to reveal key dynamic behavior, such as the resonance and anti-resonance.

Because the plant may operate under feedback during the identification process, a 3-point measurement setup was used. Three signals were recorded from the Simulink model at the same time: the injected noise disturbance $d(t)$, the plant input $u(t)$ (actuator signal), and the plant output $y(t)$ (sensor position). These signals are then used to compute an FRF that more accurately reflects the true open-loop plant dynamics.

C. *Data Processing and FRF Calculation*

These signals were then processed in MATLAB to estimate the FRF. The power spectral densities were obtained through the use of Welch's method with a Hann window 50% overlap to reduce variance and minimize the spectral leakage.

The plant FRF $H(f)$ was then computed using the indirect estimator in Equation (3):

$$H(f) = \frac{S_{dy}(f)}{S_{du}(f)} \tag{3}$$

$S_{dy}(f)$ represents the cross-power spectral density between the disturbance and the output. $S_{du}(f)$ is the cross-power spectral density between the disturbance and input signal. By utilizing this 3 point approach, the true plant dynamics become separated from any feedback effects.

To verify the estimate, the coherence between the measured input and output signals were checked, and can be seen in Figure 2 and Figure 3. In these coherence graphs, values that are close to one indicate that the response is reliable in those frequency regions.

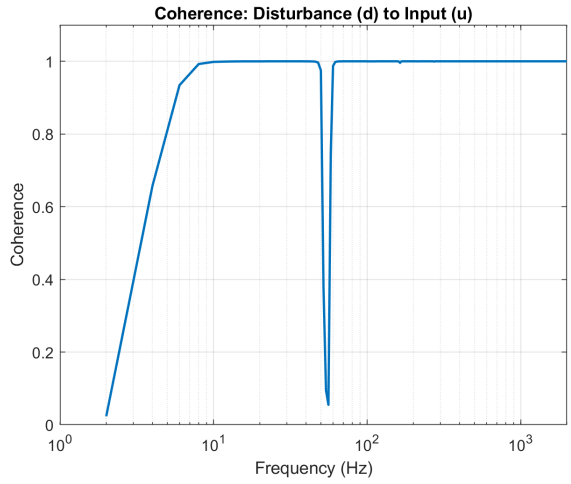

Fig. 2. Coherence of Disturbance to Input

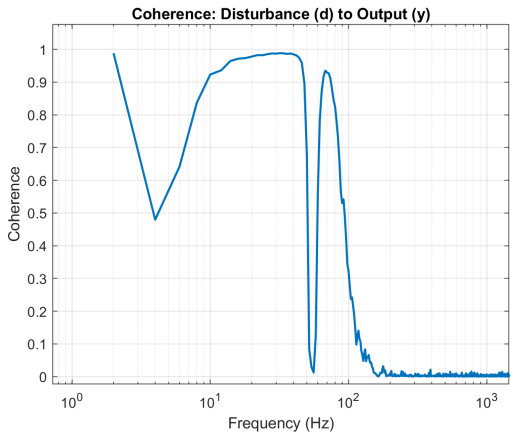

Fig. 3. Coherence of Disturbance to Output

D. *Identified System Response*

The estimated FRFs for both the collocated and non-collocated cases are shown as Bode plots in Figure 4 and Figure 5. These responses form the basis for the controller design in the following section.

III. Controller Design

A. *Initial Controller*

Once FRFs were measured, the load-side encoder was selected to close the feedback loop. This was done because the load side is also where the end-effector is, which results in a simpler and more accurate system. The motor side encoder is blind to quite a few mechanical issues/errors that can happen between the motor encoder and the end effector. The goal is to not make an estimate of the end effector based on the motor's position. This can also be seen in the difference in bode plots between the collocated and non-collocated side(4 and 5) in the higher frequencies as they capture different dynamics. An initial stabilizing controller was made as seen in Table I. The goal here was to make a stable, low-performance controller that can be used as a baseline for tuning feedforward. A low gain was kept not to excite any sensor noise or resonances. A lead filter was

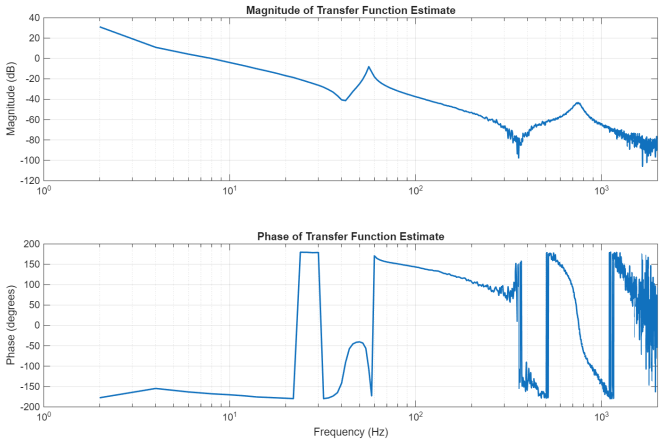

Fig. 4. Bode Plot of the Identified Collocated System ($H_c(s)$).

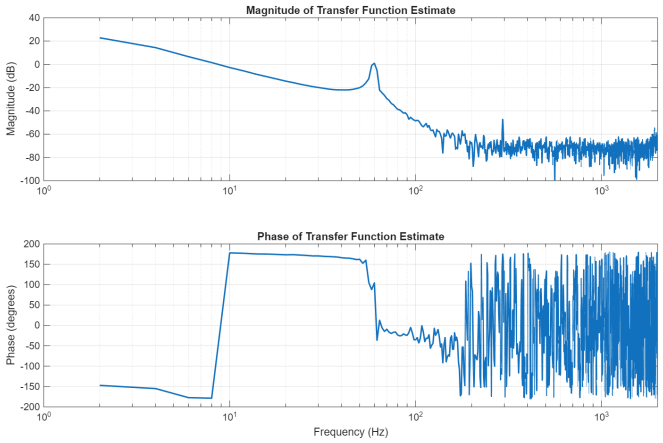

Fig. 5. Bode Plot of the Identified Non-collocated System ($H_{nc}(s)$).

added to counteract the clear phase lag around the crossover frequency(10Hz). Then a 2nd order low-pass filter was added to make sure that the controller gain rolls off sufficiently and thus attenuates high-frequency noise.

TABLE I
PARAMETERS OF THE IMPLEMENTED INITIAL CONTROLLER

Controller	Parameters	Value
Proportional Gain	Gain	0.2
Lead-lag Filter	Zero	1 Hz
	Pole	30 Hz
2nd Order Low-pass	Pole	200 Hz
	Damping pole	0.7

B. *Feedforward Design*

With this low-performance controller, a feedforward scheme and trajectory can be designed. The goal is to have the feedforward do most of the reference tracking so that the feedback controller can reject the disturbances and errors. A challenging 3rd order setpoint was made with approximately one-third for acceleration, one-third for constant velocity, and one-third for deceleration. To observe the effects of inertia and friction, a small pause was added to both the forward and backward movements. To get the shape seen in Figure 6,the following trajectory values were used:

- A 0.25-second pause was added at the beginning, with a start position at 0 rad and a final position at 250 rad. Another 0.25-second pause was added at the end.
- A maximum velocity of 250 rad/s.
- A maximum acceleration of 600 rad/s^2.
- A maximum jerk of 200,000 rad/s^3.

Initially, a focus was kept on a more conservative, slower reference, which did not allow us to see the individual components. Having a high jerk was also overlooked at first, but this allowed for sharp transitions between the acceleration, constant velocity, and deceleration phases. By plotting the position, velocity, and acceleration in addition to the error, feedforward was tuned with the classic procedure:

K_{fc} The Coulomb friction gain was adjusted until the error during the deceleration of the forward movement was equal to the error during the acceleration of the backward movement.

K_{fv} The viscous friction gain was tuned next. This was adjusted to remove as much error during the constant velocity phase as possible.

K_{fa} Lastly, the acceleration feedforward was tuned to get rid of the remaining error peaks during the acceleration and deceleration phases.

By tuning the feedforward, the peak error was effectively reduced from 1.4 rad to 0.1 rad. The final feedforward values can be seen inTable II

TABLE II
FEEDFORWARD GAIN VALUES

Gain Block	Description	Value
Kfc	Coulomb Friction Gain	0.009
Kfv	Viscous Friction Gain	0.000003
Kfa	Acceleration Feedforward	0.00039

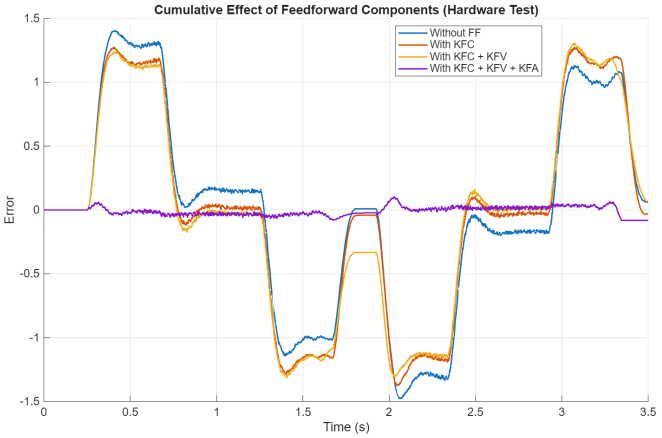

Fig. 6. Effects of Feedforward Components on Tracking Error

C. *Feedback Design*

With feedforward tuned, a higher-performance controller can be created with a higher bandwidth and stiffness. A higher bandwidth is desired to follow the reference trajectory more accurately and respond faster, which is a big limitation of the initial controller used for feedforward.

The general loop shaping procedure was followed when creating a new controller. This procedure was iterated to achieve the final values seen in Table III.

- Define a target cross-over frequency, ω_{co}, typically in the stable region between the system's anti-resonance and first resonance.
- Introduce a lead filter to provide phase margin around the crossover, with a zero placed at approximately $\omega_{co}/3$ and a pole at $3\omega_{co}$.
- Increase the proportional gain until the open-loop magnitude, $|H(j\omega)|$, is equal to 0 dB at the target crossover frequency ω_{co}.
- If required to eliminate steady-state errors, add an integrator with a zero placed well below the crossover (e.g., at $\omega_{co}/5$).
- For high-frequency noise suppression, add a second-order low-pass filter with a natural frequency significantly higher than the crossover (e.g., at $6\omega_{co}$).
- Evaluate the resulting performance in the time domain and the stability margins in the frequency domain, and iteratively re-adjust parameters as needed.

To increase the low-frequency gain, the gain parameter was iteratively increased to 2.5. Two notch filters were added, the first one at the major resonance frequency of 60Hz and the other at 3.45Hz. The 60Hz notch was first implemented with a very low damping zero (0.01) to create a sharp and

deep notch. A high-damping pole (0.7) was chosen to make the phase recover quickly outside of the notch. The notch at 3.45Hz stems from the systems low-frequency resonance caused by the masses and the rod not being ideal. This resonance caused quite a bit of instability when trying to increase the controller gain; thus a notch was placed here. A lead filter with a zero at 14Hz and a pole at 60Hz added the required phase lead at the new, higher crossover frequency. Lastly, a 2nd low-pass Filter rolled off the high-frequency gain.

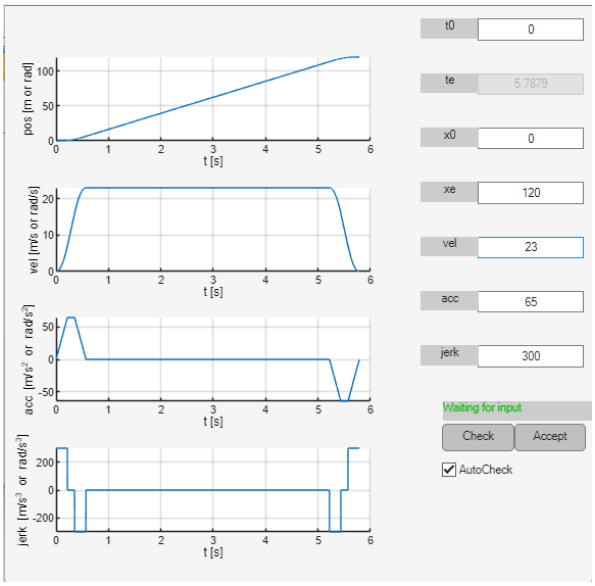

Fig. 7. Final Reference Block Used for Simulink

TABLE III
PARAMETERS OF THE IMPLEMENTED FINAL CONTROLLER

Controller	Parameters	Value
Proportional Gain	Gain	2.5
Notch Filter	Zero: Damping zero: Pole: Damping pole:	60 Hz 0.01 60 Hz 0.7
2nd Order Low-pass	Pole Damping pole:	200 Hz 0.2
Lead-lag Filter	Zero: Pole:	14 Hz 60 Hz
Notch Filter	Zero: Damping zero: Pole: Damping pole:	3.45 Hz 0.1 3.45 Hz 0.001

IV. EXPERIMENTAL RESULTS

With the final feedback and feedforward controllers implemented, the system was then ran with reference trajectories for all three regions, with the RMS error and peak error highlighted in Table IV. Region 1,2,3 trajectories consisted of a scanning time of 11.5 seconds,11 seconds, and 10 seconds, respectively. Only region 1 was fully achieved, where the rms error and peak error is below 6mrad and 12mrad, respectively. The tracking error graph for the region one reference trajectory can be seen in Figure 10. Here it can also clearly be seen that the majority of the error is between $\pm$3mrad, however, multiple peaks can be seen up to $\pm$6.5mrad, and one or two encoder values create the peak error of $\pm$10mrad. For the region 2 and 3 trajectories, very similar error profiles were seen, just amplified. The faster accelerations required for regions 2 and 3 was the culprit for higher peak errors during the acceleration phases of the trajectory.

The PSD of the tracking error can be seen in Figure 8, which shows us how the error's energy is distributed across different frequencies. The main result from this plot is that the most error power is found in the low-frequency range. More specifically, the areas of interest are around 3Hz, 10.5Hz, and 60Hz. These are the regions where further improvements can be made to the controller. The cumulative amplitude spectrum was also plotted as it provides insight into the total of the RMS error as it integrates from low to high frequencies. The graph rises quickest at low frequencies from 1-3.5Hz and 10.5Hz. This further confirms what was seen in the PSD plot.

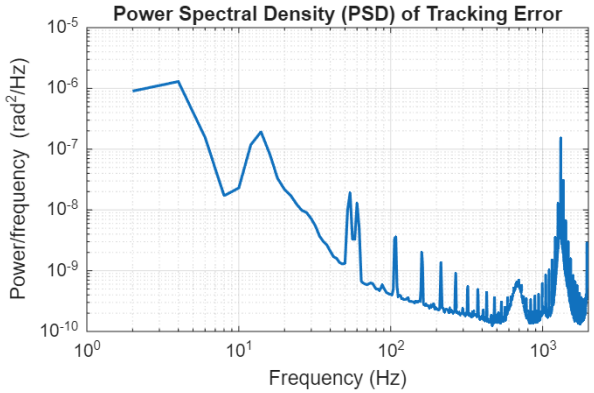

Fig. 8. Power Spectral Density of Tracking Error

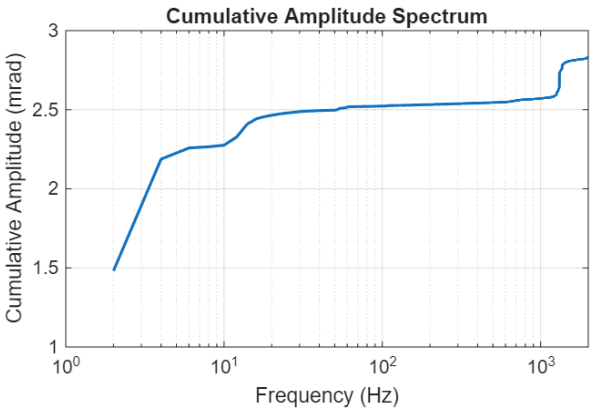

Fig. 9. Cumulative Power Spectral Density Plot

The biggest limitation of Figure 8 and Figure 9 is that these are in the time domain, meaning that the frequencies

causing these errors are known, but not when they occur in the trajectory. However, clear improvements can still be made to the controller. Here one can see that the controller struggles to reject low-frequency disturbances and errors, meaning that it could benefit from an integrator with its zero placed well below the crossover frequency to provide higher gain at lower frequencies. An integrator was experimented with; however, with initial implementations, it seemed like the integrator created substantial phase lag, which caused instability. Furthermore, further fine-tuning of the phase lead action parameters can be beneficial. It would be ideal to shift the zero and the pole to lower frequencies to allow for a higher gain crossover for better tracking performance. However, this also has to be balanced with increasing the separation between the zero and pole so that the phase margin can be increased. Lastly, another improvement is the 2nd-order low-pass filter as the damping pole could be further increased to 0.7 and the frequency could be lowered to around 150Hz. However, this becomes difficult to do as it introduces phase lag at the crossover frequency, hence reducing the phase margin.

The open-loop Bode and Nyquist plots before and after implementing the final controller can be seen in Figure 11 and Figure 12. These plots highlight the major changes made to the system's frequency response. The resonance peak at 60Hz has been successfully eliminated, as well as upping the low frequency gain, and introducing phase lead around the new crossover frequency.

TABLE IV
CONTROLLER PERFORMANCE RESULTS BY REGION

Performance Region	RMS Error (mrad)	Peak Error (mrad)
Region 1	2.252	9.425
Region 2	4.561	12.566
Region 3	5.581	15.708

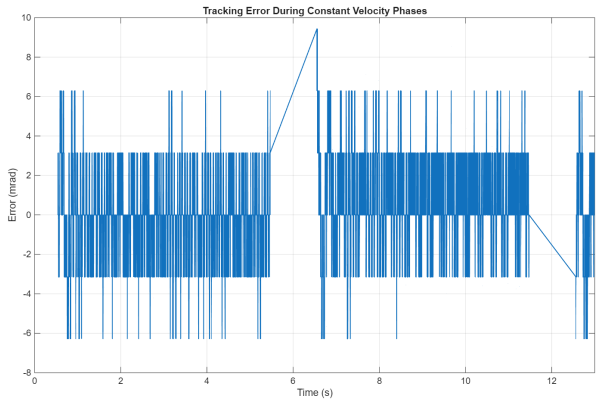

Fig. 10. Region 1 Tracking Error

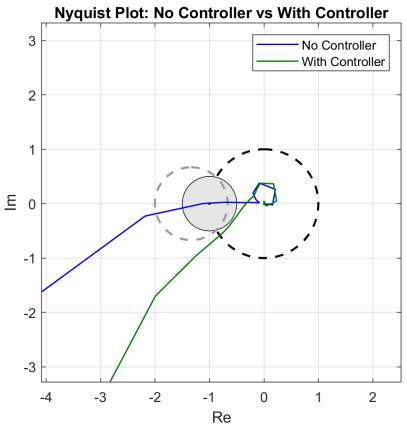

Fig. 11. Nyquist Plot: With & Without a Controller

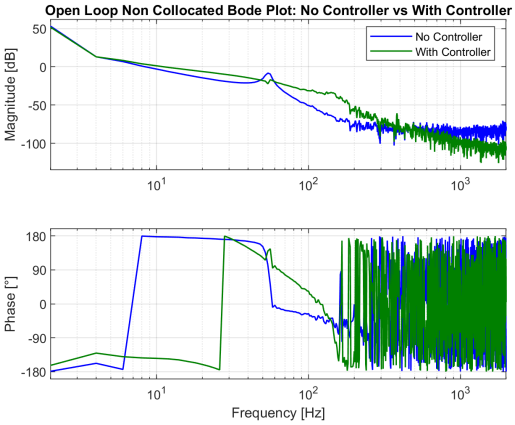

Fig. 12. Open Loop Non Collocated Bode Plot

V. CONCLUSIONS & DISCUSSION

In conclusion, Region 1 was obtained by developing and implementing both a feedforward and feedback controller. The feedback controller made use of two notch filters, a low pass, and a lead filter to eliminate system resonances and inherent phase lag around the crossover frequency. The system's performance is ultimately constrained by the bandwidth of the controller and hence the disturbances at the crossover frequency. A rms error of 2.252 mrad was achieved for region 1, which is well below the required 6 mrad. When attempting to achieve region 2, the faster acceleration and velocity required caused higher peak errors, particularly during the acceleration and deceleration phases of the trajectory. In future steps, feedforward components like K_{fa} would have to be further tuned to account for these peaks. However, beyond this, investigating a jerk feedforward term would smooth the control effort during transitions in acceleration. Input shaping would also be an option as it would pre-filter the reference trajectory of any natural frequency/resonances.

REFERENCES

[1] JPE Innovations. 4th order systems & interpretation, 2025.