Title
Precision Motion Sensing and Control Through Constrained Optimization

Author
Chang, Yen-Chi

Publication Date
2014

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Precision Motion Sensing and Control Through Constrained Optimization

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mechanical Engineering

by

Yen-Chi Chang

2014

Abstract of the Dissertation

Precision Motion Sensing and Control Through Constrained Optimization

by

Yen-Chi Chang

Doctor of Philosophy in Mechanical Engineering

University of California, Los Angeles, 2014

Professor Tsu-Chin Tsao, Chair

The precision and productivity requirements for modern manufacturing process drive precision systems to their physical limitations, such as the limitation of sensor resolution, actuator saturation, and traversing speed. Incorporating these restrictions, constrained optimization has been an active research area that can explicitly handle the physical constraints. This dissertation covers three important topics in productivity and precision improvements through constrained optimization: sub-count sensor estimation, constraint violation avoidance, and cycle-time minimization.

To solve sensor quantization and measurement synchronization problem, a model-based encoder-triggered estimation method was developed addressing state continuity. By enforcing smooth estimation in important physical states such as position and velocity, this method avoids abrupt estimation change when encoder trigger occurs. Furthermore, because continuous-time open-loop model is used to predict inter-trigger behavior, the developed sub-count estimator can be applied to asynchronous systems with untimely measurement updates. By applying this sub-count estimation method, the precision system can improve tracking performance when only quantized and untimely sensor measurement is available.

When improving tracking performance, the precision system is often pushed to its limitation. This often causes the precision system to violate system constraints (such as control signal saturation and excessive acceleration), which results in unexpected vibration and sacrifices the tracking performance. To avoid such kind of constraint violation, the system model can be used to predict the dynamic behavior of the plant given a known trajectory and plant model. A model predictive control based optimal feed-forward tracking method is proposed to be integrable with any feedback controller.

Besides tracking performance, cycle time is also an important index for modern precision motion control. The shorter the cycle time is, the more productive the machine can be. Therefore, to reduce cycle time for precision machines, a minimal time contour tracking problem is formulated to explicitly constrained receding horizon control where the trajectory (feed) profile is determined for the specified contour under system dynamics, contour error, axial velocity, signal saturation, and monotonic feed constraints. Because minimal-time operations typically drive the system near its constraint boundary, it is vital to reject measurement noise and compensate for modeling error in real-time. To accommodate the high sampling rate controllers in precision systems, an efficient real-time quadratic programming solver is proposed with robust warm start strategy in order to handle active constraints.

The dissertation of Yen-Chi Chang is approved.

Lieven Vandenberghe

Robert M'Closkey

Steve Gibson

Tsu-Chin Tsao, Committee Chair

University of California, Los Angeles

2014

Table of Contents

1 Introduction **1**

1.1 Resolution and Sensor Timing 1

1.2 Constraint Violation Avoidance 3

1.3 Time Optimal Control 4

1.4 Main Contributions of this Dissertation 7

2 Sub-count Estimation with Continuity Restoration Constraints **9**

2.1 Measurement Quantization 9

2.2 Event-triggered Correction 12

2.3 State Propagation and Observer Design 13

2.4 Intermittent Measurement and Continuity constraint 14

2.5 Optimization Formulation 15

2.6 System and Controls of the Plasmonic Imaging Lithography Machine(PILM) 19

2.7 Experiments of sub-count Estimation on PILM 21

3 Feedforward MPC with Realistic Constraints **34**

3.1 Fast Tool Servo and Model Predictive Control 34

3.2 Convergence of the Feed-forward model predictive control 41

3.3 Simulation Results for a Challenging Piston Profile 43

4 Constrained Time-optimal Control **51**

4.1 Spline Contour and Contour Error 51

4.2 Using Constrained Model Predictive Control For Minimum-time Contour Tracking 52
4.3 Convex Objective Function 54
4.4 System Dynamics 55
4.5 System Dynamics and Constraints 57
4.6 Optimization Problem Formulation 60
4.7 Real-time Robustness and Efficiency 64
4.8 Experimental Results on a Nano-precision Multi-axis Positioning Stage(MAPS) 69

5 Conclusions **77**

References **79**

List of Figures

2.1 Quadrature encoder patterns 10

2.2 Quadrature encoder triggers for the forward(F) and Backward(B) movement 11

2.3 Measurement Quantization 12

2.4 Block diagram of the contour tracking system 13

2.5 Plasmonic Imaging Lithography Machine 20

2.6 Schematic view of Plasmonic Imaging Lithography Machine . . . 21

2.7 Estimation results for non-model-based approaches 23

2.8 Estimation results for model-based approaches 24

2.9 Estimation results for triangular reference tracking 26

2.10 Triangular reference signal 27

2.11 Performance of position estimation 28

2.12 Performance of velocity estimation 30

2.13 Validated and estimated position with quantized feedback. 32

2.14 Validated and estimated position with estimation feedback. 32

2.15 Effect of quantization level on tracking error. 33

2.16 Effect of quantization level on velocity error. 33

3.1 Block diagram of Two Parameter Robust Repetitive Control . . . 35

3.2 Block diagram of TPRRC with Feedforward model predictive control 36

3.3 Illustration of the horizon length 43

3.4 Piston profile in time domain 44

3.5 Bode plot of the continuous time transfer function of the linear motor 45

3.6 Bode plot of the continuous time transfer function of the piezo actuator 45
3.7 Feedforward control signals for the primary(mm) and secondary stage(μm) 46
3.8 Tracking error(mm) 47
3.9 Control signal constraints(V) 48
3.10 Pzt stage constraints-displacement(μm) 49
3.11 Pzt stage constraints-current(A) 49
3.12 Linear motor stage constraints-acceleration(m/sec^2) 50
3.13 linear motor stage constraints-velocity(m/sec) 50

4.1 Block diagram of the contour tracking system 52
4.2 Schematic diagram of the contour projection 58
4.3 Nodal point propagation 60
4.4 Illustration of the warm start technique with residual regulation . 67
4.5 Multi-Scale Alignment and Positioning System 70
4.6 Impulse response of MAPS 70
4.7 Trajectory Profile r for Diamond-shape Contour Tracking 72
4.8 Measured Position and Boundary 72
4.9 Simulated(black) and measured(red) contour error and axial velocity when $f_{max} = 0.85um/sec$ 73
4.10 Active Constraints from simulation(black) and experiment(red) . 74
4.11 Simulated(black) and measured(red) contour error and axial velocity when $f_{max} = 0.425um/sec$ 75

4.12 Trajectory profile for $f_{max} = 0.85um/sec$ (red-dotted) and $f_{max} = 0.425um/sec$ (blue) 75

4.13 Horizon length effect on contour error and cycle time 76

List of Tables

2.1 Encoder pattern 10

2.2 PID Gains of PILM 22

2.3 RMS Error for Different Order of Continuity. 23

4.1 Model predictive control solver efficiency 69

4.2 Boundary parameters 71

Acknowledgments

I would like to acknowledge my debt to my advisor Prof. Tsu-Chin Tsao for all his guidance and support during my studies at UCLA. His dedication to his work and students will always be a model for me. During my studies at UCLA, I had the honor of making friends with many great individuals. Finally, I would like to thank my mother, my father, my brother, my sister, and my girlfriend, to whom this thesis is dedicated, for their endless love and encouragements.

Vita

2007	B.A. (Mechanical Engineering), National Taiwan University.
2008-2009	Mechanical Engineer, HTC corp.
2011–2013	Teaching Assistant, MAE Department, UCLA.
2012	M.S. (Mechanical Engineering), UCLA, Los Angeles, CA.
2013–2014	Research Assistant, MAE Department, UCLA.
2013-2014	Algorithm Engineer Intern, DYSCI.

Publications

Y.C. Chang, B.N. Berry-Pusey, R. Yasin, N. Vu, B. Maraglia, T. Tsao, and A. X. Chatziioannou, An Automated Mouse Tail Vascular Access System by Vision and Pressure Feedback, Mechatronics

Y.C. Chang and T.C. Tsao, Minimum-Time Contour Tracking with Model Predictive Control Approach, ACC, Portland, Oregon, 2014

Y.C. Chang, B.N. Berry-Pusey, T.C. Tsao, and A. X. Chatziioannou, Real-Time Image Processing for Locating Veins in Mouse Tails, DSCCMOVIC, Palo Alto, California, 2013

Y.C. Chang and T.C. Tsao, Feedfoward Model Predictive Control for Dual-Stage Fast Tool Servo with Realistic Constraints, DSCCMOVIC, Fort Lauderdale, Florida, 2012.

Y.C. Chang and T.T. Fu, "A Useful Tire Model for ATV Ride Performance on Rough Roads," SAE Technical Paper , 2010, doi:10.4271/2010-01-1922.

CHAPTER 1

Introduction

1.1 Resolution and Sensor Timing

Because of the continuous drive for high precision manufacturing, precision motion control for automation systems remains an important research area. However, the performance of precision motion control is often limited by the sensor resolution and allowable frequency of measurement update. To overcome these limitations, sub-count estimation methods have been developed to provide better estimation compared to the quantized sensor measurement, which can be generalized into two major categories: curve-fitting and model-based correction methods.

Curve-fitting sub-count estimation methods, in general, use a finite amount of past measurements to predict the upcoming position before new update arrives. Because the curve-fitting methods avoid modeling error, they are suitable for applications where an accurate model is unavailable. Linear extrapolation has been widely used for its simplicity, and has been extended to high-dimensional Spline fitting with longer past measurement horizon [1, 2]. However, to provide adequately smooth estimation, this approach requires long measurement horizon. Because the performance is sacrificed when there is abrupt change in the reference command or system states, the curve-fitting approach is restricted to the case of smooth trajectories.

To avoid the reliance on long horizon of past measurements and to achieve faster response, model-based sub-count estimation methods use a plant model

to predict the system dynamics. Discrete-time Kalman filtering is a typical approach for digitally controlled system with timely sensor update, and the sensor quantization problem is addressed by Gaussian approximation[3, 4]. However, this fixed sampling rate approach is not suitable for some commonly used sensors such as optical encoders with high quantization level. Because this type of sensors are updated by spatially distributed interrupts and remain constant between triggers, the zero-order-hold effect strongly influences Kalman filter's state estimation performance. To solve this problem, event-triggered method was developed by updating the state variables with equality constraint on position when encoder updates because such measurement is only subject to sensor noise without quantization effect at encoder triggers [5, 6, 7, 8]. The disadvantage to this method is that the event-triggered correction introduces abrupt state changes that often causes instability.

Because physical systems typically have smooth propagation in properties such as position and velocity, the continuity in such properties are vital to performing smooth and stable sub-count estimation especially with asynchronous measurement updates. Therefore, a model-based sub-count estimation method has been developed to include high-order continuity constraints on these properties. Because the values of these properties are not guaranteed available through position measurement and numerical differentiation does not provide good estimate, estimation methods have been developed for smooth estimation [9, 10]. To achieve real-time implementation, this dissertation also developed a model-based spline structure that allows efficient real-time implementation of the event-triggered sub-count estimation.

1.2 Constraint Violation Avoidance

Conventional circular machining is performed by cutting a spinning stock material with constant feed. Among many examples, combustion engine cam shafts and pistons are made by this non-circular machining process. The engine performance in power output, fuel economy, and emission depends on how these non-circular bearing surfaces conform to the design specifications. To machine non-circular cross-sections or axis asymmetric surfaces, the feed of the machining tool tip needs to be actuated corresponding to the spindle angle. This non-circular turning process is enabled by fast acting actuators also known as fast tool servos and precise control of the tool motion to track the desired trajectories. Therefore, precise tracking control of the fast tool servos for non-circular machining is critical to making modern advanced high performance engines.

In an engine piston turning operation, single-stage fast tool servo is driven by systems such as electro-magnetic or electro-hydraulic actuators to machine oval-like cross-sectional surfaces. As the piston profile becomes more complex and the work piece rotational speed becomes higher for productivity improvements, the cutting tool's trajectories may go beyond the fast tool servo's dynamic capability. To solve this problem, dual-stage actuation cascades a short stroke high bandwidth actuator such as a piezoelectric actuator to the existing longer stroke but lower bandwidth single-axis fast tool servo [29]. In there, the controller is designed through a sequence of two single-input-single-output (SISO) designs by exploiting the triangular structure of the two by two actuator system dynamics. The tracking error from the first stage actuator is used as reference for the second stage. In this master-slave control arrangement, the overall sensitivity function is the product of two sensitivity functions from each actuator's servo loop. Thus, performance is achieved by the sensitivity reduction of each loop.

The reference trajectories in non-circular machining have strong periodic com-

ponents coupled with the work piece rotation. Because a linear control could demand beyond linear ranges or limitations such as actuator's stroke, velocity, acceleration and drive's voltage and current saturation could be triggered and cause the linear control performance to degrade or even become unstable. Model predictive control (MPC) or similar optimal control approach, which explicitly takes these constraints into account, is useful for addressing this issue because it is not practical for real-time implementation at kHz sampling frequency region for electromechanical systems as the computation power is insufficient to solve constrained optimization in real-time. However, since the plant dynamics are most likely time invariant and quite accurate model can be derived by system identification, it is plausible to apply the MPC approach to off-line calculate constrained optimal feed-forward control actions and cascade it with the closed-loop controller to achieve high tracking performance without violating the actuator constraints. This is even more desirable for multi-axis systems like the dual-stage fast tool servos, where the constrained optimization for feed-forward control can determine how each actuator's dynamics should be utilized without violating constraints. Ideally, the optimization horizon should contain the entire trajectory. However, this would be impractical and unnecessary since a receding horizon would suffice to contain actuator's dynamics. Thus receding horizon constrained optimization that minimizes the tracking error is most plausible and it can be readily formulated and solved as a model predictive control problem and applied as feed-forward control action.

1.3 Time Optimal Control

Fast and accurate continuous trajectory generation is a fundamental function required for Computer Numerical Controlled(CNC) motion in modern fabrication process such as metal cutting, additive manufacturing, and nano-lithographic

manufacturing. Since mechanical motion's contour error directly relates to the part's geometric tolerance, contour tracking control, among various motion control aspects, has received continual attention and remained an active research area. Research topics have been focused on multi-axis cross coupling compensation, constraint violation, and feed-rate design. By incorporating these important aspects into minimal time contour tracking, this dissertation has developed a robust and efficient model predictive control method for feed-rate optimization on multi-input-multi-output systems (MIMO) with real-time modeling error compensation.

When a multi-axis system tracks a defined tool path, contour error is introduced as the radial position error with respect to the desired tool path contour. To improve the manufacturing precision, contour error is desired to be minimized. Because contour error is affected jointly by the system dynamics of each individual axis, axial tracking control does not provide sufficient contour tracking performance due to the coupling effect and the differences in system dynamics for each axis. To improve precision, the coupling effect was addressed by the development of cross-coupling control (CCC) [42]. By estimating contour error based on axial tracking errors, CCC generates a cross-coupling feedback signal for each individual axis to compensate for the coupling between axes.

To penalize the contour tracking error more explicitly, an optimal control scheme was introduced formulating the cross-coupling controller as a linear quadratic regulator (LQR) problem [43]. Because the LQR structure requires the weighting matrix to be time invariant, this method is only optimal for straight line tool paths. For more complex contours, the coordinate transform method was developed estimating the contour tracking error by projecting axial tracking error on the tangent of trajectory [44]. To reduce overshoot at sharp corner, a preview of contour error needs to be considered. In many applications where a preview of future behavior is important, Model Predictive Control (MPC) has been proven

a successful control method for explicit objective minimization and constraint handling [45, 46, 47, 48].

Employing model-based prediction of contour error in a finite preview, contour tracking was formulated as a MPC problem that penalizes a weighted sum of contour error, axial errors, and control signals [49]. In order to prevent the system from exceeding physical limitations, this method was improved by the constrained MPC structure [50]. This approach generates the feed-rate profile that minimizes contour error in a moving predictive horizon subject to physical constraints. While these methods optimize the contour tracking performance for a given time-domain trajectory profile, incorporating trajectory profile design for a given contour is beneficial especially at large contour error regions. A two-layer structure was proposed to adopt an additional layer of heuristic feed-rate index to the model predictive control problem [51, 52]. This structure allows feed-rate to be adjusted with respect to the predicted contour error.

In contour tracking development, explicit cycle time minimization is an important topic because it directly affects the system's productivity. The time consumption per cycle is decided by the tool-path tracking feed-rate, which is primarily determined by the machine's capability and tolerance requirement. Since tool paths are typically described by Spline functions due to differential continuity and jerk limitation [53], a feed-rate optimization method was developed for Spline contours under physical constraints for each axis [54]. This method uses geometric curvature of the Spline contour and continuous time LTI model to estimate velocity, acceleration, and jerk within each spline segment. By minimizing the sum of time consumption for each segment, it is able to generate off-line a minimal-time feed for the whole contour. When it is too computationally intensive to include the full trajectory in the optimization problem, this method proposed generating the trajectory profile with respect to only the nearby Spline contour segments.

Because time optimal control tends to drive the system near its physical limi-

tations, The performance of off-line methods is often sacrificed by modeling error and measurement noise. Real-time constrained model predictive control was proposed to handle such uncertainties: by updating the constrained optimization results on-line, it can adjust the feed-rate according to the measurement. Because the computational efficiency of real-time constrained optimization is often challenged by non-linear objective function and the use of continuous-time model, the problem is often discretized especially for digitally controlled systems. Because straightforward linearization of the time index contains the reciprocal of feed-rate, it is challenged by numerical stability at low speed [55]. Therefore, it is important to develop a numerically robust and efficient convex optimization structure in order to adjust for modeling error and measurement noise for time-optimal contour tracking.

Aiming at performing minimal time control for digitally controlled systems with robust real-time active constraint handling, an efficient constrained model predictive control algorithm is developed to minimize a time duration index subject to system dynamics, monotonic feed, contour error, signal saturation, and axial velocity constraints.

1.4 Main Contributions of this Dissertation

To summarize the main contributions of this dissertation a list is given below:

- Experimental setup and integration of a nano-precision plasmonic imaging lithography machine (PILM). A dual linear motor system controls the coarse motion with optical encoder feedback, and piezo-actuated fine resolution stage is controlled with interferometer feedback. Machine is able to perform complete lithography scans with respect to the wafer spindle.

- Matlab simulation tools and Real-time optimization system integration for

nano-scale sub-count sensor estimation, constraint handling, and minimal-time control. The developed LabVIEW Modules include complete graphical user interface and communication protocol between the host computer and the real-time target, and the optimization module compatible with any LabVIEW real-time target machines.

- Robust and efficient Quadratic Programming solver that allows real-time near-boundary estimation. Primal-dual quadratic programming solver has been widely used for its fast convergence. To improve its computational speed, an efficient matrix inversion method is developed to allow for high sampling rate calculation. On the other hand, to apply warm-start method when the system is operated near its constraint boundary, a robust warm-start method has also been developed.

CHAPTER 2

Sub-count Estimation with Continuity Restoration Constraints

2.1 Measurement Quantization

Sensors play an important role in modern precision motion control, in which optical encoders are widely used for fine position measurement due to the reliability and repeatability. Encoder measurement is composed of a light source and photo detector array that encode the optical pattern in channel A and B. The pattern reflects the relative movement between the detector and scale. As shown in Figure 2.1, the high voltage level is recognized as "true" and the low voltage level is recognized as "false". The change from true to false and false to true indicates the incremental movement of the system. Because the forward and backward movement generates different binary patterns as shown in Table 2.1, the phase difference between channel A and B can be used to indicate the direction of movement.

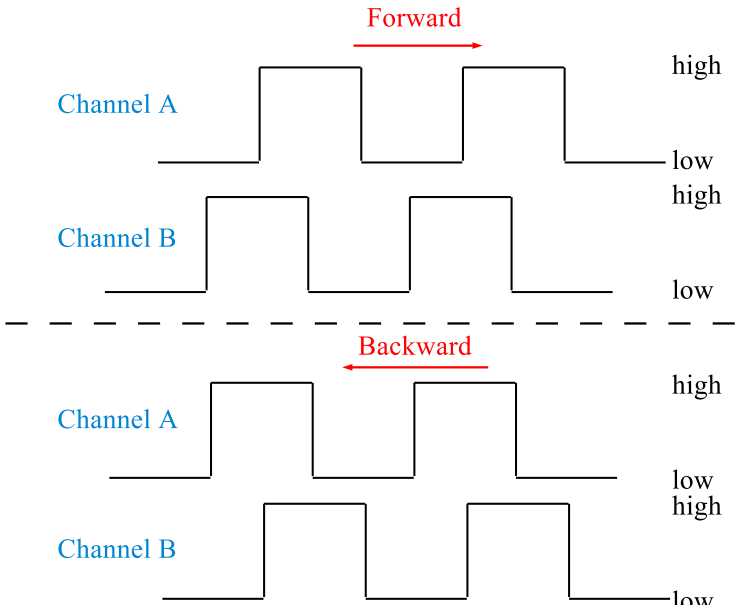


Figure 2.1: Quadrature encoder patterns

Table 2.1: Encoder pattern

Forward Direction		
Phase	**Channel A**	**Channel B**
1	0	0
2	0	1
3	1	1
4	1	0
Backward Direction		
Phase	**Channel A**	**Channel B**
1	1	0
2	1	1
3	0	1
4	0	0

Figure 2.2 shows that the measurement is updated when it detects a transition in encoder pattern, which is typically called an encoder interrupt. Before a new interrupt arrives, the digital controller uses the latest encoder output to generate the feedback signal at fixed sampling rate.

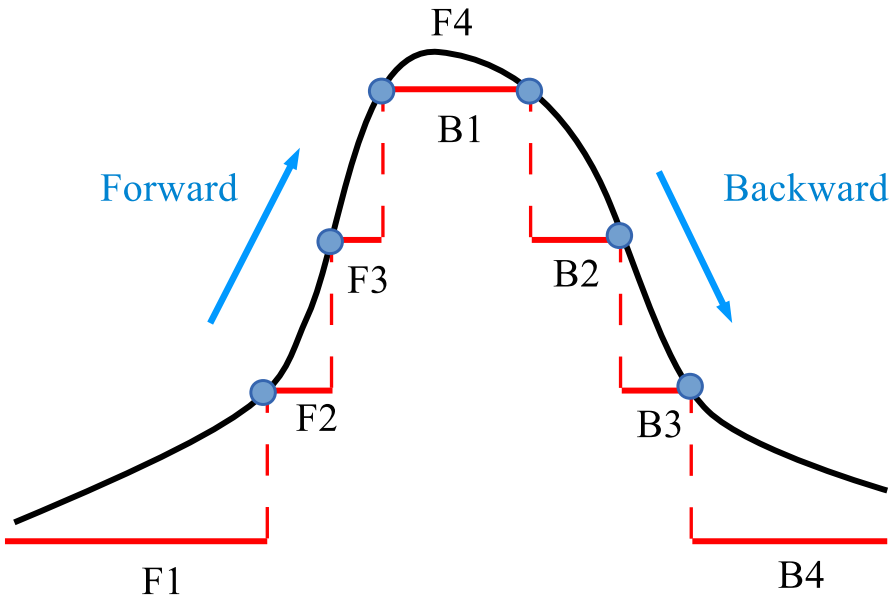


Figure 2.2: Quadrature encoder triggers for the forward(F) and Backward(B) movement

As shown in Figure 2.3, encoder interrupts have fixed spatial resolution and are asynchronous with the controller sample time. At the controller sample time, quantization and measurement noise jointly affect the error distribution. To improve sensor performance, estimation models were proposed as the fixed sampling rate and event-triggered methods. The fixed sampling rate method handles the quantization effect by using Gaussian approximation at each sample time [3, 4]. However, the computation cost of this approach prevents this method from real-time implementation. Furthermore, this approximation method is only restricted to low quantization level cases [14] and can easily diverge on unstable plants.

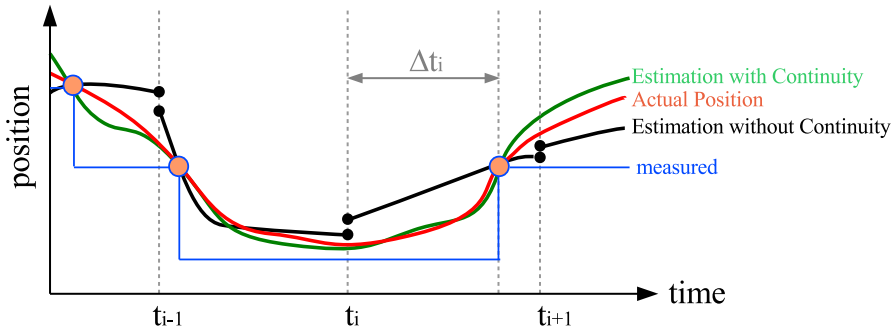


Figure 2.3: Measurement Quantization

Event-triggered method is developed to handle the high quantization cases. At each encoder interrupt, the encoder measurement is only subject to sensor noise without any quantization effect at each encoder trigger. The position measurement when encoder interrupt is triggered is shown in Equation 2.1. By enforcing the system to satisfy the position constraint at each encoder trigger, the quantization problem in fixed sampling time can be avoided [6].

$$y_{t_i+\Delta t_i} = \begin{cases} y^-_{t_i+\Delta t_i} & , if\ y^-_{t_i+\Delta t_i} \geq y^+_{t_i+\Delta t_i} \\ y^+_{t_i+\Delta t_i} & , otherwise \end{cases} \tag{2.1}$$

2.2 Event-triggered Correction

To implement the event-triggered correction with the plant model, system states are corrected when an encoder trigger occurs. As shown in Figure 2.4, between encoder updates the corrector is in the propagation mode that propagates from the initial condition corresponding to the most recent measurement. When an encoder interrupt occurs, the correcter is switched to the measurement update mode to adjust the predicted state values with respect to the sensor update.

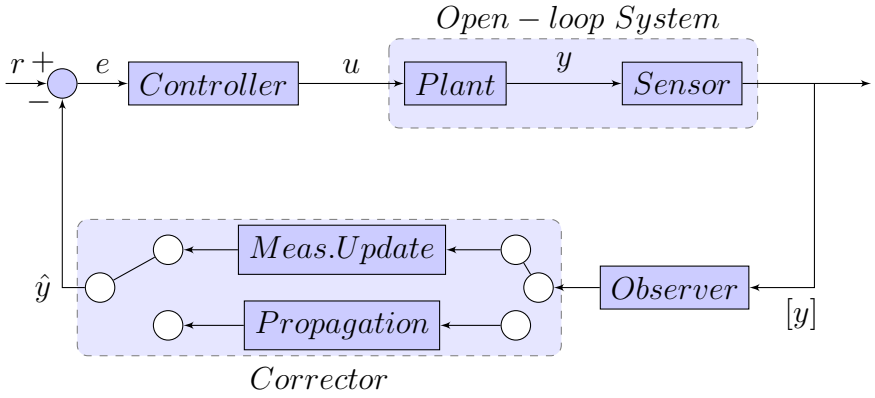


Figure 2.4: Block diagram of the contour tracking system

2.3 State Propagation and Observer Design

To allow for inter-sample prediction, continuous-time linear time-invariant model is used as shown in Equation 2.2. Initialized from a given initial condition $\bar{x}_{t_i}$, the propagation of state variables can be predicted by the open-loop plant and input u is updated at fixed sampling rate.

$$\bar{x}_{t_i+\Delta t_i} = e^{A\Delta t_i}\bar{x}_{t_i} + \int_{t_i}^{t_i+\Delta t_i} e^{A\tau}Bu(t_i-\tau)d\tau \tag{2.2}$$

For an open-loop unstable plant, the state propagation can diverge and therefore challenge the performance of the event-based state correction [6]. For fixed-sampling rate systems, Kalman filtering is widely used to retain stable system estimation with the advantage of error covariance minimization [17]:

$$\hat{x}_{t_{i+1}} = \bar{x}_{t_{i+1}} + K(y_{t_{i+1}} - C\bar{x}_{i+1}) \tag{2.3}$$

where

$$
\begin{aligned}
t_{i+1} &= t_i + \Delta t_i \\
K &= MC^T(CMC^T + R)^{-1} \\
M &= E[(x_{t_i+\Delta t_i} - \bar{x}_{t_i+\Delta t_i})(x_{t_i+\Delta t_i} - \bar{x}_{t_i+\Delta t_i})^T]
\end{aligned}
$$

Because encoder measurement does not provide sensor update at every sampling time, traditional Kalman filtering does not provide good results when quantization effect is dominant. To improve the state estimation performance for open-loop unstable plant with asynchronous update, a coarse observer was introduced to maintain bounded state estimation [7, 8]:

$$
\begin{aligned}
t_i^* &= \begin{cases} t_i + \Delta t_i, \ if \ 0 \leq \Delta t_i \leq |t_{i+1} - t_i| \\ t_i, \ else \end{cases} \\
\bar{x}_{t_{i+1}} &= e^{A(t_{i+1}-t_i^*)}\hat{x}_{t_i^*} + \int_{t_i^*}^{t_{i+1}} e^{A\tau}Bu(t_i - \tau)d\tau \\
\hat{x}_{t_{i+1}} &= \begin{cases} \bar{x}_{t_{i+1}}, \ if \ [y_{t_{i+1}}] = [\hat{y}_{t_{i+1}}] \\ \bar{x}_{t_{i+1}} + L(y_{t_{i+1}} - \hat{y}_{t_{i+1}}), \ else \end{cases}
\end{aligned}
$$

This observer formulation is equivalent to free state propagation if the predicted state variables is bounded by the measurement resolution, and the observer is active when the states exceed the measurement resolution. When the encoder is triggered, the state variables $\hat{x}_{t_i^*}$ are corrected and used as the new initial condition.

2.4 Intermittent Measurement and Continuity constraint

The intermittent state variable correction must satisfy the equality constraint shown in Equation 2.4.

$$C\hat{x}_{t_i+\Delta t_i} = y_{t_i+\Delta t_i}, if\ 0 \leq \Delta t_i \leq |t_{i+1} - t_i| \tag{2.4}$$

Because this abrupt intermittent prediction change sometimes cause estimation to diverge, continuity constraint condition is implemented by the following equality conditions:

$$\begin{aligned} C\hat{x}_{t_i} &= \bar{y}_{t_i} \\ C\dot{\hat{x}}_{t_i} &= \bar{v}_{t_i} \\ &\vdots \\ C\frac{d^n\hat{x}_{t_i}}{dt^n} &= \frac{d^n\bar{y}_{t_i}}{dt^n} \end{aligned}$$

When the corrected states satisfy these conditions, the correction maintains smooth propagation of important physical properties such as position and velocity. Because the encoder interrupt can occur anytime between sampling points, the propagation of the continuous-time model needs to be performed in real-time with respect to the encoder interrupt time stamp. Therefore, the computational efficiency is important in order to avoid any delay. The Taylor series expansion allows this algorithm to reduce the calculation of matrix exponential from $O(n^3)$ to $O(n^2)$:

$$\bar{x}_{t_i+\Delta t_i} \approx \sum_{j=0}^{l} A^j \hat{x}_i \frac{\Delta t_i^j}{j!} + \sum_{j=1}^{l} A^{j-1} B \frac{\Delta t_i^j}{j!} u_i \tag{2.5}$$

2.5 Optimization Formulation

Because error covariance is used as the indicator of prediction accuracy for linear estimators [17], it is applied as the objective function to formulate the optimization problem for sub-count continuity constraint. When the encoder interrupt is

triggered, the optimal state estimation is derived by minimizing the error covariance while satisfying both position and continuity constraints:

$$\begin{aligned}
\min P &= E[(x_{t_i+\Delta t_i} - \hat{x}_{t_i+\Delta t_i})(x_{t_i+\Delta t_i} - \hat{x}_{t_i+\Delta t_i})^T] \\
s.t. C\hat{x}_{t_i+\Delta t_i} &= y_{t_i+\Delta t_i} \\
C\hat{x}_{t_i} &= \bar{y}_{t_i} \\
C\dot{\hat{x}}_{t_i} &= \bar{v}_{t_i} \\
&\vdots \\
C\frac{d^n\hat{x}_{t_i}}{dt^n} &= \frac{d^n\bar{y}_{t_i}}{dt^n}
\end{aligned}$$

The proposed encoder-triggered estimation with continuity constraints can also be expanded to multi-trigger correction as shown in Equation 2.6. The multi-trigger correction can be seen as a model-based curve-fitting method up to r triggers, allowing the continuity constraint to be more conservative. The multi-trigger correction is beneficial when accurate model is unavailable or the system is subject to external disturbance.

$$C\hat{x}_{t_{i+k}+\Delta t_{i+k}} = y_{t_{i+k}+\Delta t_{i+k}}, for\ 0 \leq k \leq r \tag{2.6}$$

The solution to the optimization problem is shown in Equation 2.7 when $R = 0$, where K is the Kalman gain and M is the error covariance of propagated state [11]:

$$\hat{x}_{t_i+\Delta t_i} = \bar{x}_{t_i+\Delta t_i} + K(b_e - A_e\bar{x}_{t_i+\Delta t_i}) \tag{2.7}$$

where

$$K = MA_e^T(A_eMA_e^T + R)^{-1}$$

$$M = E[(x_{t_i+\Delta t_i} - \bar{x}_{t_i+\Delta t_i})(x_{t_i+\Delta t_i} - \bar{x}_{t_i+\Delta t_i})^T]$$

$$A_e = \begin{bmatrix} C \\ \sum_{j=0}^{m} CA^j \frac{(-\Delta t_i)^j}{j!} \\ CA\sum_{j=0}^{m} CA^j \frac{(-\Delta t_i)^j}{j!} \\ \vdots \end{bmatrix}, b_e = \begin{bmatrix} y_{t_i+\Delta t_i} \\ \bar{y}_{t_i} \\ \bar{v}_{t_i} \\ \vdots \end{bmatrix}$$

For a fixed sampling time system without continuity constraint, A_e is the measurement matrix C and Δt_i is $|t_{i+1} - t_i|$. In this case equation 2.7 represents the standard discrete-time Kalman filter. The solution to this optimization problem can be seen as an expansion from the standard Kalman filter to include higher order states, such as velocity and acceleration:

$$\begin{aligned} C\hat{x}_{t_i} &= \bar{y}_{t_i} = C\bar{x}_i \\ C\dot{\hat{x}}_{t_i} &= \bar{v}_{t_i} = C\dot{\bar{x}}_i \\ &\vdots \\ C\frac{d^n\hat{x}_{t_i}}{dt^n} &= \frac{d^n\bar{y}_{t_i}}{dt^n} = C\frac{d^n\bar{x}_i}{dt^n} \end{aligned}$$

When R is 0, the continuity constraints rely 100% on the propagated values. However, because the propagated values do not always provide the best continuity constraints due to modeling error and disturbances, non-zero error covariance R is added to improve the estimation robustness. The following derivation shows how non-zero R affects the continuity constraint. By using the matrix inversion lemma, R can be factored out of the matrix inversion, and results in a correction term for the estimated parameters:

$$(A_e M A_e^T + R)^{-1} = (A_e M A_e^T)^{-1} + R_s \tag{2.8}$$

where

$$\begin{aligned} R_s &= (A_e M A_e^T)^{-1}(-R^{-1} - (A_e M A_e^T)^{-1})^{-1}(A_e M A_e^T)^{-1} \\ &= (A_e M A_e^T)^{-1} r_s \end{aligned}$$

By applying Equation 2.8 to Equation 2.7, the non-zero R is shown to play the role of a correction term for both the measured and unmeasured equality constraints in b_e. When the error covariance matrix M is increased, R is weighted more when generating the offset r_s:

$$\hat{x}_{t_i+\Delta t_i} = \bar{x}_t + M A_e^T (A_e M A_e^T)^{-1}(\hat{b}_e - A_e \bar{x}_t)$$

where

$$\hat{b}_e = b_e + r_s$$

To summarize, the developed model-based sub-count estimation with continuity constraint can be implemented by using the following steps:

Model-based Sub-count Estimation with Continuity Constraint

Set initial values $\bar{x}_{t_0}$

Set i=0

Repeat until $t = t_{max}$

Record the encoder trigger value $y_{t_i+\Delta t_i}$ and time stamp $t_i + \Delta t_i$

if $0 \leq \Delta t_i \leq |t_{i+1} - t_i|$

Calculate A_e and b_e

$\hat{x}_{t_i+\Delta t_i} = \bar{x}_{t_i+\Delta t_i} + K(b_e - A_e \bar{x}_{t_i+\Delta t_i})$

$\hat{x}_{t_i^*} = \hat{x}_{t_i+\Delta t_i}$

else

$\hat{x}_{t_i^*} = \hat{x}_{t_i}$

$\bar{x}_{t_{i+1}} = e^{A(t_{i+1}-t_i^*)}\hat{x}_{t_i^*} + \int_{t_i^*}^{t_{i+1}} e^{A\tau} B u(t_i - \tau) d\tau$

$$\hat{x}_{t_{i+1}} = \begin{cases} \bar{x}_{t_{i+1}}, \ if \ [y_{t_{i+1}}] = [\hat{y}_{t_{i+1}}] \\ \bar{x}_{t_{i+1}} + L(y_{t_{i+1}} - \hat{y}_{t_{i+1}}), \ else \end{cases}$$

i:i+1

2.6 System and Controls of the Plasmonic Imaging Lithography Machine(PILM)

The developed sub-count continuity constraint algorithm was tested on a Plasmonic Imaging Lithography Machine (PILM) [15] due to its requirement for nanoscale precision. The system used for experiment is shown in Figure 2.5. PILM controls the position of a LASER write head to fabricate lithographic patterns on a spinning wafer. A schematic illustration of PILM is shown in Figure 2.6. As shown in this figure, the LASER write head is controlled by the pre-focusing

system that is installed on a moving stator. The coarse motion of the stator is controlled by two three-phase linear motors with optical encoder feedback. The encoder resolution is $5nm$ and the system is controlled at $1kHz$. The fine motion of the pre-focusing system is controlled by a multi-axis piezo-actuated system with capacitive sensor feedback.

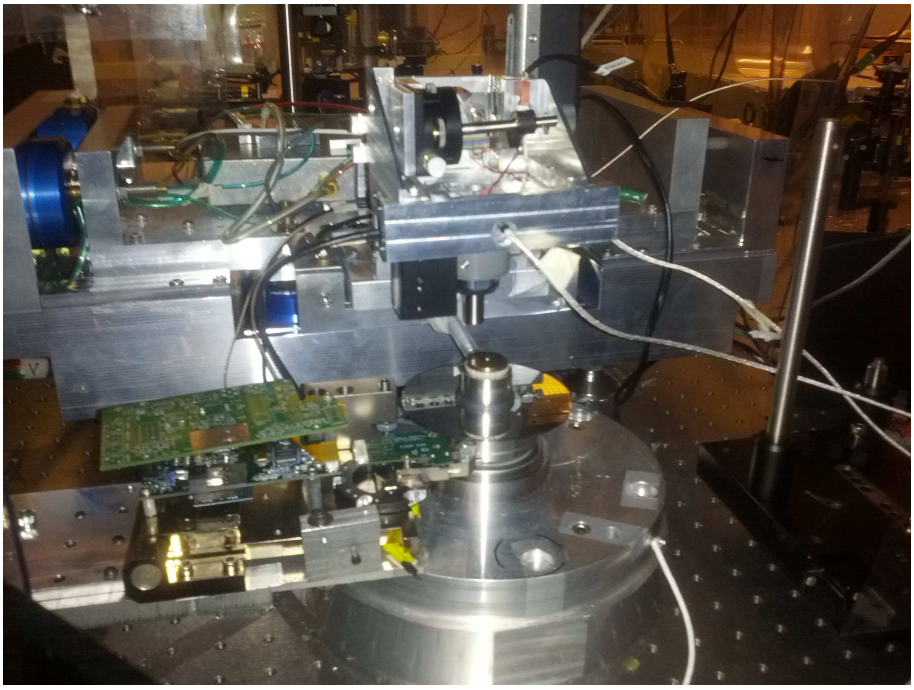


Figure 2.5: Plasmonic Imaging Lithography Machine

The coupling effect between the two linear motors is not negligible due to the parallel arrangement. Because of the system asymmetry and actuator difference, the system dynamics of the two linear motors are not identical. The difference between the two linear motors introduces non-zero yaw rotation along with the stator translational displacement. To perform lithographic patterning, the linear motor system needs to meet the nano-scale tracking performance requirement with regulated yaw motion. Because the stage is required to move at very low speed with nano-scale precision, the quantization effect is critical.

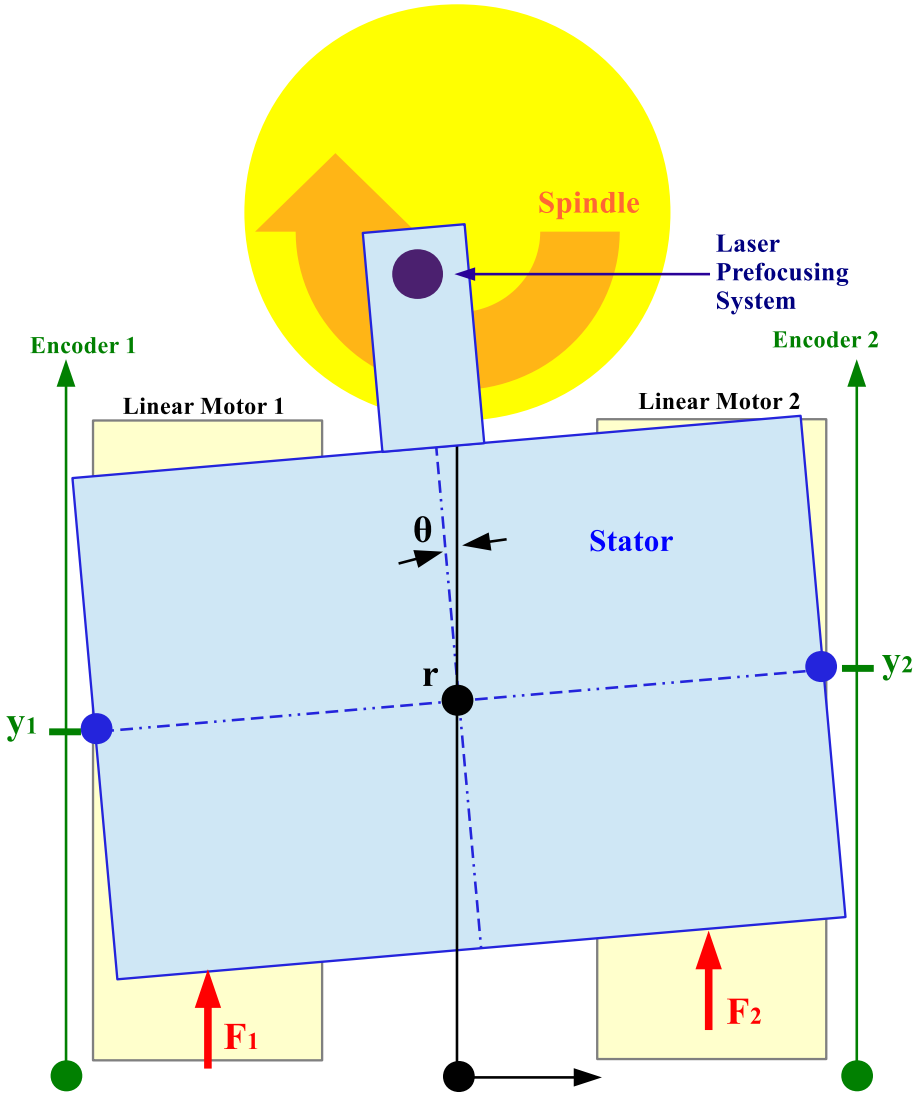


Figure 2.6: Schematic view of Plasmonic Imaging Lithography Machine

2.7 Experiments of sub-count Estimation on PILM

The proposed sub-count estimation algorithm is implemented by using the open-loop plant model shown in Equation 2.9. The quantization effect is implemented by quantizing the encoder output to have a resolution of 5 μm, which is used to close the loop with a standard PID controller with gains shown in Table 2.2. The

performance is compared to the original encoder measurement that has 5 nm of resolution.

$$G = \begin{bmatrix} G_{11} & 0 \\ 0 & G_{22} \end{bmatrix} \tag{2.9}$$

$$G_{11} = \frac{\begin{array}{r}(3.6z^6 + 11z^5 + 16z^4 \\ +15z^3 + 8.9z^2 + 4.1z - 2.8)10^{-6}\end{array}}{\begin{array}{r}z^8 - 1.4z^7 + 0.13z^6 + 0.12z^5 + 0.091z^4 \\ +0.045z^3 - 0.0097z^2 - 0.061z + 0.075\end{array}}$$

$$G_{22} = \frac{\begin{array}{r}(-4.2z^6 - 7.3z^5 + 0.91z^4 \\ +5.6z^3 - 0.61z^2 - 4.7z - 1.7)10^{-6}\end{array}}{\begin{array}{r}z^8 - 2.4z^7 + 1.6z^6 + 0.45z^5 - 0.59z^4 \\ -0.29z^3 + 0.11z^2 + 0.33z - 0.17\end{array}}$$

Table 2.2: PID Gains of PILM

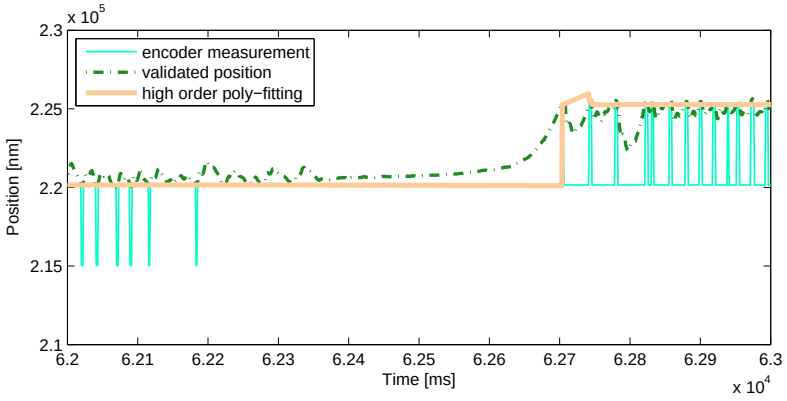
Gains	**Translational**	**Rotational**
P	5	1
I	0.1	0.1
D	10	0.1

The order of the continuity constraint depends on the relative order of the plant model. For a system with relative order of 2, continuity constraints should be applied to the stage position and velocity because acceleration is not necessarily continuous. This is verified by a simplified SISO simulation by using the translational element of the MIMO plant. The RMS error of the corresponding

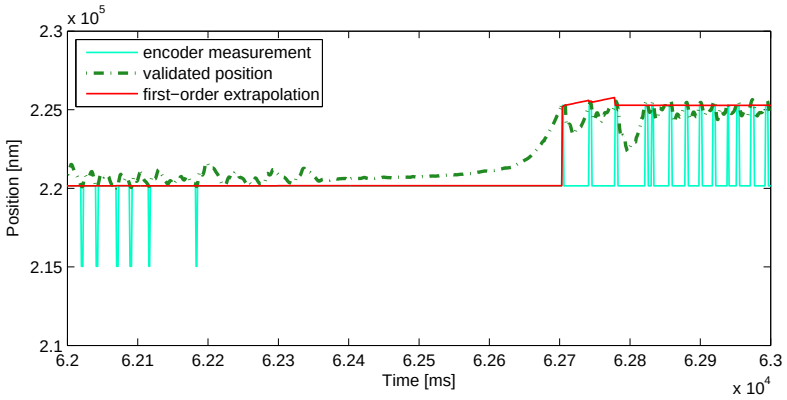
order of continuity is shown in Table 2.3. By applying continuity constraint on the acceleration state, the RMS error is dramatically increased.

Table 2.3: RMS Error for Different Order of Continuity.

Order	**RMS Error [count]**
1 (pos)	0.1649
2 (+vel)	0.1277
3 (+acc)	3.122×10^{120}

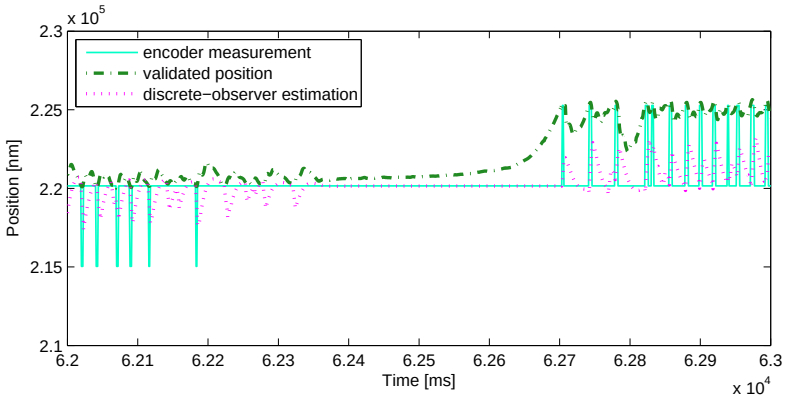


(a) 3^{rd} Order curve-fitting

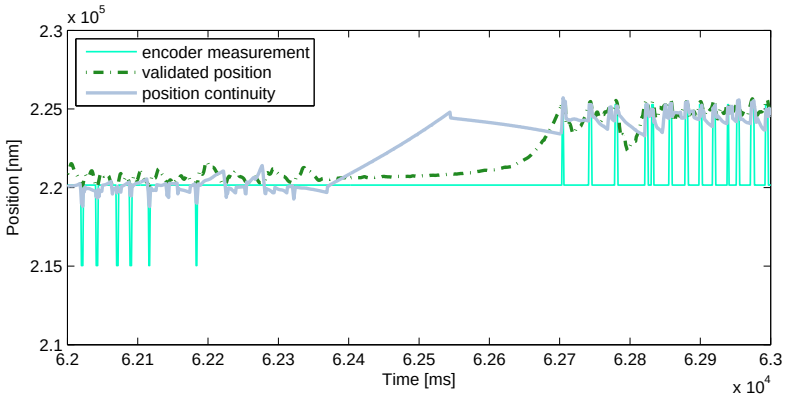


(b) Linear curve-fitting

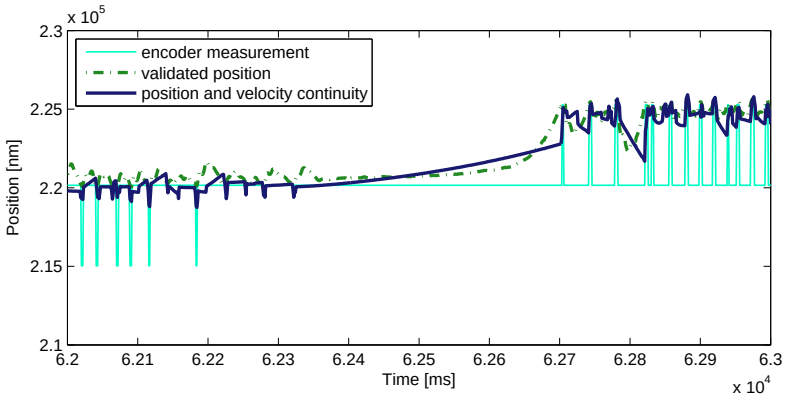
Figure 2.7: Estimation results for non-model-based approaches



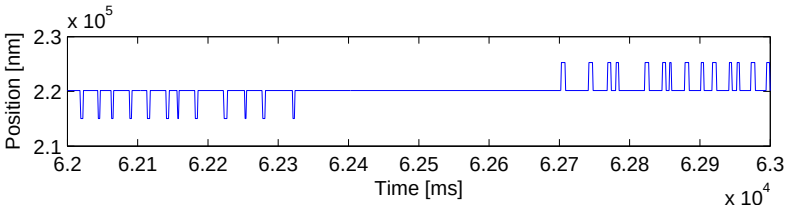
(a) Discrete time Kalman Filtering on Quantized Measurement for encoder 1



(b) Sub-count estimation with position continuity constraint for encoder 1



(c) Sub-count estimation with position and velocity continuity constraints for encoder 1



(d) Quantized position measurement from encoder 2

Figure 2.8: Estimation results for model-based approaches

Figure 2.7 presents the position estimation results for non-model-based approaches using 3 past triggers. Due to the lack of knowledge of system dynamics, non-model-based approaches are unable to properly predict the inter-sample behavior. As a result, abrupt prediction changes occur as the encoder innovates to the next quantization level. Figure 2.8 shows the position estimation results for model-based approaches. Because the traditional discrete-time Kalman filter requires measurement update at every sampling time, the estimation results are greatly influenced by the quantization effect and asynchronous measurement update. The case of sub-count estimation with position continuity constraint is equivalent to the event-based state correction method [6] augmented with the coarse observer [7, 8].

It is shown that due to the lack of higher order continuity, it does not provide good velocity estimation especially when the system propagates without any encoder update from $6.24msec$ to $6.27msec$. On the other hand, the continuity constraint on velocity is shown to generate smoother prediction, resulting in improved inter-sample propagation prediction. Also, due to the encoder update from the other encoder channel and the coupling effect, state correction can be observed from $6.22msec$ to $6.24msec$.

The transient response performance of sub-count estimation can be analyzed by tracking triangular waves. The effect of horizon length for both model-based and curve-fitting method is demonstrated by the time domain comparison in Figure 2.9. It is shown that when the proposed method outperforms the curve-fitting method especially at the tips of the triangular wave with much less overshoot.

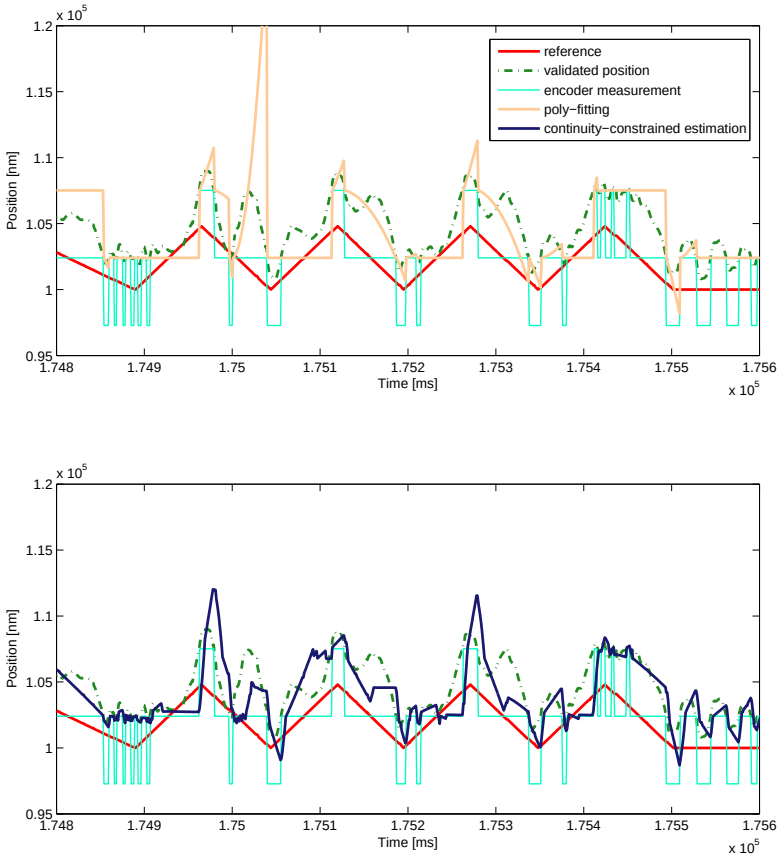


Figure 2.9: Estimation results for triangular reference tracking

Because PILM typically scans at different speed with periodic stops, its important to evaluate the estimation performance with triangular reference at different speed. Therefore, the sub-count estimation method is tested with ramp speed ranging from $0.5\mu m/msec$ to $64\mu m/msec$ as shown in Figure 2.10.

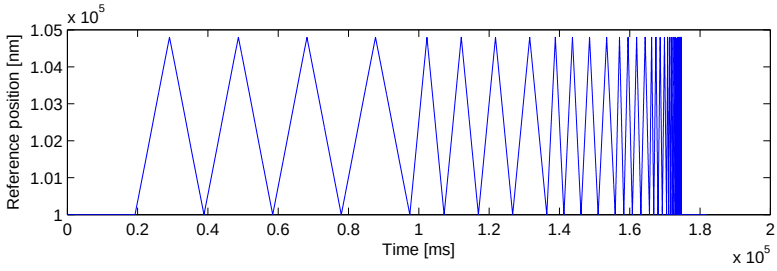
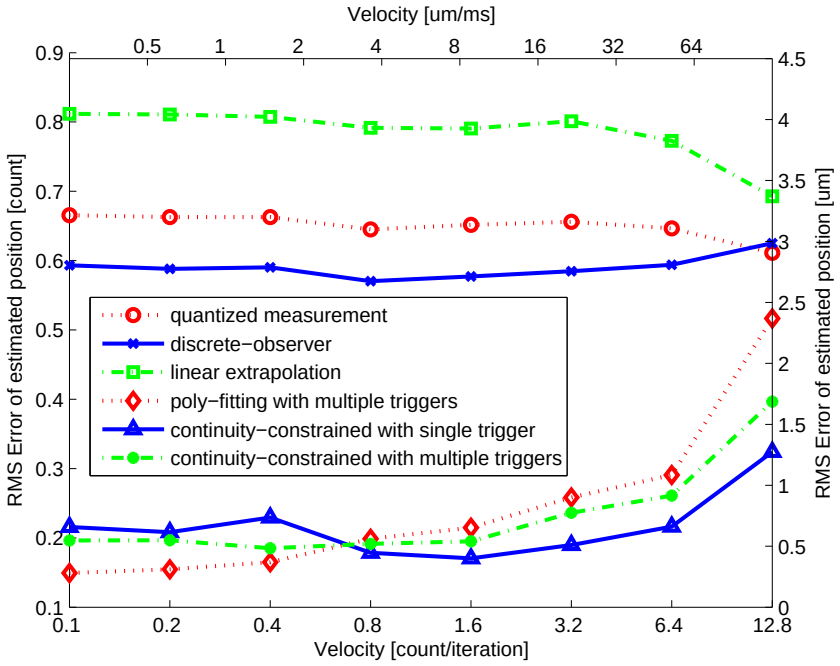


Figure 2.10: Triangular reference signal

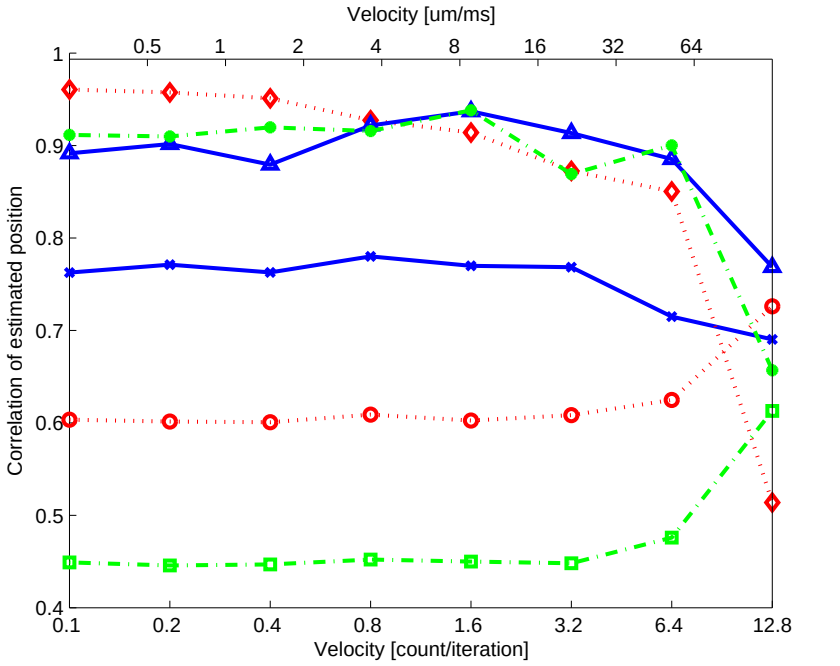
The overall estimation performance is analyzed by two indices: root-mean-square (RMS) error value and estimation correlation as defined by Equation 2.10 and 2.11, respectively. RMS error indicates the over-all tracking performance, and lower RMS error indicates better estimation performance. On the other hand, estimation correlation shows the prediction performance of sub-count behavior. Higher correlation show better prediction on the dynamic response between encoder triggers. The RMS and correlation performance are compared against the $5nm$ encoder measurement, discrete-time kalman filter, curve-fitting and continuity-constrained methods with short (1 trigger) and long horizons (3 triggers).

$$y_{rms} = \sqrt{\frac{1}{N}\Sigma_{i=0}^{N-1}(y_{t_i} - \hat{y}_{t_i})^2} \tag{2.10}$$

$$y_{cor} = \frac{\Sigma_{i=0}^{N-1}(y_{t_i} - \bar{y}_{t_i})(\hat{y}_{t_i} - \bar{\hat{y}}_{t_i})}{\sqrt{\Sigma_{i=0}^{N-1}(y_{t_i} - \bar{y}_{t_i})^2 \Sigma_{i=0}^{N-1}(\hat{y}_{t_i} - \bar{\hat{y}}_{t_i})^2}} \tag{2.11}$$



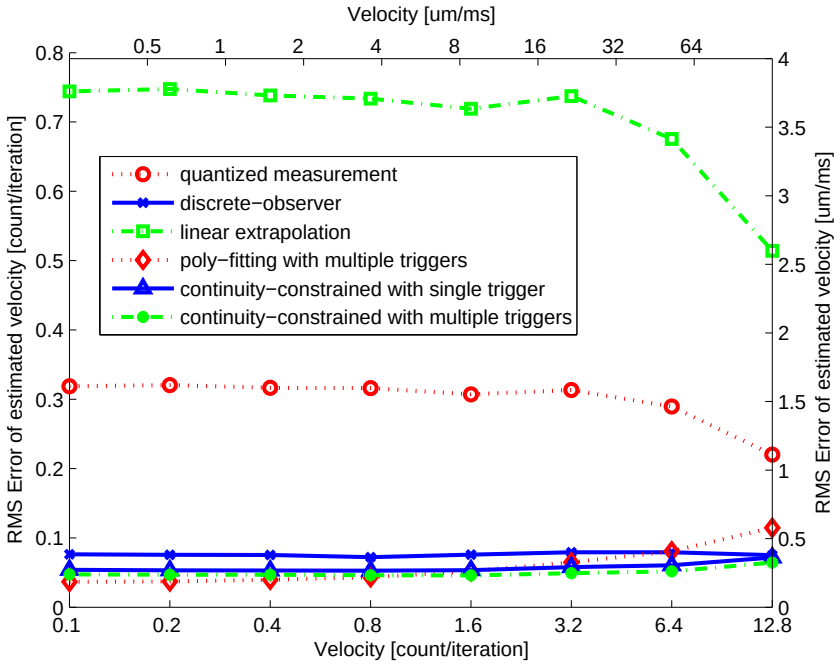
(a) RMS error of estimated position



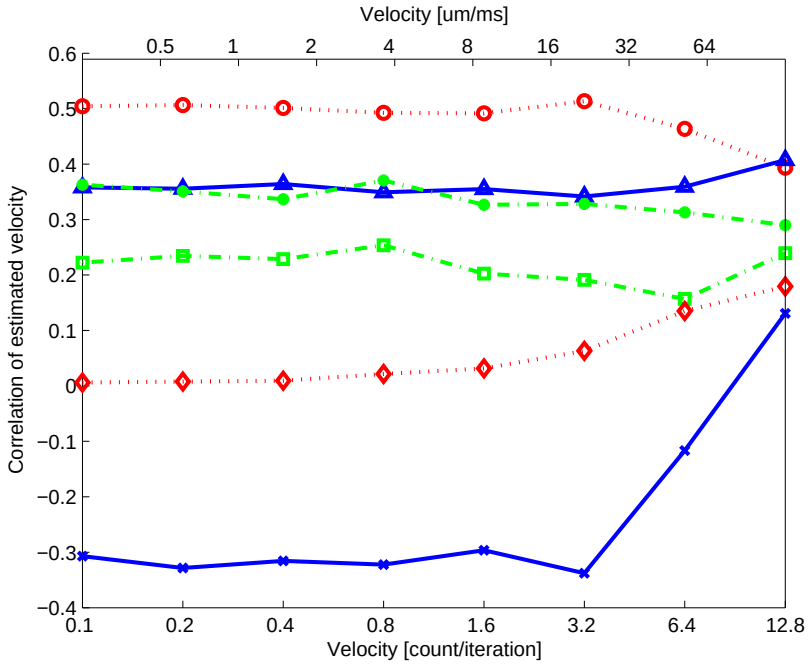
(b) Correlation of estimated position

Figure 2.11: Performance of position estimation

Figure 2.11 shows the RMS error and measurement correlation for position estimation, respectively. It is shown that the proposed approach and curve-fitting methods can improve the performance both in RMS and correlation compared to the discrete-time observer method. As seen in the time-domain comparison, curve-fitting method's performance is compromised by abrupt state changes. In contrary, the proposed method outperforms the position estimation that of the curve-fitting method at higher speeds. The effect of horizon length can also be observed in these two plots. While long horizon provides more stable prediction results, the performance is compromised when large state change occurs. For the curve fitting method, the increase in horizon length can dramatically improve the correlation and RMS error performance at lower speeds, however, causes performance drop at high speeds due to the traversing. The advantage of the proposed method is that the performance decrease is much less severe compared to the curve-fitting method, and the RMS error is better at higher speed. This phenomenon shows that when the system is nearly stationary, the more conservative method is more advantageous. However, when there are abrupt state changes such as the stop-and-go scanning motion, the proposed method shows substantial benefits when predicting inter-sample propagation.



(a) RMS error of estimated velocity



(b) Corr error of estimated velocity

Figure 2.12: Performance of velocity estimation

To make further comparison on the continuity-constrained state, the velocity RMS error and correlation are calculated by Equation 2.12 and 2.13, respectively. The values for each ramp speed is shown in Figure 2.12. It is shown that both curve-fitting and the proposed method outperform the discrete-time observer, especially with respect to the correlation performance. The continuity constraint is shown to greatly improve the prediction performance compared to other methods, resulting in smoother and more accurate estimation. The effect of horizon length can also be observed on the velocity estimation performance: longer horizon improves the velocity estimation for both the model-based and curve-fitting methods in RMS error. It is worth noting that longer horizon sacrifices the correlation performance in both cases. However, the model-based method is affected much less compared to the curve-fitting method.

$$v_{rms} = \sqrt{\frac{1}{N}\Sigma_{i=0}^{N-1}(v_{t_i} - \hat{v}_{t_i})^2} \tag{2.12}$$

$$v_{cor} = \frac{\Sigma_{i=0}^{N-1}(v_{t_i} - \bar{v}_{t_i})(\hat{v}_{t_i} - \bar{\hat{v}}_{t_i})}{\sqrt{\Sigma_{i=0}^{N-1}(v_{t_i} - \bar{v}_{t_i})^2 \Sigma_{i=0}^{N-1}(\hat{v}_{t_i} - \bar{\hat{v}}_{t_i})^2}} \tag{2.13}$$

The closed-loop performance is shown in Figure 2.13 and 2.14. When the quantized encoder measurement is used, the quantized measurement forces the stage to regulate against its current encoder reading until the measurement increments. This phenomenon creates abrupt steps in position and results in large tracking error. It is shown that by using the sensor feedback with sub-count estimation, smoother transition is observed especially when long propagation between encoder triggers occurs. The RMS error of the raw quantized encoder feedback is $2.904\mu m$, and it is improved to $1.484\mu m$ by the proposed sub-count estimation feedback.

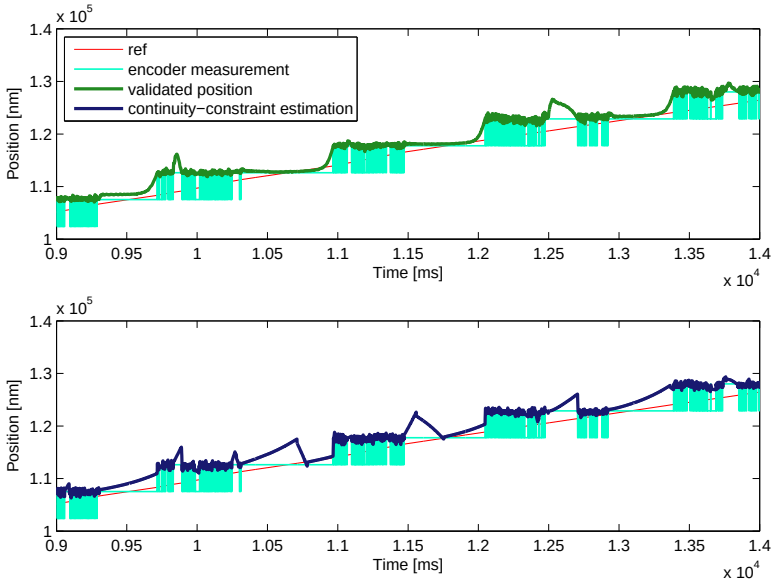


Figure 2.13: Validated and estimated position with quantized feedback.

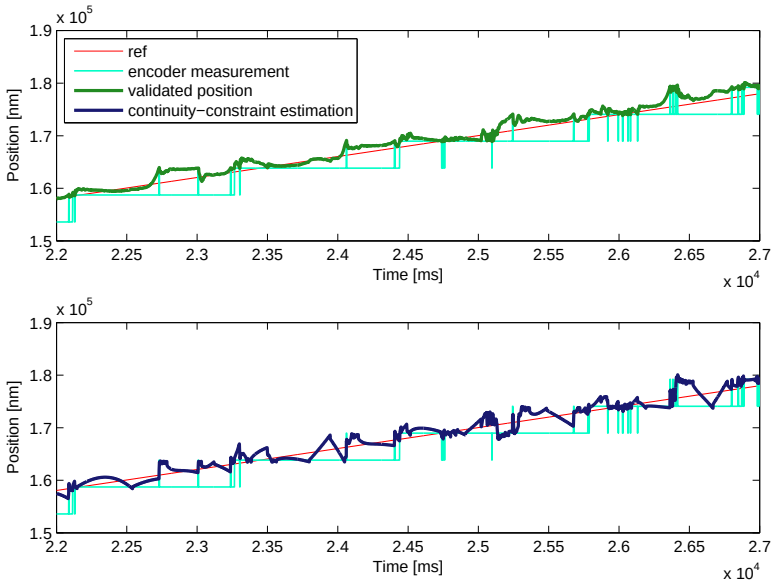


Figure 2.14: Validated and estimated position with estimation feedback.

The performance of other sensor resolution is shown in Figure 2.15 and 2.16.

It is shown that the proposed sub-count estimation method has the performance similar to half of the original resolution.

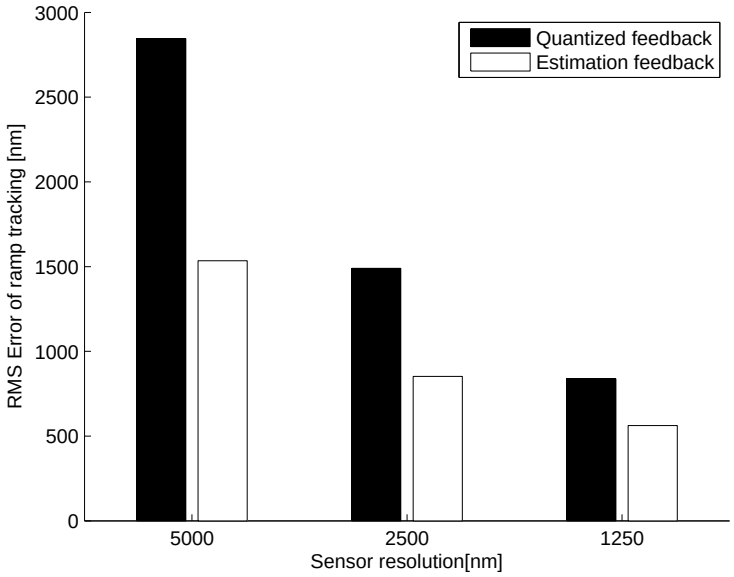


Figure 2.15: Effect of quantization level on tracking error.

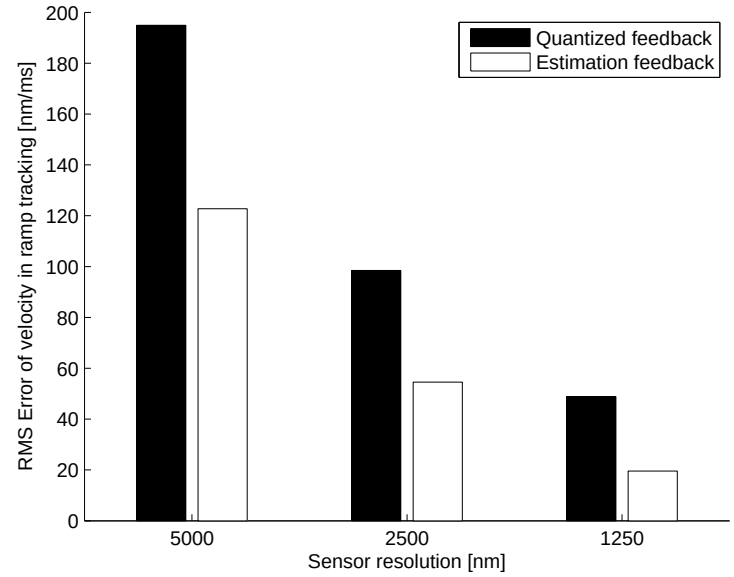


Figure 2.16: Effect of quantization level on velocity error.

CHAPTER 3

Feedforward MPC with Realistic Constraints

3.1 Fast Tool Servo and Model Predictive Control

Model predictive control (MPC) techniques have gained much attention for its ability to optimize the control signal with respect to a desired objective function, such as tracking error, motion smoothness, control signal magnitude, and the predicted values of certain state variables. This concept has been applied to contour tracking problem in its unconstrained form [20, 21]. Manipulator joint trajectory design further brings non-linear modeling into receding control as Genetic Algorithm [22, 23]. The disadvantage of pure MPC algorithm includes the accurate modeling requirement and stability concerns [24, 25, 26, 27, 28], making it vulnerable to modeling uncertainties and disturbances. Terminal constraints are commonly applied to achieve stability [33, 27], and robustness is improved by incorporating the modeling uncertainty in the objective function [24, 35]. Dual feedback controller structure and disturbance identification were also developed to keep the system in a robust control-invariant set [34, 26].

The demand for high precision non-circular manufacturing increases the need for fast tool servo machining. MPC was introduced to fast tool servo machining by cascading with an additional repetitive controller [30, 31, 32]. For challenging reference tracking, such as race car piston machining, the machining tool is desired to track a partially nearly-periodic and partially non-periodic signal. Addressing the strong periodicity, a two parameter robust and repetitive control (TPRRC)

method has been developed. Figure 3.1 shows TPRRC for dual-stage fast tool servos [29]. It shows the feedback controllers C_1 and C_2 respectively for the dual-stage actuators P_1 and P_2, where P_1 is considered a long stroke low bandwidth actuator like a linear motor and P_2 is a short stroke high bandwidth actuator like a piezoelectric actuator. The tool position z consists of the cascaded actuator position y_1 and y_2. The tracking error of the slow (P_1, C_1) loop is the reference input to the fast (P_2, C_2) loop. Each of the controllers C_1 and C_2 is a TPRRC, which consists of two compensators, one for robust feedback (K_{12}, K_{22}) and the other, with a positive feedback loop for repetitive control. The control problem is formulated numerically by μ-synthesis.

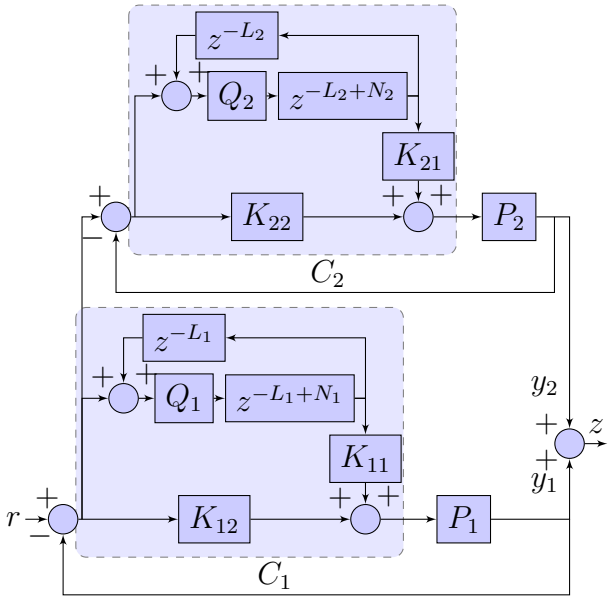


Figure 3.1: Block diagram of Two Parameter Robust Repetitive Control

Though TPRRC is able to track a challenging piston profile with strong periodic component, it is unable to handle constraint violation that frequently happens when the mechanical system is quickly traversing. To introduce MPC without computationally intensive real-time calculation, MPC can be applied as feed-forward model predictive control(FF-MPC) to be augmented with a robust

feedback controller. For a fast sampling application, computational intensity often restricts the on-line MPC from applying constraints. In addition, the length of prediction horizon needs to be limited and the complexity of the optimization problem must be carefully handled. By calculating the model predictive control gain off-line, the actuator dynamics and physical limitations of the system can be monitored without real-time computation. Analogous to the feedback form, the FF-MPC adopts the moving horizon methodology to ease computation. The calculated feedforward control signals are iteratively updated. In general, the MPC achieves better performance with longer horizon length n. However, longer prediction horizon means stronger reliance on high modeling accuracy. It is, in other words, more vulnerable to modeling error and external disturbances. Also, longer prediction horizon increases computation time, preventing the model predictive control from on-line adaptation.

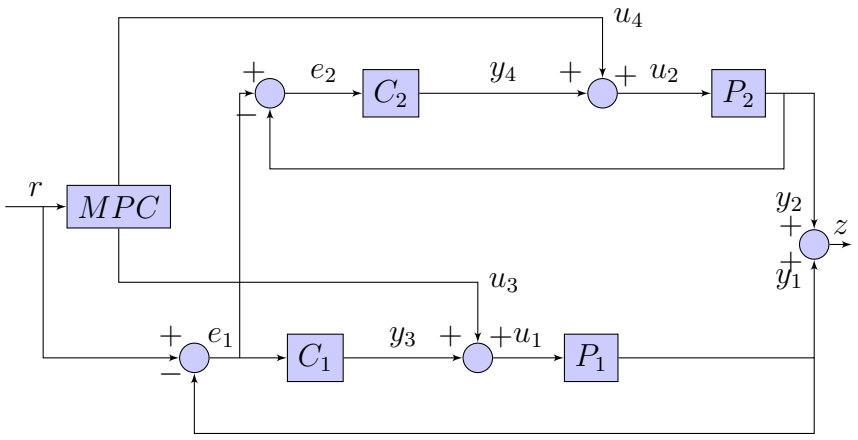


Figure 3.2: Block diagram of TPRRC with Feedforward model predictive control

Figure 3.2 shows the FF-MPC with the feedback control system. The MPC feed-forward controller signals are fed to the actuator control signal directly to preserve robustness and stability. The primal goal of the FF-MPC is to determine the feed-forward signal of the linear motor and piezo actuator, u_3 and u_4, to optimize the tracking error while restricting parameters from exceeding the

designated limitations. The model predictive control can be more aggressive in its feed-forward form, while the feedback controller handles the robustness of the system [36, 37, 38]. This hybrid control algorithm is able to preserve the benefits of MPC without sacrificing robustness or stability.

A discrete linear time invariant system can be described by the following equation:

$$\begin{aligned} x(t+\Delta t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

Within a designated horizon, the relationship between its input signal $\mathbf{u}$ and output signal $\mathbf{y}$ can be described in the following matrix form:

$$\mathbf{y} = \mathcal{B}\mathbf{u} + \mathcal{Q}\mathbf{x}_0 \tag{3.1}$$

where

$$\mathcal{B} = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{n-2}B & \dots & CB & D \end{bmatrix}$$

$$\mathcal{Q} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Because the actuators and controllers are modeled as discrete linear time invariant plants, this matrix form expression allows the whole system to be represented as a linear matrix equation:

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{bmatrix} = \bar{\mathcal{B}} \begin{bmatrix} \mathbf{u}_3 \\ \mathbf{u}_4 \\ \mathbf{r} \end{bmatrix} + \bar{\mathcal{Q}}\bar{\mathbf{x}}_{\mathbf{0}} \tag{3.2}$$

The input signal is consisted of the reference trajectory r, feed-forward signals u_3 and u_4. The output signal is consisted of the position measurement of the primary stage y_1 and secondary stage y_2. The vector form representation of these variables within the moving horizon is defined below:

$$\begin{aligned} \mathbf{r} &\equiv r(t : \Delta t : t + (n-1)\Delta t) \\ \mathbf{u}_i &\equiv u_i(t : \Delta t : t + (n-1)\Delta t) \\ \mathbf{y}_i &\equiv y_i(t : \Delta t : t + (n-1)\Delta t) \\ i &= 1 : 4 \end{aligned}$$

Model predictive control iteratively solves a predefined optimization problem, which is consisted of an objective function, equality and inequality constraints. The objective function can be any convex function consisted of the interested parameters. This dissertation represents the fast tool servo accuracy by a weighted quadratic sum of the tool tip tracking error:

$$\min \left(\begin{bmatrix} I & I & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{bmatrix} - \mathbf{r} \right)^T W^T W \left(\begin{bmatrix} I & I & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{bmatrix} - \mathbf{r} \right) \tag{3.3}$$

While the system dynamics can be expressed as the equality constraints of the optimization problem, the hardware limitations of the system can be represented

by the equality constraints on the position measurement output. For this dual-stage fast tool servo application, the linear motor acceleration maximizes at 100 m/sec^2, and the velocity constraint of the linear motor is set to be 0.25 m/sec. The travel range of the piezo actuator position is limited to within $\pm$ 10 um, and the average location of the piezo actuator should remain close to zero due to linearity concerns. The mean value is designed to be within $\pm$ 0.01 um range in this dissertation because strictly restricting the average value to zero is impractical.

The input signal limitation can also be handled by linear inequality constraints. According to the actuator specifications, control signal saturates at 10 V and 5 V for the linear motor and piezo actuator stage, respectively. Another important constraint is the electrical current limitation to the piezo actuator stage. The relationship between the drive current and the input voltage is shown in the equation below. The capacitance C_p and maximum current i_2 value for the piezo actuator in this dissertation are 30 nF and 100 mA, respectively. Since the current constraint is placed on the input signal to the amplifier of the piezo actuator, the amplifier gain K_p must be considered. In this application, the amplifier gain is 100.

Similar to the equality constraints, these inequality constraints can be augmented into a matrix form as shown in the following equation. In equation 3.4, H_a, H_v, and H_m represent the acceleration, velocity and mean value filter, respectively. The limitations on acceleration, velocity, displacement, mean value, and control signal are denoted as a, v, d, m, and s, respectively. R represents the compensation vector of the

$$-\left[\begin{array}{c} a_1\mathbf{l} \\ v_1\mathbf{l} \\ d_2\mathbf{l} \\ m_2\mathbf{l} \\ \hline s_1\mathbf{l} \\ s_2\mathbf{l} \\ v_2\mathbf{l} \end{array}\right] \leq \left(\underbrace{\left[\begin{array}{cccc} H_a & 0 & 0 & 0 \\ H_v & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & H_m & 0 & 0 \\ \hline 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & H_v \end{array}\right]}_{\bar{H}} \bar{\mathcal{B}} + \left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \hline I & 0 & 0 \\ 0 & I & 0 \\ 0 & H_v & 0 \end{array}\right] \right) \left[\begin{array}{c} \mathbf{u}_3 \\ \mathbf{u}_4 \\ \mathbf{r} \end{array}\right] + \bar{U} \leq \left[\begin{array}{c} a_1\mathbf{l} \\ v_1\mathbf{l} \\ d_2\mathbf{l} \\ m_2\mathbf{l} \\ \hline s_1\mathbf{l} \\ s_2\mathbf{l} \\ v_2\mathbf{l} \end{array}\right] \tag{3.4}$$

where

$$v_2 = \frac{i_2}{K_p C_p}, \bar{U} = \bar{H}\bar{Q}\bar{\mathbf{x}}_0 + R$$

and

$$\begin{aligned} D_j &= \left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ 1 & & & \\ \vdots & \ddots & & \\ 1 & \dots & 1 & \\ \underbrace{1 \quad \dots \quad 1 \quad 1}_{j} & & & \end{array}\right]_{n\times n}, \mathbf{l} = \left[\begin{array}{c} 1 \\ 1 \\ \vdots \\ 1 \end{array}\right]_{n\times 1} \\ H_v &= \frac{1}{\Delta t}\left(D_n - 2D_{n-1} + D_{n-2}\right) \\ H_a &= \frac{1}{\Delta t^2}\left(D_n - 3D_{n-1} + 3D_{n-2} - D_{n-3}\right) \\ H_m &= \frac{1}{m}\left(D_n - D_{m-n}\right) \end{aligned}$$

3.2 Convergence of the Feed-forward model predictive control

Signal convergence is an important issue for model predictive control. An explicit solution to the finite horizon optimization problem can be derived when the inequality constraints are not violated. To solve this quadratic optimization problem, the equality constraints can be incorporated in the objective function:

$$\mathbf{y}_1 + \mathbf{y}_2 - \mathbf{r} = \underbrace{\left[[I\ I\ 0\ 0]\bar{\mathcal{B}} - [0\ 0\ I]\right]}_{\mathcal{B}_e} \underbrace{\begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{r} \end{bmatrix}}_{\mathbf{u}_e} + \underbrace{[I\ I\ 0\ 0]\bar{\mathcal{Q}}\bar{\mathbf{x}}_0}_{-\mathbf{x}_e} \tag{3.5}$$

The objective function is thereby reformulated as an unconstrained quadratic function with coordinate transformations. The constrained optimization problem can, therefore, be represented as an unconstrained quadratic optimization problem:

$$\begin{aligned} &(\mathbf{y}_1 + \mathbf{y}_2 - \mathbf{r})^T W^T W (\mathbf{y}_1 + \mathbf{y}_2 - \mathbf{r}) \\ &= (\mathcal{B}_e \mathbf{u}_e - \mathbf{x}_e)^T W^T W (\mathcal{B}_e \mathbf{u}_e - \mathbf{x}_e) \end{aligned}$$

The solution to the unconstrained optimization problem, $\mathbf{u}_e$, must be in the range space shown below:

$$\mathcal{B}_e^T W^T W \mathcal{B}_e \mathbf{u}_e^* = \mathcal{B}_e^T W \mathbf{x}_e \tag{3.6}$$

When the weighting function W is purely an identity matrix with n_d look-ahead steps, the objective of this model predictive control is equivalent to optimizing the absolute tracking performance after n_d steps. The convergence analysis is then dependent on the amount of look-ahead steps chosen.

$$W = diag([\underbrace{0\ 0\ ...\ 0}_{n_d}\ \ 1\ 1\ ...\ 1]) \tag{3.7}$$

The convergence of the designed model predictive control can be evaluated by observing the mapping from the input signal $\mathbf{r}$ to the output measurement $\mathbf{y}$. From linear system theory, the least square least norm solution must satisfy the following condition.

$$\begin{aligned} \mathcal{B}_e^T W \mathcal{B}_e \quad & (\mathcal{B}_e^T W \mathcal{B}_e)^+ \mathcal{B}_e^T W \mathcal{B}_e = \mathcal{B}_e^T W \mathcal{B}_e \\ col \qquad & : W(\mathcal{B}_e(\mathcal{B}_e^T W \mathcal{B}_e)^+ \mathcal{B}_e^T W - I) \subset null(\mathcal{B}_e^T)^T \end{aligned}$$

Because W is a pure delay matrix, the mapping from $\mathbf{r}$ to $\mathbf{y}^*$ can be derived:

$$\begin{aligned} W \quad & (\mathcal{B}_e(\mathcal{B}_e^T W \mathcal{B}_e)^+ \mathcal{B}_e^T W - I) = \phi \\ W \quad & \mathbf{y}^* = W(\mathcal{B}_e \mathbf{u}_e^* - \mathbf{x}_e + \mathbf{r}) = W\mathbf{r} \end{aligned}$$

For the case when there's no look-ahead steps, the model predictive control performs equivalently to a pole-zero canceling controller. However, for a system with non-minimum-phase zeros, this operation introduces unstable pole-zero cancellation. Therefore the delay steps must be chosen based on the unstable zeros [40, 41].

For this dual-stage fast tool servo application, convergence is maintained when the look-ahead steps are greater or equal to the number of unstable plant zeros. On the other hand, unnecessarily long delay steps compromises the tracking performance. Therefore, in this dissertation, the delay step is chosen to be two, which correspond to the two unstable zeros from the piezo actuator plant.

The following figure shows the definition of horizon length. As previously discussed, while there are n steps spanning from time t_1 to $t_1 + (n-1)\Delta t$, only

the measurements after n_d steps are penalized. The overall prediction horizon n not only must be greater than the look-ahead steps, it also needs to capture the nominal dynamics of the system. In other words, its dimension must be greater or equal to the order of the linear motor and piezo actuator plant. In this dissertation, the prediction horizon n is 20. After the solution for the optimization is derived, the first element of the optimization results are used to update the FF-MPC output.

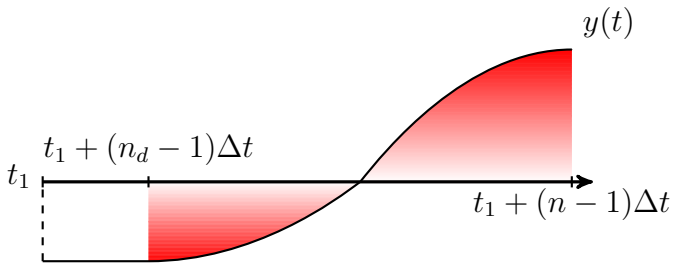


Figure 3.3: Illustration of the horizon length

3.3 Simulation Results for a Challenging Piston Profile

A typical piston tooling profile example is shown in Figure 3.4. It is composed of nearly periodic oscillations and a drastically changing profile, which well demonstrates the benefit of the FF-MPC. The performance of the feed-forward model predictive control is compared to the case without the feed-forward signal for a piston machining application by tracking the piston profile illustrated in Figure 3.4 at 10 kHz of sampling rate.

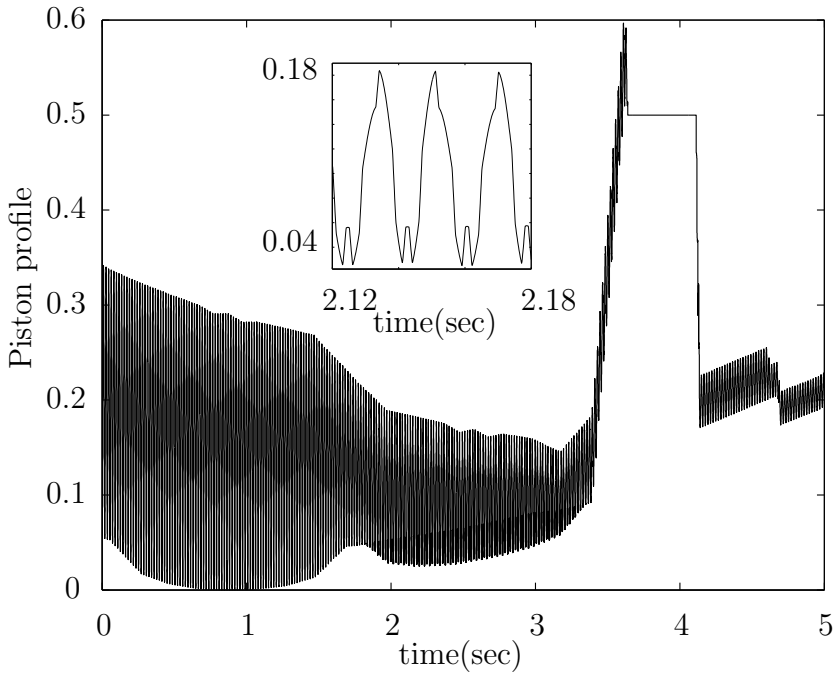


Figure 3.4: Piston profile in time domain

The bode plots for the linear motor and piezo actuator are shown in Figures 3.5 and 3.6, respectively. The linear motor has much lower resonant frequency and smaller resonant peak compared to the piezo actuator stage. Therefore, the piezo actuator is used as the secondary stage to compensate for transient error response, while the linear motor is used as the primary stage tracking the nominal trajectory defined by the piston profile. Figure 3.7 shows the feed-forward control signals of the linear motor and pzt stage u_3 and u_4 from the FF-MPC method.

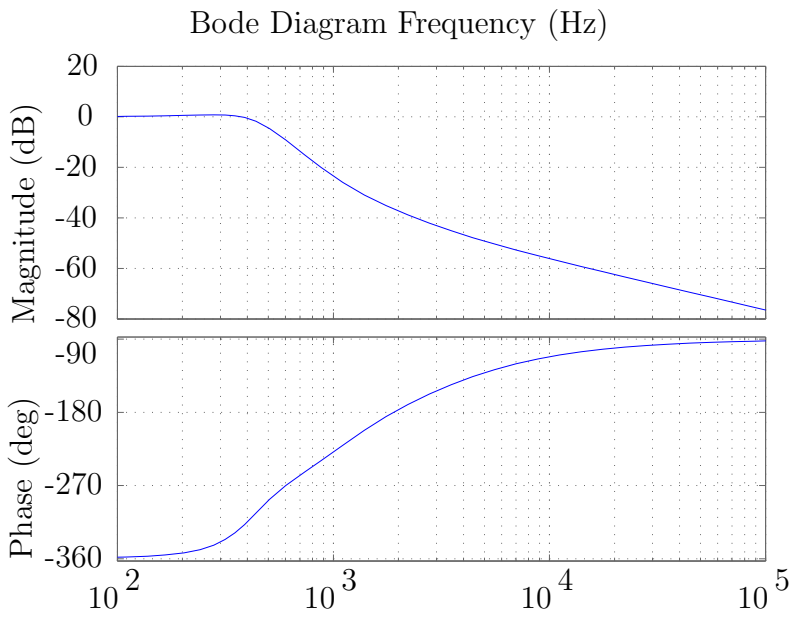


Figure 3.5: Bode plot of the continuous time transfer function of the linear motor

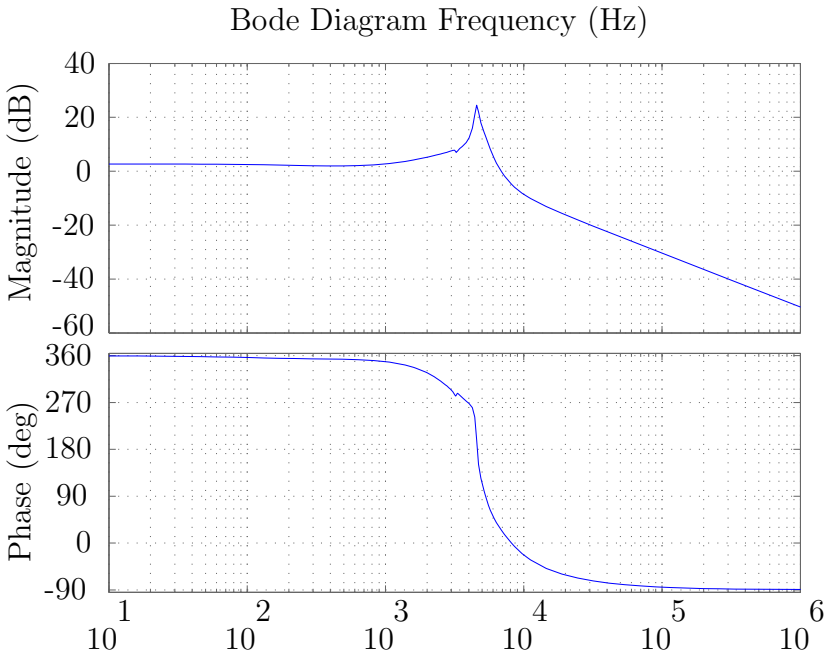


Figure 3.6: Bode plot of the continuous time transfer function of the piezo actuator

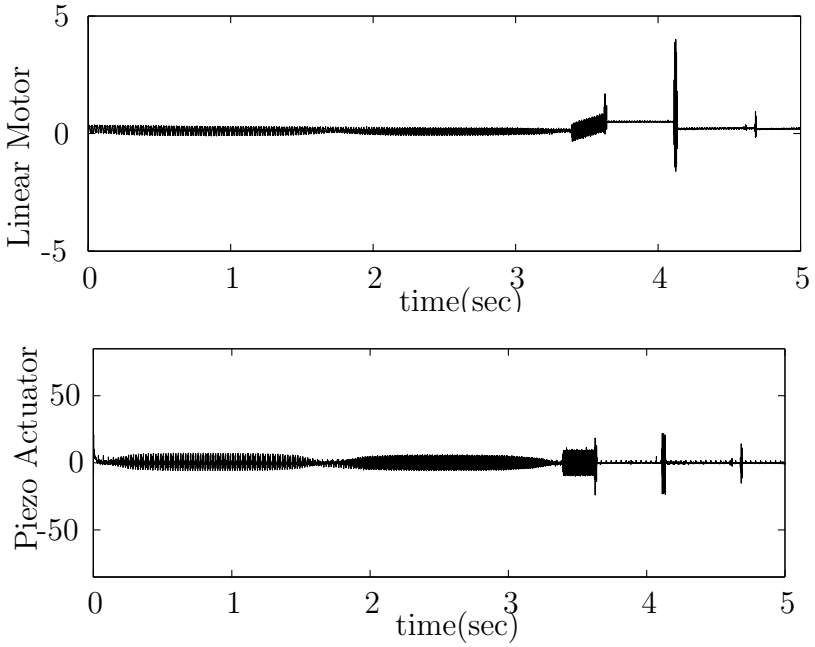


Figure 3.7: Feedforward control signals for the primary(mm) and secondary stage(μm)

The tracking error performance is compared to TPRRC in Figure 3.8. Comparing the single-stage and dual-stage performance, both TPRRC and FF-MPC show that the dual-stage structure is able to suppress the transient tracking error response when the reference signal changes drastically. The FF-MPC tracking performance performs better than the TPRRC especially when the reference profile leaves the nearly-periodic region. TPRRC drives the secondary stage by passively compensating for the error of the primary stage, resulting in 6.18 mm of the 2-norm of the absolute error and much larger error transient responses. The FF-MPC sends feed-forward control signals to drive the primary and secondary stages with respect to their dynamic properties and limitations. As a result, the FF-MPC improves the 2-norm of error signal to 2.19% of the TPRRC method.

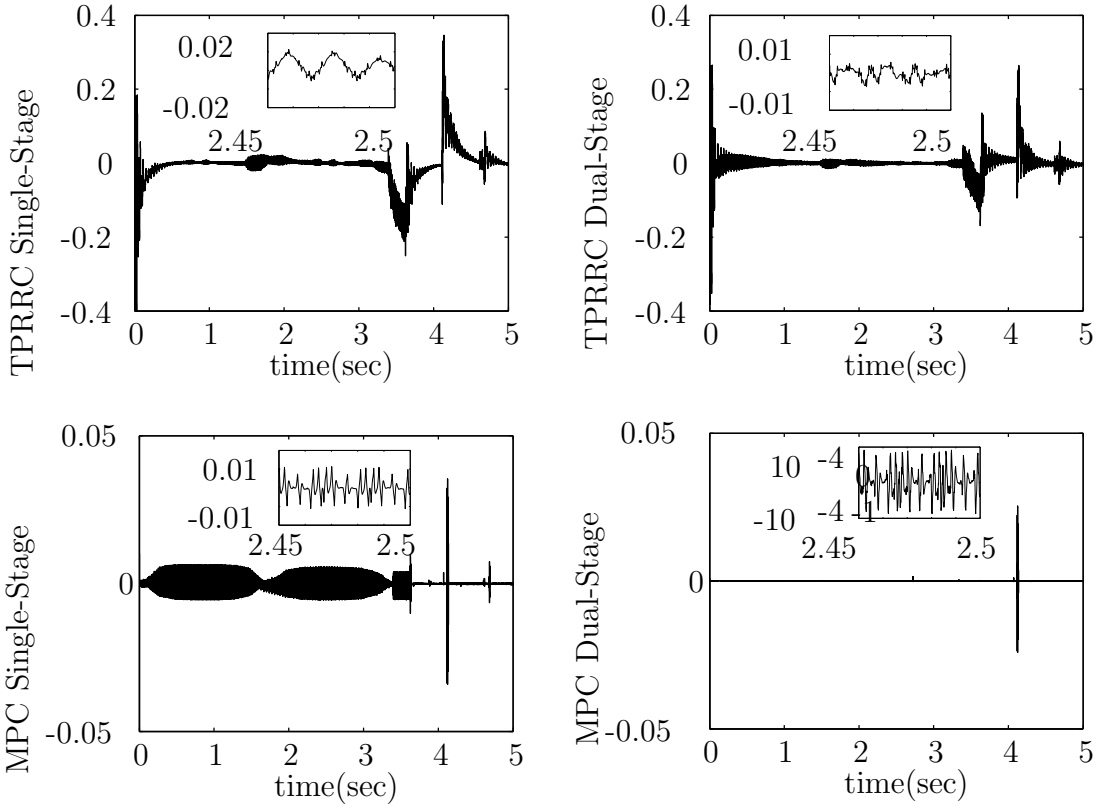


Figure 3.8: Tracking error(mm)

Figures 3.9 to 3.13 show the variations of the parameters restricted by the inequality constraints. In these Figures, the saturation limits are indicated by the red lines. Figure 3.9 shows the control signal variation of both stages in the case without FF-MPC and with FF-MPC. The input signal saturation of the piezo stage occurs when the reference trajectory enters and leaves the drastically changing state. In contrast, the FF-MPC controller is able to constrain the signal from reaching the saturation limit.

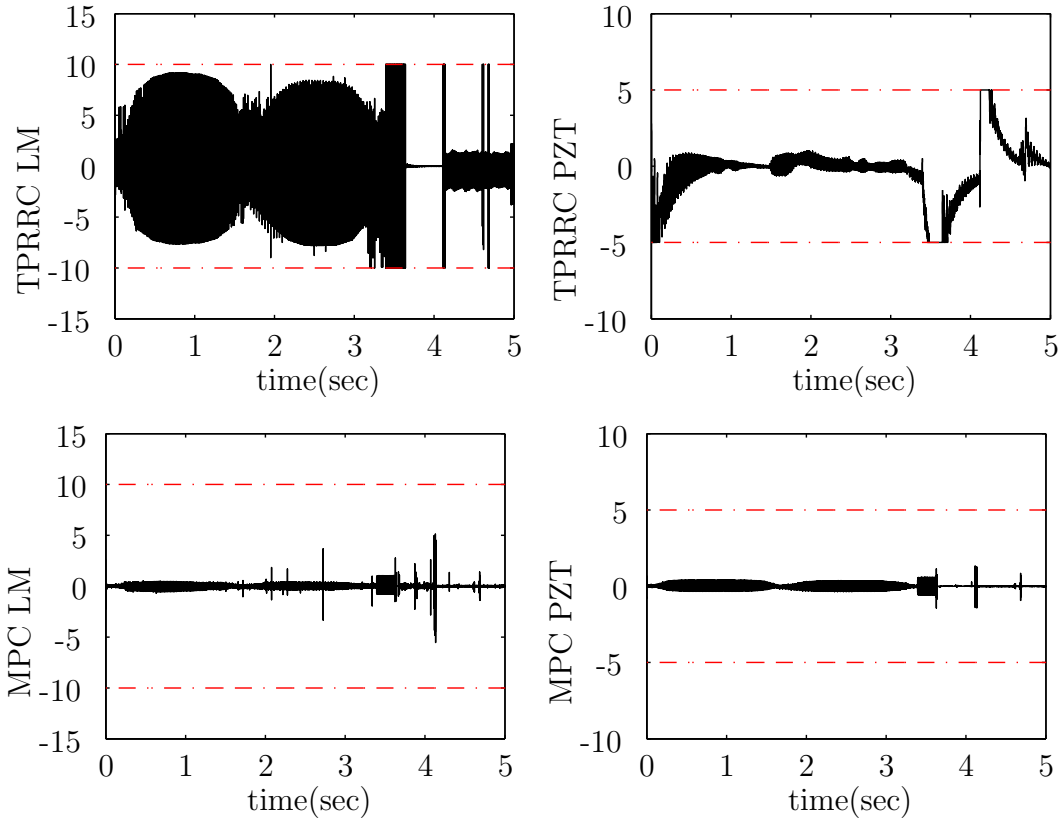


Figure 3.9: Control signal constraints(V)

Figure 3.10 and 3.11 show that the TPRRC method fails to keep the piezo stage displacement within the travel range limitations, which demonstrates a fundamental benefit of the FF-MPC. The FF-MPC is able to monitor physical limitations other than signal saturation. For example, by keeping the average displacement of the piezo actuator small, the secondary stage can be better kept in its linear region. Figure 3.10 shows that the pure closed-loop method fails to keep the piezo stage displacement within the travel range limitations, which demonstrates a fundamental benefit of the FF-MPC. The FF-MPC is able to monitor physical limitations other than signal saturation. For example, by keeping the average displacement of the piezo actuator small, the secondary stage can be better kept in its linear region.

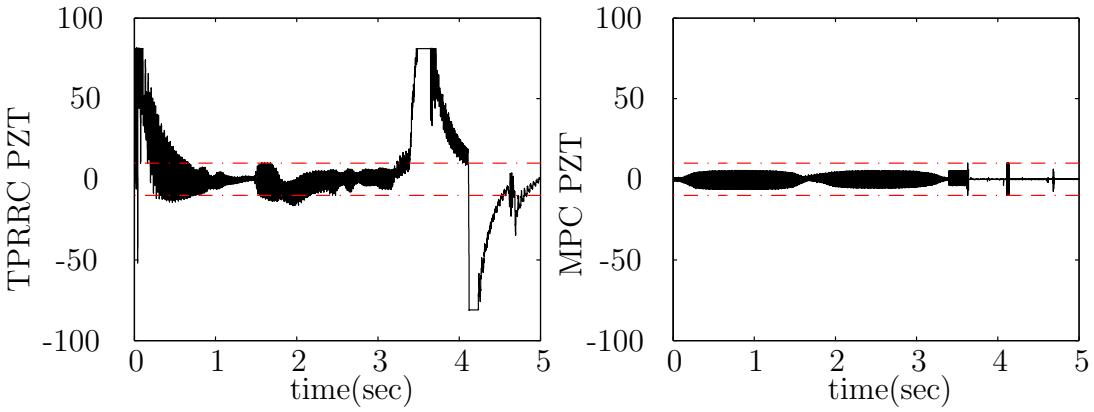


Figure 3.10: Pzt stage constraints-displacement(μm)

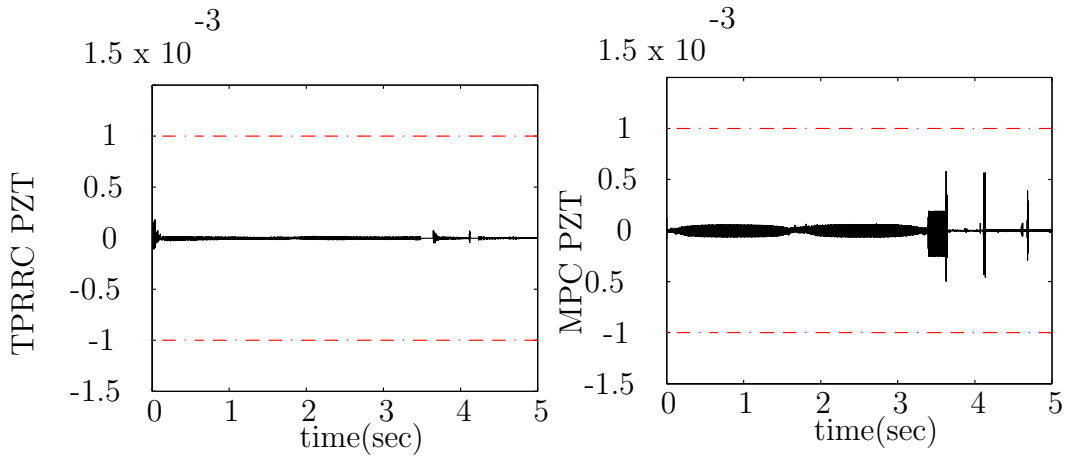


Figure 3.11: Pzt stage constraints-current(A)

Figures 3.12 and 3.13 show the velocity and acceleration constraints on the linear motor stage. The pure closed-loop method exceeds the acceleration limit during the time when the reference signal changes drastically. The FF-MPC, on the other hand, is able to remain both velocity and acceleration within its physical limitations.

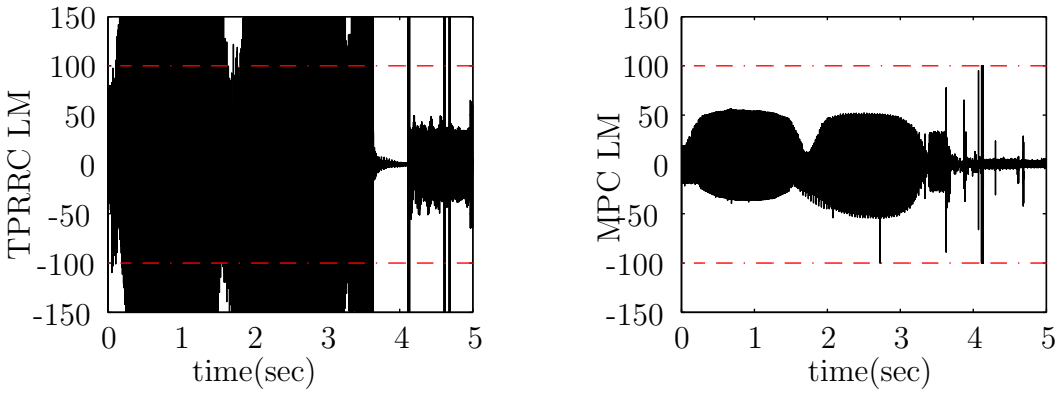


Figure 3.12: Linear motor stage constraints-acceleration(m/sec^2)

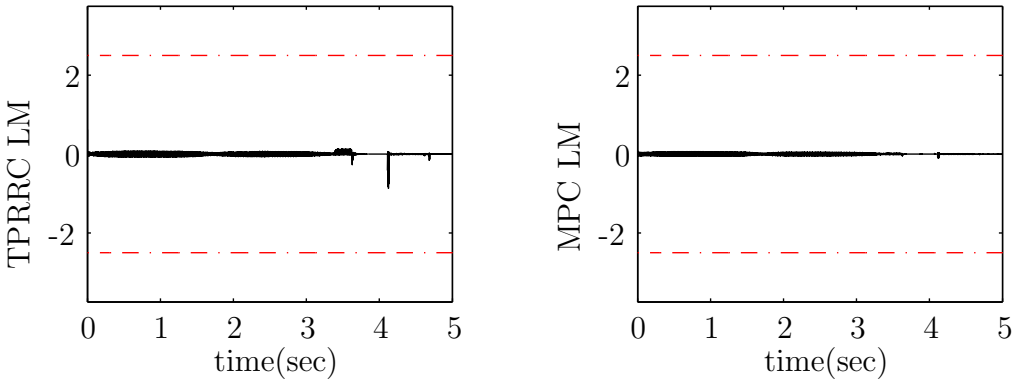


Figure 3.13: linear motor stage constraints-velocity(m/sec)

CHAPTER 4

Constrained Time-optimal Control

4.1 Spline Contour and Contour Error

Many precision motion control applications, such as CNC and lithography, use Spline functions to parametrically describe the tool path for the benefit of inter-sample behavior and data reduction [54]. To describe complex contours, the Spline model can be divided into multiple segments. The points N_k are the predetermined segment boundaries that the contour must pass:

$$S(\tau) = S_k(\tau_k) = K_{0,k} + K_{1,k}\tau_k + \ldots + K_{m-1,k}\tau_k^{m-1}$$

$$\begin{cases} \tau_k = \tau - k + 1 \\ 0 \leq \tau_k < 1 \\ S_k(1) = N_{k+1} \\ \frac{d^l}{d\tau^l} S_k(1) = \frac{d^l}{d\tau^l} S_{k+1}(0), 1 \leq l \leq q, q \in N \end{cases}$$

For contour tracking applications such as machining tool path design, cubic spline was first applied with knot speed control based on the geometric curvature. This was improved by the quintic Spline configuration and continuous time model to provide velocity, acceleration, and jerk estimation [54]. Though this off-line design method is convenient and flexible, the effect of modeling error, external disturbance, and measurement noise are not accounted for and the true system states may deviate from the estimated values when the system is driven near its physical limitation. Therefore, it is necessary to compensate such uncertainties by

real-time state estimation, rendering importance to on-line constrained feed-rate optimization [64, 65].

To utilize minimal-time contour tracking in real time, a contour tracking feed-rate optimization structure is proposed as shown by the block diagram in Figure 4.1. A multi-stage plant P is controlled by a controller C to track reference signal r, where r is derived by a reference governor that considers the contour profile S and system limitations h. For a known contour S, reference r can be determined given system dynamics and specified constraints. When calculated off-line, this method can avoid possible real-time implementation latency [66]. For the situation when modeling error and measurement noise are not negligible, the observer provides real-time state estimation to account for the difference between the measurement and prediction.

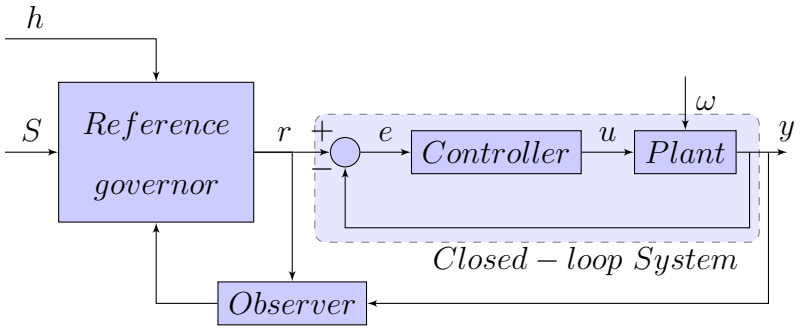


Figure 4.1: Block diagram of the contour tracking system

4.2 Using Constrained Model Predictive Control For Minimum-time Contour Tracking

To maintain system constraints within their limitation, model predictive control(MPC) has been widely used for explicit constraint handling and system dynamics incorporation. Because robotics and automation systems are typically controlled by a digital system at fixed sampling rate, the values of position mea-

surement y, reference signal r, contour S, and constraints h are structured into discrete finite vectors in the prediction horizon at each sampling time t_j:

$$\mathbf{y} = \begin{bmatrix} y(t_j) \\ y(t_j + \Delta t) \\ \vdots \\ y(t_j + (n-1)\Delta t) \end{bmatrix} \quad \mathbf{r} = \begin{bmatrix} r(t_j) \\ r(t_j + \Delta t) \\ \vdots \\ r(t_j + (n-1)\Delta t) \end{bmatrix}$$

$$\mathbf{S} = \begin{bmatrix} S(\tau(t_j)) \\ S(\tau(t_j + \Delta t)) \\ \vdots \\ S(\tau(t_j + (n-1)\Delta t)) \end{bmatrix} \quad \mathbf{h} = \begin{bmatrix} h(t_j) \\ h(t_j + \Delta t) \\ \vdots \\ h(t_j + (n-1)\Delta t) \end{bmatrix}$$

MPC generates the system input that optimizes an objective function based on the predicted system behavior and contour model within a finite moving horizon [49]. For the minimal-time contour tracking application, a time consumption index F is desired to be minimized, while function G models the system behavior that is constrained by $\mathbf{h}$:

$$\begin{aligned} \min_{\mathbf{r}} \quad & F(\mathbf{r}, \mathbf{S}, \mathbf{y}) \\ s.t. \quad & G(\mathbf{r}, \mathbf{S}, \mathbf{y}) \preceq \mathbf{h} \end{aligned}$$

The solution to the optimization problem is updated iteratively as the system propagates. At each iteration, the system behavior is predicted up to n steps ahead and the derived control signal is applied up to n_c steps. The solution to the constrained optimization problem updates the corresponding reference signal r to the n_s-dimensional multi-stage system iteratively within the control horizon:

$$r(t_j : t_j + (n_c - 1)\Delta t)$$
$$= [\mathbf{r}(1 : n_s : n_c)\ \mathbf{r}(2 : n_s : n_c) \ldots \mathbf{r}(n_s : n_s : n_c)]$$

4.3 Convex Objective Function

The contour tracking task is desired to solve a time-optimal problem by driving the system at maximum allowable feed-rate. Automatic feed-rate generation is first approached by the heuristic method shown in equation 4.1, where ϵ^p_{max} denotes the maximum predicted contour error from previous iteration. μ and σ are the adaptive feed-rate scheme parameters to tune [52].

$$f(\epsilon^p_{max}) = \begin{cases} \epsilon^p_{max} & , \epsilon^p_{max} \geq \mu \\ e^{-((\epsilon^p_{max} - \mu)^2 / 2\sigma^2)} \epsilon^p_{max} & , \epsilon^p_{max} < \mu \end{cases} \tag{4.1}$$

This approach introduces non-linearity and one-step delay to the model predictive controller. Also, this type of feed-rate generation is excluded from the optimization process. These restrictions result in conservative feed-rate design. To achieve explicit feed-rate optimization, explicit cycle time minimization for the whole trajectory was proposed [55] as shown in equation 4.2. Though this objective function explicitly minimizes time consumption, it needs to be discretized by grid points and the reciprocal of feed-rate creates singularity at low speed.

$$T = \int_0^T 1dt = \int_{s(0)}^{s(T)} \frac{1}{\dot{s}} ds \approx \Sigma_{k=0}^{K-1} \frac{2\Delta s^k}{|\dot{s}^k| + |\dot{s}^{k+1}|} \tag{4.2}$$

To allow real-time optimization to compensate for measurement noise, disturbances, and modeling error, the constrained minimization structure is desired to use a computationally efficient structure. By formulating the time index into a quadratic norm and system constraints into the linear inequality form, this dis-

sertation realizes time-optimal control for high sampling rate systems.

Because aggressive feed-rate shortens the time consumption, the objective function can be designed to maximize the feed $\bar{y}_i$ within the prediction horizon. However, maximizing the end feed directly results in a non-convex objective function and requires barrier method to implement [61]. For convexity and real-time implementation, a convex objective function is proposed to achieve this goal by minimizing the distance between the end feed $\bar{y}_i$ and the maximum allowable feed $S(\tau_{max})$ as shown in Equation 4.3:

$$\min_{\mathbf{r}} ||S(\tau_{max}) - y(t_j + (n-1)\Delta t)||_2^2 \tag{4.3}$$

The value of τ_{max} determines the maximum propagation of the contour index τ within the prediction horizon, which can be related to the maximum physical feed-rate by the Spline function:

$$\begin{aligned} f_{max} = \dot{S}(\tau_{max}) &= \left.\frac{dS}{d\tau}\frac{d\tau}{dt}\right|_{\tau=\tau_{max}} \\ &\approx \frac{dS(\tau_{max})}{d\tau}\frac{(\tau_{max} - \tau(t_j + (n-2)))}{\Delta t} \end{aligned}$$

4.4 System Dynamics

To estimate the system dynamics within the finite preview, the system plant is modeled as a discrete linear-time invariant (LTI) model as shown in equation 4.4.

$$\begin{aligned} x(t + \Delta t) &= Ax(t) + Br(t) \\ y(t) &= Cx(t) + Dr(t) \end{aligned} \tag{4.4}$$

Based on this LTI model, the relation between the input $\mathbf{r}$ and the output $\mathbf{y}$

within the finite interval for a given initial condition x_0 can be mapped by matrix $\mathcal{H}_1$. Similarly, the control signal $\mathbf{u}$ can be determined by a mapping matrix $\mathcal{H}_2$ with the initial condition c_0.

$$\begin{aligned} \mathbf{y} &= \mathcal{H}_1\mathbf{r} + \underbrace{\mathcal{X}_1 x_0}_{R_3} \\ \mathbf{u} &= \underbrace{\mathcal{C}_1(I - \mathcal{H}_1)}_{\mathcal{H}_2}\mathbf{r} + \underbrace{\mathcal{X}_2 c_0 - \mathcal{C}_1\mathcal{X}_1 x_0}_{-R_2} \end{aligned}$$

where

$$\begin{aligned} \mathcal{H}_1 &= \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{n-2}B & \dots & CB & D \end{bmatrix} \\ \mathcal{C}_1 &= \begin{bmatrix} D_c & 0 & \dots & 0 \\ C_cB_c & D_c & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_cA_c^{n-2}B_c & \dots & C_cB_c & D_c \end{bmatrix} \\ \mathcal{X}_1 &= \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}, \ \mathcal{X}_2 = \begin{bmatrix} C_c \\ C_cA_c \\ \vdots \\ C_cA_c^{n-1} \end{bmatrix} \end{aligned}$$

The prediction error can be estimated by the Luenberger observer, which provides the real-time estimation of the initial system states x_0 based on sensor measurements:

$$\hat{x}_0^+ = A\hat{x}_0 + Br + L(y - C\hat{x}_0) \tag{4.5}$$

4.5 System Dynamics and Constraints

For minimal-time contour tracking applications, restrictions are formulated into inequality constraints within the finite preview horizon. The model predictive controller generates trajectory profile subject to important physical limitation of the system, including control signal saturation, velocity limitation, maximum contour error, and monotonic feed. Signal saturation is important to consider due to amplifier limitations, which can be monitored by voltage command $\mathbf{u}$. Maximum velocity is also important and is monitored by the numerical differentiation of position measurement $\mathbf{y}$. These constraints are formulated into inequality constraints:

$$-\mathbf{u_{max}} \leq \mathbf{u} \leq \mathbf{u_{max}} \tag{4.6}$$

$$-\mathbf{v_{max}} \leq D_p\mathbf{y} \leq \mathbf{v_{max}} \tag{4.7}$$

Geometric tolerance can be determined by the maximum contour error allowed. Defined as the radial positioning error, contour error can be determined by the distance between the current machine position y and the closed point on the reference contour S as shown in equation 4.8. For computational efficiency, the solution to this minimization problem is often approximated by the projection method as shown in Figure 4.2 [44, 49]. By linearizing the Spline function with respect to the nodal point $\bar{\tau}$, the contour error in the neighborhood can be determined by projecting the absolute error onto the orthogonal line of the linearized contour function.

$$\begin{aligned}\epsilon &= \min_{\tau} ||S(\tau) - y|| \\ &\approx \frac{q_\perp}{|q_\perp|} \cdot \underbrace{(S(\tau) - y)}_{e} \\ &= q_\tau e\end{aligned} \tag{4.8}$$

where

$$\begin{aligned} q_\perp \cdot \frac{d}{d\tau} S(\tau) &= 0 \\ S(\tau) &= \Sigma_{j=0}^{m} K_{j,k} \tau^j \\ &\approx \Sigma_{j=0}^{m} K_{j,k} \bar{\tau}^j + \Sigma_{j=0}^{m} K_{j,k} j \bar{\tau}^{j-1} \delta\tau \end{aligned}$$

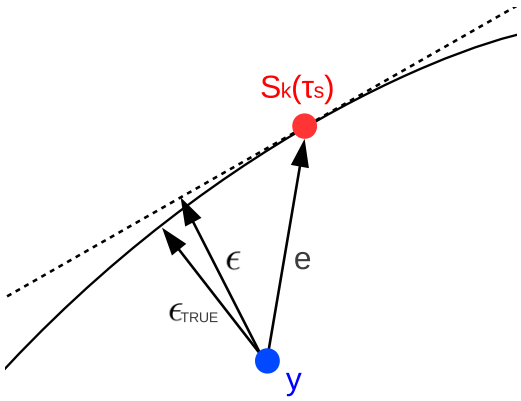


Figure 4.2: Schematic diagram of the contour projection

The linearized Spline contour within the finite preview can then be expressed by the vector form. The contour matrices K_τ and K_v are updated by the next segment's Spline parameters when the prediction horizon crosses segments. Based on the contour trajectory and system dynamics, tracking error **e** and contour

tracking error $\boldsymbol{\epsilon}$ can be derived by the contour data $\mathbf{S}$ and position $\mathbf{y}$ within the moving horizon:

$$\begin{aligned}
\boldsymbol{\epsilon} &= Q_p(\mathbf{S}-\mathbf{y}) \\
&= Q_p(K_\tau + K_v\boldsymbol{\delta\tau} - \mathcal{H}_1\mathbf{r} - \mathcal{X}_1 x_0) \\
&= Q_p\begin{bmatrix} K_v & -\mathcal{H}_1 \end{bmatrix}\begin{bmatrix} \boldsymbol{\delta\tau} \\ \mathbf{r} \end{bmatrix} + \underbrace{Q_pK_\tau - Q_p\mathcal{X}_1x_0}_{R} \\
&\leq \boldsymbol{\epsilon_{max}}
\end{aligned} \tag{4.9}$$

where

$$\boldsymbol{\delta\tau} = \begin{bmatrix} \tau(t_j) - \bar{\tau}(t_j) \\ \tau(t_j+\Delta t) - \bar{\tau}(t_j+\Delta t) \\ \vdots \\ \tau(t_j+(n-1)\Delta t) - \bar{\tau}(t_j+(n-1)\Delta t) \end{bmatrix}$$

Because the stage is only desired to move in the progressive direction, monotonic feed constraint is formulated to allow the τ index to only increase at each iteration as shown in Equation 4.10.

$$\begin{aligned}
\tau(t_j) &\geq \bar{\tau}(t_j) \\
\tau(t_j+i\Delta t) &\leq \tau(t_j+(i+1)\Delta t) \\
\tau_(t_j+(n-1)\Delta t) &\leq \tau_{max}
\end{aligned} \tag{4.10}$$

Due to the monotonic feed constraint and MPC structure, index τ increases gradually when it propagates by the sampling time Δt. Therefore, the nodal points can be selected as previous iteration's results with one step of lookahead as shown in Figure 4.3.

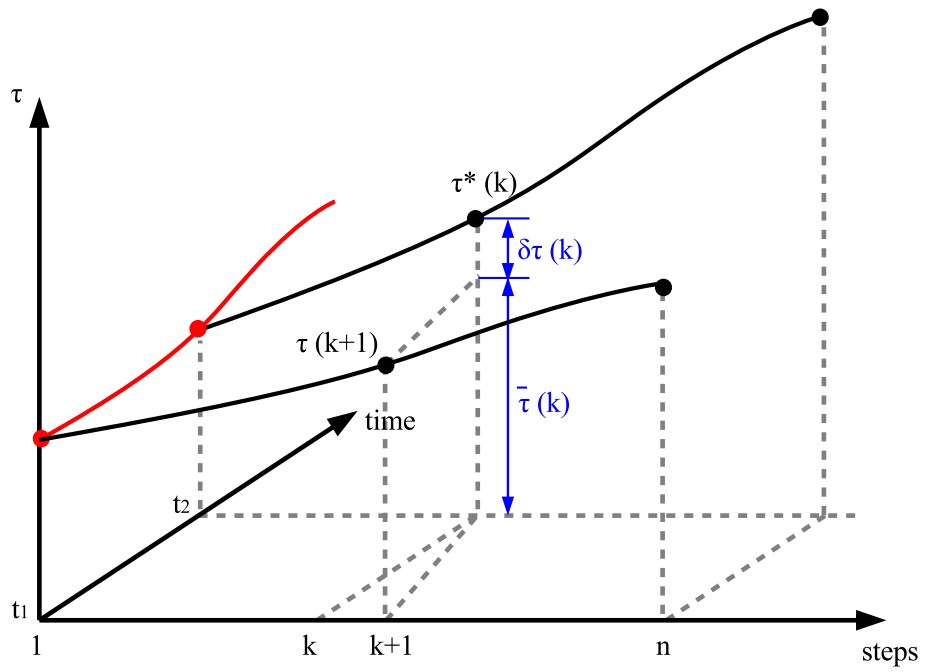


Figure 4.3: Nodal point propagation

4.6 Optimization Problem Formulation

By incorporating the LTI system dynamics and linearized Spline function, the minimal-time objective function becomes an affine function of the reference $\mathbf{r}$ and contour index $\boldsymbol{\delta\tau}$:

$$
\begin{aligned}
& S(\tau_{max}) - y(t_j + (n-1)\Delta t) \\
& = Q_n(K_\tau + K_v \boldsymbol{\delta\tau_{max}} - \mathcal{H}_1 \mathbf{r} - \mathcal{X}_1 x_0) \\
& = \begin{bmatrix} 0 & -Q_n\mathcal{H}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta\tau} \\ \mathbf{r} \end{bmatrix} + \underbrace{Q_n \begin{bmatrix} K_v & K_\tau & -\mathcal{X}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta\tau_{max}} \\ I \\ x_0 \end{bmatrix}}_{R_o}
\end{aligned}
\tag{4.11}
$$

The constrained feed-rate optimization problem can be formulated by using the

objective function as the norm of Equation 4.11 subject to the control signal saturation, maximum velocity, contour error bound, and monotonic feed constraints shown in equation 4.6, 4.7, 4.9, and 4.10, respectively:

$$\min_{\mathbf{r},\delta\tau} || \begin{bmatrix} 0 & -Q_n\mathcal{H}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta\tau} \\ \mathbf{r} \end{bmatrix} + R_o||_2^2$$

subject to

$$\underbrace{\left[\begin{array}{cccc|c} & & & & \\ -1 & & & & \\ 1 & -1 & & & \mathbf{0} \\ & \ddots & \ddots & & \\ & & 1 & -1 & \\ \hline 0 & \dots & 0 & 1 & \mathbf{0} \\ \hline & & & & \\ Q_pK_v & & & & -Q_p\mathcal{H}_1 \\ -Q_pK_v & & & & Q_p\mathcal{H}_1 \\ 0 & & & & D_p\mathcal{H}_1 \\ 0 & & & & -D_p\mathcal{H}_1 \\ 0 & & & & \mathcal{H}_2 \\ 0 & & & & -\mathcal{H}_2 \end{array}\right]}_{G} \underbrace{\begin{bmatrix} \boldsymbol{\delta\tau} \\ \hline \\ \mathbf{r} \end{bmatrix}}_{x_p} \preceq \underbrace{\begin{bmatrix} \tau_{s,1} - \tau_0 \\ \tau_{s,2} - \tau_{s,1} \\ \vdots \\ \tau_{s,n} - \tau_{s,n-1} \\ \hline \\ \delta\tau_{max} \\ \hline \\ \boldsymbol{\epsilon_{max}} - R \\ \boldsymbol{\epsilon_{max}} + R \\ \mathbf{v_{max}} - R3 \\ \mathbf{v_{max}} + R3 \\ \mathbf{u_{max}} + R_2 \\ \mathbf{u_{max}} - R_2 \end{bmatrix}}_{h}$$

The scale of this constrained optimization problem is decided by the horizon size n. Ideally, the horizon length is desired to incorporate the whole trajectory. However, long horizon length requires more computation and limits the feasible sampling rate of the real-time implementation. Because n determines the length

of system dynamics incorporated in the model predictive controller, it needs to be long enough to capture the system dynamics in time domain. When long horizon length is too computationally intensive to be implemented on-line but real-time adjustment is desired, a down-sampling interpolation scheme is implemented, where x_p is interpolated by a coarser vector $\tilde{x}_p$ by a ratio of α:

$$x_p(i) = \tilde{x}_p(k) + \frac{i-(k-1)\alpha+1}{\alpha}(\tilde{x}_p(k+1) - \tilde{x}_p(k))$$
$$k = floor((i-1)/\alpha + 1$$

The down-sampling operator creates a mapping matrix S_d between the full-size vector x_p and the down-sampled vector $\tilde{x}_p$, allowing the quadratic cone problem to be solved with reduced amount of states. This structure allows the time-optimal contour tracking to be solved as a quadratic cone programming problem [61]:

$$\min_{x_p}(1/2)\tilde{x}_p^T S_d^T Q S_d \tilde{x}_p + q^T S_d \tilde{x}_p$$

subject to

$$\begin{aligned} S_d^T G S_d \tilde{x}_p + \tilde{s} &= S_d^T h \\ \tilde{s} &\succeq 0 \end{aligned}$$

For constraints related to LTI system dynamics, this down-sampling method is equivalent to the sampling period adjustment in time domain. The constraint values at the original sampling period Δt can be estimated by the inter-sample behavior at down-sampled period ΔT. Demonstrated by the continuous-time dynamics (A_c, B_c, C_c, D_c) subject to constraint h_s, this formulation applies to the contour error, signal and velocity limitations. When the system strictly satisfy the inequality constraint at sampling period ΔT, the following inequality conditions must hold:

$$C_s x_s \leq h_s$$
$$C_s e^{A_c \Delta T} x_s + C_s \int_0^{\Delta T} e^{A_c \tau} d\tau B_c r \leq h_s$$

Equation 4.12 describes systems where average velocity within ΔT is less than in Δt when given a step input. This condition is satisfied by many machines with high inertia. If the system satisfies the property shown in Equation 4.12, Equation 4.13 holds when the system is sufficiently near a positive boundary h_s. A special case is when $C_s e^{A_c t} x_s$ is convex, K_s is $\frac{1-\Delta t}{\Delta T}$.

$$\frac{1}{\Delta T} \underbrace{C_s \int_0^{\Delta T} e^{A_c \tau} B_c d\tau}_{\beta_{\Delta T}} \leq \frac{1}{\Delta t} \underbrace{C_s \int_0^{\Delta t} e^{A_c \tau} B_c d\tau}_{\beta_{\Delta t}} \tag{4.12}$$

$$\begin{aligned} C_s e^{A_s \Delta t} x_s &\leq K_s C_s x_s + \frac{\Delta t}{\Delta T} C_s e^{A_c \Delta T} x_s \\ &\leq K_s C_s x_s + \frac{\beta_{\Delta t}}{\beta_{\Delta T}} C_s e^{A_c \Delta T} x_s \end{aligned} \tag{4.13}$$

Given the properties above, the inter-sample behavior can be evaluated and constraints can be monitored as shown in Equation 4.14. K_s reflects the transient response the system propagates since the last sampling point, while $\beta_{\Delta t}/\beta_{\Delta T}$ approaches 1 when the next sampling time occurs. This inter-sample behavior analysis can be applied to the system dynamics constraints, including contour error, control signal, and velocity constraints, while monotonic feed constraint is always satisfied due to linear interpolation.

$$
\begin{aligned}
& C_s e^{A_c \Delta t} x_s + C_s \int_0^{\Delta t} e^{A_c \tau} d\tau B_c r \\
& = C_s e^{A_s \Delta t} x_s + \beta_{\Delta t} r \\
& \leq K_s C_s x_s + \frac{\beta_{\Delta t}}{\beta_{\Delta T}} C_s e^{A_c \Delta T} x_s + \beta_{\Delta t} r \\
& = K_s C_s x_s + \frac{\beta_{\Delta t}}{\beta_{\Delta T}} \left(C_s e^{A_c \Delta T} x_s + \beta_{\Delta T} r \right) \\
& \leq \left(K_s + \frac{\beta_{\Delta t}}{\beta_{\Delta T}} \right) h_s
\end{aligned}
\tag{4.14}
$$

4.7 Real-time Robustness and Efficiency

For real-time optimal control, the efficiency of the constrained optimization solver is important for sampling rate improvement. Primal-dual solver has been widely used for the benefit of computation efficiency and convergence rate [61]. The primal-dual solver iterates until the error residual converges to the defined tolerance or when the maximum iteration is reached:

Primal-dual Interior Point Algorithm

Set initial $\mathbf{x}$ $\mathbf{z}$ $\mathbf{s}$ values

Set k=1

Repeat until $|r| \leq r_{max}$ or $k \geq k_{max}$ or $\mathbf{s}^T\mathbf{z} \leq e_{max}$

1. Calculate the updating step

$$\begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{z} \\ \Delta\mathbf{s} \end{bmatrix} = \begin{bmatrix} -Q & -G^T & 0 \\ G & 0 & I \\ 0 & \lambda^0 W & \lambda^0 W^{-T} \end{bmatrix}^{-1} \begin{bmatrix} q + Q\mathbf{x} - G^T\mathbf{z} \\ h - \mathbf{s} - G\mathbf{x} \\ -\lambda^0\lambda + \sigma\mu e \end{bmatrix}$$

2. Update parameters

$$\begin{aligned} \lambda &:= W^{-T}\mathbf{s} = W\mathbf{z} \\ \mu &= \lambda^T\lambda/\theta = \mathbf{s}^T\mathbf{z}/\theta \\ W &= diag\left[\sqrt{\mathbf{s}./\mathbf{z}.}\right] \end{aligned}$$

3. Update values

$\alpha_p = min(1, -1/min(\Delta\mathbf{s}/\mathbf{s}))$

$\alpha_d = min(1, -1/min(\Delta\mathbf{z}/\mathbf{z}))$

$\alpha = min(\alpha_p, \alpha_d)$

$\sigma = ((\mathbf{s} + \Delta\mathbf{s})^T(\mathbf{z} + \Delta\mathbf{z})/\mathbf{s}^T\mathbf{z})^3$

$(\mathbf{x}, \mathbf{z}, \mathbf{s}) := (\mathbf{x}, \mathbf{z}, \mathbf{s}) + min(1, 0.99\alpha)(\Delta\mathbf{x}, \Delta\mathbf{z}, \Delta\mathbf{s})$

4. Calculate residual r

$$r = \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} - \begin{bmatrix} Q & G^T \\ -G & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} - \begin{bmatrix} q \\ h \end{bmatrix}$$

k:k+1

The primal-dual solver efficiency is mainly determined by the hardware limitations, matrix dimensions, solver efficiency, and initial condition. Given the same hardware and problem formulation, the less iteration it requires, the faster the maximum allowable sampling rate can be. Computation time can be shortened by initializing the primal-dual solver at the estimated $\mathbf{x}$, $\mathbf{z}$, and $\mathbf{s}$ values. This warm start technique has been used and studied to improve computational efficiency of constrained optimization problems [62]. Because the prediction updates iteratively, the change in boundary condition is gradual. Therefore, initiating the algorithm from solution from last iteration can greatly shorten computation time [63]. For the minimal-time task, the system inevitably operates near the limitation boundaries especially at high curvature areas. When the warm start initializes the primal-dual solver near constraint boundaries, directly applying solutions from last iteration can possibly be initialized at infeasible points, resulting in negative $\mathbf{s}$ and $\mathbf{z}$.

To avoid such infeasible initial condition, a residual regulation method is developed by initiating the warm-start algorithm at the position of the same error residual r with the updated parameters. Because the residual decreases as the algorithm propagates on the central path as shown in Figure 4.4, the error residual can be regulated within the desired residual bound. Equation 4.15 shows the derivation of error residual.

$$\begin{bmatrix} r_z \\ r_s \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} - \begin{bmatrix} Q & G^T \\ -G & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} - \begin{bmatrix} q \\ h \end{bmatrix} \tag{4.15}$$

By applying warm-start $\mathbf{x}$ value from previous iteration, the corresponding warm-start $\mathbf{s}$ and $\mathbf{z}$ can be derived. Using this residual regulation method, the primal-dual solver is guaranteed to start from a feasible point:

$$\mathbf{s} = r_s + h + G\mathbf{x}$$

$$\mathbf{z} = \mathbf{z}^- - (G^T)^+(r_z + q + Q\mathbf{x} + G^T\mathbf{z}^-)$$

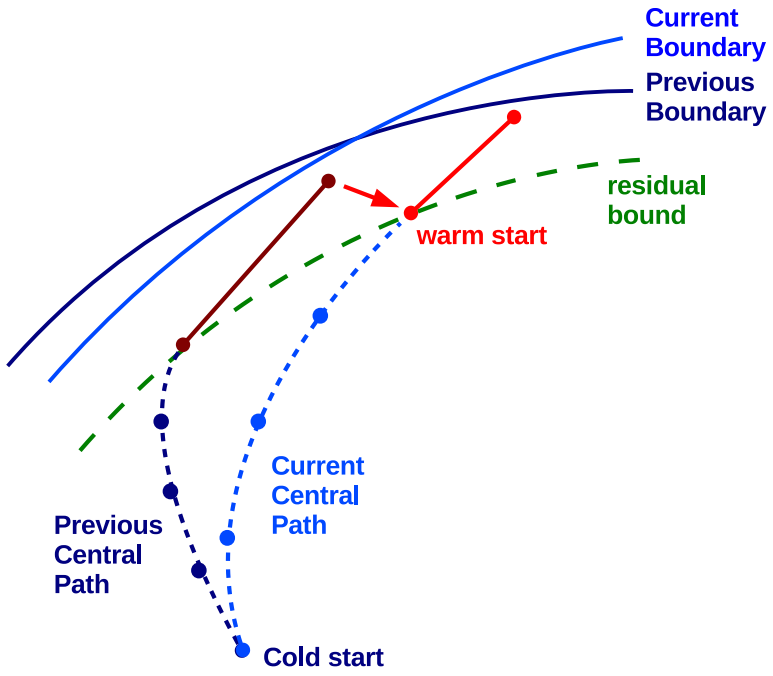


Figure 4.4: Illustration of the warm start technique with residual regulation

Another important aspect of real-time constrained optimization is solver efficiency. The more efficient the solver is, the higher the sampling rate can be. While the most time consuming step is the matrix inversion in step 1, an algebraic matrix inversion method is proposed as shown in Equation 4.16.

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} -Q & -G^T & 0 \\ G & 0 & I \\ 0 & \lambda^0 W & \lambda^0 W^{-T} \end{bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \tag{4.16}$$

where

$$M = Q + G^T W^{-1} W^{-T} G$$

$$\begin{aligned}
A_{11} &= -M^{-1} \\
A_{21} &= -W^{-1} W^{-T} G M^{-1} \\
A_{31} &= G M^{-1} \\
A_{12} &= -A_{11} G^T W^{-1} W^{-T} \\
A_{22} &= -A_{21} G^T W^{-1} W^{-T} - W^{-1} W^{-T} \\
A_{32} &= -A_{31} G^T W^{-1} W^{-T} + I \\
A_{13} &= A_{11} G^T W^{-1} \lambda^{-0} \\
A_{23} &= W^{-1} \lambda^{-0} + A_{21} G^T W^{-1} \lambda^{-0} \\
A_{33} &= A_{31} G^T W^{-1} \lambda^{-0}
\end{aligned}$$

The dimension of matrix inversion is reduced to the inversion of matrix M. The advantage of the algebraic matrix inversion can be evaluated by the FLOP counts comparison with the generic primal-dual solver algorithm: for a problem with n_c states and m_c inequality constraints the generic matrix inversion requires approximately $(n_c + 2m_c)^3$ of multiplication, while the proposed matrix inversion requires only $n_c^3 + 5m_c n_c^2$ computation. This approach prevents the increase in constraints to affect the size of matrix inversion, which can greatly improve computational efficiency.

The improvements in solver efficiency and iteration reduction are demonstrated on a dual-core 2.4 GHz real-time target running LabVIEW Real-time with 4G of RAM as shown in Table 4.1. By incorporating analytical matrix inversion into the down-sampled (5 points) constrained minimal-time contour tracking problem with warm start, the real-time target is able to achieve 1 kHz of sampling rate

for a two-dimensional contour. This improvement allows minimal-time contour tracking control to be applied on systems controlled at high sampling rate, which is common for robotic systems.

Table 4.1: Model predictive control solver efficiency

Solver	Frequency[Hz]
Primal-dual(pd)	0.0088
pd with warm start(ws)	0.088
pd with ws and analytical inversion(ai)	1
pd with ws, ai, and down-sampling	1000

4.8 Experimental Results on a Nano-precision Multi-axis Positioning Stage(MAPS)

The time-optimal contour tracking algorithm is verified on a Multi-Scale Alignment and Positioning System (MAPS) controlled at 1 kHz sampling rate. The MAPS used in this dissertation was designed for nano-manufacturing such as atomic force microscopy or imprint lithography [57, 58]. As shown in Figure 4.5, the wafer holder is levitated primarily by the air bearing system with passive in-plane damping. The in-plane motion allows MAPS to freely move in a two-dimensional plane with rotational regulation. The system dynamics can be evaluated by the impulse response shown in Figure 4.6. The coupling effect exist between x and y due to the rotational degree of freedom.

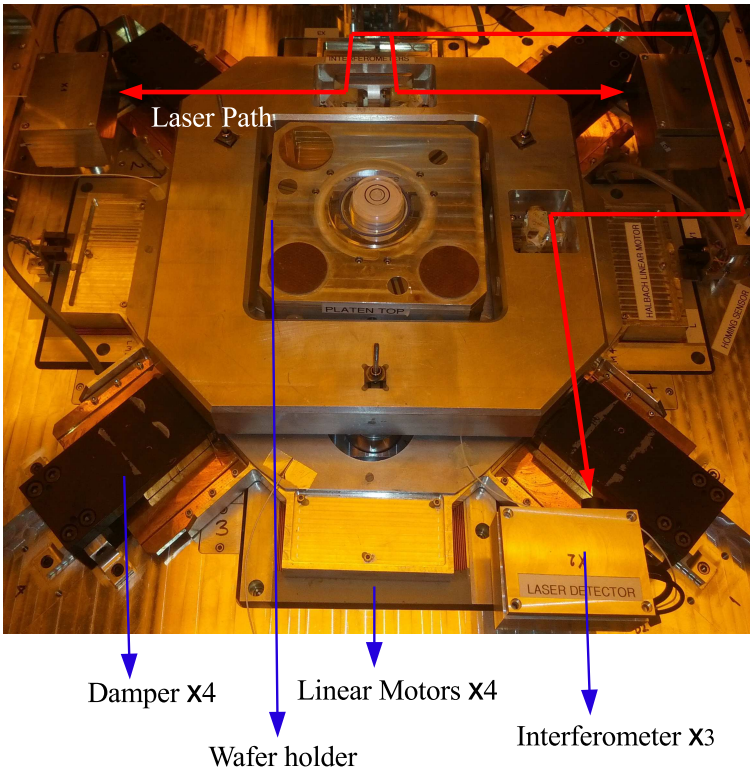
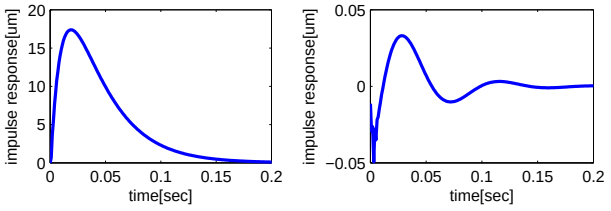
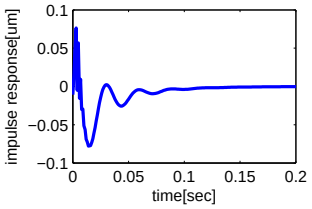


Figure 4.5: Multi-Scale Alignment and Positioning System

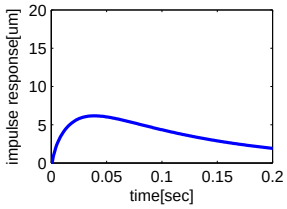


(a) input 1 to output 1

(b) input 1 to output 2



(c) input 2 to output 1



(d) input 2 to output 2

Figure 4.6: Impulse response of MAPS

In order to perform fast AFM and high throughput nano-lithography, MAPS is desired to track a two-dimensional in-plane contour while regulating the rotational angle. The faster the stage travels, the higher the throughput of the system is. The in-plane motion is measured by the interferometer sensor as the distance between the target mirrors on the wafer holder and the receivers. Because it is sensitive to misalignment as the stage travels [59, 60], axial velocity measurement of the stage is limited by the functional range of the interferometer. The maximum allowable contour error, velocity, feed-rate, and command signal saturation are defined by the values shown in table 4.2.

Table 4.2: Boundary parameters

Parameters	ϵ	v_{max}	u_{max}
Values	$10nm$	$0.5\ um/sec$	$0.2\ V$

Diamond-shape contour is chosen to demonstrate the most critical condition for minimal-time contour tracking. At straight line regions the system needs to propagate as fast as possible, which naturally drives the system near the velocity boundary. At sharp corners the system needs to decelerate predictively to avoid large contour error overshoot. The performance of the contour tracking feed-rate optimization is demonstrated by tracking a 12-segment diamond-shape contour. By using the defined constraints, 0.05 seconds of the prediction horizon, identified plant model, and Spline contour, the constrained model predictive control problem can be solved and generates reference signal r iteratively at 1 kHz. Figure 4.7 shows the minimal-time contour tracking results from both simulation and experiment. It is shown that though off-line calculation generates smooth feed-rate, the experimental result demonstrates the effect of real-time correction when the system is subject to modeling error and measurement noise.

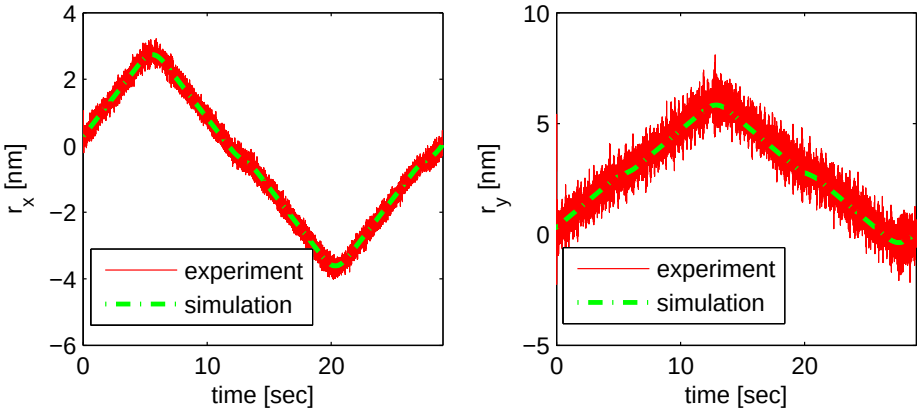


Figure 4.7: Trajectory Profile r for Diamond-shape Contour Tracking

Figure 4.8 shows the zoomed-in views for the straight-line and corner areas of the diamond shape contour tracking experiment. It is shown that the stage is able to track the diamond-shape contour both at the straight-line region and high-curvature area with bounded contour error.

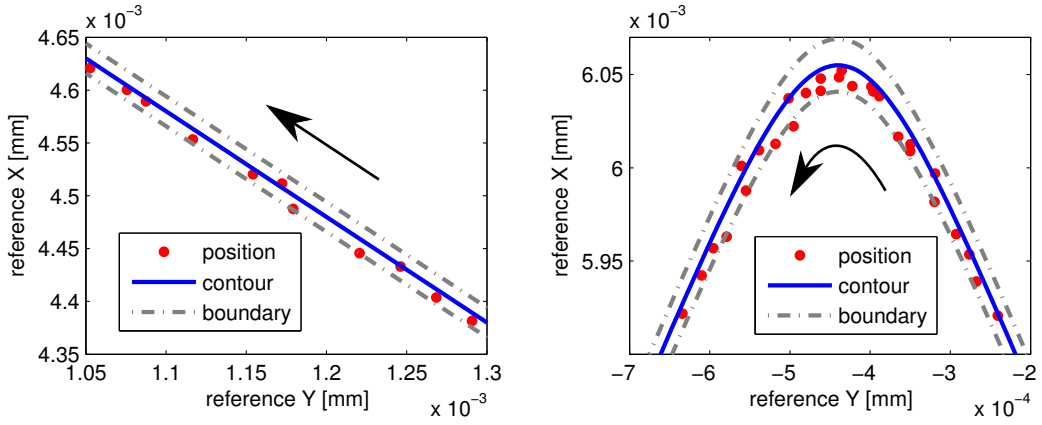


Figure 4.8: Measured Position and Boundary

The contour error and axial velocity profile corresponding to the diamond-shape contour tracking task are demonstrated in Figure 4.9. By comparing the simulation and experimental results, it is shown that the real-time model predictive controller is able to reject measurement noise and disturbance. The contour

error is suppressed to be near the given limit at the presence of measurement noise and disturbance. The feed-rate profile is shown able to slow down the stage before it reaches the high curvature areas, and allows the system to approach the maximum velocity boundary at straight line region to maximize the feed-rate.

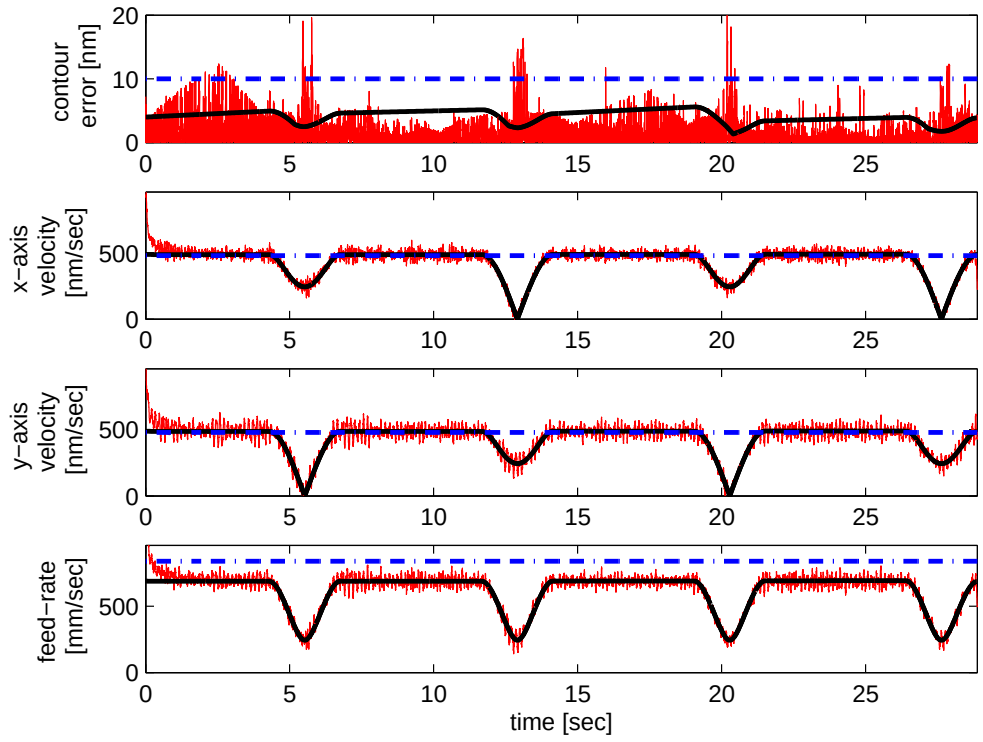


Figure 4.9: Simulated(black) and measured(red) contour error and axial velocity when $f_{max} = 0.85um/sec$

The constraint violation results are shown in Figure 4.10. At straight line regions simulation shows the system is driven near the velocity constraint. During experiments, however, modeling error and measurement noise would cause the system to oscillate near the constraint boundary and the system relies on the real-time correction to compensate for the difference between the model-based prediction and measurement.

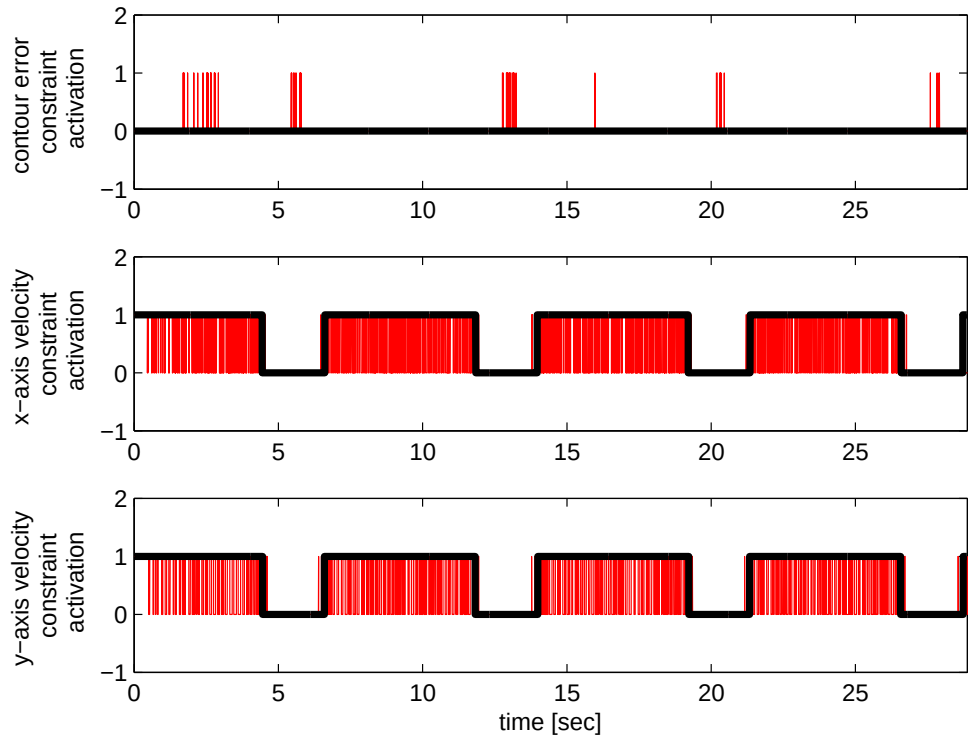


Figure 4.10: Active Constraints from simulation(black) and experiment(red)

Figure 4.11 shows the system response when the maximum feed-rate f_{max} is feasible. It is shown that when the feed index τ_{max} can be reached within the finite preview, the velocity constraint would not be triggered. The trajectory profiles of the two maximum feed-rates are shown in Figure 4.12. It is shown that feed-rate slows down at sharp corners, and the total cycle time is increased by constraining the maximum feed-rate. This offers an extra degree of freedom for feed-rate design, which is useful for applications that are desired to have bounded maximum feed-rate due to measurement noise or local feed-rate constraints.

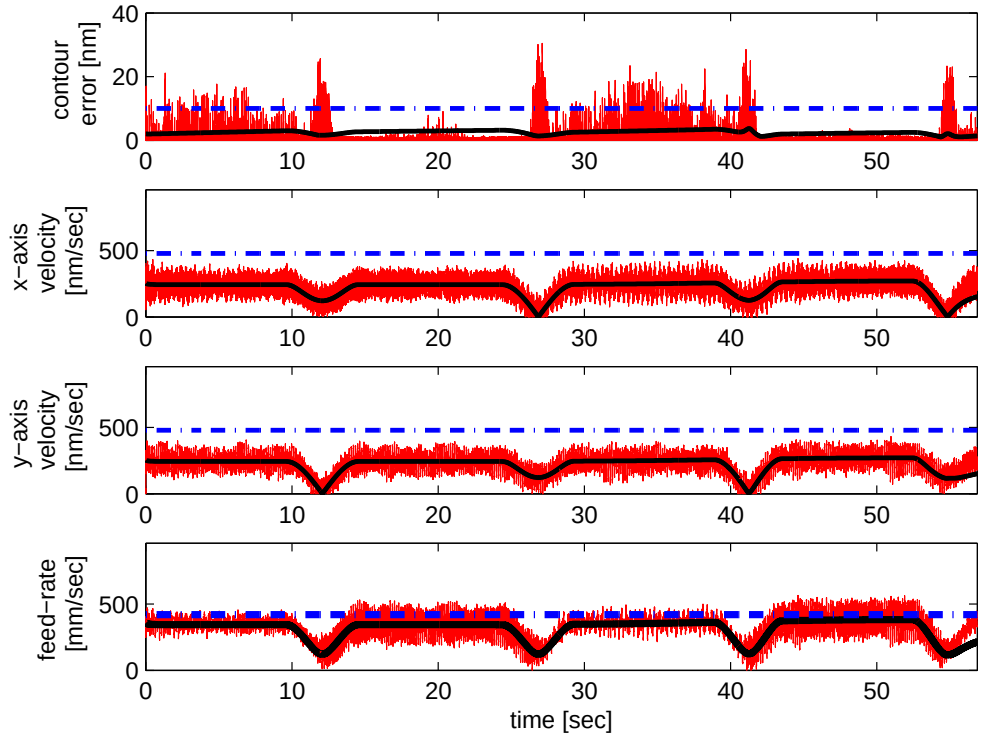


Figure 4.11: Simulated(black) and measured(red) contour error and axial velocity when $f_{max} = 0.425um/sec$

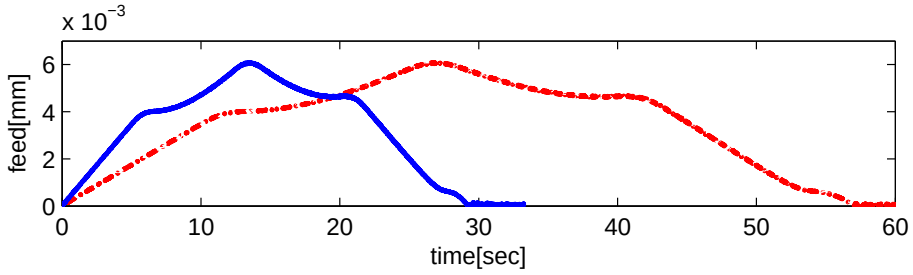


Figure 4.12: Trajectory profile for $f_{max} = 0.85um/sec$ (red-dotted) and $f_{max} = 0.425um/sec$ (blue)

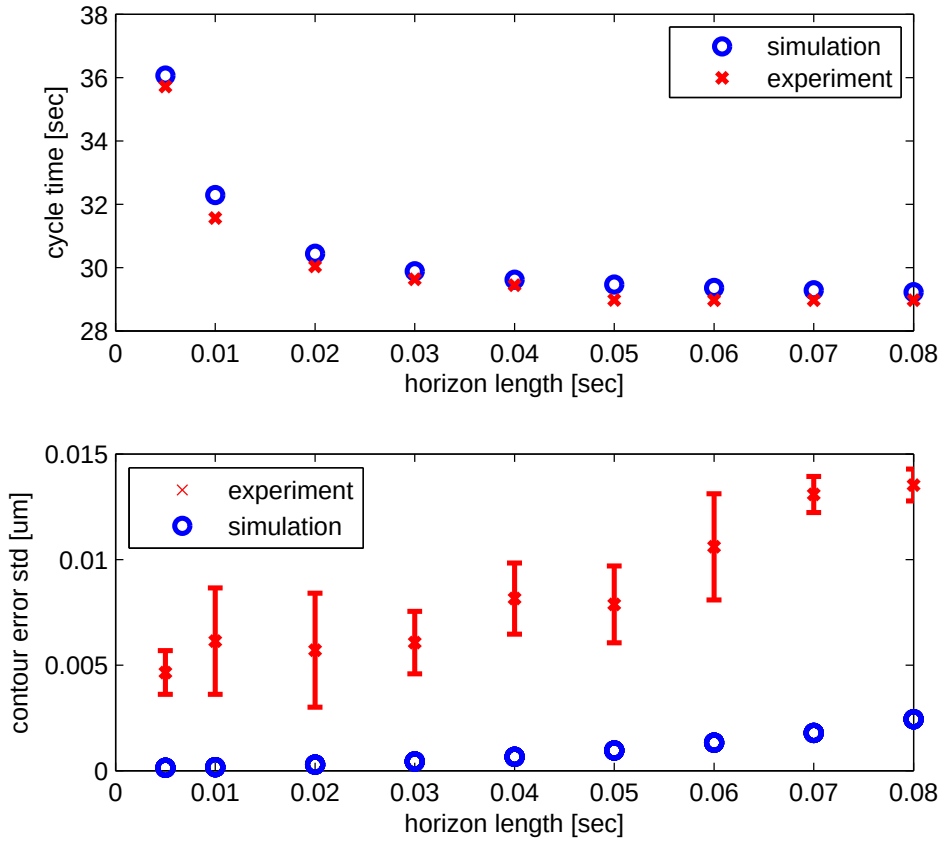


Figure 4.13: Horizon length effect on contour error and cycle time

Figure 4.13 shows the effect of horizon length on cycle time and contour error measurement, while each condition is repeated 10 times to validate the repeatability of the system. As shown in this figure, when the horizon length increases, the cycle time is improved while the contour error remains bounded. Because of the down-sampling effect, contour error exceeds the assigned boundary at 10 nm for the horizon longer than $0.06sec$. However, this improvement saturates after the horizon length exceeds $0.05sec$. Because shorter cycle time means faster transversing, the corresponding modeling error drives the system near the neighborhood of the assigned contour error constraint.

CHAPTER 5

Conclusions

In this dissertation important topics in precision motion sensing and control have been studied through constrained optimization, including sub-count sensing, constraint handling, and cycle time minimization. These algorithms were implemented and verified on nano-precision positioning stages.

In chapter 2 the sub-count estimation beyond sensor resolution is studied to improve precision motion tracking performance especially when synchronous measurement update is unavailable. To improve robustness and achieve fast convergence, a model-based estimation method with continuity restoration. By performing continuity correction on important measured and unmeasured states at each encoder trigger. This approach provides more accurate estimation beyond sensor resolution without relying heavily on the past data measurements as the non-model-based methods. The performance was verified on a multi-axis stage used on Plasmonic Imaging Lithography Machine (PILM). The algorithm is able to provide sub-count estimation with $25\% - 50\%$ of error RMS in position and $13\% - 26\%$ in velocity, and $1.1 - 1.6$ times of correlation in position and $0.8 - 1.1$ in velocity compared to the quantized measurement.

Constraint handling for challenging reference tracking is discussed in chapter 3. To prevent the system from exceeding their physical limitations, constrained model predictive control was proposed. Due to the high computational requirement for constrained optimization with long horizon, a feedforward model predictive control method is formulated to be augmented with a robust feedback

controller, which is demonstrated, by but not restricted to, a dual-stage fast tool servo system. The FF-MPC algorithm can be applied to other multi-axis applications, such as positioning and contour tracking problems. By applying the FF-MPC on a system with a robust feedback controller, modeling uncertainty can be well handled.

To improve productivity for fabrication processes, a time-optimal convex model predictive control method has been developed and is discussed in chapter 4. Cycle time is minimized subject to contour error, velocity, control saturation, and monotonic feed constraints. Because time-optimal control often drives the system near its physical limitation, a robust and efficient quadratic programming solver is developed for real-time adjustments. This proposed algorithm is verified on a Multi-Scale Alignment and Positioning System tracking a diamond-shape quintic Spline contour at 1 kHz. It is shown able to regulate the error residual when the system is driven near its physical limitation with nano-scale contour precision.

As a summary, optimization and constraint handling becomes more and more important for precision motion sensing and control. Because of the advancement in computing systems, real-time optimization and constraint handling have been made possible. This allows precision systems to compensate for un-modeled dynamics and reject external disturbances. This dissertation has demonstrated the use of constrained optimization on improving tracking error and cycle time by sub-count estimation, feedforward MPC, and minimal-time contour tracking.

References

[1] Merry, R. J. E., M. J. G. van de Molengraft, and M. Steinbuch. "Optimal higher-order encoder time-stamping." Mechatronics 23, no. 5 (2013): 481-490.

[2] Merry, R. J. E., M. J. G. Van de Molengraft, and Maarten Steinbuch. "Velocity and acceleration estimation for optical incremental encoders." Mechatronics 20, no. 1 (2010): 20-26.

[3] J. Xu, J. X. Li, and S .Xu, "Quantized innovations Kalman filter: stability and modification with scaling quantization,"Journal of Zhejiang University Science C: Computer and Electronics, vol.13, no.2, pp.118- 130, 2012.

[4] Xu, Jian, Jianxun Li, and Jiayun Wu. "Convergence of Kalman filter with quantized innovations." In Control Automation Robotics and Vision (ICARCV), 2010 11th International Conference on, pp. 1703-1708. IEEE, 2010.

[5] Sur, Joono, and Brad Paden. "Observers for linear systems with quantized outputs." In American Control Conference, 1997. Proceedings of the 1997, vol. 5, pp. 3012-3016. IEEE, 1997.

[6] Sur, Joono, and Brad E. Paden. "State observer for linear time-invariant systems with quantized output." Journal of dynamic systems, measurement, and control 120, no. 3 (1998): 423-426.

[7] Zheng, Jinchuan, and Minyue Fu. "A reset state estimator using an accelerometer for enhanced motion control with sensor quantization." Control Systems Technology, IEEE Transactions on 18.1 (2010): 79-90.

[8] Zheng, Jinchuan, and Minyue Fu. "A reset state estimator for linear systems to suppress sensor quantization effects." Proc. 17th IFAC World Congress. 2008.

[9] Bascetta, Luca, Gianantonio Magnani, and Paolo Rocco. "Velocity estimation: Assessing the performance of non-model-based techniques." Control Systems Technology, IEEE Transactions on 17, no. 2 (2009): 424-433.

[10] Jacobsen, M. P., T-C. Tsao, "Optimal Velocity Estimation of Slowly Moving System with Quantized Position Output," DSC-Vol. 64, Proc. ASME Dynamic Systems and Control Division, 479-486 , Anaheim, CA, Nov. 1998.

[11] N. Gupta, and R. Hauser. "Kalman filtering with equality and inequality state constraints." arXiv preprint arXiv:0709.2791 (2007).

[12] Sinopoli, B., Schenato, L., Franceschetti, M., Poolla, K., Jordan, M. I., and Sastry, S. S. (2004). Kalman filtering with intermittent observations. Automatic Control, IEEE Transactions on, 49(9), 1453-1464.

[13] Widrow, Bernard, Istvan Kollar, and Ming-Chang Liu. "Statistical theory of quantization." IEEE Transactions on Instrumentation and Measurement 45, no. 2 (1996): 353-361.

[14] Xu, Jian, Jianxun Li, and Sheng Xu. "Analysis of quantization noise and state estimation with quantized measurements." Journal of Control Theory and Applications vol. 9(1),pp. 66-75, 2011.

[15] Pan, L., Park, Y., Xiong, Y., Ulin-Avila, E., Wang, Y., Zeng, L., ... and Zhang, X. (2011). Maskless plasmonic lithography at 22 nm resolution. Scientific reports, 1.

[16] Bas, Omur Y., Bahram Shafai, and Stephen P. Linder. "Design of optimal gains for the proportional integral Kalman filter with application to single particle tracking." Decision and Control, 1999. Proceedings of the 38th IEEE Conference on. Vol. 5. IEEE, 1999.

[17] Speyer, Jason L., and Walter H. Chung. Stochastic processes, estimation, and control. Vol. 17. Siam, 2008.

[18] Li,J. and Tsao, T., 2001, "Robust Performance Repetitive Control Systems", Journal of Dynamic Systems, Measurement, and Control, 123, pp.330-337.

[19] Li, J. and Tsao, T., "A Two Parameter Robust Repetitive Control Design Using Structured Singular Values", 1998, Proceedings of the 37th IEEE Conference on Decision and Control, Tampa, Florida USA, pp.1230-1235.

[20] Uchiyama, N., Sano, S., Takagi, S., and Yamazaki, K., 2007, "Precision Contouring Control of Multi-Axis Feed Drive Systems", Towards Synthesis of Micro-/Nano-Systems, A1, Part 2, pp.41-46.

[21] Mcnab, R. and Tsao, T., 2000, "Receding Time Horizon Linear Quadratic Optimal Control for Multi-Axis Contour Tracking Motion Control", Journal of Dynamic Systems, Measurement, and Control, **122**, pp.375-381.

[22] Marcos, M., Machado, J. and Azevedo-Perdicoulis, T., 2009, "Trajectory planning of redundant manipulators using genetic algorithms", Commun Nonlinear Sci Numer Simulat, **14**, pp.2858-2869.

[23] Fahham, H., Farid, M. and Khooran, M., 2011, "Time Optimal Trajectory Tracking of Redundant Planar Cable-Suspended Robots Considering Both Tension and Velocity Constraints", Journal of Dynamic Systems, Measurement and Control, **133**, pp.011004-1 - 011004-14.

[24] Cocuzza, S., Pretto, I. and Debei, S., 2011, "Novel reaction control techniqu es for redundant space manipulators: Theory and simulated microgravity tests", Acta Astron Autica, **68**, pp. 1712-1721.

[25] Cuzzola, F., Geromel, J. and Morari, M., 2002, "An improved approach for constrained robust model predictive control", Automatica, **38**, pp. 1183-1189.

[26] Susanu, M. and Dumur, D., 2005, "Using Predictive Techniques within CNC Machine Tools Feed Drives", Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference, Seville, Spain, pp.5150-5155.

[27] Wan, Z. and Kothare, M., 2003, "Efficient robust constrained model predictive control with a time varying terminal constraint set", Systems and Control Letters, **48**, pp.375-383.

[28] Ozbay, U., Sahin, H., and Zergeroglu, E., 2008, "Robust tracking control of kinematically redundant robot manipulators subject to multiple self-motion criteria", Robotica, **26**, pp. 711-728.

[29] Kim, B., Jianwu Li, and Tsao, T., 2002, "Two-Parameter Robust Repetitive Control With Application to a Novel Dual-Stage Actuator for Noncircular Machining", IEEE/ASME Transactions on Mechatronics, **9**(4), pp. 644 - 652.

[30] Lin, C. and Liu, Y., 2011, "Precision Tracking Control and Constraint Handling of Mechatronic Servo Systems Using Model Predictive Control", IEEE/ASME Transactions on Mechatronics, 99, pp.1 - 13.

[31] Liu Y. and Lin, X., 2010, "Model Predictive Control with Repetitive Control for Periodic Signal Tracking and Constraint Handling of Fast Sampled -Data Control Systems", 2010 IEEE International Conference on Control Applications, Yokohama, Japan, pp. 626-631.

[32] Cao, R. and Low, K., 2009, "A Repetitive Model Predictive Control Approach for Precision Tracking of a Linear Motion System", IEEE Transactions on Industrial Electronics,**56**(6), pp.1955-1962.

[33] Gu, D. and Hu, H., 2006, "Receding Horizon Tracking Control of Wheeled Mobile Robots", IEEE Transactions on Control Systems Technology, **14**(4), pp.743-749.

[34] Scokaert, P. and Mayne, D., 1998, "Min-Max Feedback Model Predictive Control for Constrained Linear Systems", IEEE Transactions on Automatic Control, **43**(8), pp. 1136-1142.

[35] Kim, J. and Crassidis, J., 2001, "Model-Error Control Synthesis Using Approximate Receding-Horizon Control Laws", American Institute of Aeronautics and Astronautics, 2001-4220, pp.1-11.

[36] Brus, L., Wigren, T., and Zambrano, D., 2010,"Feedforward model predictive control of a nonlinear solar collector plant with varying delays", IET Control Theory and Applications,**4**(8),pp.1421-1435.

[37] Carrasco, D. and Goodwin, G., 2011, "Preview and Feedforward in Model Predictive Control: A Preliminary Robustness Analysis", 18th IFAC World Congress Milano (Italy), pp.185-190.

[38] Carrasco, D. and Goodwin, G., 2011, "Feedforward model predictive control", Annual Reviews in Control,**35**, pp.199-206.

[39] Gross, E., Tomizuka, M., and Messner, W., 1994, "Cancellation of Discrete Time Unstable Zeros by Feedforward Control", Journal of Dynamic Systems, Measurement, and Control, **116**(1), pp.33-38.

[40] Garca-Gabn, W., Camacho E.F., and Zambrano, D., 2000, "Improving GPC Tuning for non-minimum phase systems", Controlo'2000: 4th Portuguese Conference on Automatic Control, Guimares, Portugal, pp. 66-71.

[41] Garca-Gabn, W., Zambrano, D.,and Camacho E.F., 2002, "Multivariable Model Predictive Control of processes with unstable transmission zeros", Proceedings of the American Control Conference, Anchorage, USA, pp. 4189-4190.

[42] Y. Koren, 1980, "Cross-Coupled Biaxial Computer Control for Manufacturing Systems", J. of Dynamic Systems, Measurement, and Control, **102**, 265-271.

[43] P. K. Kulkarni and K. Srinivasan, 1989, "Optimal Contouring Control of Multi-Axial Feed Drive Serwomechanisms", Transactions of the ASME, **111**, pp.140-148.

[44] G. Chiu and B. Yao, 1997, "Adaptive Robust Contour Tracking of Machine Tool Feed Drive Systems- A Task Coordinate Frame Approach", Proceedings of the American Control Conference, Albuquerque, 2731-2735.

[45] Bemporad, Alberto, Alessandro Casavola, and Edoardo Mosca. "Nonlinear control of constrained linear systems via predictive reference management." Automatic Control, IEEE Transactions on 42.3 (1997): 340-349.

[46] Morari, Manfred, and Jay H Lee. "Model predictive control: past, present and future." Computers and Chemical Engineering 23.4 (1999): 667-682.

[47] Dufour, Pascal, Francoise Couenne, and Youssoufi Tour. "Model predictive control of a catalytic reverse flow reactor." Control Systems Technology, IEEE Transactions on 11.5 (2003): 705-714.

[48] Kang, Yeonsik, and J. Karl Hedrick. "Linear tracking for a fixed-wing UAV using nonlinear model predictive control." Control Systems Technology, IEEE Transactions on 17.5 (2009): 1202-1210.

[49] R. McNab and T. Tsao, 2000, "Receding Time Horizon Linear Quadratic Optimal Control for Multi Axis Contour Tracking Motion Control", Journal of Dynamic Systems, Measurement, and Control, **122**, pp.375-381.

[50] D. Lam, C. Manzie, M. Good, and R. Bitmead, 2012 "Model Predictive Contouring Control for Biaxial Systems", IEEE Transactions of Control Systems Technology, **21**(2), 552-559.

[51] M. Susanu and D. Dumur, 2005, "Using Predictive Techniques within CNC Machine Tools Feed Drives", IEEE Conference of Decision and Control, and European Control Conference, Seville, Spain, pp.5150-5155.

[52] Lie Tang and Robert G. Landers, 2012, "Predictive Contour Control With Adaptive Feed Rate", IEEE/ASME Transactions of Mechatronics, **17**(4), pp.669-679.

[53] K. Erkorkmaz and Y. Altintas, 2001, "High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation", Int'l J. of Machine Tools and Manufacture,**41**, 1323-1345.

[54] Y. Altintas and K. Erkorkmaz, 2003, "Feedrate Optimization for Spline Interpolation in High Speed Machine Tools", CIRP Annals - Manufacturing Technology, **52**(1), pp.297-302.

[55] D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl, 2009, "Time-optimal path tracking for robots: a convex optimization approach", Automatic Control, IEEE Transactions on, 54(10), 2318-2327.

[56] D. Lam, C. Manzie, and M. Good, 2013 "Receding Horizon Time-Optimal Control for a Class of Differentially Flat systems", Automatica.

[57] R. Fesperman, O. Ozturk, R. Hocken, S. Ruben, T. Tsao, J. Phipps, T. Lemmons, J. Brien, and G. Caskey. "Multi-scale Alignment and Positioning System-MAPS." Precision Engineering 36, no. 4 (2012): 517-537.

[58] S. Ruben, "Modeling, Control, and Real-Time Optimization for a Nano-Precision System", Ph.D. dissertation, MAE, UCLA, Los Angeles, CA, 2010.

[59] Bobroff N, Critical alignments in Plane Mirror Interferometry, Precision Engineering, Jan 1993, Vol 15, No 1, p.33-38

[60] Ruben, Shalom D., et al. "Mechatronics and control of a precision motion stage for nano-manufacturing." ASME 2009 Dynamic Systems and Control Conference. American Society of Mechanical Engineers, 2009.

[61] S. Boyd and L. Vandenberghe, 2009, "Convex Optimization", Cambridge University Press.

[62] E. John and E. Alper Yldrm. "Implementation of warm-start strategies in interior-point methods for linear programming in fixed dimension." Computational Optimization and Applications 41, no. 2 (2008): 151-183.

[63] Y. Wang and S. Boyd. "Fast model predictive control using online optimization." Control Systems Technology, IEEE Transactions on 18, no. 2 (2010): 267-278.

[64] H. Zhang and R. Landers, 2007, "Precision Motion Control Methodology for Complex Contours", American Control Conference, New York, pp.4333-4338.

[65] A. El Khalick M. and N. Uchiyama, 2011, "Discrete-time model predictive contouring control for biaxial feed drive systems and experimental verification", Mechatronics, **21**, pp.918-926.

[66] Chang, Yen-Chi, and Tsu-Chin Tsao. "Minimum-time contour tracking with model predictive control approach." American Control Conference (ACC), 2014. IEEE, 2014.