L_1 ADAPTIVE CONTROL IN AN ITERATIVE LEARNING CONTROL FRAMEWORK FOR PRECISION NANOPOSITIONING

Berk Altın[1] and Kira Barton[2]
[1]**Department of Electrical Engineering and Computer Science**
University of Michigan
Ann Arbor, Michigan, USA
[2]**Department of Mechanical Engineering**
University of Michigan
Ann Arbor, Michigan, USA

INTRODUCTION

Iterative learning control (ILC) is a feedforward control strategy aimed towards systems that execute the same task repetitively [1]. ILC is based on the idea that the performance of such systems can be improved by using information from previous trials. ILC modifies the control input rather than the controller itself for better tracking performance [1, 2]. As such, ILC can be thought of as feedback in the iteration domain. Naturally, this property equips iterative learning controllers with simplicity, robustness and fast convergence to iteration domain equilibria with a significant decrease in error metrics up to several orders of magnitude.

Research on robust ILC has focused on disturbance rejection, stochastic effects, transient growth, μ synthesis, and robustness to high frequency modeling uncertainties [3, 4]. Previous work by the authors presented the design of ILC algorithms for systems with large parametric uncertainties [5, 6]. Robustness of control algorithms in the time and iteration domains is especially important as applications with parametric uncertainties (multi-agent systems, precision motion controllers, prosthetics) requiring monotonic behavior and high tracking performance can benefit from it.

Recent work by Parmar et al. [7] presented the development of a single axis flexure bearing based large range nanopositioner. In [7], nonlinearities associated with the actuator and its driver, endogenous noise components, and the low open loop bandwidth prevented conventional feedback controllers from achieving nanometric tracking performance. Subsequently, ILC was employed with a feedback controller to achieve a large working range, high scanning speeds, and nanometric tracking performance [8]. The combination of linear ILC and feedback resulted in a 10nm root mean squared error (RMSE) over a 4mm range 2Hz bandlimited triangular profile. Nevertheless, although traditional ILC enabled compensation of the aforementioned limitations; parametric uncertainties due to changing payloads and position dependent nonlinearities must be addressed to fully realize the potential of large range nanopositioners.

To overcome iteration varying factors such as these, the combination of an L_1 adaptive feedback controller with an ILC feedforward controller into a single framework was proposed [5]. In [5], the L_1 adaptive controller was designed to compensate for nonrepetitive, low frequency (parametric) uncertainty in the time domain, and ensure that the plant as seen from the feedforward input was sufficiently close to its nominal value. On the other hand, the iterative learning controller was utilized to compensate for repetitive system uncertainties in the iteration domain. L_1 adaptive control was preferred over more conventional forms (e.g. model reference adaptive control) because of its guaranteed robustness bounds (stability of the feedback loop is a necessary condition for ILC), along with a priori known, quantifiable, steady state and transient performance.

In this paper, we present a modified L_1 adaptive control architecture to accommodate parallel ILC signals [6] and prevent the trade-off between time and iteration domains previously found in [5]. We then compare the combined framework against conventional ILC on the large range nanopositioner [7, 8] under parametric uncertainties through simulation.

L_1 ADAPTIVE CONTROL

L_1 adaptive control theory is a recently developed methodology [9] with guaranteed transient performance and robustness in the presence of fast adaptation. The critical feature of L_1 adaptive control theory is the *decoupling of estimation and control*, realized by the insertion of a bandlimited filter at a *particular* point in the architecture. In L_1 adaptive control, adaptation rates can be increased arbitrarily; although practical concerns such as hardware speed and noise may limit achievable performance. The performance-robustness trade-off of L_1 systems is defined by the bandwidth of the filter and can be adressed with tools from classical and robust control. Consequently, uniform performance bounds on all system signals can be enforced without resorting to gain scheduling, persistency of excitation or high gain feedback.

L_1 adaptive control algorithms have been developed for a wide variety of classes. We now present the L_1 architecture for single-input single-output (SISO) linear time invariant (LTI) systems with unknown constant parameters. To put the L_1-ILC problem into a meaningful format, we augment the original controller [9] with a bounded feedforward signal. This makes sure that the ILC signal does not act as a disturbance to the L_1 controller and overcomes the trade-off between closed loop bandwidth and ILC performance observed in [5].

Problem Formulation

We consider the following class of systems

$$\begin{aligned} &\dot{x}(t) = Ax(t) + B(u(t) + \theta^T x(t)),\ x(0) = x_0, \\ &y(t) = Cx(t). \end{aligned} \tag{1}$$

where $x(t)$ is the n dimensional measured state vector; $u(t)$ is the control input; B, C^T are known constant vectors; A is a known constant matrix, with (A, B) controllable; θ is an unknown constant vector such that $\|\theta\|_\infty \leq m$ for some known m; and $y(t)$ is the output signal. The L_1 adaptive controller ensures transient and steady-state behavior in the input and output channels in relation to the L_1 reference system. The reference system is described by the triple (A_m, B, C), the uncertain parameter θ, and the strictly proper bounded-input bounded-output (BIBO) stable transfer function $D(s)$ with DC gain 1 and zero state space initialization. A_m is a stable matrix which describes the desired dynamics and can be achieved through state feedback. $D(s)$ is subject to the L_1 norm stability condition

$$\lambda \triangleq \|G(s)\|_{L_1} mn < 1, \tag{2}$$

where $G(s) \triangleq H(s)(1 - D(s))$, and $H(s) \triangleq (s\mathbb{I} - A_m)^{-1}B$, which guarantees bounded-input bounded-state (BIBS) stability of the reference system. The feedforward augmented closed loop reference system is then defined as

$$\begin{aligned} &\dot{x}_{ref}(t) = Ax_{ref}(t) + B(u_{ref}(t) + \theta^T x_{ref}(t)), \\ &y_{ref}(t) = Cx_{ref}(t), \\ &U_{ref}(s) = -K^T X_{ref}(s) + U_i(s) \\ &\qquad\qquad + D(s)(K_g R(s) - \theta X_{ref}(s)), \end{aligned} \tag{3}$$

with initial condition $x_{ref}(0) = x_0$, where K is the state feedback gain such that $A_m = A - BK^T$; $U_i(s)$ is a bounded input signal; $K_g = 1/(CH(0))$ is a static precompensator; and $R(s)$ is the reference signal.

L_1 Adaptive Controller

The L_1 adaptive controller is based on the fast estimation scheme which makes use of a state predictor, the bounded feedforward input $u_i(t)$ and the bandlimited filter $D(s)$.

State Predictor

The control law relies on the following state predictor

$$\dot{\hat{x}}(t) = A_m\hat{x}(t) + B(\hat{\theta}^T(t)x(t) + u(t)) - K_{sp}\tilde{x}(t), \tag{4}$$

with initial condition $\hat{x}(0) = x_0$, where $\hat{x}(t)$ is the state prediction vector; $\hat{\theta}(t)$ is the estimate of the unknown vector θ; $\tilde{x}(t) \triangleq \hat{x}(t) - x(t)$ is the prediction error; and K_{sp} is gain matrix that can be used to assign faster poles to $(A_m - K_{sp})$ [10].

Adaptation Law

The adaptation law that estimates θ is

$$\dot{\hat{\theta}}(t) = \Gamma Proj(\hat{\theta}(t), -\tilde{x}^T(t)VBx(t)), \tag{5}$$

where the initial condition $\hat{\theta}(0) = \hat{\theta}_0$ is arbitrary provided $\|\hat{\theta}_0\|_\infty \leq m$, ; $Proj(.,.)$ is the "smooth projection" operator defined in [11]; $\Gamma > 0$ is the adaptation rate; and $V = V^T > 0$ is the solution to the algebraic Lyapunov equation $A^T V + VA = -Z$, for arbitrary positive definite symmetric matrix Z. The projection operator ensures that $\|\hat{\theta}(t)\|_\infty \leq m$ by definition and thus keeps the parameter estimate bounded.

Control Law

The control input is defined as

$$\begin{aligned} &u(t) = u_i(t) + u_m(t) + u_{ad}(t), \\ &u_m(t) \triangleq -K^T x(t), \\ &U_{ad}(s) \triangleq D(s)(K_g R(s) - \hat{\eta}(s)), \end{aligned} \tag{6}$$

where $u_i(t)$, $u_m(t)$ and $u_{ad}(t)$ are the feedforward, static feedback and adaptive feedback signals, respectively, and $\hat{\eta}(s)$ is the Laplace transform of $(\hat{\theta}^T(t)x(t))$.

The closed loop system with control (6) defined according to (4) and (5), together with the stability condition (2), is stable. In addition, the system has uniform performance bounds on *both the input and the output*:

$$\begin{aligned}\|x_{ref} - x\|_{L_\infty} &\leq \tfrac{\phi_1}{\sqrt{\Gamma}}, \lim_{t\to\infty}(x_{ref}(t) - x(t)) = 0,\\ \|u_{ref} - u\|_{L_\infty} &\leq \tfrac{\phi_2}{\sqrt{\Gamma}}, \lim_{t\to\infty}(u_{ref}(t) - u(t)) = 0,\end{aligned} \tag{7}$$

where ϕ_1 and ϕ_2 are constants dependent on system parameters. In other words, arbitrary close model tracking can be achieved by increasing Γ. As ILC uses information from input and output channels, this property enables the use of the reference model in designing the ILC update law. Moreover, the reference system can be made arbitrarily close to the desired system [9] by increasing the bandwidth of $D(s)$. This, however, comes at the expense of reduced robustness. Further details of the stability analysis and derivation of (7) for the nonaugmented system can be found in [9].

ITERATIVE LEARNING CONTROL

ILC architectures can be broadly classified in two groups as parallel and series. In essence, the two architectures can be found to be equivalent when the input signals are rearranged. The parallel architecture, which we use in our controller, divides the input signal into feedback and feedforward components. In this approach, the learning controller outputs the feedforward signal for the next iteration by processing the error and the feedforward input at the current iteration.

ILC design methods are numerous and include frequency domain, plant inversion, H_∞ and norm optimization techniques. Frequency methods, whilst only approximating the system (due to finite trial duration), offer simplicity, flexibility and tunability as in classical control. The learning controllers that we use in this paper are designed using frequency domain methods.

ILC Update Law

A common first order frequency domain ILC algorithm, which we will employ in our controller, is the Q filter and learning function approach:

$$U_{i+1}(s) = Q(s)(U_i(s) + L(s)E_i(s)). \tag{8}$$

In (8), $U_i(s)$ is the ILC input; $Q(s)$ is the Q filter; $L(s)$ is the learning function; $E_i(s)$ is the reference tracking error; and i is the iteration index. In this algorithm, $L(s)$ is designed to maximize learning, while $Q(s)$ is used to limit the bandwidth to robustify the system and for other practical purposes.

Stability and Robustness

The ILC problem can be simplified by designing the ILC update law for the reference model (3). Nevertheless, due to the fact that the L_1 controller aims to compensate for the system uncertainty *within the bandwidth of* $D(s)$, system uncertainty will still exist. For simplicity of notation, we drop the subscript ref. Then, assuming zero initial conditions, the reference model dynamics for ILC are defined as

$$Y_i(s) = P'(s)U_i(s) + P'(s)K_g D(s)R(s), \tag{9}$$

where $P'(s) \triangleq P(s)W(s)$, with $P(s) \triangleq CH(s)$ and $W(s) \triangleq \frac{1}{1-\theta G(s)}$. The ILC update law (8) is monotonically stable for the nominal system when

$$\gamma = \|Q(s)(1 - L(s)P(s))\|_\infty < 1, \tag{10}$$

where $\gamma < 1$ is the convergence rate. The condition implies $\|e_\infty - e_{i+1}\|_{L_2} < \gamma\|e_\infty - e_i\|_{L_2}$, where $e_\infty(t)$ is the converged error. Robustness of the update law against parametric uncertainty is guaranteed by the bound

$$\alpha < \frac{\gamma - |Q(j\omega)||1 - L(j\omega)P(j\omega)|}{|Q(j\omega)||L(j\omega)||P(j\omega)|}, \tag{11}$$

where $\alpha = \frac{m\sqrt{n}\|G(s)\|_\infty}{1-m\sqrt{n}\|G(s)\|_\infty}$. If (11) is satisfied, the system is monotonically stable with rate γ for any θ with $\|\theta\|_\infty \leq m$.

Assuming a stable update law, the iteration domain equilibrium can be expressed as

$$\begin{aligned}U_\infty(s) &= \tfrac{Q(s)L(s)}{1-Q(s)(1-L(s)P'(s))}(1 - F(s))R(s),\\ E_\infty(s) &= \tfrac{1-Q(s)}{1-Q(s)(1-L(s)P'(s))}(1 - F(s))R(s),\end{aligned} \tag{12}$$

where $F(s) \triangleq P'(s)K_g D(s)$. The tracking error is minimized within the bandwidth of the Q filter. The readers are referred to [1] for further details on frequency domain ILC design methods and [6] for a rigorous analysis of the combined L_1-ILC scheme.

DESIGN TRADE-OFFS

The trade-offs between iteration and time domain properties can be observed via the inequality $|L(j\omega)||P(j\omega)| \leq \left(\frac{\gamma}{|Q(j\omega)|}+1\right)(|1-\theta H(j\omega)| + |D(j\omega)||\theta H(j\omega)|)$ [6]. Since $D(s)$ and $Q(s)$ describe the performance-robustness trade-offs in their respective domains, generally speaking, the following design trade-offs can be deduced:

1. Increasing the bandwidth of $D(s)$ decreases the convergence rate γ, i.e. faster iteration domain performance. Indirectly, a higher bandwidth also results in better iteration domain robustness, thereby leaving the possibility of higher gain Q filters for enhanced performance: As the L_1 filter bandwidth increases, the minimum γ in (11) becomes bounded further away from 1 and naturally, α decreases since $G(s) \triangleq H(s)(1-D(s))$. As a result, the designer can tune $Q(s)$ to increase its bandwidth and minimize the converged error.

2. Decreasing the bandwidth of $Q(s)$ decreases the minimum allowable γ that would satisfy (11), which signifies increased iteration domain robustness. This further implies that one can use a lower gain $D(s)$ for a feedback system with better stability margins: Because $Q(s)$ has a lower gain, there exists a higher value of α satisfying (11) for the original value of γ.

Thus, the design trade-offs for the combined adaptive-learning controller can be summarized as that of performance versus robustness. Intuitively, this is to be expected as increasing the passband of $D(s)$ decreases parametric uncertainty ($W(s) = [1-\theta H(s)(1-D(s))]^{-1}$), which is the desired result from an ILC perspective.

PRACTICAL CONSIDERATIONS AND DESIGN OF THE COMBINED CONTROLLER

In the design of the combined controller, the first step is the selection of the desired dynamics. The readers should note that the use of a static feedback gain is not necessary and A_m can be simply chosen to be A (assuming stable dynamics) for simplicity. However, satisfying the L_1 norm condition in (2) can be difficult and might lead to high gain feedback when the desired dynamics are far from the actual system dynamics. A good strategy would be to take A to describe the nominal open loop dynamics and use static feedback to achieve A_m. Additionally, it should be observed that $\|G(s)\|_{L_1}$ can also be decreased by increasing the damping ratio of the desired pole locations.

After A_m is chosen, the next step is deciding on the structure and bandwidth of $D(s)$. While there is no rigorous relationship between the order of $D(s)$ and the L_1 norm of $G(s)$, the reference system can be made arbitrarily close to the desired system by increasing the bandwidth at the expense of reduced robustness.

For a given filter, the L_1 adaptive controller has guaranteed (bounded away from zero) robustness margins. Thus, in order to track the reference system, Γ should be increased as much as hardware permits. However, large values of the adaptive gain might also amplify noise and thus negatively affect closed loop performance. Also note that when the adaptive gain and consequently tracking performance is impeded by these practical concerns, ILC can be more effective for large reference amplitudes: As the signals increase in amplitude, the signal to "noise" (tracking error, see (7)) ratio increases.

Once the adaptive controller parameters are selected, the learning function can be designed on the nominal system via the well known Nyquist tuning method. The Q filter can then be used to robustify the system against high frequency dynamics and noise, and to ensure monotonic robust stability by satisfying (10) and (11).

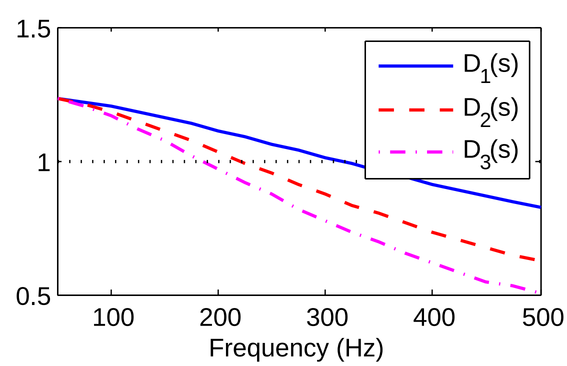

FIGURE 1. L_1 stability condition as a function of cutoff frequency for different filters. Here, $D_1(s) = 1-(\frac{s}{s+2\pi f_c})$, $D_2(s) = 1-(\frac{s}{s+2\pi f_c})^2$, *and* $D_3(s) = 1-(\frac{s}{s+2\pi f_c})^3$.

SIMULATION RESULTS

We now present the simulation results of the L_1-ILC scheme on the large range nanopositioner from [7, 8]. The nominal open loop transfer function from the actuator input has been identified in [8] as

$$P(s) = \frac{9\times10^9(s^2+5.63s+3.34\times10^5)}{(s+141.5)(s^2+159.50s+5.01\times10^4)} \times \frac{1}{(s^2+12.43s+3.87\times10^5)}. \quad (13)$$

In [8], the authors used the following output compensator to improve the dynamics of the nanopositioner:

$$C(s) = \frac{1.57\times10^4(s+141.5)}{s(s+4000)} \times \frac{(s^2+159.50s+5.01\times10^4)}{(s^2+6700s+1.92\times10^7)}. \quad (14)$$

Since the transfer function of the plant was obtained through system identification, the MATLAB® function *balreal* was used to come up with the state matrices A, B and C, and m was set to 1.5. The desired closed loop eigenvalues were then selected to be the poles of the reduced order (5th) closed loop system, which results in near identical step responses. It can be easily seen that the unknown feedback gain θ with $\|\theta\|_\infty \leq m$ has negligible effects on the closed loop time and frequency responses for the state feedback case. However, the uncertainty results in an over 250 Hz variation on the fastest closed loop pole (nominal frequency of 891 Hz) and over 50 Hz variation on the faster complex pole pair (nominal frequency of 339 Hz with a damping ratio of 0.24).

For the L_1 controller, the low-pass filter $D(s)$ was chosen to be $1-(\frac{s}{s+2\pi f_c})^3$, where the cutoff frequency f_c is 400 Hz. $D(s)$ was selected as a 3rd order filter to better attenuate the high frequencies in the input signal and satisfy the L_1 norm condition without having excessive bandwidth in the adaptive controller (see figure 1). The desired dynamics and feedback filter results in the robustness parameter α to be equal to 0.196. K_{sp} was taken to be 0 for simplicity, and Γ was set to 10^{12} for close tracking of the reference system.

The results concerning ILC stability and robustness can be identically extended to the discrete time case. Hence, for simplicity, we design the

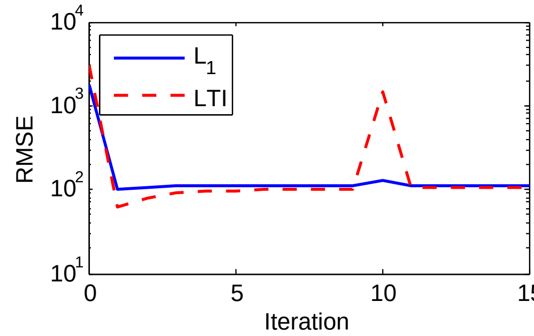

FIGURE 2. Transients in the learning controllers due to change of parameters. Scenario 1.

update law in the z domain at a sampling frequency of 10 kHz. We choose a phase lead type learning function and normalize with the DC gain of the nominal system, letting $L(z) = K_g z^{-}10$. The Q filter is chosen as a 5th order discrete Butterworth filter. To avoid phase lag in the control signal, $Q(z)$ is employed as a zero phase filter using the MATLAB® function *filtfilt*. For comparison, we also consider the output feedback learning controller on the closed loop system in [8]. On the output feedback based ILC, we let $L(z) = z^{-10}$ and use the same Q filter, injecting the ILC input before the feedback compensator. We present two different scenarios to expose the advantages of the L_1-ILC scheme over conventional ILC with output feedback. In the first scenario, the initial value of the parameter θ is set to $\begin{bmatrix}0.1 & 0 & 0 & 0 & 0\end{bmatrix}^T$ and changed to $\begin{bmatrix}-0.1 & 0 & 0 & 0 & 0\end{bmatrix}^T$ at the 10th iteration. For the second scenario, we begin with $\theta = \begin{bmatrix}1.5 & 0 & 0 & 0 & 0\end{bmatrix}^T$ and switch to $\theta = \begin{bmatrix}-1.5 & 0 & 0 & 0 & 0\end{bmatrix}^T$, again at the 10th iteration. We choose to look at the effects of the variations in the first element since x_1 (state 1) has the highest contribution to the output: Hankel singular value of x_1 is 52% more in the closed loop system when compared to other states.

The results of the first case can be seen in figure 2. While the LTI output feedback based system shows slightly lower converged RMSE values, it suffers an 1383% increase in RMSE due to a very slight change in parameters, whereas the L_1-ILC scheme only shows an 18% increase in the same iteration. In the second scenario (see figure 3), the parameter change causes a 825% increase in RMSE for L_1 feedback but converges back to a close equilibrium. In the same case, the output feedback system shows an increase of 29168% before going unstable. It should be kept in mind that in both cases, better results can be achieved through less agressive learning.

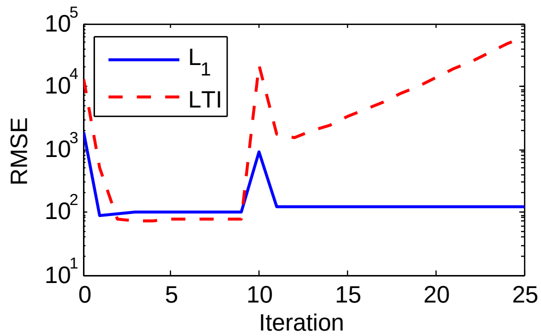

FIGURE 3. Transients in the learning controllers due to change of parameters. Scenario 2.

We remind the readers that neither the adaptive controller nor the learning controllers were optimized for a set of design specifications and hardware constraints, but rather designed to make a rough comparison under a similar set of conditions. A comparison of the L_1-ILC scheme with unknown input gain and disturbance against conventional *state feedback* was previously done in [5]. Despite the direct "plug-in" L_1-ILC approach (the adaptive controller was not modified), it was shown that the scheme was far superior to nonadaptive feedback based ILC. In [6], we did extended numerical simulations showcasing the properties of the L_1-ILC scheme in this paper. We would also like to note that [6] gives a better example in showing the effectiveness of *adaptive feedback* since the system in question is open loop with slow dynamics (mass-spring-damper system with a natural frequency of 1 rad/s), wherein the uncertain parameter is large enough to destabilize the closed loop system.

CONCLUSIONS

While the state feedback L_1-ILC scheme shows promising results, extension to the output feedback case is necessary for feasible implementation on applications without full state measurement; such as the large range nanopositioner. Future work will concentrate on experimental verification and extending the results to the output feedback and multiple-input multiple-output (MIMO) cases with additional uncertainties.

ACKNOWLEDGMENTS

This work was supported by startup funds from the University of Michigan.

REFERENCES

[1] Bristow D, Tharayil M, Alleyne A. A Survey of Iterative Learning Control. IEEE Control Systems Magazine. 2006; 26(3): 96-114.

[2] Moore K. Iterative Learning Control for Deterministic Systems. Springer-Verlag, London: 1993.

[3] De Roover D, Bosgra O. Synthesis of Robust Multivariable Iterative Learning Controllers with Application to a Wafer Stage Motion System. International Journal of Control. 2000; 73(10); 968-979.

[4] Moon JH, Doh TY, Chung MJ. A Robust Approach to Iterative Learning Control for Uncertain Systems. Automatica. 1998; 34(8): 1001-1004.

[5] Barton K, Mishra S, Xargay E. Robust Iterative Learning Control: L1 Adaptive Feedback Control in an ILC Framework. American Control Conference. 2011; 3663-3668.

[6] Altın B and Barton K. L_1 Adaptive Control in an Iterative Learning Control Framework: Stability, Robustness and design Trade-Offs. American Control Conference. 2013, submitted for publication.

[7] Parmar G, Hiemstra DB, Chen Y, Awtar S. A Moving Magnet Actuator for Large Range Nanopositioning. ASME Dynamic Systems and Control Conference. 2011; 41-48.

[8] Parmar G, Awtar S, Barton K. Large Dynamic Range Nanopositioning Using Iterative Learning Control. Precision Engineering. 2013; under review.

[9] Cao C and Hovakimyan N. L_1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation. Philedelphia, PA: Society for Industrial and Applied Mathematics, 2010.

[10] Cao C and Hovakimyan N. Design and Analysis of a Novel L_1 Adaptive Controller, Part I: Control Signal and Asymptotic Stability. American Control Conference. 2006; 3397-3402.

[11] Pomet JB and Praly L. Adaptive Nonlinear Regulation: Estimation from the Lyapunov Equation. IEEE Transactions on Automatic Control. 1992; 37(6); 729-740.