Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Optimal Normalized Weighting Factor in Iterative Feedback Tuning of Step Input Responses

Charles Z. Lu* Sheng Q. Xie* Chao Deng**

** Dept. of Mechanical Engineering, The University of Auckland, Auckland, New Zealand (e-mail: zlu013@aucklanduni.ac.nz, s.xie@auckland.ac.nz)*

*** Sch. of Mechanical Science & Engineering, Huazhong University of Science and Technology, Wuhan, China (e-mail: dengchao@mail.hust.edu.cn)*

Abstract: Iterative feedback tuning (IFT) is a model-free tuning method that has been proven to work well in various applications since its introduction in 1994. Several factors affect the performance of the optimization process, and one of which is the design criterion. Historically, the weighting factor for each element in the design criterion is chosen by trial and error and results in a different value for each system tested. In this paper, a normalized design criterion is presented with a weighting factor that allows the tuning performance to be assessed across different systems. This new design criterion is then applied to various test systems using the Monte Carlo method to determine the optimal range of values of this normalized weighting factor in tuning for step input responses.

***Keywords**:* Automatic control, Self-optimizing control, Iterative methods, Monte Carlo simulation, Numerical algorithms

1 INTRODUCTION

Iterative feedback tuning (IFT) is an automatic model-free tuning method that relies solely on information obtained from the closed-loop response of the control system. This tuning method was first introduced by Hjalmarsson et al. in 1994 optimizes the controller parameters by running a set of specially designed experiments aimed to minimize a specific design criterion (Hjalmarsson et al., 1994). IFT as a black box approach has several advantages over traditional tuning approaches. Most classical tuning methods rely on accurately determining the model of the system using system identification processes. However, the identification of an accurate model of a system is often difficult to achieve due to its complexity. Even when determined, system models have been shown to exhibit bias and variance (Forssell and Ljung, 1999). Additionally, in actual application through wear and tear, the model of the system will change over time, which means the system model identified is no longer accurate, and the identification process will have to be repeated. IFT can also be tuned online in real-time as demonstrated in McDaid et al. (2011), which is a significant advantage over traditional tuning methods as they usually require the operation to be stopped for tuning. Furthermore, traditional process based tuning methods require the loop to be opened, making the tuning process sensitive to disturbances (Åström and Hägglund, 1995). This is a non-issue in IFT as it is a closed-loop in situ tuning method.

There have been a lot of developments and applications utilizing IFT since its introduction. IFT has been shown to have a performance advantage when compared with classical tuning methods in Lequin et al. (2003). It has also been adapted for multi variable control (Hjalmarsson, 1999, Gunnarsson et al., 2003), and has shown that it can deal with some nonlinearities (Sjöberg et al., 2003). In terms of application, it has been shown to perform well in tuning of a wide range of systems, including the control of a profile cutting machine in Graham et al. (2007), the control of smart materials (Liu et al., 2011, McDaid et al., 2010, Aw et al., 2011), and on a two-mass-spring system with friction in Ho et al. (2003).

The subsequent sections are organized as follow: Section 2 introduces the general IFT algorithm, and factors that affect the performance of the optimization process, where a normalized IFT design criterion is introduced to enable the comparison of optimization performance across different systems; in Section 3, the Monte Carlo method is used on four sets of simulations to find an optimal setup for the IFT design criterion for tuning step input responses; and conclusions are drawn in Section 4.

2 ITERATIVE FEEDBACK TUNING PERFORMANCE

2.1 *General Algorithm*

First a design criterion is selected to be minimized. In Hjalmarsson et al. (1994), the following design criterion was adopted:

$$J(\rho) = \frac{1}{2N}\sum_{t=1}^{N}\left(L_y\tilde{y}_t(\rho)^2 + \lambda L_u u_t(\rho)^2\right), \qquad (1)$$

where ρ is the vector of controller parameters to be optimized, $\tilde{y}_t(\rho)$ is the error between the output signal $y_t(\rho)$ and the desired reference signal $y_d(\rho)$, $u_t(\rho)$ is the control signal into the plant, L_y and L_u are frequency weighting filters, λ is a weighting factor that determines the balance between the two parts minimized, and N is the number of sample collected for each experiment. The optimized controller parameter vector ρ^* is defined as:

$$\rho^* = \arg\min_\rho J(\rho). \tag{2}$$

The local minimum of the design criterion $J(\rho)$ can be found by iteratively approximating its gradient $\partial J/\partial\rho$, and uses the following equation to update the controller parameters for the next iteration:

$$\rho_{i+1} = \rho_i - \gamma_i R_i^{-1} \frac{\partial J(\rho_i)}{\partial \rho}. \tag{3}$$

Here γ_i is a positive scalar step size, and R_i is some positive definite matrix indicating the search direction. Commonly a modified Newton's method such as the Gauss-Newton approximation of the Hessian of $J(\rho)$ is used.

In order to compute the approximation to $\partial J(\rho_i)/\partial\rho$ and the Hessian R_i, three experiments needs to be completed in each iteration of optimization. First, a test signal is applied to the input of the closed-loop system and the output signal is collected. Second, the output signal from the first experiment is re-applied as the reference input of the control system, and the output is recorded in order to calculate the gradient of the controller parameters. A final experiment is performed where the test signal from the first experiment is used as the reference. This is done to ensure that data from the second and third experiments are independent of each other to remove bias (Hjalmarsson et al., 1998).

2.2 *Factors That Affect Tuning Performance*

The performance of the IFT process can be affected by several factors. Different search algorithms can be used to calculate R_i, with better and more robust algorithms achieving a faster convergence, or enables some form of global convergence resulting in a lower design criterion $J(\rho)$.

The reference/input waveform also affects the result. A square or a step input contains discontinuity and is impossible for the response to track perfectly. This results in a design criterion minimum that is far away from the zero value, and the algorithm tries to compensate for this "poor" result by increasing the gains excessively in order to minimize the error. A continuous signal such as a sinusoidal waveform makes it easier for the controller to track the reference signal. However square waves and step inputs are commonly studied as the step response is the most basic type of response, with the system performance figures such as the rise time and the settling time can be quantified easily.

The design criterion can also be customized to highlight the performance parameters that are deemed important to the system, such as the error and the control signal. It also enables the system designer to emphasize particular system characteristics such as the settling time or some specific bandwidth. The choice of the design criterion is indeed a flexible one. Apart from the design criterion presented in Hjalmarsson et al. (1994), several others have been presented in previous studies. The simplest form of design criterion is the sum of the square of the error values collected during the experiment:

$$J(\rho) = \frac{1}{2N}\sum_{t=1}^{N}(\tilde{y}_t(\rho)^2). \tag{4}$$

Lequin (1997) presented a time weighted design criterion shown in (5) where w_y and w_u are nonnegative factors, in order to put more emphasis on certain parts of the response that are deemed important.

$$J(\rho) = \frac{1}{2N}\sum_{t=1}^{N}\left(w_y(t)L_y[\tilde{y}_t(\rho)]^2 + \lambda w_u(t)L_u[u_t(\rho)]^2\right). \tag{5}$$

Similarly, Lequin et al. (1999) introduced a mask of length t_0 on the error part of the response designed to ignore the transient part of the response, as only the steady state error and the speed of the response are important, and not the tracking error during the transient. The design criterion in (1) is replaced by:

$$J(\rho) = \frac{1}{2N}\sum_{t=t_0}^{N}\left(L_y\tilde{y}_t(\rho)^2\right) + \lambda\sum_{t=1}^{N}(L_u u_t(\rho)^2). \tag{6}$$

It is important to note that whatever the scheme of the design criterion, if it involves more than one element to be minimized, a weighting factor is present to determine the balance between each performance measure. However, there are no rules on how the value of λ is chosen, as the value is system dependent and is always determined by trial and error.

2.3 *The Normalized Design Criterion*

In order to determine an optimal value of λ, it is important to consider the relative magnitudes of each variable in the design criterion $J(\rho)$. As different systems will have desired and actual output, error, as well as control signal that span across different range of possible values, there needs to be a method to normalize these ranges so the weighting factor λ can be meaningful and can be compared across different systems. Equation (7) presents a normalized design criterion where a normalizing factor K_n is introduced, and λ_n replaces λ to be the normalized weighting factor that remains comparable across different systems.

$$J(\rho) = \frac{1}{2N}\sum_{t=1}^{N}\left(L_y\tilde{y}_t(\rho)^2 + \lambda_n K_n L_u u_t(\rho)^2\right), \tag{7}$$

where

$$K_n = \frac{y_{d,max} - y_{d,min}}{u_{t,max} - u_{t,min}}. \tag{8}$$

The choice for y_d represents the range of output in an ideal situation, and ensures that the value of λ_n stays meaningful across different tuning iterations for the same system, as the

range of y_d should remain constant throughout the tuning process. The range of u_t is limited by the available output (saturation) of the control signal, therefore it should also remain similar across iterations for the same system, as the control signal often saturates when the reference input changes suddenly by a large amount. However, generally it is not a good idea to saturate the control signal for a long time as this introduces discontinuities and additional nonlinearities into the control parameter surface, adding to the difficulties in determining the downhill direction for the optimization algorithm. Thus with normalized design criterion, it becomes possible to determine a set of λ_n values that represent the optimal range for IFT tuning.

3 SIMULATIONS

3.1 *Simulation Setup*

A total of four system sets were tested under simulation using the Monte Carlo method, where the system plants and λ_n were stochastically tested. The control system structure used for the tests is shown in Fig. 1, where G represents the unknown system plant being optimized, the controller $C = [C_r(\rho), C_y(\rho)]$, where $C_r(\rho)$ is a standard proportional-integral controller and $C_y(\rho)$ is a standard PID controller sharing the same P and I gains with $C_r(\rho)$. r_t, u_t, and y_t are the reference input, the control signal and the output signal respectively. A unit step function was used as the system input. The elimination of the derivative component in the reference controller $C_r(\rho)$ also lessens the issue of sudden changes in the setpoint such as a step input, and is a common practice in the industry.

Starting positions for the optimization process were set to relatively low gains for the best chance of giving an overdamped response. Due to the variability of the plant transfer function, it was not possible to guarantee an overdamped starting position for every test point. The IFT algorithm should still be able to tune towards an optimal solution, however an underdamped starting position is not recommended as system resonance may come into play (Hjalmarsson, 2002).

The design criterion in (7) was used for all IFT optimization tests. No frequency filters were applied, therefore $L_y = L_u = 1$. Up to 10 tuning iterations were performed for each setup. The reason for this choice is that if the system is convergent, even without an optimized λ_n, the system response can still be tuned to have an adequate performance eventually. By limiting the tuning process to 10 iterations maximum, it allows the optimal λ_n to be identified.

For each test point, a value of λ_n ranging between 1×10^{-7} and 1×10^{0} was generated randomly over a logarithmic distribution. The four sets of plant transfer functions were modified versions of benchmarking transfer functions from Åström and Hägglund (2000), with two coefficients for each plant set also varying stochastically to create a new transfer function for each test point. These transfer functions were chosen as they are well suited to parametric studies of PID control. 5000 optimization test points were completed for each plant set. The settling time of the system to the unit step response was used to give a rating to the final performance of a tuning result. As error is already part of the design criterion in (7), it automatically tends towards a lower value during the course of optimization. Settling time on the other hand naturally factors in the error as well as the smoothness of the response. Therefore, by observing the correlation between settling time and λ_n, with large enough sample sizes one can determine the optimal value range of λ_n.

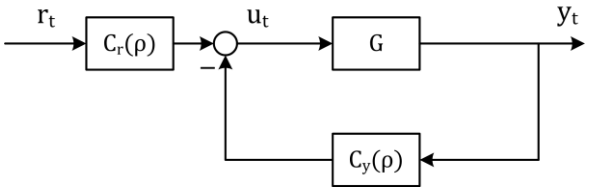

Fig. 1 Two degrees-of-freedom closed-loop control system for simulation.

3.2 *Simulation Set 1*

The simulated system G_1 is:

$$G_1(s) = \frac{1}{(s+1)^n} e^{-ds}, \quad (9)$$

where n is an integer value from 1 to 8, and d is a integer delay value from 1 to 20. The correlation between the settling time T_s and the normalized weighting factor λ_n is shown in Fig. 2(a) where only the convergent results are plotted. Data points where the algorithm did not converge after 10 iterations or remained in a local minimum with T_s that is higher than 200 seconds are not shown. As the plant coefficients were selected at random, it is possible that for some combinations of the plant transfer function the control structure is ill suited for the situation, or the starting position generated a response that is divergent. As a result the IFT algorithm was unable to locate the downhill direction on the control parameter surface for those configurations. The large range in settling times shown in the raw data can also be attributed to instances of difficult control parameter surfaces where the IFT algorithm was not very effective in the optimizing process.

Data from Fig. 2(a) were filtered using a robust implement of a weighted moving average filter, where any values over 6 standard deviations were ignored, and the span size is 10 samples. The result of this operation is shown in Fig. 2(b). It is immediately obvious that the average settling time remains constant as λ_n is increased from 1×10^{-7} to around 1×10^{-3}, but increases exponentially when λ_n increases further. To further highlight the trend and obtain a numeric representation of the optimal range of values for λ_n, resulting data from Fig. 2(b) was further fitted using a 5th order polynomial. The optimal range of λ_n was calculated using a 5% tolerance from the minimum value in the fitted polynomial. For system set G_1, the calculated optimal λ_n range is from 1×10^{-7} to 1.307×10^{-3}.

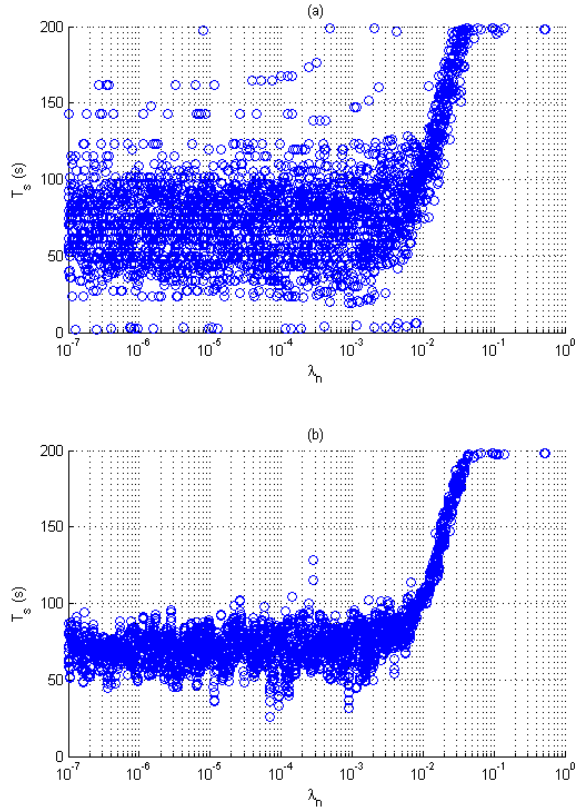

Fig. 2 (a) Settling time vs. weighting factor after 5000 optimization routines for plant G_1. (b) Correlation clarified with a weighted moving average filter using a span of 10 samples.

3.3 *Simulation Set 2*

The simulated system G_2 is:

$$G_2(s) = \frac{1}{(s+1)(1+as)(1+a^2s)\cdots(1+a^ns)}, \tag{10}$$

where n is an integer from 1 to 3, and a is a real number ranging from 1 to 5. For this system set there is no delay specified for the plant transfer function. Due to the absence of any delay, a lower limit for settling time T_s of 100 s is used. Fig. 3(a) shows a scatter plot of the convergent data points. It can be seen that there is a distinct separate outlier group around $T_s = 5$ s, which does not follow the main trend. These outlier points indicate that the value of λ_n does not affect the tuning results for those systems sets, possibly due to the transfer function being too simple to tune. These outlier data points were then removed and the result is shown in Fig. 3(b). Again the scatterplot was then clarified using the same weighted moving average method, and the result is shown in Fig. 3(c). It can be seen that the trend is different to the one in Simulation 1, with the settling time value T_s at a constant value towards the lower end of the spectrum, and dipping to a minimum value at around $\lambda_n = 1 \times 10^{-3}$, before rising sharply when λ_n is increased further. When the data points in Fig. 3(c) were further fitted using the automated polynomial fitting function however, the result was not satisfactory using any order, so it was decided to manually calculate the optimal range of λ_n. The minimum settling time determined was 16.875 s, and the range of λ_n value within a 5% tolerance value was calculated to be from 2.398×10^{-4} to 1.960×10^{-2}. Additionally, it can also be noted from all three plots in Fig. 3 that when outside of the optimal λ_n zone, the spread of the settling time increases. This denotes that the performance of the IFT algorithm is less consistent outside of the optimal λ_n zone.

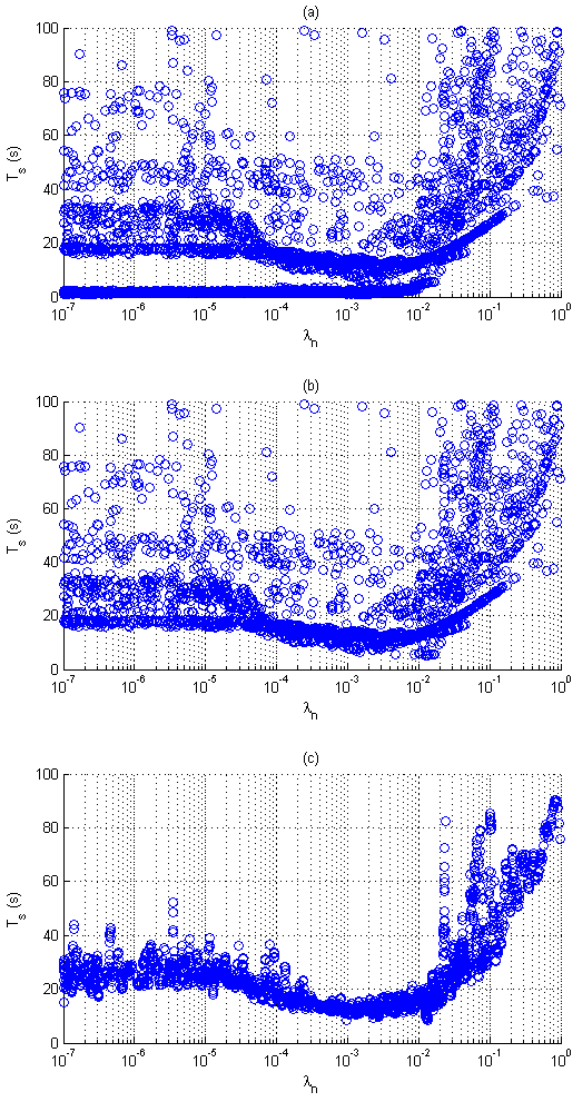

Fig. 3 (a) Settling time vs. weighting factor after 5000 optimization routines for plant G_2. (b) Settling time vs. weighting factor with lower outliers removed. (c) Correlation clarified with a weighted moving average filter using a span of 10 samples.

3.4 *Simulation Set 3*

The simulated system G_3 is:

$$G_3(s) = \frac{1-as}{(s+1)^3}e^{-ds}, \tag{11}$$

where a is a real number from 0.1 to 4, and d is an integer delay from 0 to 5. The limit for the settling time is set at 150

s. Note that for this system set the plant has a nonminimum phase zero. The convergent data points are shown in Fig. 4(a), and one can see that for this set the data points are clustered tightly together due to the plant transfer function denominator staying the same throughout the test. These data points were smoothed out using the same weighted moving average filter as previous simulations, with the result shown in Fig. 4(b). Again a 5th order polynomial was fitted in order to give a numeric answer for the optimal range of λ_n, which was calculated to be from 1×10^{-7} to 1.598×10^{-3}. It is also worth noting that the spread of data points in Fig. 4(a) increases when λ_n is below 1×10^{-5}, indicating some instability with a design criterion that focuses almost purely on the error component. This behavior is consistent with the observation made in simulation set 2.

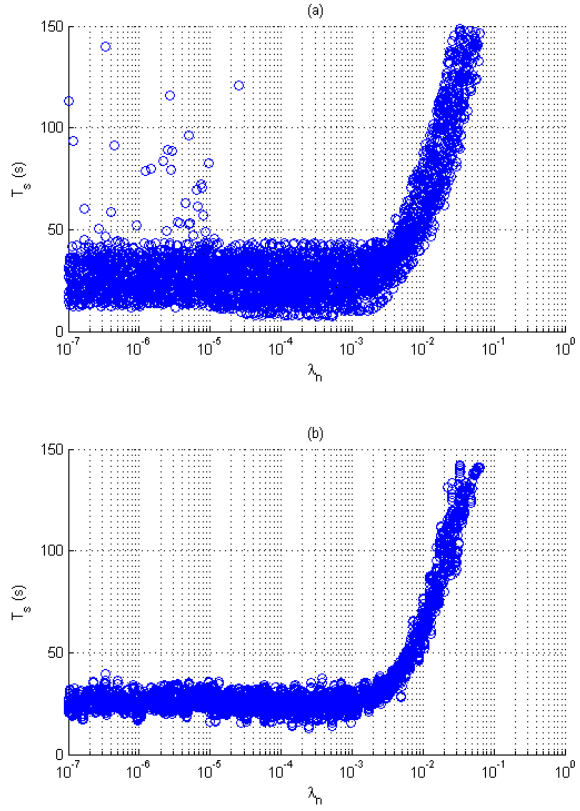

Fig. 4 (a) Settling time vs. weighting factor after 5000 optimization routines for plant G_3. (b) Correlation clarified with a weighted moving average filter using a span of 10 samples.

3.5 *Simulation Set 4*

The simulated system G_4 is:

$$G_4(s) = \frac{1}{(1+as)^2} e^{-ds}, \qquad (12)$$

where a is a real number from 0.1 to 10, and d is an integer delay from 0 to 10. The limit for the settling time is set at 175 s. Fig. 5(a) shows the convergent data points. It can be seen that the variance for the settling time for this plant set is much larger than other system sets, indicating that the balance between the error and the control signal component is not as significant on the settling time T_s when compared to other simulations. While the overall trend is less obvious, an outlier group similar to the one in simulation 2 is present. These data points, defined as $T_s < 15$ s were removed and the result of this operation is shown in Fig. 5(b). Even though the trend is less clear for this system set, the usage of the 10 sample span weighted moving average filter was able to distill the correlation between λ_n and T_s, which is shown in Fig. 5(c). As in previous simulations, a 5th order polynomial was fitted to the smoothed data points and an optimal λ_n range was calculated to be from 1×10^{-7} to 1.057×10^{-2}. Also note that even though the automated polynomial fit gave the bottom limit value of 1×10^{-7}, in Fig. 5(c) it can be seen that the data points trend slightly upwards towards the lower limit of λ_n, indicating a possible rebound similar to Simulation 2, though this may just be due to the stochastic nature of the Monte Carlo Method.

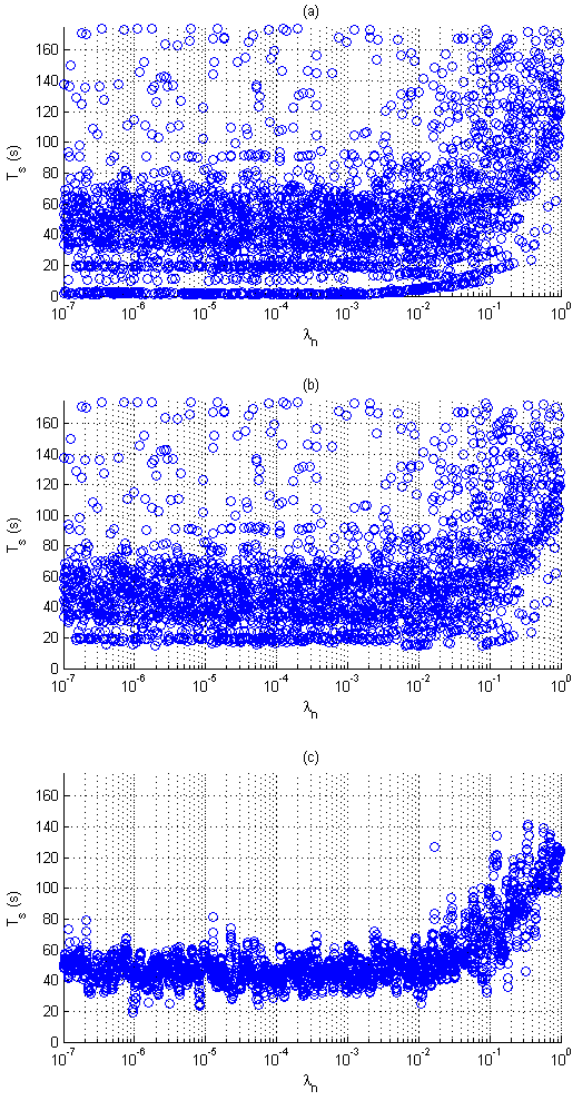

Fig. 5 (a) Settling time vs. weighting factor after 5000 optimization routines for plant G_4. (b) Settling time vs. weighting factor with lower outliers removed. (c) Correlation clarified with a weighted moving average filter using a span of 10 samples.

4 CONCLUSION

In this paper, a normalized design criterion for IFT with a normalized weighting factor λ_n is presented in order to compare the performance of the tuning process across different systems. Four sets of transfer functions were analyzed using the Monte Carlo method, where two variables from each plant transfer functions, along with the variable λ_n were varied stochastically to create a new test point for each sample. 5000 test samples for each simulation set were run and the calculated optimal λ_n range is shown in Table 1. While the lower limit of λ_n set for the simulations was 1×10^{-7}, the settling time trend is steady around that limit and is not expected to change significantly as by that point the design criterion becomes almost purely error based. By considering the results from all four sets of simulations, it is recommended that the weighting factor λ_n range for testing using the normalized design criterion is from 1×10^{-4} to 1×10^{-3}, for a step input.

Table 1. Summary of Simulation Results

Set	Optimal $\boldsymbol{\lambda_n}$ Range		
	Lower value	Upper value	Note
1	1×10^{-7}	1.31×10^{-3}	-
2	2.40×10^{-4}	1.96×10^{-2}	Increased variance outside the optimal λ_n range
3	1×10^{-7}	1.60×10^{-3}	Increased variance when $\lambda_n < 1 \times 10^{-5}$
4	1×10^{-7}	1.06×10^{-2}	Slight upwards trend towards lower limit

ACKNOWLEDGMENT

The authors would like to acknowledge the project "Health assessment and residual life prediction of CNC equipment based on the map of multi-performance and multi-failure modes", supported by the Natural Science Foundation of China (Grant No. 51375181) and the Fundamental Research Funds for the Central Universities of HUST (Grant No. 2013TS027).

REFERENCES

Åström, K. J. and Hägglund, T. (1995). *PID controllers: theory, design, and tuning.* (2nd ed.) Instrument Society of America, North Carolina.

Åström, K. J. and Hägglund, T. (2000). Benchmark systems for PID control. *Proceedings of IFAC workshop on Digital Control*, 165-166.

Aw, K. C., Yu, W., McDaid, A. J. and Xie, S. Q. (2011). An IPMC driven micropump with adaptive on-line iterative feedback tuning. *Proc. of SPIE Vol 8409*, 84090I–1.

Forssell, U. and Ljung, L. (1999). Closed-loop identification revisited. *Automatica,* **35**, 1215-1241.

Graham, A. E., Young, A. J. and Xie, S. Q. (2007). Rapid tuning of controllers by IFT for profile cutting machines. *Mechatronics,* **17**, 121-128.

Gunnarsson, S., Collignon, V. and Rousseaux, O. (2003). Tuning of a decoupling controller for a 2 x 2 system using iterative feedback tuning. *Control Engineering Practice,* **11**, 1035-1041.

Hjalmarsson, H. (1999). Efficient tuning of linear multivariable controllers using iterative feedback tuning. *International Journal of Adaptive Control and Signal Processing,* **13**, 553-572.

Hjalmarsson, H. (2002). Iterative feedback tuning - an overview. *International Journal of Adaptive Control and Signal Processing,* **16**, 373-395.

Hjalmarsson, H., Gevers, M., Gunnarsson, S. and Lequin, O. (1998). Iterative feedback tuning: theory and applications. *IEEE Control Systems,* **18**, 26-41.

Hjalmarsson, H., Gunnarsson, S. and Gevers, M. (1994). A convergent iterative restricted complexity control design scheme. *Proceedings of the 33rd IEEE Conference on Decision and Control*, 1735 -1740.

Ho, W. K., Hong, Y., Hansson, A., Hjalmarsson, H. and Deng, J. W. (2003). Relay auto-tuning of PID controllers using iterative feedback tuning. *Automatica,* **39**, 149-157.

Lequin, O. (1997). Optimal closed loop PID tuning in the process industry with the Iterative Feedback Tuning scheme. *CD-Rom of European Control Conference.* Brussels, Belgium.

Lequin, O., Gevers, M., Mossberg, M., Bosmans, E. and Triest, L. (2003). Iterative feedback tuning of PID parameters: comparison with classical tuning rules. *Control Engineering Practice,* **11**, 1023-1033.

Lequin, O., Gevers, M. and Triest, L. (1999). Optimizing the settling time with iterative feedback tuning. *Proceedings of the 14th IFAC World Congress.* Beijing, P. R. China.

Liu, D., McDaid, A. J., Aw, K. C. and Xie, S. Q. (2011). Position control of an Ionic Polymer Metal Composite actuated rotary joint using Iterative Feedback Tuning. *Mechatronics,* **21**, 315-328.

McDaid, A. J., Aw, K. C., Haemmerle, E. and Xie, S. Q. (2011). Control of IPMC actuators for microfluidics with adaptive "online" iterative feedback tuning. *IEEE/ASME Transactions on Mechatronics,* **17**, 789-797.

McDaid, A. J., Aw, K. C., Xie, S. Q. and Haemmerle, E. (2010). Gain scheduled control of IPMC actuators with 'model-free' iterative feedback tuning. *Sensors and Actuators A: Physical,* **164**, 137-147.

Sjöberg, J., De Bruyne, F., Agarwal, M., Anderson, B. D. O., Gevers, M., Kraus, F. J. and Linard, N. (2003). Iterative controller optimization for nonlinear systems. *Control Engineering Practice,* **11**, 1079-1086.