Technical Report, Dept. of Electrical Engineering, University of Souterhn Calif., Sept 15, 2002.

The Comparison of Unfalsified Control and Iterative Feedback Tuning[†]

Rengrong Wang[1]
University of Southern California
Los Angeles, CA, 90089-2563, USA
Michael G. Safonov[1]
University of Southern California
Los Angeles, CA, 90089-2563, USA

Abstract

In this paper, we compare two data-driven model-free adaptive control design schemes: gradient-based myopic unfalsified control (MUC) and iterative feedback tuning (IFC). Both have drawn much attention in recent years, each with its own advantages. Based on our analysis and simulation results, we conclude that advantages of MUC are that it requires few assumptions on the plant and seems to be simpler to implement and faster to adapt.

Keywords: unfalsification, iterative feedback tuning, adaptive control, model-free control, parameter adaptation, gradient method

1 Introduction

Indirect adaptive control methods, like multi-model and self-tuning, usually require some kind of identification for the plant dynamics. This contributes to a number of

---

[†] Research supported in part by AFOSR Grant F49620-01-1-0302.
[1] Rengrong Wang, email: rengronw@usc.edu
[2] Michael G. Safonov, email: msafonov@usc.edu Fax: +1-213-740-4449

fundamental problems, such as (a) the amount of offline training required, (b) the assumption on the plant structure, (c) the system stability issue, and (d) the difficulties on dealing with nonlinear, large time delayed and time variant plant.

To overcome these problems, people have proposed plant-model-free control schemes by abandoning the identification for plant altogether and instead focusing on the problem of controller identification. One school of thought is to address these problems as parameter optimizations to be carried directly on the controller parameters so as to directly minimize a closed-loop control performance criterion.

Iterative Feedback Tuning (IFT) is in this school ([1], [2]). It is based on an iterative tuning of the controller parameter vector along the steepest-descent gradient direction of a given control criterion. The construction of the gradient requires a "special experiment" in which a finite record of the output of the closed system is recycled as the reference input of the system.

Another related, but different approach to direct controller parameter tuning is Unfalsified Control ([3], [4], [5]). This is an adaptive control theory that permits learning by an online process of elimination of candidate controllers as they are falsified by evolving open-loop plant data. Its goal is to determine a control law K for the unknown plant P such that the closed-loop system response, say T, satisfies certain given specifications.

In the usual approach to unfalsified control, it is essential that falsification procedure starts with a large candidate controller set. Then, as plant data accumulates, one iteratively performs a global optimization over this entire set to determine which candidate controllers still remain unfalsified. If the scheme is to lead to a non-empty set of unfalsified controllers, the class of candidate controllers must be sufficiently large to include at least one candidate that is capable of meeting the performance goal. Depending on the performance goals and the candidate controller parameterization, the computation load may be too big. To reduce this potential computational burden, Jun and Safonov ([4]) proposed a new simplified gradient-based version of unfalsified control which, unlike the usual global approach, only examines the local gradient of the performance function at each time. We call this simplified approach *Myopic Unfalsified Control* (MUC), because it only does 'near-sighted' local gradient-based optimization at each time, rather than attempting the usual global optimization.

As stated above, MUC and IFT are similar in some sense. Both are model-free, data-driven methods with similar control criterion. Both employ a "myopic" gradient-based steepest descent approach to parameter optimization. Aroused by their similarity, in this paper, we will have a close look at the theory behind these two methods, and will compare their performance by a simulation.

This paper is organized as follows: The theoretical background is given in Section 2. The design of our simulation and the results are reported in Section 3. An analysis is presented in Section 4. Finally, summary is given in Section 5.

2 Theory Background

In this section, a brief introduction of unfalsified control and iterative feedback tuning theory is described. To make latter comparison straightforward, at the end of this section, a lemma is given to show that the algorithm in IFT under deterministic circumstance is equivalent to that under stochastic circumstance.

2.1 Myopic Unfalsified Control

Definition 1:Unfalsification:([3]) A controller is said to be ***falsified*** by measurement information if this information is sufficient to deduce that the performance specification $(r, y, u) \in T_{spec}\ \forall r \in R$ would be violated if that controller were in the feedback loop. Otherwise, the controller is said to be ***unfalsified***.

Definition 2: Fictitious Reference Signal $\tilde{r}$: Given measured data (u, y), we define for each candidate controller K the ***fictitious reference signal*** $\tilde{r}(K)$ to be any signal with the property that if the candidate controller K had been in the control system during the period when the input-output data (u, y) was collected, and the signal $\tilde{r}(K)$ had been applied to the system, then the measured data (u, y) would have been reproduced.

Because the data (u, y) may have been collected when one or more controllers other than K were in the feedback loop, $\tilde{r}(K)$ is in general not the same as the actual reference signal r. This is why it is called a *fictitious* reference signal.

By computing the fictitious reference signal or signals associated with each candidate controller in real time, we can do online controller falsification and parameter tuning without inserting every candidate controller into the closed-loop system.

Based on Theorem in [1], a candidate controller $K(\theta)$ is unfalsified at time t by plant data $u(\tau), y(\tau), (\tau \in [0,t])$ if and only if,

$$\tilde{J}(\theta,\tau) \le 0, \ \forall\tau \in [0,t] \qquad (1)$$

where $\tilde{J}(\theta,\tau) \overset{\Delta}{=} -\rho(\tau) + \int_0^\tau T_{spec}(\tilde{r}_i(\theta,\zeta), y(\zeta), u(\zeta))d\zeta$, $u(\zeta), y(\zeta), (\zeta \in [0,t])$ are measured past plant data, and $\tilde{r}_i(\theta,\zeta)$ denotes the fictitious reference signal for the controller $K(\theta)$.

Since the objective of unfalsified control is to adjust the controller parameter vector θ so as to satisfy the given performance specification (1), one approach to achieving this objective is to adjust θ in the steepest-descent direction $-\nabla\tilde{J}(\theta,t)$ so that the performance specification $\tilde{J}(\theta,t)$ tends to decrease whenever the currently active controller's parameter vector θ is falsified, viz.,

$\dot{\theta} = -\gamma\nabla\tilde{J}(\theta,t)$ where $\gamma > 0$ is a design constant that determines the rate of adaptation

$$\nabla\tilde{J}(\theta,t) \overset{\Delta}{=} [\frac{\partial\tilde{J}(\theta,t)}{\partial\theta_1} \ \frac{\partial\tilde{J}(\theta,t)}{\partial\theta_2} \cdots \frac{\partial\tilde{J}(\theta,t)}{\partial\theta_n}]^T = \int_0^\tau T_{spec}(\tilde{r}_i(\theta,\zeta), y(\zeta), u(\zeta)) \cdot \nabla\tilde{r}(\theta,\zeta)d\zeta$$ is

the gradient of $\tilde{J}(\theta,t)$ with respect to θ ,where

$$\nabla\tilde{r}(\theta,\zeta) = -K(\theta)^{-1}\nabla K(\theta)K(\theta)^{-1}u(\zeta)$$

$$\nabla K(\theta) \stackrel{\Delta}{=} [\frac{\partial K(\theta)}{\partial\theta_1}\ \frac{\partial K(\theta)}{\partial\theta_2} \cdots \frac{\partial K(\theta)}{\partial\theta_n}]^T$$

since $u(t) = K(\theta)(\tilde{r}(t) - y(t))$.

Therefore, controller parameter adaptation rule can be expressed as:

$$\dot{\theta} = \begin{cases} \gamma\int_0^t \frac{\partial T_{spec}(\tilde{r},\zeta)}{\partial\tilde{r}} K(\theta)^{-1}\nabla K(\theta)K(\theta)^{-1}u(\zeta)d\zeta, & \text{if } \tilde{J}(\theta,\tau) > 0, \forall\tau \in [t_0,t] \\ 0, & \text{if } \tilde{J}(\theta,\tau) \le 0, \forall\tau \in [t_0,t] \end{cases}$$

2.2 Iterative Feedback Tuning ([2])

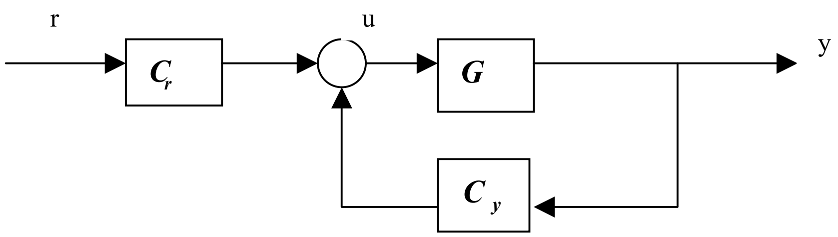

Figure 1: Block diagram for IFT

Consider a system with a controller $C(\theta_i) \stackrel{\Delta}{=} \{C_r(\theta_i), C_y(\theta_i)\}$ in Figure 1. The objective of IFT is to minimize the expected value $E[J(\theta)]$ of some criterion $J(\theta)$:

$$\text{J}(\theta) = \frac{1}{2\text{N}}E[\sum_{t=1}^{N}(L_y\tilde{y}_t(\theta))^2 + \lambda\sum_{t=1}^{N}(L_u u_t(\theta))^2]$$

with respect to controller parameter vector θ. Here $\tilde{y}_t(\theta) \stackrel{\Delta}{=} y(\theta) - y^d$ is the error between the output $y(\theta)$ and the desired output y^d. L_y and L_u are filters and λ is a positive constant. The form of $E[\cdot]$ denotes the expectation with respect to the weakly stationary plant disturbance. To obtain the minimum of $\text{J}(\vartheta)$ in the deterministic case, the IFT procedure for updating controller parameter vector ϑ employs the following iterative algorithm:

$$\theta_{i+1} = \theta_i - \gamma_i R_i^{-1} \frac{\partial J}{\partial \theta}(\theta_i)$$

In the stochastic case with non-zero plant disturbances the IFT update is obtained by replacing the gradient $\frac{\partial J}{\partial \theta}$ with an unbiased estimate

$$est\left[\frac{\partial J}{\partial \theta}(\theta_i)\right] = E\left[\frac{\partial J}{\partial \theta}(\theta_i)\right] \tag{2}$$

To calculate (2), in each iteration i, the IFT procedure requires data from three experiments, each of duration N, with a fixed controller operating on the actual plant We denote N-length reference signals by $r_i^j, j = 1,2,3,$ and the corresponding output signals by $y^i(\theta_i), j = 1,2,3.$ Then during each iteration i, we design:

$$r_i^1 = r, \quad r_i^2 = r - y^1(\theta_i), \quad r_i^3 = r.$$

With the gathered data $(u^1(\theta_i), y^1(\theta_i)), (u^2(\theta_i), y^2(\theta_i)),$ and $(u^3(\theta_i), y^3(\theta_i))$ from the experiments, we can compute:

$$\tilde{y}(\theta_i) = y^1(\theta_i) - y^d \tag{3}$$

$$est[\frac{\partial y}{\partial \theta}(\theta_i)] = \frac{1}{\mathrm{C_r}(\theta_i)}\left[(\frac{\partial \mathrm{C_r}}{\partial \theta}(\theta_i) - \frac{\partial \mathrm{C_y}}{\partial \theta}(\theta_i))\mathrm{y}^3(\theta_i) + \frac{\partial \mathrm{C_y}}{\partial \theta}(\theta_i)\mathrm{y}^2(\theta_i)\right] \tag{4}$$

$$u(\theta_i) = u^1(\theta_i). \tag{5}$$

$$est[\frac{\partial u}{\partial \theta}(\theta_i)] = \frac{1}{\mathrm{C_r}(\theta_i)}\left[(\frac{\partial \mathrm{C_r}}{\partial \theta}(\theta_i) - \frac{\partial \mathrm{C_y}}{\partial \theta}(\theta_i))\mathrm{u}^3(\theta_i) + \frac{\partial \mathrm{C_y}}{\partial \theta}(\theta_i)\mathrm{u}^2(\theta_i)\right] \tag{6}$$

Then, the estimate of the gradient becomes:

$$est\left[\frac{\partial J}{\partial \theta}(\theta_i)\right] = \frac{1}{\mathrm{N}}\sum_{t=1}^{N}\left(\tilde{y}_t(\theta_i) est\left[\frac{\partial y_t}{\partial \theta}(\theta_i)\right] + \lambda u_t(\theta_i) est\left[\frac{\partial u_t}{\partial \theta}(\theta_i)\right]\right) \tag{7}$$

Finally, the controller parameters are updated by:

$$\theta_{i+1} = \theta_i - \gamma_i R_i^{-1} est\left[\frac{\partial J}{\partial \theta}(\theta_i)\right] \tag{8}$$

Lemma1: With IFT method, the updating algorithms under stochastic circumstance and under deterministic circumstance are exactly the same if everything else is the same.

Proof: The result follows immediately, since the IFT algorithm (3)-(8) does not depend on the plant disturbance or its properties.

3 Simulation Design and Results

Noting that by Lemma1 the stochastic IFT algorithm does not actually involve the noise, we perform a deterministic simulation example for both MUC and IFT to compare them. The following are used in both simulations:

- unknown plant $P(s) = \frac{1}{s(s+2)}$
- controller $K(\theta) = \theta > 0$ with initial gain $\theta(0) = 8$
- all initial conditions at time 0 are zero
- reference signal: step signal (in [6], it is said that to do identification with one parameter, step signal is a signal with persistent excitation)
- sampling time $\Delta t = 0.1s$

3.1 Controller Parameter Adaptation by MUC:

The parameter θ is tuned such that the cost function:

$$J(K) = \begin{cases} \frac{1}{2}\int_{t-T}^{t} (\tilde{r}(\zeta) - y(\zeta))^2 d\zeta - 0.005 \le 0, \ t > 10s \\ \frac{1}{2}\int_{0}^{t} (\tilde{r}(\zeta) - y(\zeta))^2 d\zeta - 0.005 \le 0, \ t < 10s \end{cases}$$ is satisfied.

So, according to section2.1, the adaptation rule becomes:

$$\dot{\theta} = \begin{cases} \frac{\gamma}{\theta^3}\int_0^t u(\zeta)^2 d\zeta, & \text{if } \tilde{J}(\theta,\tau) > 0,\ \exists\tau \in [t_0, t] \\ 0, & \text{if } \tilde{J}(\theta,\tau) \le 0, \forall\tau \in [t_0, t] \end{cases}, \text{ where } \gamma = 5$$

And, we get the following results:

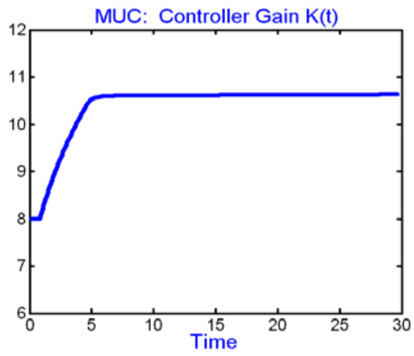

Figure 2: Controller Gain of MUC

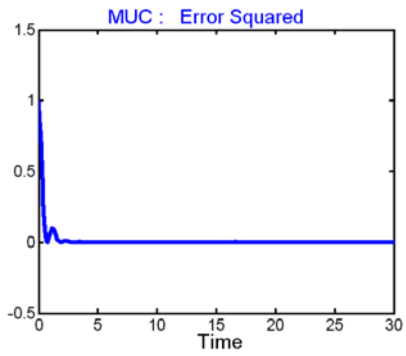

Figure 3: Error Squared of MUC

3.2 Controller Parameter Adaptation by IFT:

The parameter θ is tuned such that the cost function $J(K) = \frac{1}{2N}\left(\sum_{t=1}^{N}(r_t - y_t)^2\right)$ is decreasing into an acceptable range.

So, according to section2.2, the adaptation rule becomes:

$$\theta_{i+1} = \theta_i - \gamma_i R_i^{-1}\frac{\partial J}{\partial\theta}(\theta_i), \text{ where } \gamma = 2.5 \text{ and } R_i = \frac{1}{N}\sum_{t=1}^{N}\left(\frac{\partial y_t}{\partial\theta}\right)^2$$

Since we are talking about deterministic condition and the controller $C_r(\theta_i) = C_y(\theta_i) = \theta$ is of one degree of freedom, so we only need two experiment1 and experiment 2 in one iteration. The procedure is illustrated as follows:

At last, we get the following results:

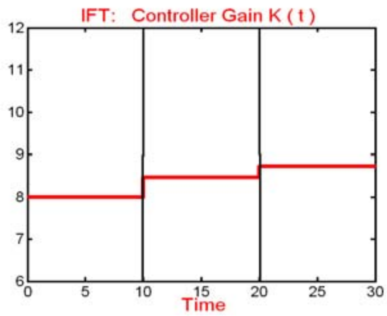

Figure 4: Controller Gain of IFT

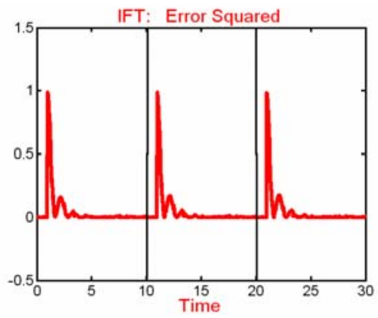

Figure 5: Error Squred of IFT

4 Discussion

As stated above, MUC and IFT are similar in many aspects. Both of them are model-free, data-driven, gradient-based local methods with similar cost function. They are both reliable, and fairly easy to implement.

But, there are still some fundamental differences between them. First, in cost function of MUC, u and y are viewed as measured data, and only $\tilde{r}$ is taken as a function of the controller parameter vector θ. That is, MUC is doing optimization with one variable. So, we can do continuously online tuning. Quite differently, the cost function in IFT is viewed as a function of two variables --u and y. Both u and y are viewed as function of θ. So with IFT, one is doing local optimization of the cost J with respect to perturbations in θ that affect *(u,y)* but not *r*, whereas in MUC one examines only perturbations in θ that affect *r* but leaves *(u,y)* to be the actual observed data. To compute the derivative of u and y respect to θ for a quadratic cost, at least two separate experiments are required in IFT, one with a special probing signal. Therefore, the process of IFT becomes run-to-run and takes time, which is not the case with MUC. And unlike MUC, the IFT algorithm requires the application of a specially designed test signal to the plant. What's more, if we use another kind of cost function with IFT instead of quadratic cost, sometimes much more work are required to get the gradient. So, one may expect IFT to be typically slower to adapt and more complicated to implement than MUC. Finally, IFT requires an assumption not required by MUC, namely that the plant should be time-invariant. By contrast, no such assumption is required in MUC.

We conclude that the MUC approach seems to be superior to IFT in that it is simpler, easier to implement, requires fewer assumptions on the plant and seems to converge more quickly without noticeable differences in the final performance.

5 Conclusion

This paper compared two adaptive parameter-tuning schemes, MUC and IFT. Both of them are model-free, data-driven and gradient-based methods. Our analysis and simulations suggest that MUC controllers tend to be easier to implement, require fewer assumptions, and adapt more quickly then IFT controllers.

References

[1] H. Hjalmarsson, S.Gunnarsson and M.Gevers. "A Convergent Iterative Restricted Complexity Control Design Scheme". *Proceedings of the 33rd Conference on Decision and Control*, pages 1735-1740, Lake Buena Vista, FL, Dec. 1994.

[2] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, "Iterative Feedback Tuning: Theory and Applications ", *IEEE Control Systems Magazine* , **18**(4):26 –41, Aug. 1998.

[3] M. G. Safonov and T.-C. Tsao. " The Unfalsified Control Concept and Learning". *IEEE Trans. Automat. Contr.*, **42**(6):843-847, Jun. 1997.

[4] M. Jun and M. G. Safonov. "Controller Parameter Adaptation Algorithm Using Unfalsified Control Theory and Gradient Method". In *Proceedings of IFAC World Congress*, Barcelona, Spain, July 21-26, 2002.

[5] M. Jun and M.G. Safonov. “Automatic PID Tuning: An Application of Unfalsified Control”, *Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design*, pages 328–333, 1999.

[6] J.E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, 1991.