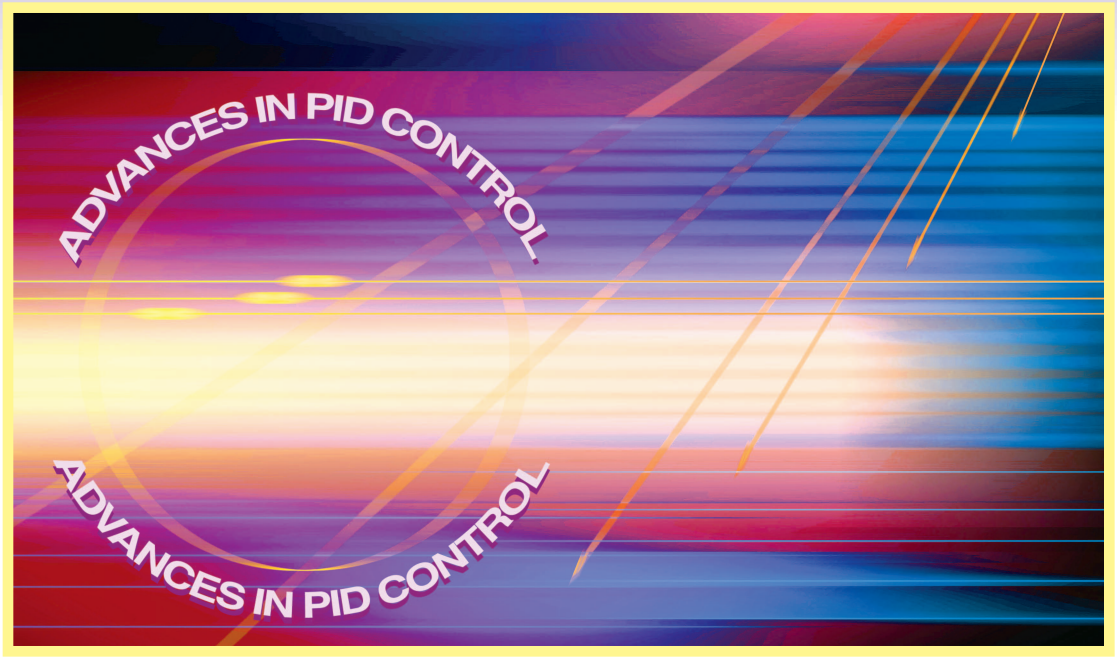
PID Tuning Using Extremum Seeking

ONLINE, MODEL-FREE PERFORMANCE OPTIMIZATION

By NICK J. KILLINGSWORTH and MIROSLAV KRSTIĆ



Although proportional-integral-derivative (PID) controllers are widely used in the process industry, their effectiveness is often limited due to poor tuning. The manual tuning of PID controllers, which requires optimization of three parameters, is a time-consuming task. To address this difficulty, much effort has been invested in developing systematic tuning methods. Many of these methods rely on knowledge of the plant model or require special experiments to identify a suitable plant model. Reviews of these methods are given in [1] and the survey paper [2]. In many situations, however, a plant model is not known, and it is not desirable to open the process loop for system identification. Thus, a method for tuning PID parameters within a closed-loop setting is advantageous.

In relay feedback tuning [3]–[5], the feedback controller is temporarily replaced by a relay. Relay feedback causes most systems to oscillate, thus determining one point on the Nyquist diagram. Based on the location of this point, PID parameters can be chosen to give the closed-loop system a desired phase and gain margin.

An alternative tuning method, which does not require either a modification of the system or a system model, is unfalsified control [6], [7]. This method uses input-output data to determine whether a set of PID parameters meets performance specifications. An adaptive algorithm is used to update the PID controller based on whether or not the controller falsifies a given criterion. The method requires a finite set of candidate PID controllers that must be initially specified [6]. Unfalsified control for an infinite set of PID controllers has

1066-033X/06/$20.00©2006IEEE

been developed in [7]; this approach requires a carefully chosen input signal [8].

Yet another model-free PID tuning method that does not require opening of the loop is iterative feedback tuning (IFT). IFT iteratively optimizes the controller parameters with respect to a cost function derived from the output signal of the closed-loop system (see [9]). This method is based on the performance of the closed-loop system during a step-response experiment [10], [11].

In this article, we present a method for optimizing the step response of a closed-loop system consisting of a PID controller and an unknown plant with a discrete version of extremum seeking (ES). Specifically, ES minimizes a cost function similar to that used in [10] and [11], which quantifies the performance of the PID controller. ES, which is a nonmodel-based method, iteratively modifies the arguments of a cost function (in this application, the PID parameters) so that the output of the cost function reaches a local minimum or local maximum.

COST FUNCTION AND PID CONTROLLERS

ES is used to tune the parameters of a PID controller so as to minimize a given cost function. The cost function, which quantifies the effectiveness of a given PID controller, is evaluated at the conclusion of a step-response experiment. We use the integrated square error (ISE) cost function

$$J(\theta) \triangleq \frac{1}{T - t_0} \int_{t_0}^{T} e^2(t, \theta) dt, \tag{1}$$

where the error $e(t, \theta) \triangleq r(t) - y(t, \theta)$ is the difference between the reference and the output signal of the closed-loop system, and

$$\theta \triangleq [K, T_i, T_d]^T \tag{2}$$

contains the PID parameters. The PID controller structure and the meaning of K, T_i, and T_d are given below.

The cost function $J(\theta)$ defined in (1) takes into account the error over the time interval $[t_0, T]$. By setting t_0 to approximate the time T_{peak} at which the step response of the closed-loop system reaches the first peak, the cost function $J(\theta)$ effectively places zero weighting on the initial transient portion of the response [10]. Hence, the controller is tuned to minimize the error beyond the peak time T_{peak} without constraints on the initial transient.

We use a standard PID controller, with the exception that the derivative term acts on the measured plant output but not on the reference signal. This PID controller avoids large control effort during a step change in the reference signal. Figure 1 shows a block diagram of the closed-loop system, where G is the unknown plant, the controller is parameterized as

$$C_r(s) = K\left(1 + \frac{1}{T_i s}\right), \tag{3}$$

$$C_y(s) = K\left(1 + \frac{1}{T_i s} + T_d s\right), \tag{4}$$

and r, u, and y are the reference signal, control signal, and output signal, respectively.

ES TUNING SCHEME

The cost function $J(\theta)$ should be understood as a mapping from the PID parameters K, T_i, and T_d to the tracking performance. ES seeks to tune the PID controller by finding a minimizer of $J(\theta)$. However, since ES is a gradient method, the PID parameters found by ES are not necessarily a global minimizer of $J(\theta)$.

The overall ES PID tuning scheme is summarized in Figure 2. The step-response experiment, which is contained within

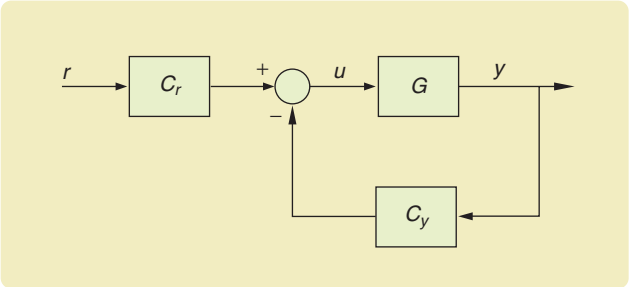


FIGURE 1 Closed-loop servo system. The output signal y of the unknown plant G is regulated to the reference signal r by the two-degree-of-freedom controller C_r and C_y.

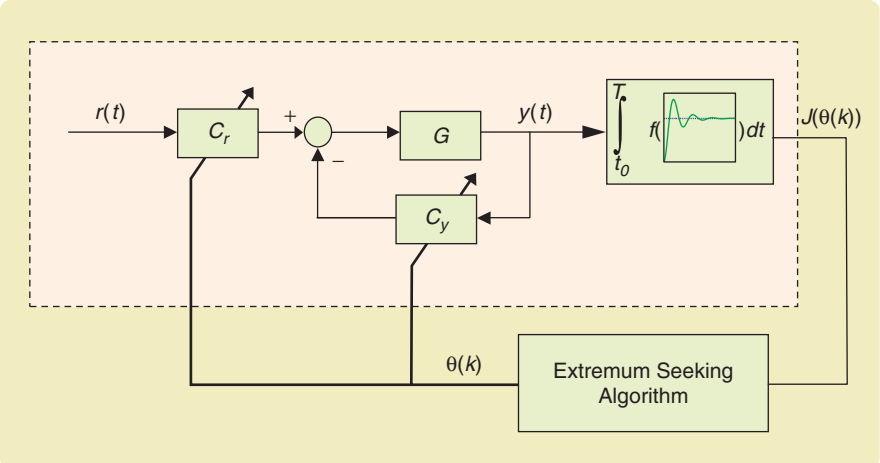


FIGURE 2 The overall ES PID tuning scheme. The ES algorithm updates the PID controller parameters $\theta(k)$ to minimize the cost function $J(\theta)$, which is calculated from a step-response experiment carried out within the dashed box.

We present a method for optimizing the step response of a closed-loop system consisting of a PID controller and an unknown plant with a discrete version of extremum seeking.

the dashed box, is run iteratively. The cost $J(\theta(k))$ is calculated at the conclusion of the step-response experiment. The ES algorithm uses the value $J(\theta(k))$ of the cost function to compute new controller parameters $\theta(k)$. Another step function experiment is then performed with the new controller parameters, and the process continues iteratively.

ES is a nonmodel-based method that iteratively modifies the input θ of the cost function $J(\theta)$ to reach a local minimizer. As shown in Figure 3, ES achieves this optimization by sinusoidally perturbing the input parameters $\theta(k)$ of the system and then estimating the gradient $\nabla J(\theta(k))$. Note that k is the index of the step-response experiment, whereas t is the continuous-time variable within an individual step-response experiment. The gradient is determined by highpass filtering the discrete time signal $J(\theta(k))$ to remove its dc portion and then demodulating it by multiplication with a discrete-time sinusoid of the same frequency as the perturbation signal. This procedure estimates the gradient by picking off the portion of $J(\theta(k))$ that arises due to perturbation of the parameter estimate $\hat{\theta}(k)$ (see "How Extremum Seeking Works"). The gradient information is then used to modify the input parameters in the next iteration; specifically, the gradient estimate is integrated with a step size γ, yielding a new parameter estimate $\hat{\theta}(k)$. The integrator both performs the adaptation function and acts as a lowpass filter.

The time-domain implementation of the discrete-time ES algorithm in Figure 3 is

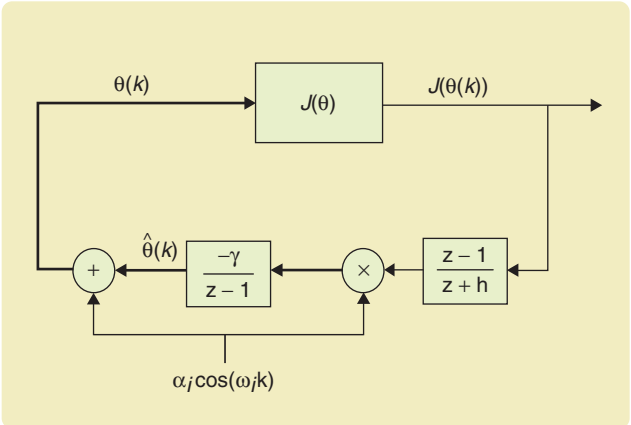


FIGURE 3 Discrete ES scheme. The input parameters $\theta(k)$ are perturbed by the signal $\alpha_i \cos(\omega_i k)$. The output of the cost function $J(\theta(k))$ is then highpass filtered, demodulated, and finally lowpass filtered to yield new input parameters.

$$\zeta(k) = -h\zeta(k-1) + J(\theta(k-1)), \tag{5}$$

$$\hat{\theta}_i(k+1) = \hat{\theta}_i(k) - \gamma_i \alpha_i \cos(\omega_i k)[J(\theta(k)) - (1+h)\zeta(k)], \tag{6}$$

$$\theta_i(k+1) = \hat{\theta}_i(k+1) + \alpha_i \cos(\omega_i(k+1)), \tag{7}$$

where $\zeta(k)$ is a scalar and the subscript i indicates the ith entry of a vector. γ_i is the adaptation gain and α_i is the perturbation amplitude. Stability and convergence are influenced by the values of γ, α, and the shape of the cost function $J(\theta)$ near the minimizer, as explained in "How Extremum Seeking Works." The modulation frequency ω_i is chosen such that $\omega_i = a^i\pi$, where a satisfies $0 < a < 1$. Additionally, the highpass filter $(z-1)/(z+h)$ is designed with $0 < h < 1$ and a cutoff frequency well below the modulation frequency ω_i.

An overview of ES theory and some state-of-the-art applications are given in [12]. The PID tuning in this article comprises a novel hybrid application, where the plant dynamics are continuous time and the ES dynamics are discrete time.

EXAMPLES OF ES PID TUNING

We now demonstrate ES PID tuning and compare this method with IFT and two classical PID tuning methods, namely, Ziegler-Nichols (ZN) tuning rules and internal model control (IMC). In particular, we use the ultimate sensitivity method [13] version of the ZN tuning rules, which consists of a closed-loop experiment with only proportional feedback, where the feedback gain is increased to a critical value until the system begins to oscillate. PID parameters are then prescribed based on the critical gain K_c and the period T_c of oscillation to give the closed-loop system response approximately a quarter amplitude decay ratio, corresponding to a damping ratio of about 0.2. The amplitude decay ratio is the ratio of two consecutive maxima of the error e after a step change of the reference signal. Specifically, the PID parameters given by ZN are $K = K_c/1.7$, $T_i = T_c/2$, and $T_d = T_c/8$.

Details of IMC can be found in [1], where the plant is assumed to have the form

$$G(s) = \frac{K_p}{1+sT}e^{-sL}. \tag{8}$$

Based on (8), the PID parameters are chosen to be of the form $K = (2T+L)/(2K_p(T_f+L))$, $T_i = T + L/2$, and $T_d = (TL)/(2T+L)$, where T_f is a design parameter that affects the tradeoff between performance and robustness. When the

plant is unknown, a step-response experiment can be used to obtain an estimate of the form (8), as explained in [1]. Although variations of IMC that can deal with alternative model structures are available in [14] and [15], these methods are not considered here. We note that ZN and IMC are derived for a PID structure with derivative action on both the reference signal and the output signal rather than the structure (3), (4), which does not have derivative action on the reference signal.

In [11], IFT, ZN, and IMC are applied to the models

$$G_1(s) = \frac{1}{1+20s}e^{-5s}, \tag{9}$$

$$G_2(s) = \frac{1}{1+20s}e^{-20s}, \tag{10}$$

$$G_3(s) = \frac{1}{(1+10s)^8}, \tag{11}$$

$$G_4(s) = \frac{1-5s}{(1+10s)(1+20s)}. \tag{12}$$

How Extremum Seeking Works

The first documented use of ES is Leblanc's 1922 application to electric railway systems [18]. In the 1950s and 1960s, ES was widely studied and used in applications in both the former Soviet Union [19]–[24] and the West [25]–[28]. The ability of this technique to force $\hat{\theta}(k)$ to converge to a local minimizer θ^* of $J(\theta)$ is the subject of stability proofs obtained in the late 1990s [29]. Subsequently, ES has become a useful tool for real-time applications [30]–[34] as well as an active area of theoretical research [12]. Here we give an intuitive argument that explains the convergence of ES.

For simplicity, we consider the single-parameter case in which $\theta(k)$ and $\hat{\theta}(k)$ are scalar and only one probing signal $\alpha\cos(\omega k)$ is used (see Figure 3). We also assume a quadratic cost function $J(\theta)$ of the form

$$J(\theta) = f^* + \frac{f''}{2}\left(\theta^* - \theta\right)^2,$$

where f is positive. Letting $\tilde{\theta} \triangleq \theta^* - \hat{\theta}$, we expand $J(\theta)$ as

$$J \approx \left(f^* + \frac{\alpha^2 f''}{4}\right) + \frac{\alpha^2 f''}{4}\cos(2\omega k) - \left(\alpha f''\cos(\omega k)\right)\tilde{\theta},$$

where a trigonometric identity is used to replace $\cos^2(\omega k)$. The term $(f''/2)\tilde{\theta}^2$ is omitted since it is quadratic in $\tilde{\theta}$ and we focus on local analysis only. The role of the washout filter $(z-1)/(z+h)$ in Figure 3 is to filter out the dc component of the output signal $J(\theta(k))$. Thus,

$$\frac{z-1}{z+h}[J] \approx \frac{\alpha^2 f''}{4}\cos(2\omega k) - \left(\alpha f''\cos(\omega k)\right)\tilde{\theta}. \tag{16}$$

Multiplying (16) by $\alpha\cos(\omega k)$ yields

$$\alpha\cos(\omega k)\frac{z-1}{z+h}[J] \approx -\frac{\alpha^2 f''}{2}\tilde{\theta}, \tag{17}$$

where trigonometric identities are used for $\cos(2\omega k)\cos(\omega k)$ and $\cos^2(\omega k)$. Moreover, the higher frequency terms with $\cos(\omega k)$, $\cos(2\omega k)$, and $\cos(3\omega k)$ are attenuated by the integrator $1/(z-1)$ and thus omitted.

Feeding the signal (17) into the integrator $(-\gamma)/(z-1)$ in Figure 3 results in

$$\tilde{\theta}(k+1) \approx \left(1 - \frac{\gamma\alpha^2 f''}{2}\right)\tilde{\theta}(k).$$

Hence, the estimation error $\tilde{\theta}(k)$ decays exponentially provided the adaptation gain γ and the probing amplitude α are chosen such that the positive quantity $(\gamma\alpha^2 f'')/2$ is small. The complete proof of stability presented in [35] is considerably more involved, and is based on two-time-scale averaging [36] for the system

$$\tilde{\theta}(k+1) = \tilde{\theta}_k + \gamma\alpha\cos(\omega k)\left(e + \frac{f''}{2}\left(\tilde{\theta} - \alpha\cos(\omega k)\right)^2\right), \tag{18}$$

$$e(k+1) = -he(k) - (1+h)\frac{f''}{2}\left(\tilde{\theta} - \alpha\cos(\omega k)\right)^2 \tag{19}$$

where $e = f^* - ((1+h)/(z+h))[J]$, with the assumption that γ and α are small. The proof guarantees exponential convergence of $J(\theta(k))$ to $f^* + O(\alpha^3)$.

Another intuitive point of view is to observe that the term $f''\tilde{\theta}$ in the signal (17) at the output of the multiplier is the gradient (derivative) of $J = f^* + (f''/2)(\tilde{\theta} - \alpha\cos(\omega k))^2$ with respect to $\tilde{\theta}$ for $\alpha = 0$. Hence, the role of the additive probing term $\cos(\omega k)$ and the multiplicative term of the same form (along with the filtering effects of the washout filter and the integrator) is to estimate the gradient of J, which is then fed into the integrator, employing classical gradient-based optimization with step size γ. While gradient-based methods usually require a model to determine the gradient, ES estimates the gradient in a nonmodel-based manner.

An interesting aspect of ES is the role of the signal $\cos(\omega k)$, which mimics amplitude modulation (AM) in analog communications. This similarity is not obvious since ES employs one addition and one multiplication block rather than two multipliers. The addition block is used because the nonlinearity $J(\theta)$ provides the effect of multiplication since its quadratic part generates a product of $\cos(\omega k)$ and $\tilde{\theta}$ that carries the gradient information discussed above. The modulation, demodulation, and filtering serve to extract the gradient information $f''\tilde{\theta}(k)$ from the signal $J(\theta(k))$.

Extremum seeking converges to parameters that yield performance comparable to the best achievable with other popular PID tuning methods.

Notice that G_1 and G_2 have time delays, G_3 has repeated poles, and G_4 is nonminimum phase. We apply ES to (9)–(12) to facilitate comparison with the IFT, ZN, and IMC PID controllers found in [11].

The closed-loop systems are simulated using a time step of 0.01 s, and the time delays are approximated using a third-order Padé approximation to be consistent with [11]. The PID controller parameters given by ZN are used as a starting point for ES tuning. For all simulations, the parameters a and h in the ES scheme (5)–(7) are set to 0.8 and 0.5, respectively.

Tuning for G_1

ES PID tuning is applied to G_1 in (9), which has a time delay of 5 s. For these simulations, the cost function spans from $t_0 = 10$ s to $T = 100$ s, $\alpha = [0.1, 1, 0.1]^T$, $\gamma = [200,\ 1, 200, 200]^T$, and $\omega_i = a^i\pi$. Figure 4 shows that ES minimizes the cost function (1) with convergence in less than ten iterations to PID parameters that produce a local minimum. ES achieves this step response by increasing the value of the integral time T_i to almost three times that given by the ZN tuning rules, thereby reducing the influence of the integral portion of the controller (see Table 1). The performance of the PID parameters obtained from ES tuning is roughly equivalent to the IFT performance. This similarity is expected since both methods attempt to minimize the same cost function. Figure 4 shows that IFT and ES yield closed-loop systems with less overshoot and smaller settling times than ZN and IMC.

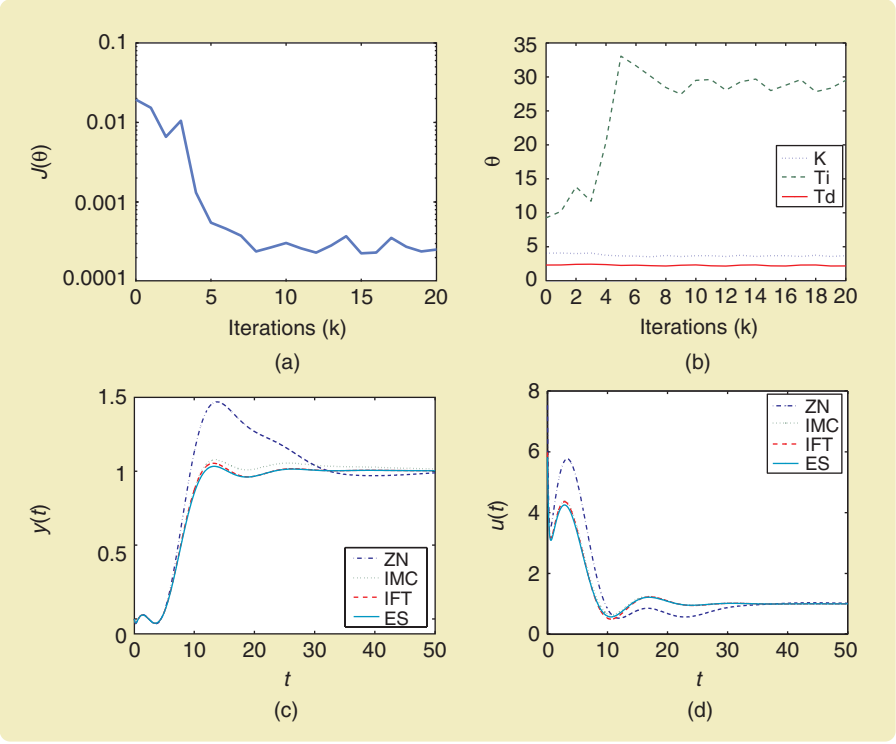


FIGURE 4 ES PID tuning of G_1 illustrated by (a) the evolution of the cost function and (b) the PID parameters during ES tuning of the closed-loop system with $G_1(s)$. The lower plots present (c) the output signal and (d) the control signal during step-response experiments of the closed-loop systems with $G_1(s)$ and the PID controllers obtained from the four methods. ES reduces the cost function in (a) by increasing the integral time in (b), which produces a more favorable step response similar to that given by IFT in (c).

Tuning for G_2

For G_2, which is identical to G_1 except with a longer time delay of 20 s, we set $t_0 = 50$ s, $T = 300$ s, $\alpha = [0.06, 0.3, 0.2]^T$, $\gamma = [2{,}500, 2{,}500, 2{,}500]^T$, and $\omega_i = a^i\pi$. Figure 5 shows that ES reduces the cost function by an order of magnitude in less than ten iterations. Moreover, ES yields a closed-loop system whose step response is similar to that produced by IMC and IFT and, thus, has improved overshoot and settling time compared to ZN tuning. The PID parameters determined by the four tuning methods are presented in Table 2.

TABLE 1 **PID parameters for G_1. The PID parameters given by IFT in [11] and ES in the present article are similar. Both methods increase the integral time T_i markedly over ZN.**

Tuning Method	K	T_i	T_d
ZN	4.06	9.25	2.31
IMC	3.62	22.4	2.18
IFT	3.67	27.7	2.11
ES	3.58	27.8	2.15

TABLE 2 **PID parameters for G_2. Although ES and IFT yield different parameters, the resulting responses are similar, as shown in Figure 5.**

Tuning Method	K	T_i	T_d
ZN	1.33	31.0	7.74
IMC	0.935	30.5	6.48
IFT	0.930	30.1	6.06
ES	1.01	31.5	7.16

Tuning for G_3

For G_3 with a single pole of order eight, we use $\alpha = [0.06, 1.1, 0.5]^T$, $\gamma = [800, 3{,}500, 300]^T$, $\omega_1 = \omega_2 = a\pi$ (with $\alpha_2 \cos(\omega_2 k)$ replaced by $\alpha_2 \sin(\omega_2 k)$ in Figure 3), and $\omega_3 = a^3\pi$. Furthermore, the cost function accounts for the error from $t_0 = 140$ s to $T = 500$ s. Figure 6 shows that ES improves the step-response behavior obtained by the ZN tuning rules and returns a response that is similar to that achieved by IFT, yet with a smaller settling time than the IMC controller. Table 3 indicates that ES reduces the integral time T_i and controller gain K to reduce the value of the cost function. This plant, which is more challenging than G_1 and G_2, requires roughly 30 iterations for parameter convergence.

Tuning for G_4

The PID controller for the closed-loop system with nonminimum phase G_4 in (12) is tuned using ES. We set $t_0 = 30$ s, $T = 200$ s, $\alpha = [0.05, 0.6, 0.2]^T$, $\gamma = [2, 000, 10, 000, 2, 000]^T$, $\omega_1 = \omega_2 = a\pi$ (with $\alpha_2 \cos(\omega_2 k)$ replaced by $\alpha_2 \sin(\omega_2 k)$ in Figure 3), and $\omega_3 = a^3\pi$. Figure 7 shows that ES produces a step response similar to IFT; both ES and IFT yield no overshoot and a smaller settling time than the ZN and IMC controllers. However, ES produces a slightly larger initial control signal than IFT. Table 4 shows that an increased integral time improves the system response.

COST FUNCTION COMPARISON

The cost function dictates the performance of the PID controller obtained from ES. It is therefore important to choose a cost function that emphasizes the relevant performance aspects, such as settling time, overshoot, and rise time. To illustrate the dependence of the optimal PID parameters θ^* on the cost function, we use ES for plant $G_2(s)$ to minimize the ISE cost function (1) with $t_0 = 0$ and $t_0 = T_{\text{peak}}$ as well as the cost functions

$$\text{IAE} = \frac{1}{T}\int_0^T |e|dt, \tag{13}$$

$$\text{ITAE} = \frac{1}{T}\int_0^T t|e|dt, \tag{14}$$

$$\text{ITSE} = \frac{1}{T}\int_0^T te^2dt. \tag{15}$$

Note that (14) and (15) involve a time-dependent weighting, which de-emphasizes the transient portion of the response. Figure 8 shows that ISE with $t_0 = T_{\text{peak}}$ produces the response with the smallest overshoot and fastest settling time. Integrated time absolute

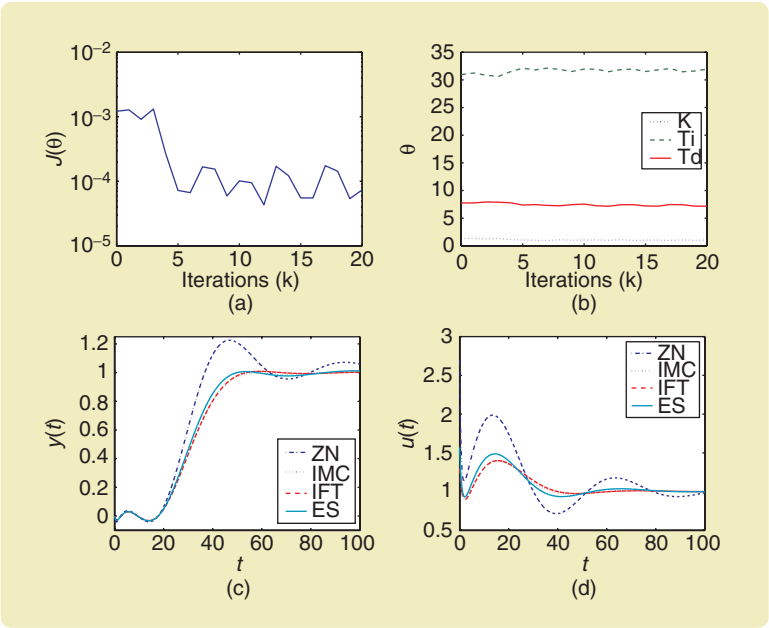


FIGURE 5 ES PID tuning of G_2 illustrated by (a) the evolution of the cost function and (b) the PID parameters during ES tuning of the closed-loop system with $G_2(s)$. The lower plots present (c) the output signal and (d) the control signal during step-response experiments of the closed-loop systems with $G_2(s)$ and PID controller parameters obtained using the four methods. ES reduces the cost function in (a) after a few iterations and finds PID parameters in (b), which produce a step response similar to the IFT and IMC controllers in (c).

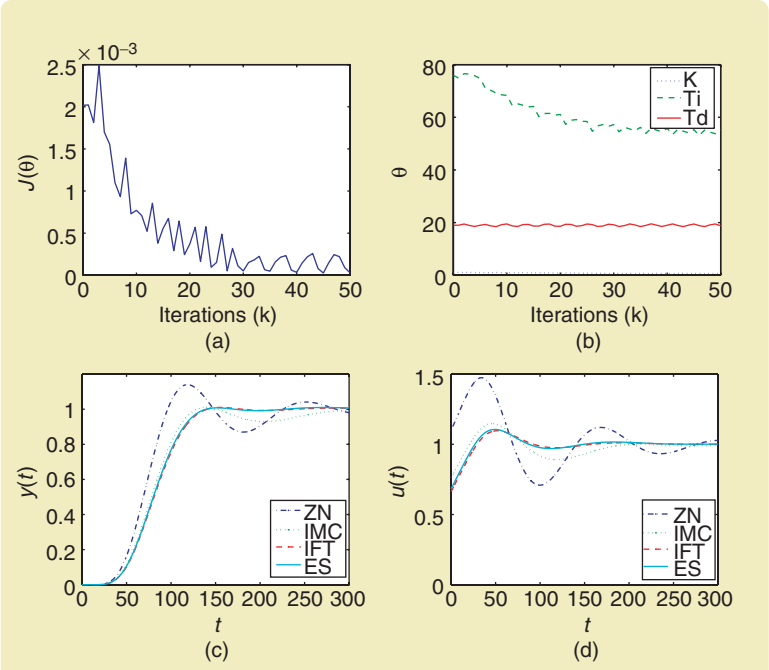


FIGURE 6 ES PID tuning of G_3 illustrated by (a) the evolution of the cost function and (b) the PID parameters during ES tuning of the closed-loop system with $G_3(s)$. The lower plots present (c) the output signal and (d) the control signal during step-response experiments of the closed-loop systems with $G_3(s)$ and the PID controllers obtained by means of the four methods. ES reduces the cost function in (a), although not as quickly as for the other plants, by decreasing the integral time T_i in (b), which produces a more favorable step response in (c).

error (ITAE) and integrated absolute error (IAE) perform slightly worse than ISE with $t_0 = T_{peak}$, whereas ISE with $t_0 = 0$ and integrated time square error (ITSE) are similar to ZN in terms of overshoot and settling time. However, Figure 8 also indicates that using a cost function comprised of the squared error (ISE and ITSE) versus the absolute error (IAE and ITAE) decreases the time required for the output of a closed-loop system to initially reach the setpoint.

Thanks to the flexibility of ES, the cost function can be modified on the fly, allowing the PID parameters to be retuned whenever it is desirable to emphasize a different performance aspect. However, stability of ES must be maintained for the new cost function through the choice of the ES parameters.

TABLE 3 PID parameters for G_3. IMC, IFT, and ES decrease the proportional gain K and the integral time T_i versus the parameters found using ZN. Furthermore, IMC reduces the derivative time T_d more so than IFT and ES.

Tuning Method	K	T_i	T_d
ZN	1.10	75.9	19.0
IMC	0.760	64.7	14.4
IFT	0.664	54.0	18.2
ES	0.684	54.9	19.5

TABLE 4 PID parameters for G_4. IMC, IFT, and ES progressively decrease the influence of the integral term while increasing the effect of the derivative term.

Tuning Method	K	T_i	T_d
ZN	3.53	16.8	4.20
IMC	3.39	31.6	3.90
IFT	3.03	46.3	6.08
ES	3.35	49.2	6.40

CONTROL SATURATION

Many applications of PID control must deal with actuator saturation. Actuator saturation can result in integrator windup, in which the feedback loop becomes temporarily disconnected since the controller output is no longer affected by the feedback signal. During saturation, the integral term grows while the error remains either positive or negative. Hence, the integrator is slow to recover when the actuator desaturates.

To examine ES tuning in the presence of saturation, we apply ES with and without the tracking antiwindup scheme [1] depicted in Figure 9, which modifies the integral control signal using a feedback signal proportional to $\tilde{u}$, the difference between the requested control signal $u_{\text{requested}}$ and the actual control signal u_{actual} produced by the actuator. The tracking time constant T_t for the case of ES is set to $T_t = (T_i T_d)^{1/2}$. For IMC, this choice of T_t results in a slow controller response; thus, we use $T_t = 18$.

We compare ES and IMC in the presence of saturation with and without antiwindup. Figure 10 shows that overshoot is a problem for the IMC controller without antiwindup, whereas ES increases the integral time (see Table 5) to improve the performance of the controller. ES finds controller parameters that perform almost as well as the systems with antiwindup. However, for small changes in the reference signal, the actuator will not saturate and the ES controller without antiwindup, with its large integral time, may demonstrate inferior performance. It is, therefore, preferable to always employ antiwindup rather then rely on the ability of ES to tune the PID controller without antiwindup at a fixed reference size to attain similar performance.

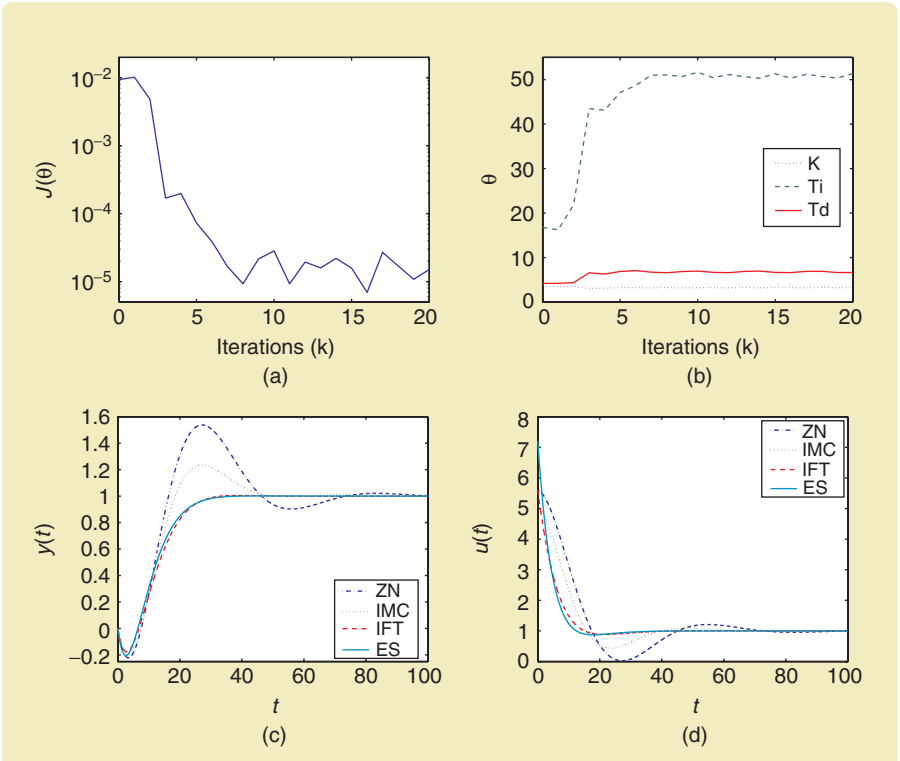


FIGURE 7 ES PID tuning of G_4 illustrated by (a) the evolution of the cost function and (b) the PID parameters during ES tuning of the closed-loop system with $G_4(s)$. The lower plots present (c) the output signal and (d) the control signal during step-response experiments of the closed-loop systems with $G_4(s)$ and PID controllers obtained using the four methods. ES reduces the cost function in (a) by increasing the integral time T_i and the derivative time T_d in (b), which produces a more favorable step response similar to that found using IFT in (c).

SELECTING PARAMETERS FOR ES TUNING

Implementation of ES requires the choice of several parameters, namely, the perturbation amplitudes α_i, adaptation gains γ_i, perturbation frequencies ω_i, and the parameter h in the highpass filter. However, it turns out that the minimizer found by ES is fairly insensitive to the ES parameters. To investigate this sensitivity, we use ES to tune the closed-loop system with G_2 in (10) while varying α and γ. The parameters h and ω_i are chosen to be $h = 0.5$ and $\omega_i = 0.8^i\pi$.

For the plant G_2, Figure 11 shows the evolution of the cost function during tuning with various ES parameters. Table 6 shows that ES yields almost identical PID parameters even though α is varied by 50% and γ is reduced by an order of magnitude relative to the values we use in the section "Tuning for G_1." However, the convergence is slower due to the reduced perturbation amplitudes α_i and adaptation gains γ_i. The tradeoff between the speed of convergence and the domain of initial conditions that yield the minimizer θ^* is quantified in [16], where the ability of ES to avoid getting trapped in local minima, when its parameters are chosen appropriately, is demonstrated analytically.

COMPARISON OF TUNING METHODS

ES and IFT use the same cost function and, thus, yield similar results. Therefore, it is interesting to compare how these methods minimize the cost function. Both methods are nonmodel based and estimate the gradient of the cost function with respect to the controller parameters. The estimated gradient is then used in a gradient search scheme to find a local minimizer of the cost function. The difference lies in how these algorithms estimate the gradient. IFT uses signal information from three experiments, including

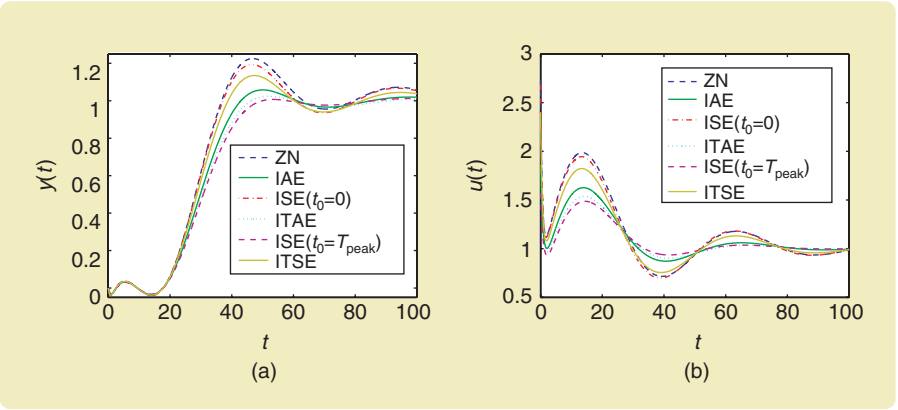


FIGURE 8 The effect of the cost function illustrated by the output signal (a) and the control signal (b) during step-response experiments of the closed-loop systems with $G_2(s)$ and PID controllers obtained using ES with various cost functions. The use of different cost functions in ES yields different step responses, with the ISE ($t_0 = T_{peak}$) cost function producing the best result.

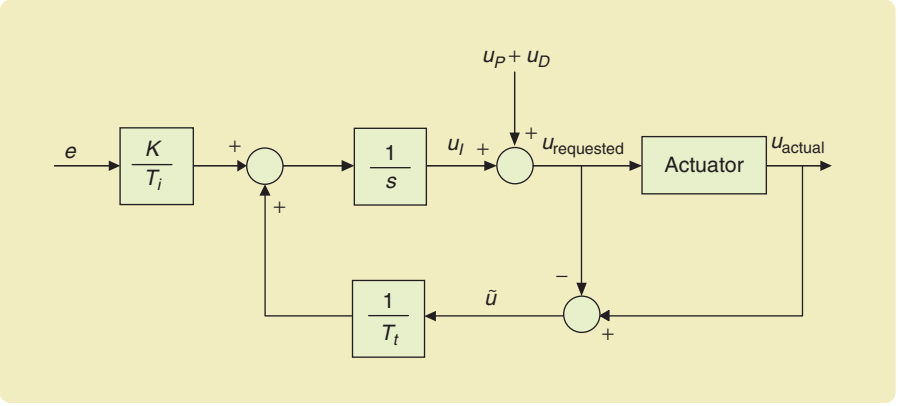


FIGURE 9 Tracking antiwindup scheme. This approach reduces integrator windup by feeding back the error signal $\tilde{u} = u_{actual} - u_{requested}$, which is the difference between the requested control signal $u_{requested}$ and the actual control signal u_{actual}.

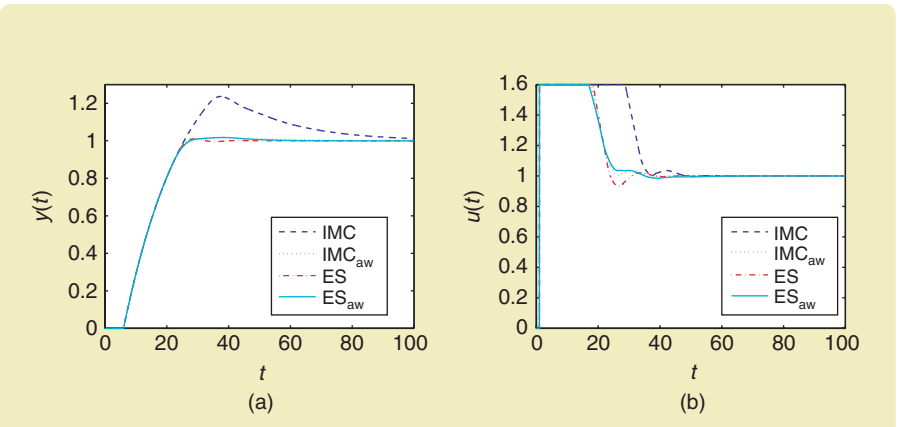


FIGURE 10 The effect of actuator saturation illustrated by the output signal (a) and the control signal (b) during step-response experiments of the closed-loop systems with $G_1(s)$, control saturation of 1.6, and PID controllers obtained using IMC and ES both with and without antiwindup. ES finds PID parameters that produce a step response with little overshoot even without the aid of antiwindup. Furthermore, the step response for ES without antiwindup is comparable to IMC and ES with antiwindup.

Extremum seeking produces favorable results in the presence of actuator saturation.

a special feedback experiment, and assumes that the system is linear time invariant to estimate the gradient. Although IFT is based on linear theory, the technique can be applied to nonlinear systems [17].

On the other hand, ES requires only one experiment per iterative gradient estimate, and its derivation does not assume that the system is linear. ES uses simple filters along with modulation by sinusoidal signals to estimate the gradient. However, ES requires a choice of several design parameters, whereas IFT requires that only the step size be specified.

While both ES and IFT are more difficult to implement than ZN and IMC, ES and IFT often yield improved performance. For G_3, which has repeated poles, these benefits can be seen in Figure 6; benefits can also be seen for the nonminimum phase plant G_4 in Figure 7. Additionally, ES outperforms IMC in the presence of control saturation, as shown in Figure 10.

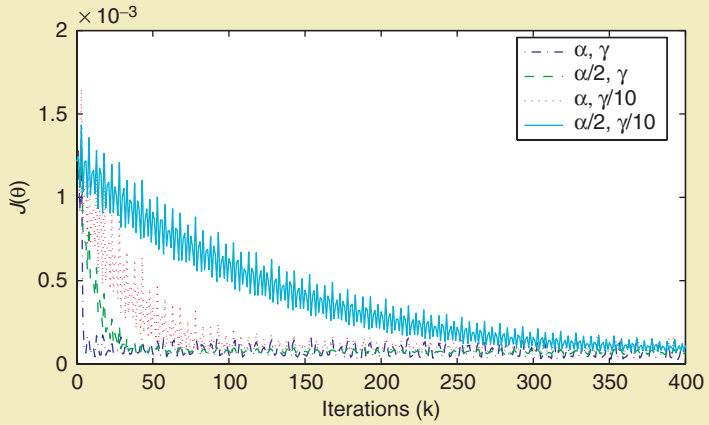


FIGURE 11 Sensitivity of ES to α and γ illustrated by the evolution of the cost function during ES tuning of the PID parameters for the plant $G_2(s)$ with various values of α and γ. In each case, ES converges to a similar cost with slower convergence for reduced gains.

CONCLUSIONS

ES tunes PID controllers by minimizing a cost function that characterizes the desired behavior of the closed-loop system. This tuning method is demonstrated on four typical plants and found to give parameters that yield performance better than or comparable to that of other popular tuning methods. Additionally, ES produces favorable results in the presence of actuator saturation. The ES method thus has an advantage over model-based PID tuning schemes in applications that exhibit actuator saturation. However, since ES requires initial values of the PID parameters, the method can be viewed as a complement to another PID parameter design method. Furthermore, the ES cost function can be chosen to reflect the desired performance attributes.

ACKNOWLEDGMENTS

This research was partly supported by Lawrence Livermore National Laboratory, Ford Motor Company, National Science Foundation Grant ECS-0501403, and the GEM and AGEP fellowships.

AUTHOR INFORMATION

Nick J. Killingsworth received his B.S. degree in mechanical and materials science engineering in 2000 and his M.S. degree in mechanical and aeronautical engineering in 2002 from the University of California, Davis. He is currently pursuing a Ph.D. at the University of California, San Diego, in the Department of Mechanical and Aerospace Engineering. His research interests include control of internal combustion engines, combustion instabilities, and extremum seeking.

Miroslav Krstić (krstic@ucsd.edu) received his Ph.D. degree in electrical

TABLE 5 PID parameters for G_1 with saturation. ES without antiwindup increases the integral time to decrease the effect of integral windup, whereas ES with antiwindup can use a smaller integral time because of the antiwindup scheme.

Tuning Method	K	T_i	T_d
IMC	3.62	22.4	2.18
ES	3.61	47.6	1.81
ES_{aw}	4.07	12.8	2.20

TABLE 6 PID Parameters for G_2 with different values of α and γ. ES arrives at similar PID parameters for reduced values of the perturbation amplitude α and the adaptation gain γ.

ES Tuning Parameters	K	T_i	T_d
α, γ	1.01	31.5	7.16
$\frac{\alpha}{2}, \gamma$	1.00	31.1	7.60
$\alpha, \frac{\gamma}{10}$	1.01	31.3	7.54
$\frac{\alpha}{2}, \frac{\gamma}{10}$	1.01	31.0	7.65

engineering in 1994 at the University of California at Santa Barbara. After two years at the University of Maryland, College Park, he joined the University of California, San Diego (UCSD), in 1997. In 2005 he received the UCSD Award for Excellence in Research and was appointed the inaugural Harold W. Sorenson Distinguished Professor. He coauthored the books *Nonlinear and Adaptive Control Design* (Wiley 1995), *Stabilization of Nonlinear Uncertain Systems* (Springer, 1998), *Flow Control by Feedback* (Springer, 2002), and *Real-Time Optimization by Extremum Seeking Control* (Wiley, 2003). He was associate editor for *IEEE Transactions on Automatic Control*, *International Journal of Adaptive Control and Signal Processing*, and *Systems and Control Letters*. He is editor for adaptive and distributed parameter systems for *Automatica*. He was vice president for technical activities and a member of the Board of Governors of the IEEE Control Systems Society, as well as vice chair of the UCSD Department of Mechanical and Aerospace Engineering. He is a Fellow of the IEEE and has received the NSF Career Award, the NR Young Investigator Award, the Presidential PECASE Award, and the Axelby and Schuck best paper awards. He can be contacted at Department of Mechanical and Aerospace Engineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0411 USA.

REFERENCES

[1] K.J. Åström and T. Hägglund, *PID Controllers: Theory, Design and Tuning* 2nd ed. Research Triangle Park, NC: Instrum. Soc. Amer., 1995.

[2] K.J. Åström, T. Hägglund, C.C. Hang, and W.K. Ho, "Automatic tuning and adaptation for PID controllers—A survey," *Contr. Eng. Pract.*, vol. 1, no. 4, pp. 699–714, 1993.

[3] K.J. Åström and T. Hägglund, "Automatic tuning of simple regulators with specifications on phase and amplitude margins," *Automatica*, vol. 20, no. 5, pp. 645–651, 1984.

[4] A. Leva, "PID autotuning algorithm based on relay feedback," *IEEE Proc. D Contr. Theory Applicat.*, vol. 140, no. 5, pp. 328–338, 1993.

[5] A.A. Voda and I.D. Landau, "A method for the auto-calibration of PID controllers," *Automatica*, vol. 31, no. 1, pp. 41–53, 1995.

[6] M. Jun and M.G. Safonov "Automatic PID tuning: An application of unfalsified control," in *Proc. IEEE Int. Symp. CACSD*, Hawaii, 1999. pp. 328–333.

[7] M. Saeki, "Unfalsified control approach to parameter space design of PID controllers," in *Proc. 42nd IEEE Conf. Decision and Control*, Maui, HI, 2003, pp. 786–791.

[8] M. Saeki, A. Takahashi, O. Hamada, and N. Wada, "Unfalsified parameter space design of PID controllers for nonlinear plants," in *Proc. IEEE Int. Symp. CACSD*, Taipei, Taiwan, 2004, pp. 1521–1526.

[9] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, "Iterative feedback tuning: Theory and applications," *IEEE Contr. Syst. Mag.*, vol. 18, no. 4, pp. 26–41, 1998.

[10] O. Lequin, M. Gevers, and T. Triest, "Optimizing the settling time with iterative feedback tuning," in *Proc. 14th IFAC World Congress*, Beijing, P.R. China, 1999, pp. 433–437.

[11] O. Lequin, E. Bosmans, and T. Triest, "Iterative feedback tuning of PID parameters: Comparision with classical tuning rules," *Contr. Eng. Pract.*, vol. 11, no. 9, pp. 1023–1033, 2003.

[12] K.B. Ariyur and M. Krstić, *Real-Time Optimization by Extremum Seeking Feedback*. Hoboken, NJ: Wiley, 2003.

[13] K.J. Åström and B. Wittenmark, *Computer Controlled Sytems: Theory and Design*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1997.

[14] D.E. Rivera and M. Morari, "Control relevant model reduction problems for SISO H_2, H_∞, and μ-controller synthesis," *Int. J. Contr.*, vol. 46, no. 2, pp. 505–527, 1987.

[15] A.J. Isaksson and S.F. Graebe, "Analytic PID parameter expressions for higher order systems," *Automatica*, vol. 35, no. 6, pp. 1121–1130, 1999.

[16] Y. Tan, D. Nešić, and I.M.Y. Mareels, "On non-local stability properties of extremum seeking control,'' *Automatica*, to be published.

[17] H. Hjalmarsson, "Control of nonlinear systems using iterative feedback tuning: Theory and applications,'' in *Proc. American Control Conf.*, Philadelphia, PA, 1998, pp. 2083–2087.

[18] M. Leblanc, "Sur l'électrification deschemins de fer au moyen de courants alternatifs de fréquence élevée,'' *Revue Générale de l'Electricité*, 1922.

[19] P.I. Chinaev, Ed., *Self-Tuning Systems Handbook*. Kiev: Naukova Dumka, 1969.

[20] A.A. Feldbaum, *Computers in Automatic Control Systems*. Moscow: Fizmatgiz, 1959.

[21] A.A. Krasovskii, *Dynamics of Continuous Self-Tuning Systems*. Moscow: Fizmatgiz, 1963.

[22] S.M. Meerkov, "Asymptotic methods for investigating quasistationary states in continuous systems of automatic optimization,'' *Automat. Remote Contr.*, no.11, pp. 1726–1743, 1967.

[23] S.M. Meerkov, "Asymptotic methods for investigating a class of forced states in extremal systems,'' *Automat. Remote Contr.*, no. 12, pp. 1916–1920, 1967.

[24] S. M. Meerkov, "Asymptotic methods for investigating stability of continuous systems of automatic optimization subjected to disturbance action,'' *Avtomatika i Telemekhanika*, no. 12, pp. 14–24, 1968.

[25] P.F. Blackman, "Extremum-seeking regulators,'' in *An Exposition of Adaptive Control*, J. H. Westcott, Ed. New York: Macmillan, 1962, pp. 36–50.

[26] C.S. Draper and Y. Li, *Principles of Optimizing Control Systems*. New York: ASME, 1954.

[27] H.S. Tsien, *Engineering Cybernetics*. New York: McGraw-Hill, 1954.

[28] D.J. Wilde, *Optimum Seeking Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1964.

[29] M. Krstić and H.-H. Wang, "Design and stability analysis of extremum seeking feedback for general nonlinear systems,'' *Automatica*, vol. 36, no. 2, pp. 595–601, 2000.

[30] A. Banaszuk, S. Narayanan, and Y. Zhang, "Adaptive control of flow separation in a planar diffuser,'' in *Proc. 41st Aerospace Sciences Meeting and Exhibit*, Reno, NV, 2003, paper 2003-617.

[31] K. Peterson and A. Stefanopoulou, "Extremum seeking control for soft landing of an electromechanical valve actuator,'' *Automatica*, vol. 40, no. 6, pp. 1063–1069, 2004.

[32] D. Popovic, M. Jankovic, S. Manger, and A. R. Teel, "Extremum seeking methods for optimization of variable CAM timing engine operation,'' in *Proc. American Control Conf.*, Denver, CO, 2003, pp. 3136–3141.

[33] Y. Li, M.A. Rotea, G.T.-C. Chiu, L.G. Mongeau, and I.-S. Paek, "Extremum seeking control of a tunable thermoacoustic cooler,'' *IEEE Trans. Contr. Syst. Technol.*, vol. 13, no. 4, pp. 527–536, 2005.

[34] X.T. Zhang, D.M. Dawson, W.E. Dixon, and B. Xian, "Extremum seeking nonlinear controllers for a human exercise machine,'' in *Proc. 2004 IEEE Conf. Decision and Control*, Atlantis, Bahamas, 2004, pp. 3950–3955.

[35] J. Y. Choi, M. Krstic, K.B. Ariyur, and J.S. Lee, "Extremum seeking control for discrete-time systems,'' *IEEE Trans. Automat. Control*, vol. 47, no. 2, pp. 318–323, 2002.

[36] E.-W. Bai, L.-C. Fu, and S. Sastry, "Averaging analysis for discrete time and sampled data adaptive systems,'' *IEEE Trans. Circuits Syst.*, vol. 35, no. 2, pp. 137–148, 1988.