

# Auto-Tuning of PID Controllers via Extremum Seeking

Nick Killingsworth\* and Miroslav Krstić  
 Department of Mechanical and Aerospace Engineering  
 University of California, San Diego  
 La Jolla, CA 92093-0411, U.S.A.  
 nkillingsworth@ucsd.edu and krstic@ucsd.edu

**Abstract**—The proportional-integral-derivative (PID) controller is widely used in the process industry, but to various degrees of effectiveness because it is sometimes poorly tuned. The goal of this work is to present a method using extremum seeking (ES) to tune the PID parameters such that optimal performance is achieved. ES is a non-model based method which searches on-line for the parameters which minimize a cost function; in this case the cost function is representative of the controllers performance. Furthermore, this method has the advantage that it can be applied to plants in which there is no knowledge of the model. We demonstrate the ES tuning method on a cross section of plants typical of those found in industrial applications. The PID parameters are tuned based on the results of step response simulations to produce a response with minimal settling time and overshoot. Additionally, we have compared these results to those found using other tuning methods widely used in industry.

## I. INTRODUCTION

Controllers used in industry, especially in process control, are almost exclusively proportional-integral-derivative (PID) controllers. Despite the wide spread use of PID controllers they are sometimes poorly tuned impeding their effectiveness. Manually tuning such a controller is a tedious task since it requires the optimization of three parameters. To remedy this, much effort has been put into devising ways to systematically tune PID controllers. Many of these methods rely on knowledge of the plant model or require special experiments to be performed to identify a model of the plant. Reviews of these methods are given in the book by Astrom and Hagglund (1995) [1] and in the survey by Astrom et al. (1993) [2]. However, in many situations a model for the plant is not known and it is not desirable to open the process loop. Thus a method of tuning the parameters by means of the closed loop system is advantageous.

One such method is relay feedback tuning; however, this method requires that a relay is added to the system and is used as a feedback into the plant bypassing the controller. For most plants relay feedback will cause the system to oscillate, this allows the determination of one point on the Nyquist diagram. Based on the location of this point PID parameters can be formulated to give the closed loop system a desired phase and amplitude margin, see for example, [3]-[5]. Another non-model based method which has been applied to PID tuning and does not require opening of

the loop is iterative feedback tuning (IFT). This technique iteratively optimizes the controller parameters with respect to a cost function derived from the output signal of the closed loop system, and is discussed in detail in [6]. This method has been applied to tune a PID controller with a cost function based on the performance of the closed loop system during a step response, [7], [8].

In this work we apply a discrete version of extremum seeking (ES) from [9] to the task of optimizing the step response of a closed loop system in a fashion similar to that in [7], [8]. However, it should be noted that other forms of the cost function could easily be adapted to the ES tuning method. ES is a non-model based method which iteratively modifies the input of a function such that the output of the function is taken to a local minimum or maximum. ES achieves this by perturbing the input parameters of the system and then making an on-line estimation of the gradient of the output relative to the inputs. This gradient information is then used to modify the input parameters in the next iteration towards the minimum or maximum of the output. A block diagram of the discrete version of ES is depicted in Fig. 1 and will be discussed further in the following section. An overview of the ES theory as well as many state of the art applications is provided in [10].

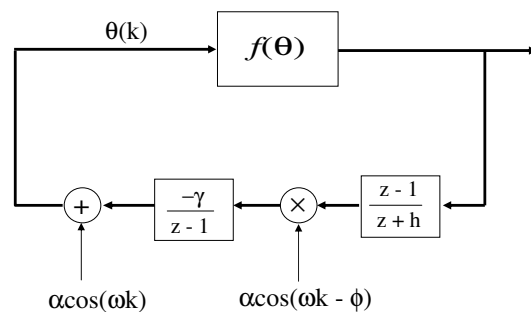


Fig. 1. Block diagram of discrete extremum seeking scheme.

## II. IMPLEMENTATION

ES is used to tune the parameters of a PID controller such that the tracking error of the closed loop system with an unknown plant is minimized. We use a standard PID controller, however the derivative term only acts on the

\*supported by NSF AGEP HRD 9978892 and the GEM program

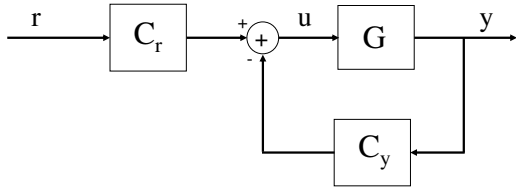


Fig. 2. Block diagram of closed-loop system.

measured plant output and not the reference signal. This reduces large control efforts during a step change in the reference signal. A block diagram of the controller is shown in Fig. 2, where

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

$$C_y = K \left( 1 + \frac{1}{T_i s} + T_d s \right). \quad (2)$$

Note  $r$ ,  $u$ , and  $y$  are the reference, control, and output signal, respectively. The effectiveness of the above controller is quantified using the following cost function evaluated at the conclusion of a step response experiment,

$$J(\theta_k) = \frac{1}{T - t_0} \int_{t_0}^T e^2(\theta_k, t) dt, \quad (3)$$

where the error is defined as the difference between the reference and the output signal of the closed loop system,  $e(\theta, t) = r - y(\theta, t)$ , and  $\theta$  is a vector containing the PID parameters,  $\theta = [K, T_i, T_d]^T$ .

In (3),  $t_0$  and  $T$  are the times during the step response experiment at which we begin and end taking into account the error during calculation of the cost function, respectively. Setting  $t_0$  equal to the settling time of the closed loop system effectively puts zero weights on the transient portion of the response. This focuses all available degrees of freedom of the controller on minimizing the error beyond the settling time and none on the transient portion. From principles of optimality we know that this will always yield a cost beyond  $t_0$  which is equal if not better than that of a controller minimized using a value earlier than  $t_0$ . This method of using zero weights in the cost function is suggested by [7] where it has been applied to the IFT method of PID tuning.

The discrete ES algorithm has been implemented using the following equations, which are equivalent to the system representation shown in Fig. 1,

$$\xi(k) = -h\xi(k-1) + J(k-1) \quad (4)$$

$$\hat{\theta}_i(k+1) = \hat{\theta}_i(k) - \gamma_i \alpha_i \cos(\omega_i k) [J(k) - (1+h)\xi(k)] \quad (5)$$

$$\theta_i(k+1) = \hat{\theta}_i(k+1) + \alpha_i \cos(\omega_i(k+1)), \quad (6)$$

where  $\xi(k)$  is a scalar and the subscript  $i$  indicates the  $i^{th}$  entry of the vectors  $\hat{\theta}(k+1)$  and  $\theta(k+1)$ .  $\gamma$  is the estimation gain and  $\alpha$  is the perturbation step size. The modulation frequency  $\omega$  should be chosen such that  $\omega = a\pi$ ,  $0 < |a| < 1$ , where  $a$  is rational, this guarantees that

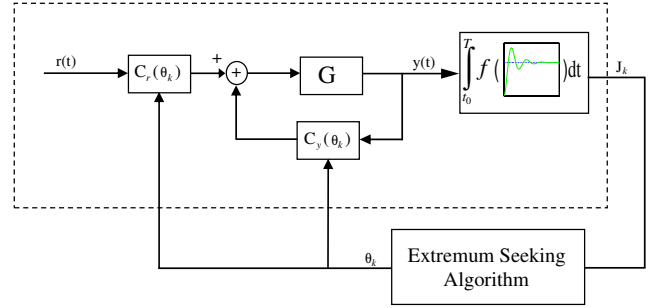


Fig. 3. Overall ES PID tuning scheme.

the Nyquist criterion is satisfied. Additionally, the high-pass filter  $\frac{z-1}{z+h}$  is designed with  $0 < h < 1$ .

The overall ES PID tuning scheme is presented in Fig. 3. The step function experiment portion of the system, which is contained within the dashed box is run iteratively. The ES algorithm uses the discrete value of the cost function,  $J_k$  generated at the completion of each iteration to compute the next set of controller parameters,  $\theta_k$ . A new step function experiment is then performed with the new controller parameters and the process continues iteratively in this fashion.

### III. SIMULATIONS

In this section we demonstrate the ES method of PID tuning, as well as, compare it with the IFT method and two classical PID tuning methods: the Ziegler-Nichols tuning rules (ZN) and the internal model control (IMC) method. The ZN method consists of performing an open loop experiment and increasing the system gain until the point where the Nyquist plot crosses the negative axis. PID parameters are then prescribed based on this gain and the frequency at which the system oscillates [1]. The IMC method assumes the plant is of the following form,  $G(s) = \frac{K_p}{1+T_s} e^{-Ls}$ ; based on this model PID parameters are then determined to give the closed-loop system a desired response. If the actual plant model is unknown a step response experiment can be used to estimate it in the above form. These four methods have been applied to the following four models with dynamics typical of many industrial plants.

$$G_1(s) = \frac{1}{1+20s} e^{-5s} \quad (7)$$

$$G_2(s) = \frac{1}{1+20s} e^{-20s} \quad (8)$$

$$G_3(s) = \frac{1}{(1+10s)^8} \quad (9)$$

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

Notice that plants  $G_1$  and  $G_2$  have time delays,  $G_3$  has multiple poles, and  $G_4$  is non-minimum phase. These models have been taken from [8] to allow comparison with the IFT method, additionally controller parameter values for the ZN and IMC methods have also been used from this work.

TABLE I  
PID PARAMETERS FROM FOUR TUNING METHODS FOR SIMULATIONS  
OF  $G_1$ .

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

Simulations of the closed loop systems have been performed using a time step of 0.01 s and the time delays have been approximated using a third-order Padé approximation. The values for the PID controller parameters found from the ZN tuning rules have been used as a starting point for the ES tuning method in all simulations. For all simulations the ES parameters,  $a$  and  $h$  in equations (4) - (6) were set to 0.8 and 0.5, respectively.

#### A. Simulation of $G_1$

The ES PID tuning method has been applied to the first transfer function  $G_1$  in (7), which has a time delay of 5 s. For these simulations we set  $\alpha = [0.1, 1, 0.1]^T$ ,  $\gamma = [200, 1200, 200]^T$ , and  $\omega_i = a^i \pi$  where  $a = 0.8$  as mentioned previously. The results of these simulations are presented in Fig. 4. We can see from the upper two plots of this figure that the ES method quickly minimizes the cost function (3) and converges in less than 10 iterations to PID parameters which produce a local minimum. ES achieves this response by reducing the influence of the integral portion of the controller and increasing the value of the integral time to almost three times that given by the ZN tuning rules. The performance of the PID parameters found via the ES tuning method compared to those of the other three methods is also shown, we can see that ES produces a controller with performance that is roughly equivalent to that found using IFT. This is expected since both methods attempt to minimize the same cost function. It can also be seen that these two methods yield closed loop systems with less overshoot as well as smaller settling times than the other tuning methods evaluated in this study.

#### B. Simulation of $G_3$

Here we have used the ES PID tuning method to tune the PID parameters of the closed-loop system with  $G_3$  in (9), which has a single pole of order eight. In contrast, for this plant we found that using a cost function based on the integrated absolute error (IAE)

$$J(\theta_k) = \frac{1}{T - t_0} \int_{t_0}^T |e(\theta_k, t)| dt, \quad (11)$$

instead of the one based on the integrated squared error (ISE) in (3) produced a smoother map and thus the behavior of the ES algorithm was similarly more well behaved.

The following ES parameters were used in these simulations:  $\alpha = [0.1, 0.7, 0.4]^T$ ,  $\gamma = [22, 200, 22]^T$ ,  $\omega_1 = a\pi$ ,

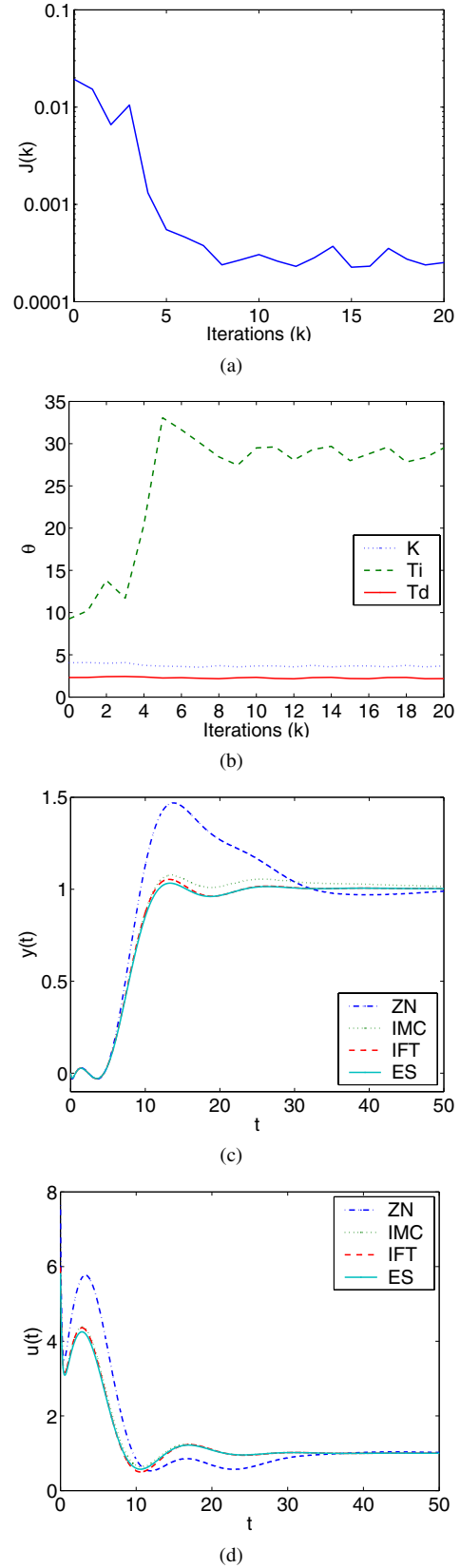


Fig. 4. Evolution of the cost function (a) and the PID parameters (b) during ES tuning of the closed-loop system with  $G_1(s)$ . The output signal (c) and control signal (d) during a step response experiment of the closed-loop system with  $G_1(s)$  and PID controller parameters obtained by means of the four different methods.

TABLE II  
PID PARAMETERS FROM FOUR TUNING METHODS FOR SIMULATIONS  
OF  $G_3$ .

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	0.727	55.5	19.3

TABLE III  
PID PARAMETERS FROM FOUR TUNING METHODS FOR SIMULATIONS  
OF  $G_4$ .

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

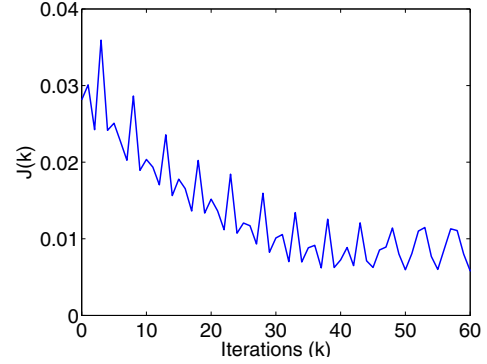
$\omega_2 = a\pi + \frac{\pi}{2}$ , and  $\omega_3 = a^3\pi$ . The results are presented in Fig. 5 and the PID parameters are listed in Table 3. From these plots it can be seen that the ES algorithm vastly improves upon the step response behavior given by the ZN tuning rules, and returns a response that is once again similar to that achieved by the other two methods. In this case ES finds that a reduction in both the integral time  $T_i$  and controller gain  $K$  produce a smaller value of the cost function. This plant proves more of a challenge and requires roughly 50 iterations until the PID parameters converge.

### C. Simulation of $G_4$

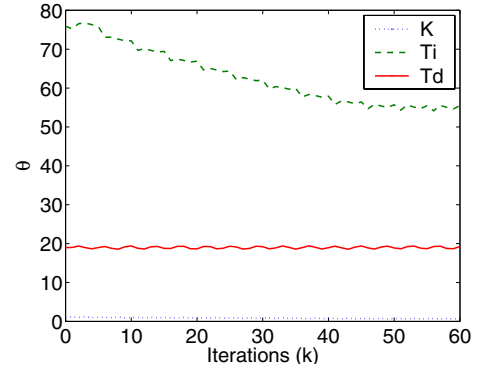
The PID controller of the closed-loop system with non-minimum phase  $G_4$  in (10) has also been tuned using the ES method. For these simulations we set  $\alpha = [0.05, 0.6, 0.2]^T$ ,  $\gamma = [2000, 10000, 2000]^T$ ,  $\omega_1 = a\pi$ ,  $\omega_2 = a\pi + \frac{\pi}{2}$  and  $\omega_3 = a^3\pi$ . The results can be found in Fig. 6, additionally the PID parameters are listed in Table 4. These plots show that once again the ES tuning method produces a step response very similar to that obtained using the IFT method both of which yield no overshoot and a settling time much smaller than those controllers tuned using the ZN or IMC methods. However, the ES method in this case does produce a slightly larger control signal initially than the IFT method. Also note that larger values of the integral and derivative times yield a better system response over that found using the ZN method.

### D. Cost function comparison

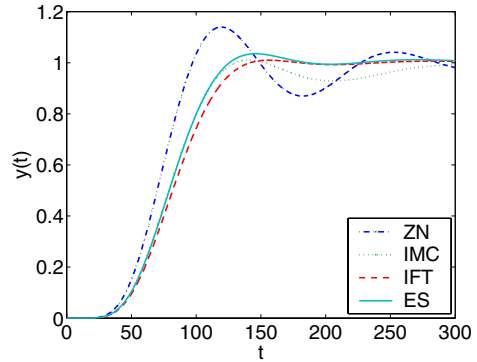
As we found for the plant  $G_3$ , choosing the right cost function is key to getting good results and in this section we will expand upon this topic. We will use the ES algorithm to minimize the following cost functions obtained from simulations of plant  $G_2(s)$ : IAE, which is equal to (11) with  $t_0 = 0$ , ISE, which is equal to (3) with  $t_0 = 0$ , Window, which is (3) with  $t_0$  set to some positive “window time”,



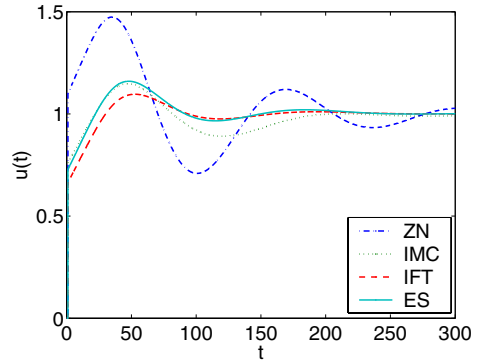
(a)



(b)

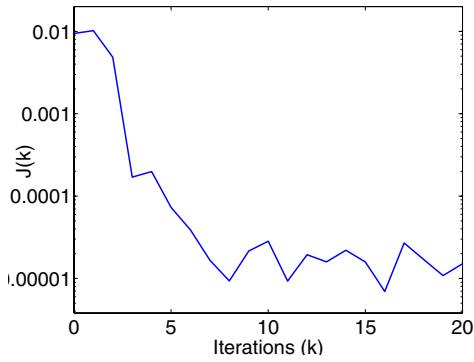


(c)

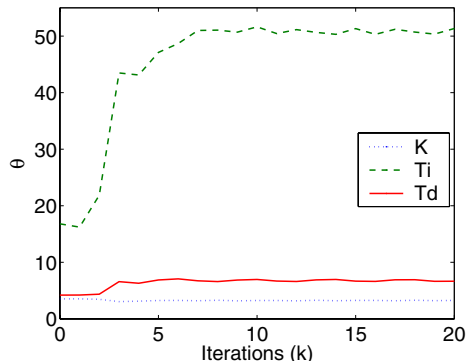


(d)

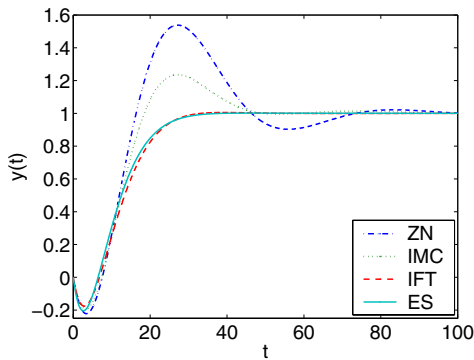
Fig. 5. Evolution of the cost function (a) and the PID parameters (b) during ES tuning of the closed-loop system with  $G_3(s)$ . The output signal (c) and control signal (d) during a step response experiment of the closed-loop system with  $G_3(s)$  and PID controller parameters obtained via the four different methods.



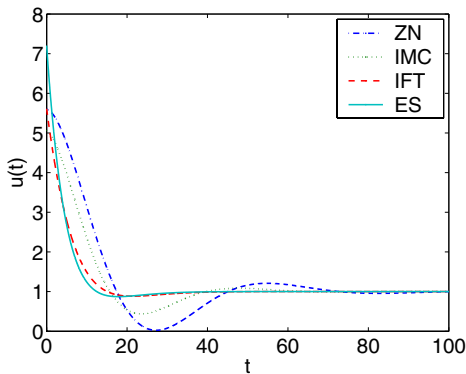
(a)



(b)

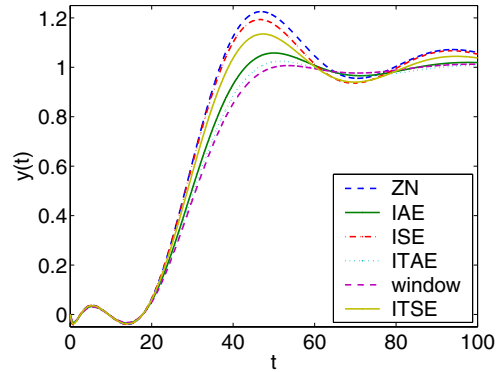


(c)

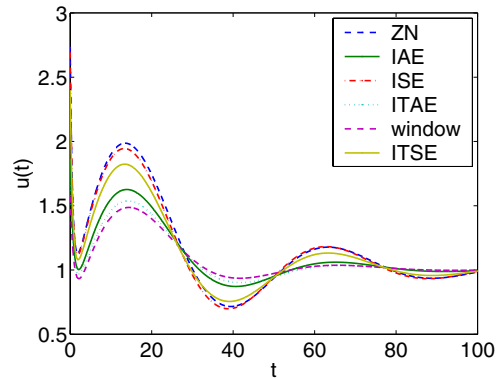


(d)

Fig. 6. Evolution of the cost function (a) and the PID parameters (b) during ES tuning of the closed-loop system with  $G_4(s)$ . The output signal (c) and control signal (d) during a step response experiment of the closed-loop system with  $G_4(s)$  and PID controller parameters obtained via the four different methods.



(a)



(b)

Fig. 7. The output signal (a) and control signal (b) during a step response experiment of the closed-loop system with  $G_2(s)$  and PID controller parameters obtained using ES with various cost functions.

and

$$ITAE = \frac{1}{T} \int_0^T t |e(\theta_k, t)| dt \quad (12)$$

$$ITSE = \frac{1}{T} \int_0^T t e^2(\theta_k, t) dt. \quad (13)$$

Note the time-dependent weighting in (12) and (13), which de-emphasizes the transient portion of the response. Fig. 7 illustrates the importance of using a cost function which correctly characterizes the desired system behavior. We can see that the Window cost function produces a response with the smallest overshoot and fastest settling time. The ITAE and IAE have similar performance, which is slightly worse than that produced by using the Window cost function, whereas ISE and ITSE do not offer much of an improvement over that found using the ZN tuning rules.

#### E. Control saturation

In many applications of PID control we must deal with actuator saturation. Actuator saturation can result in integrator windup, in which case the feedback loop becomes temporarily disconnected since the controller no longer has an effect on the system. In this section we investigate how the ES tuning method handles this nonlinearity. We

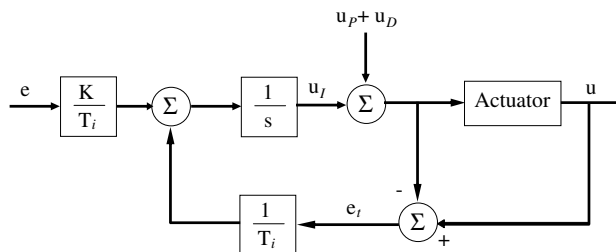


Fig. 8. Tracking anti-windup scheme.

TABLE IV

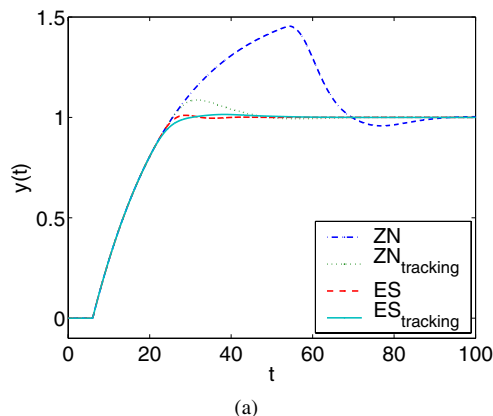
PID PARAMETERS FROM THREE TUNING METHODS FOR SIMULATIONS OF  $G_1$  WITH SATURATION.

Tuning method	K	$T_i$	$T_d$
ZN	4.06	9.25	2.31
ES	3.61	47.6	1.81
ES <sub>tracking</sub>	4.07	12.8	2.20

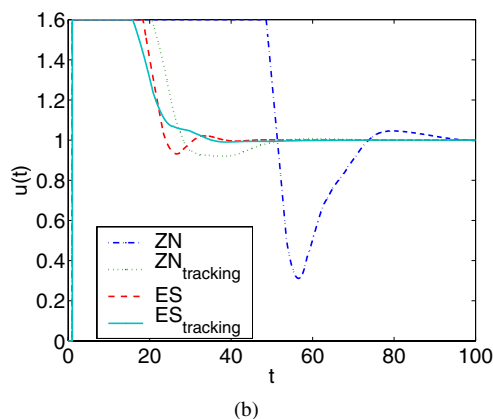
will tune the same PID controller using ES, but in the presence of control saturation, additionally we tune this PID controller with a tracking anti-windup scheme. Tracking modifies the integral control signal using a feedback which is proportional to the difference between the requested control effort and the control effort produced by the actuator  $e_t$ , as depicted in Fig. 8. This scheme is explained in more detail in [1]. We compare these results to a standard PID controller tuned using ZN and the same controller with tracking in Fig. 9 for plant  $G_1(s)$  with control saturation of 1.6. We can see from these plots that there is quite a bit of overshoot for the system tuned using ZN, but that even without tracking the system tuned using ES has minimal overshoot. ES is able to do so well because it increases the integral time such that the system is more robust to saturation as can be seen in Table 5. The ES method even finds controller parameters which outperform the ZN method with tracking. And the controller does almost as well as the system with tracking tuned using ES. This illustrates the downfall of using an open-loop tuning scheme such as ZN when actuator saturation is not accounted for.

#### IV. CONCLUSIONS

We have shown that ES provides an effective and efficient method to tune PID controllers that minimize a cost function which characterizes some desired behavior. This tuning method has been demonstrated on four typical plants and found to give parameters that yield performance better than or comparable to that of other popular tuning methods. However it should be noted that initial values of the PID parameters are needed, thus ES tuning is best used as a complement to another PID parameter design method. It should also be made clear that the cost function can be modified such that different performance attributes are emphasized.



(a)



(b)

Fig. 9. The output signal (a) and control signal (b) during a step response experiment of the closed-loop system with  $G_1(s)$ , control saturation of 1.6, and PID controller parameters obtained using ZN and ES both with and without tracking.

#### REFERENCES

- [1] K. J. Astrom, and T. Hagglund, *PID Controllers: Theory, Design and Tuning (2nd ed.)*. North Carolina, USA: Instrument Society of America, 1995.
- [2] K. J. Astrom, T. Hagglund, C. C. Hang, and W. K. Ho, "Automatic tuning and adaptation for PID controllers - A survey," *Control Eng. Practice*, vol. 1, no. 4, pp. 699-714, 1993.
- [3] K. J. Astrom, and T. Hagglund, "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 Proceedings*, 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] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, "Iterative feedback tuning: Theory and applications," *IEEE Control Systems Magazine*, vol. 18, no. 4, pp. 26-41, 1998.
- [7] O. Lequin, M. Gevers, and T. Triest, "Optimizing the settling time with iterative feedback tuning," in *Proceedings of the 14th IFAC world congress*, Beijing, P.R. China, pp. 433-437, 1999.
- [8] O. Lequin, E. Bosmans, and T. Triest, "Iterative feedback tuning of PID parameters: Comparison with classical tuning rules," *Control Eng. Practice*, vol. 11, no. 9, pp. 1023-1033, 2003.
- [9] J. Y. Choi, M. Krstic, K. B. Ariyur, and J. S. Lee "Extremum Seeking control for discrete-time systems," *IEEE Transactions on Automatic Control*, vol. 47, pp. 318-323, 2002.
- [10] K. B. Ariyur, and M. Krstic, *Real-Time Optimization by Extremum Seeking Feedback*, Hoboken, N.J. : Wiley-Interscience, 2003.