PI Tuning via Extremum Seeking Methods for Cruise Control

Yiyao(Andy) Chang, Scott Moura

ABSTRACT

In this study, we reproduce the results from an existing paper on PID tuning using extremum seeking (ES) methods. In addition to analyzing performance with respect to existing tuning methods, we investigate the ES algorithm parameters and how their values impact stability and convergence speed. To demonstrate the efficacy of ES on tuning classical controllers, we apply the algorithm to a simple cruise control problem. The results indicate the effectiveness of ES as an adaptive control algorithm. Moreover, sensitivity analysis on the ES algorithm parameters and controller gains reveal several potential areas for improvement, such as controller gain constraints, variable ES parameters, and dynamic cost functions. As a result, we suggest general guidelines for tuning controllers with ES.

1. INTRODUCTI0N

The Proportional-Integral-Derivative (PID) controller, a generic feedback control mechanism, is used widely in industrial control systems. In classical PID tuning methods, engineers typically satisfy requirements based on Ziegler-Nichols (ZN) conditions or open-loop frequency responses. These are proven methods but require substantial knowledge of the plant and use trial-and-error to achieve satisfactory system performance. In many cases, however, the plant model is not known, thus a closed-loop PID tuning method is beneficial.

There are various model-free tuning schemes suitable for PID controllers. In relay feedback tuning [1-3], the feedback controller is replaced by a relay which causes the system to oscillate, thus defining one point on the Nyquist plot. Based on the location of this point, PID gains can be determined to achieve the desired phase and gain margins. However, this method requires a system model, which may not be known. Unfalsified control [4,5] updates the PID gains based on if the controller falsifies a criterion or performance requirement. However, it requires a finite set of candidate PID gains, which can be time consuming or impractical to generate.

Extremum seeking (ES) offers a model-free, online control synthesis method to achieve optimal performance in some sense; hence it is able to tune PID gains in a more efficient fashion without requiring any knowledge or modification of the plant. In the paper we reproduce here [6], Killingsworth and Krstic explore the ES algorithm and apply it to several plants to compare system performance with classical tuning methods.

2. PAPER REPRODUCTION

2.1 EXTREMUM SEEKING ALGORITHM

When applying ES for PID tuning, one can implement the ES algorithm as an outer feedback loop as demonstrated in Figure 1 with plant transfer function G, PID controllers C_y, C_r, and the associated cost function, $J(\theta(k))$.

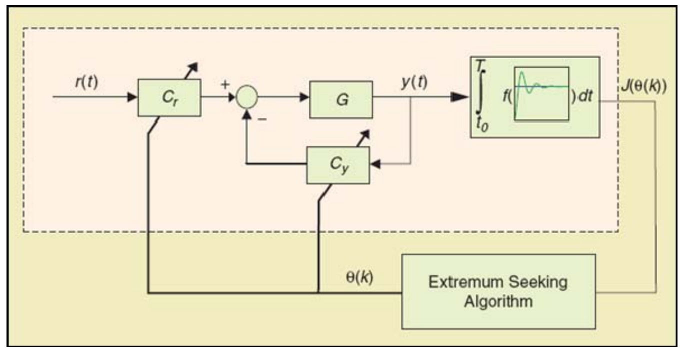


Figure 1: ES PID tuning scheme [6].

For a given input to the system, the cost function evaluates the effectiveness of the PID gains (θ). ES approximates the cost function gradient with respect to the PID gains in two steps. First, it applies a high-pass filter to remove the dc gain. Second, it demodulates the signal to remove the perturbation applied in the previous iteration. Information from the cost gradient is then used to calculate PID gains for the next iteration. Specifically, ES integrates the estimated cost gradient and then perturbs the resulting signal by a harmonic function to give new PID input gains. Iteratively, ES modifies these gains until the cost function has reached a local minimum.

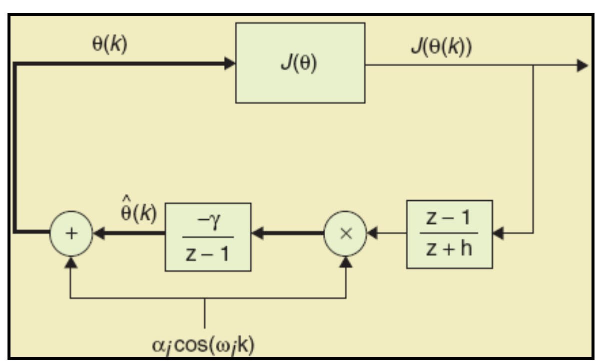


Figure 2: Discrete-time ES algorithm [6].

2.2 PID TUNING ON SAMPLE PLANT

Following the paper by Killingsworth and Krstic [6], we apply ES on the following plant:

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

The PID controller is split into a pre-compensator C_r and feedback controller C_y with expressions given by

$$C_r(s) = K\left(1+\frac{1}{T_i s}\right) \tag{2}$$

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

Notice the only difference between the two controllers is that the pre-compensator lacks a derivative term T_d. If this term were included, a step input would induce an infinite output from the controller. As a result, we apply the derivative term on the feedback signal only.

In this particular application, we use the integrated squared error (ISE) cost function,

$$J(\theta) = \frac{1}{T-t_0}\int_{t_0}^{T} e^2(t,\theta)dt \tag{4}$$

where the error $e(t,\theta) = r(t) - y(t,\theta)$ takes the difference between the reference and output signals. The PID tuning gains are defined as

$$\theta = \begin{bmatrix} K & T_i & T_d \end{bmatrix}^T \tag{5}$$

From an initial set of PID gains found using ZN, the ES algorithm reaches a minimum after 20 iterations, as shown in Figure 3. The PID gains change by 25%, 2.2%, and 7.6% for K, T_i, and T_d, respectively. This result suggests that the gain K has the greatest effect on minimizing the cost for this specific plant. The cost function surface plot, shown in Figure 4 as a function of K and T_d, (T_i is set constant), confirms this observation. Moreover, it may be possible to achieve satisfactory performance while significantly reducing computational time by applying ES on gain K only. We discuss this idea more thoroughly in Section 4.2.

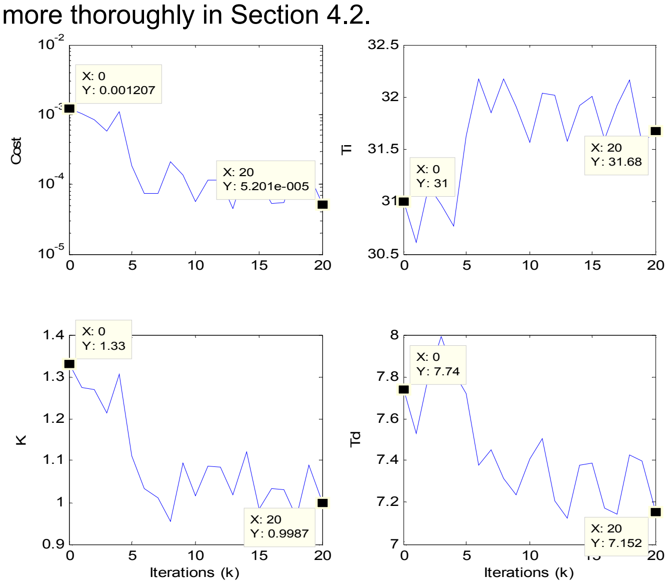


Figure 3: Cost function values and PID Gains

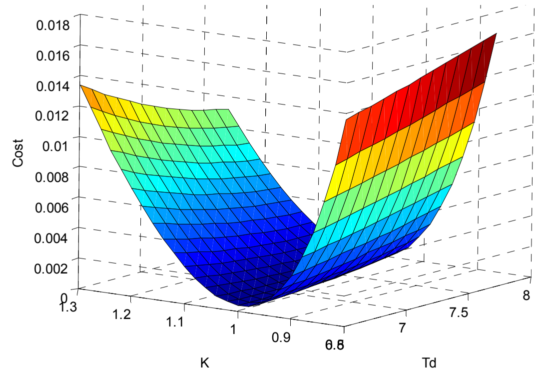
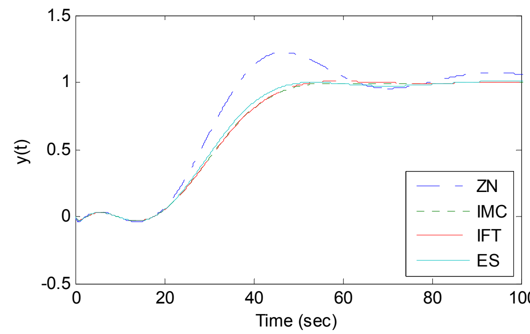


Figure 4: Cost function values vs. K and T_d.

As shown in Figure 5, ES achieves significantly better performance than ZN and roughly equal performance to more sophisticated tuning methods, such as internal model control (IMC) and iterative feedback tuning (IFT).



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	0.9987	31.68	7.152

Figure 5: System output (top) and PID gains (bottom).

2.3 SENSITIVITY OF ES PARAMETERS

Killingsworth and Krstic argue that the solution found by ES is fairly insensitive to the perturbation amplitude (α_i) and adaptation gain (γ) parameters, shown in Figure 2. Based on the cost trajectories shown in Figure 6, all four combinations of ES parameters are able to reach the same minimum cost, but at different convergence rates. This result speaks to the robustness of ES, however the authors do not include the case when the ES parameters are increased.

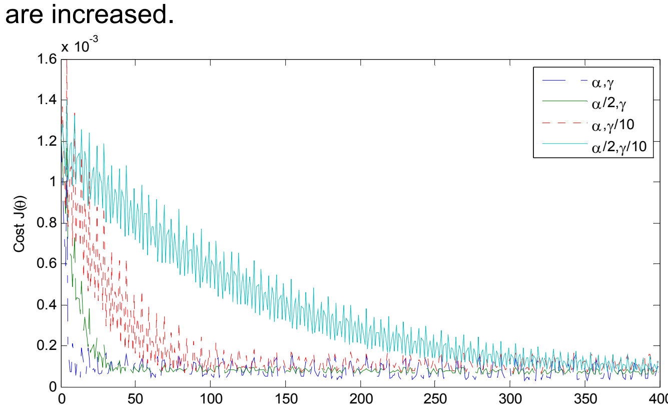


Figure 6: Sensitivity to ES parameters

As the ES parameters increase, the perturbation amplitude and adaptation gain increase, demonstrated conceptually by Figure 7. Depending on the initial PID gains, it is possible to drive the overall system into an unstable region. Therefore we conclude that ES is insensitive to smaller or more conservative parameters, but extremely sensitive to larger or more aggressive parameters.

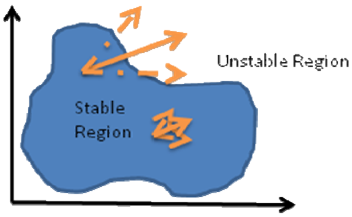


Figure 7: Stability of ES

3. CRUISE CONTROL VIA ES METHODS

The cruise control problem contains nonlinear longitudinal vehicle dynamics due to viscous air drag effects and road grade. As a result, a PI controller with fixed gains can be optimally tuned for only one operating point. We propose to utilize ES to adapt the PI gains online to achieve optimal performance for the entire operating range.

3.1 LONGITUDINAL VEHICLE DYNAMICS MODEL

We assume the vehicle can be modeled as a disk rolling without slip. The body experiences forces due to gravity, viscous air drag, rolling friction, ground normal forces, and traction force. The free-body diagram is presented below:

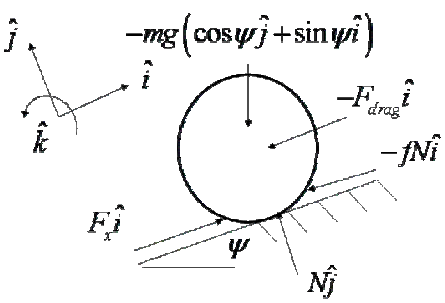


Figure 8: Rolling Disk Model.

Forming Newton's 2nd law produces the following system dynamics, in which velocity (v) is the state variable and traction force (F_x) is the control input. Other model parameters include vehicle mass (m), road grade angle (ψ), rolling friction coefficient (f), air density (ρ), frontal cross-sectional area (A), drag coefficient (C_d), and wind speed (v_w).

$$m\frac{dv}{dt} = F_x - mg\sin\psi - fmg\cos\psi - 0.5\rho A C_d\left(v+v_w\right)^2 \quad (6)$$

Note that the system is nonlinear with respect to the road grade angle and velocity. We linearize the system about a nominal operating point to apply the ZN method. The PI gains acquired from ZN will serve as the initial values for the ES algorithm.

3.2 LINEARIZATION

A first order Taylor series expansion about a nominal operating point produces the following linear system, given in terms of perturbation variables:

$$\begin{aligned}\delta\dot{v} = &-\frac{1}{m}\rho A C_d\left(v_0+v_w\right)\delta v + \frac{1}{m}\delta F_x \\ &+\frac{1}{m}\left[-mg\cos\psi_0 + fmg\sin\psi_0\right]\delta\psi\end{aligned} \quad (7)$$

where $\delta(\bullet) = (\bullet) - (\bullet)_0$ represents the perturbation from the nominal value. In this set of coordinates δv, δF_x, $\delta\psi$ may be interpreted as the state variable, control input, and disturbance input, respectively. Suppose we select the operating point given in Table 1. Furthermore, we use the following model parameters: $m = 1000kg$, $f = 0.015$, $\rho = 1.202\ kg/m^3$, $A = 1m^2$, $C_d = 0.5$, $v_w = 2m/s$. The resulting system is then represented by the first-order transfer function of traction force to vehicle speed, shown in Equation 8.

Variable	Nominal Value
Traction Force, F_x	293 N
Road Grade, ψ	0 radians
Wind Speed, v_w	2 m/s
Vehicle Speed, v	20 m/s ≈ 45 mph

Table 1: Nominal Operating Point [7].

$$\frac{Y(s)}{U(s)} = \frac{0.0758}{75.75s+1} = \frac{K}{\tau s+1} \quad (8)$$

The time constant, $\tau = 75.75\ sec$, is quite slow and the steady-state error (with respect to step inputs) is 92.42%. Both of these characteristics are unacceptable for cruise control applications. A suitable controller to resolve both of these issues is a PI controller [1].

$$C(s) = K_{PI}\frac{\tau_{PI}s+1}{s} \quad (9)$$

Next, we must determine appropriate values for K_{PI} and τ_{PI}. Although this is a non-trivial task and the greater subject of discussion for this paper, one simple option is the ZN method:

$$C(s) = (0.3845)\frac{43s+1}{s} \quad (10)$$

These controller gains serve as the initial values for the ES algorithm and also provide a baseline comparison to evaluate performance.

3.3 CASE STUDY: VARIABLE REFERENCE SPEEDS

We now demonstrate the utility of ES PI tuning and compare this algorithm to more commonly used techniques, namely fixed gains generated by ZN and gain scheduling. The ES algorithm performs one discrete-time iteration using the ISE cost function formulation for each second of the continuous-time closed-loop system. As previously stated, the controller gains obtained from ZN provide initial values for the algorithm.

$$\theta(0) = [K_{PI} \quad \tau_{PI}]_0^T = [0.3845 \quad 43]^T \tag{11}$$

Finally, we set the ES algorithm parameters to $\alpha = [0.05, 0.05]^T$, $\gamma = [600, 200]^T$, and $\omega = 0.8^2\pi$. Conservative (small) values for α and γ may cause slow convergence rates, but ensure stability for large cost function values.

The first case study considers zero road grade with a ramp up and ramp down in desired velocity, demonstrated in Figure 9. As expected, gain scheduling with ZN gains (ZN GS) produces slightly better reference tracking results than fixed ZN gains (ZN Fix). ES, however, produces significantly better results than both methods. This result is supported by the speed error responses shown in Figure 10.

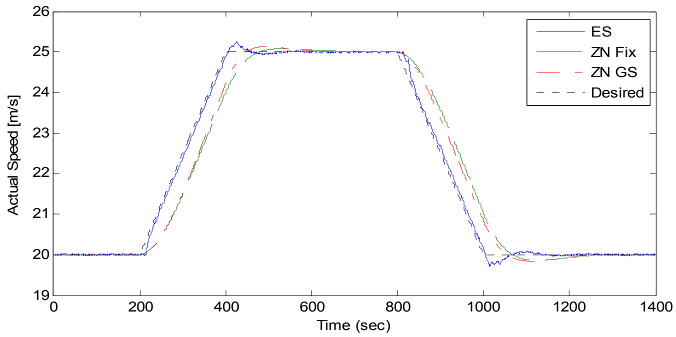


Figure 9: Reference tracking performance.

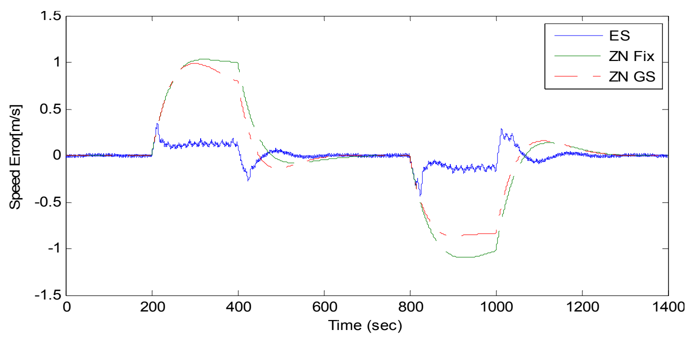
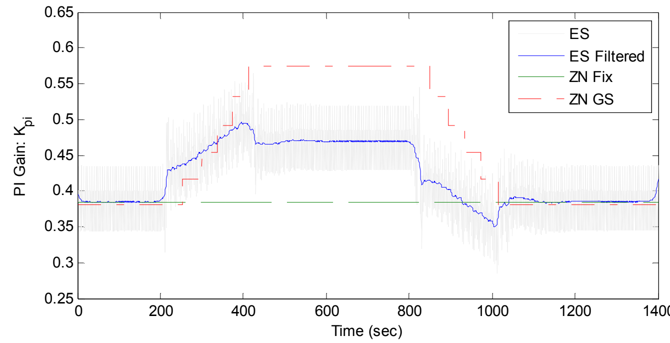


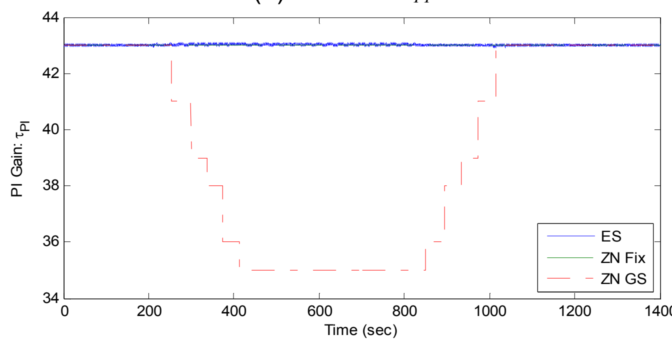
Figure 10: Speed error for each control algorithm

The most interesting results occur in the time-responses of the controller gains, K_{PI} and τ_{PI}, given in Figures 11a, b. In Figure 11a, the response associated with ES is low-pass filtered to provide a qualitative interpretation and does not represent the actual response. At the plateau ZN GS selects a 22% greater value for K_{PI} than ES, which intuitively implies a faster response. However, ZN GS also selects a 29% slower zero placement τ_{PI}, thereby partially nullifying the speed acquired from a higher gain. Conversely, ES keeps τ_{PI} nearly constant compared to ZN GS, which implies the cost function is insensitive to variations in τ_{PI} relative to variations in K_{PI}.

Another key observation arises after the ramp down, where ES appears to converge to the same controller gains as originally selected by ZN. This result suggests the original ZN gains were optimal with respect to the cost function. We do not expect this since ZN does not consider optimality conditions or our specific cost function formulation. As it turns out, the gains will slowly converge to new values for a very large number of iterations. This occurs because our conservative selection of ES parameters (α, γ), required to ensure stability, produce very slow convergence speeds.



(a) PI Gain: K_{PI}



(b) PI Gain: τ_{PI}

Figure 11: Controller parameter time-responses.

The control input (traction force) provides additional insight to ES. Again, this signal is low-pass filtered to acquire a qualitative interpretation. ES produces a faster, more aggressive reaction to variations in desired speed, hence achieving better performance. In practical systems, however, actuator saturation may not allow such aggressive control efforts. Also, the sinusoidal perturbations within the ES algorithm produce oscillations in the control input. While some degree of oscillations may be tolerable with respect to amplitude and frequency, the oscillations shown in Figure 12 degrade drivability in cruise control applications.

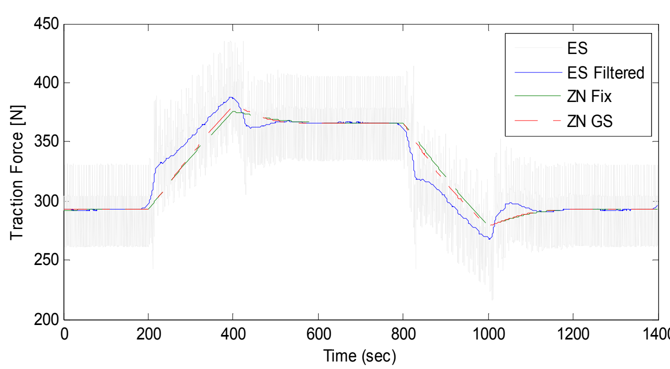


Figure 12: Traction force time-responses.

3.4 CASE STUDY: VARIABLE ROAD GRADE

For the second case study, we consider a vehicle maintaining constant speed as it climbs a hill, defined by a step-up and step-down in road grade. Here, we do not include gain scheduling and instead focus our attention on how the PI controller gains adjust as road conditions change. The speed error in Figure 13 clearly indicates that ES regulates velocity more effectively than fixed ZN gains. Specifically, the overshoot is reduced by 75% and settling time is reduced by 80%. These improvements are significant considering the modest road conditions investigated here.

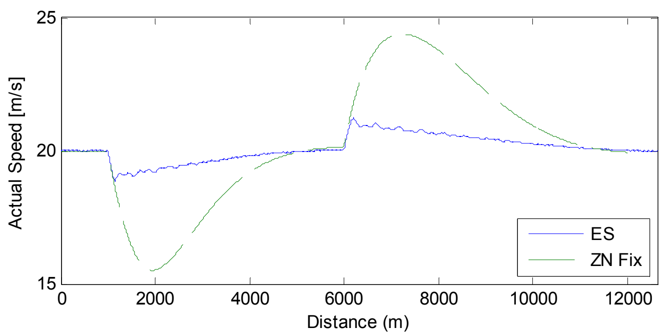
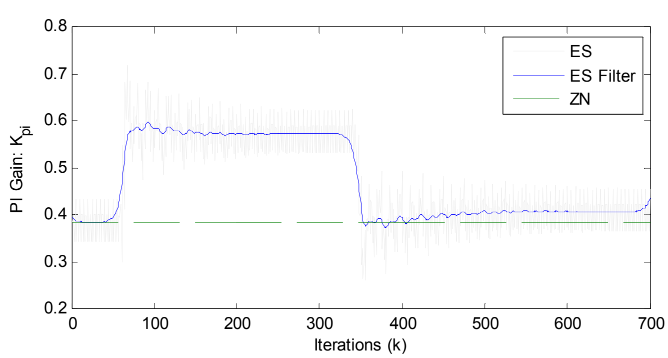
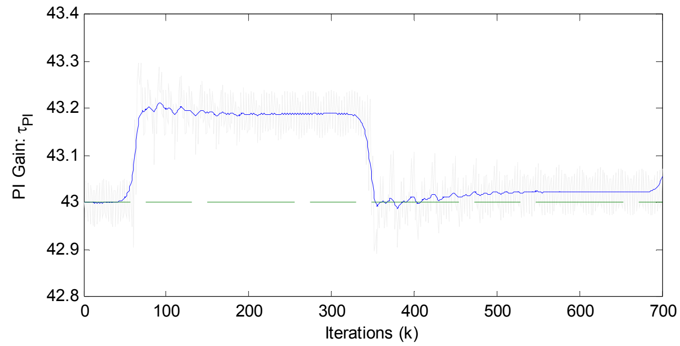


Figure 13: Speed error for varying road grade

Figures 14a,b demonstrate the controller parameter trajectories for the road grade step profile. Two noteworthy observations provide insight toward controller parameter sensitivity and convergence. First, the optimal values for K_{PI} and τ_{PI} increase by 49% and 0.45%, respectively during ascent. As in the first case study, the insignificant change for τ_{PI} indicates the cost function is not very sensitive to variations in this parameter. Therefore, it may be possible to simplify the PI tuning scheme to a single degree of freedom without significant sacrifices in performance. This result has significant implications on computational power and calibration, discussed further in Section 4.2. Secondly, the controller gains converge to values different than those given by ZN for a large number of iterations. In particular, the optimal value for K_{PI} is 30% greater, therefore confirming the claim made in the first case study.



(a) PI Gain: K_{PI}



(b) PI Gain: τ_{PI}

Figure 14: Controller parameter time-responses.

4. DISCUSSION OF RESULTS

4.1 ES PARAMETER CONSTRAINTS

The sinusoidal perturbation amplitude, α, and the integration step size, γ, can have significant impacts on stability, rate of convergence, and the solution itself. Killingsworth and Krstic argue that decreasing these values does not affect stability, but may reduce convergence speed [6]. Here we have conceptually shown that increasing these parameters may increase the convergence rate, but destabilize the system. Therefore, a potential improvement to this algorithm is to provide upper-bound constraints on these parameters which guarantee stability. This enhancement necessarily assumes a plant model is readily available to analyze for stability conditions. However, recall that ES is a model-free optimization algorithm in the first place, therefore rendering parameter constraints impractical. Nonetheless, this is an interesting area for future research which may provide useful insight to extend the stability of ES without compromising its model-free approach.

Two additional drawbacks of ES are that it does not find global solutions and large amplitude oscillations may be intolerable for certain systems. To alleviate both of these issues, one could allow the ES parameters, α and γ, to vary dynamically. In particular, these parameters could be configured to perform a global search initially, and then switch to a local optimization approach. Moreover, allowing these values to vary with time provides the ability to reduce oscillation amplitudes when a suitable solution has been achieved. In the case of cruise control, this could potentially improve drivability.

4.2 CONTROLLER GAIN SENSITIVITY

The two cruise control case studies provided here indicate that ES impacts only one of two controller gains. This fact arises from the cost function's sensitivity to variations in each of the gains. If it is known that the cost function is relatively insensitive to variations in one (or many) gains, these variables can be removed from the tuning algorithm, hence eliminating degrees of freedom. Additionally, consider the scenario where multiple equivalent control systems must be calibrated. If ES on one system indicates that tuning a subset of all controller gains is required to achieve satisfactory performance, calibration time for the remaining systems can be significantly reduced. Although this idea is not pursed in depth here, we propose a rigorous proof for this hypothesis as an area for future work.

4.3 COST FUNCTION FORMULATION

The cost function establishes the performance of the controller derived from ES, and therefore impacts how new controller gains are computed. Throughout the study, we have only considered the ISE cost function formulation to minimize system output error. Other cost functions may be applied to capture desired characteristics, such as minimizing overshoot or rise time, to give some examples. Although we maintained the same cost function throughout the simulations, it is possible to adjust this function in real time to emphasize different performance aspects. However, the cost function values directly correlate to how new gains are selected. As a result, the ES parameters must also be changed simultaneously to ensure stability.

4.4 CONVERGENCE SPEED & APPLICATIONS

The longitudinal vehicle dynamics are considerably slower than the ES dynamics. This property is extremely important to ensure improved performance. In fact, this principle is similar to pole-placement in Luenberger observer design, where the error dynamics must be significantly faster than plant dynamics to acquire correct state variable values. Unfortunately, applying this principle to ES is not as straight forward as state feedback observer design, since it requires careful tuning of the ES algorithm parameters. One naïve, yet simple approach used in existing ES literature [9] is to start with very conservative values and gradually increase them until the desired convergence speed is reached. Due to this property, controller tuning via ES methods is limited to applications with relatively slow dynamics.

5. CONCLUSIONS

Extremum seeking introduces a new approach for optimally tuning PID controllers with respect to a cost function developed by the control engineer. We demonstrate this tuning method on an arbitrary plant model to characterize its performance vis-à-vis other tuning methods. Observations from ES parameter sensitivity analysis suggest that these parameters must be chosen carefully to balance stability against convergence speed. As an extension to work performed by Killingsworth and Krstic [6], we apply ES to adapt PI controller gains for a simple velocity tracking cruise control problem. This methodology investigates an innovative approach for non-model based adaptive control. The results indicate performance that exceeds the ZN method and gain scheduling techniques. Nevertheless, ES is limited to systems with relatively slow dynamics and may introduce undesirable oscillatory responses. Consequently, this tuning scheme is applicable for a very specific class of problems.

REFERENCES

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

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

[3] 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

[4] 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

[5] 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

[6] N.J. Killingsworth, M. Krstic, "PID tuning using extremum seeking," *IEEE Control Systems Magazine*, pp 70-79, Feb. 2006.

[7] A. Stefanopoulou, "Lecture 13: Cruise Control," ME 569, University of Michigan, Ann Arbor, Fall 2007.

[8] 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.