PID Tuning using Extremum Seeking

PID Control

Proportional-Integral-Derivative (PID) Control

- Ubiquitous controller in practice
- Often poorly tuned (Astrom [1995], etc.)
- Proper tuning can yield large benefits

- Consists of the sum of three control terms

  - Proportional term: $u_P(t) = Ke(t)$
  - Integral term: $u_I(t) = \dfrac{K}{T_I}\int^t e(s)ds$
  - Derivative term: $u_D(t) = KT_D \dfrac{de(t)}{dt}$

where,

$e(t) = r(t) - y(t)$
$r(t)$ is the reference signal
$y(t)$ is the measured output
K is the controller gain
T_I is the integral time
T_D is the derivative time

PID Tuning

- Many methods have been developed
  - Some require a plant model or special experiment
    - Ziegler-Nichols (ZN)
    - Kappa-Tau Tuning
    - Internal Model Control method (IMC)
  - Closed loop methods
    - Desirable since process loop need not be disturbed
      - Relay Feedback tuning
      - Unfalsified control tuning
      - Iterative Feedback Tuning (IFT)

Talk Outline

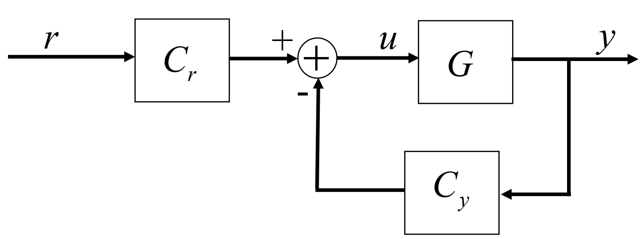
- We will apply discrete version of ES to tune PID parameters
- Goal to optimize step response of a closed loop system
- Method can tune controller for many plants in only a few iterations
- Yields performance at least at good as many popular PID tuning methods

Talk Outline

- PID controller
- ES tuning scheme
- Examples
- Cost function
- Saturation
- Noise
- ES tuning parameters
- Conclusion

PID Controller

- In this work we use a two degree of freedom controller
- And the derivative term only acts on y(t)
  - This avoids large control effort when there is a step change in the reference signal



$$C_r = K\left(1 + \frac{1}{T_I s}\right) \qquad C_y = K\left(1 + \frac{1}{T_I s} + T_D s\right)$$

Cost Function, $J(\theta_k)$

- Used to quantify the controller's performance
- Mapping of PID parameters to tracking performance
- Constructed from the output error of the plant during a step response experiment

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

where, T is the total sample time of each step response experiment

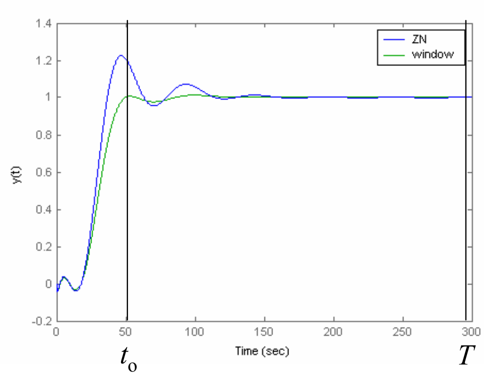
θ is a vector containing the PID parameters:

$$\theta = \left[K, T_I, T_D \right]$$

Cost Function, $J(\theta_k)$

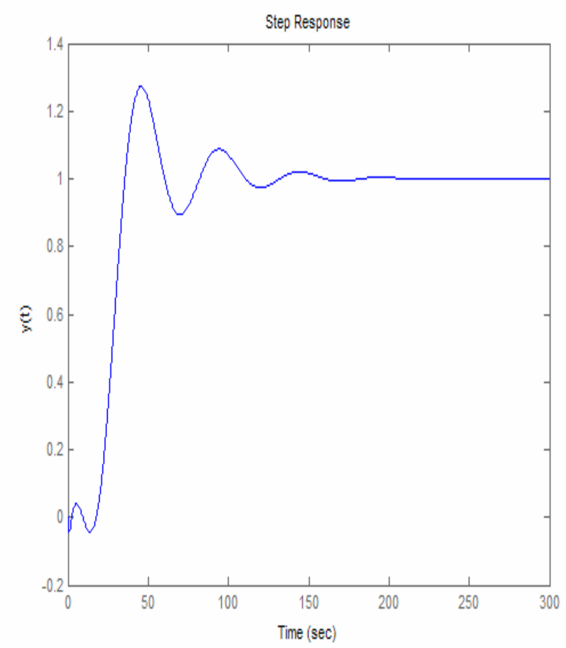
- t_0 time up until which zero weightings are placed on the error.
- Shifts emphasis of PID controller from transient phase of response to that of minimizing tracking error after initial transient portion of response
- By optimality principle cost after t_o is always less than or equal to when zero weightings before t_o are used
- ES used to minimize the cost function

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



Extremum Seeking Tuning Scheme

- Implementation
  1. Run Step response experiment with ZN PID parameters



Extremum Seeking Tuning Scheme

- Implementation
  1. Run Step response experiment with ZN PID parameters
  2. Calculate *J*

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

Extremum Seeking Tuning Scheme

- Implementation
  1. Run Step response experiment with ZN PID parameters
  2. Calculate J
  3. Calculate next set of PID parameters using discrete ES tuning method

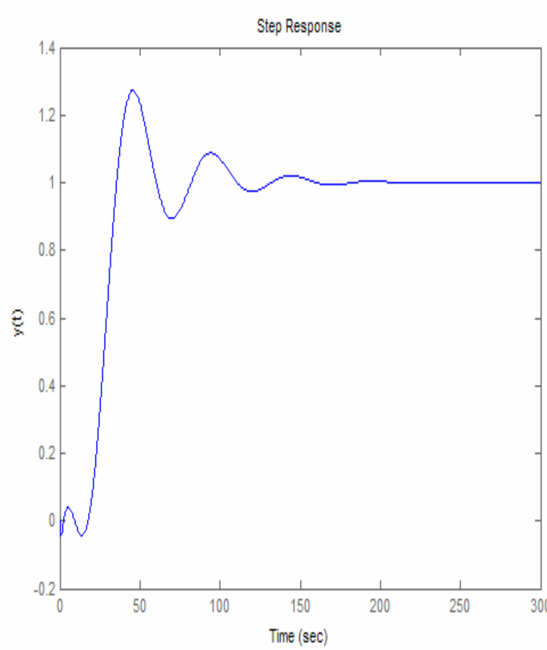
$$\xi(k) = -h\xi(k-1) + J(k-1)$$

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

$$\boxed{\theta_i(k+1) = \hat{\theta}_i(k+1) - \alpha_i \cos(\omega_i (k+1))}$$

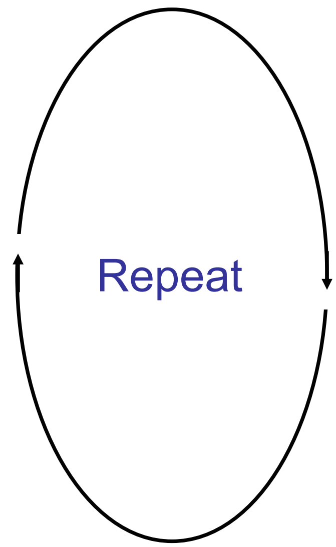
Extremum Seeking Tuning Scheme

- Implementation
  1. Run Step response experiment with ZN PID parameters
  2. Calculate J
  3. Calculate next set of PID parameters using discrete ES tuning method
  4. Run another step response experiment with new PID parameters

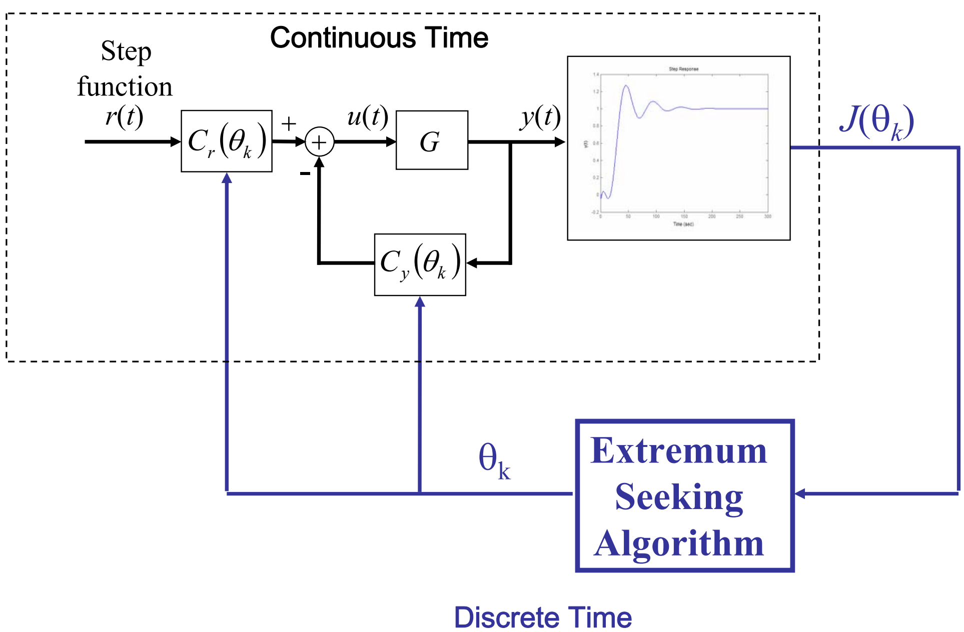


Extremum Seeking Tuning Scheme

- Implementation
  1. Run Step response experiment with ZN PID parameters
  2. Calculate *J*
  3. Calculate next set of PID parameters using discrete ES tuning method
  4. Run another step response experiment with new PID parameters
  5. Repeat 2-4 set number of times or until algorithm converges



Extremum Seeking Tuning Scheme



Examples

- The ES PID tuning algorithm has been used in simulations to find the optimal PID parameters for four plants
  - PID parameters based on Ziegler-Nichols tuning rules have been used as initial conditions in ES tuning algorithm
  - These results have been compared to three other popular PID tuning methods
    - The Ziegler-Nichols (ZN) tuning rules
    - The internal model control (IMC) method
    - The iterative feedback tuning (IFT) method (Gevers, '94, '98)

Example Plants

Four systems have been used to test the ES PID tuning method

1. Time delay

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

2. Large time delay

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

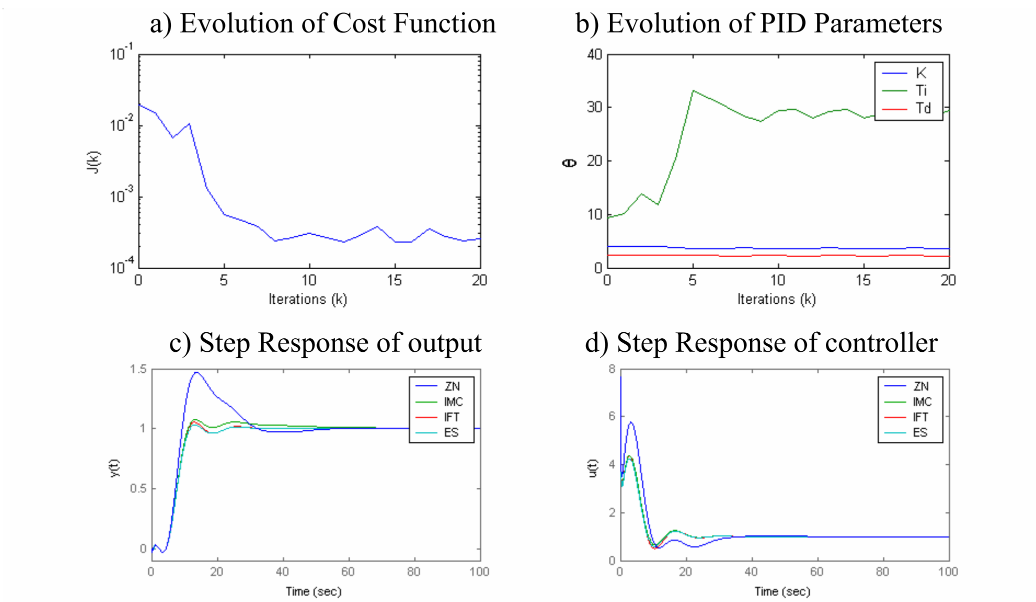
3. Single pole of order eight

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

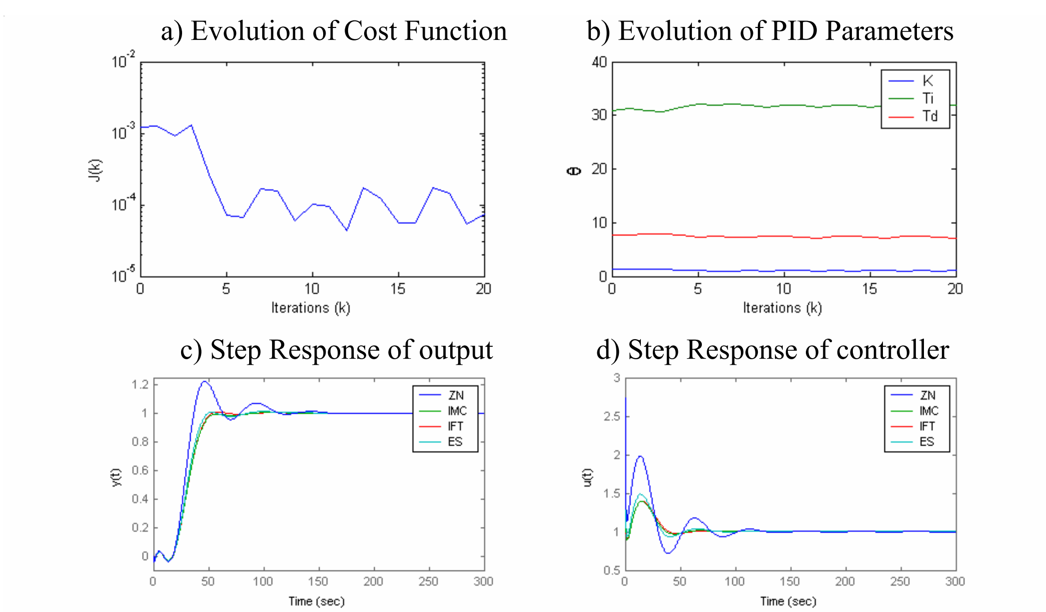
4. Unstable zero

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

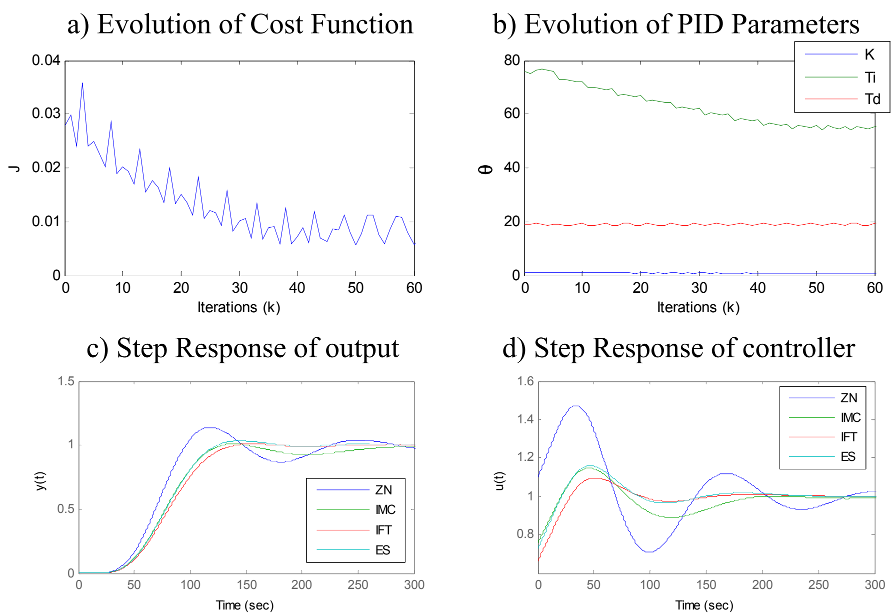
$$G_1(s) = \frac{1}{1+20s} e^{-5s}$$



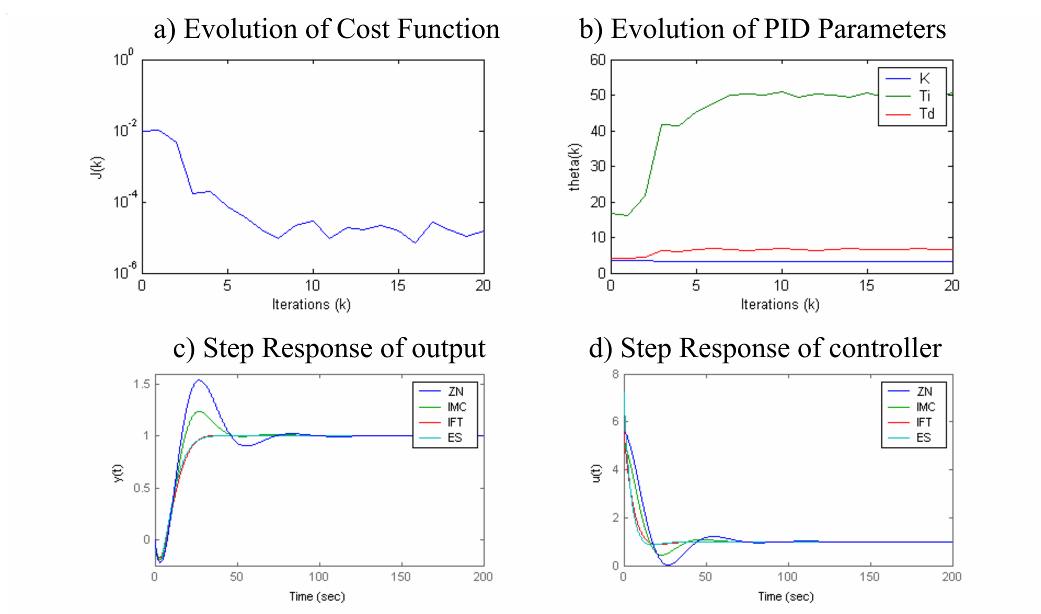
$$G_2(s) = \frac{1}{1+20s} e^{-20s}$$



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



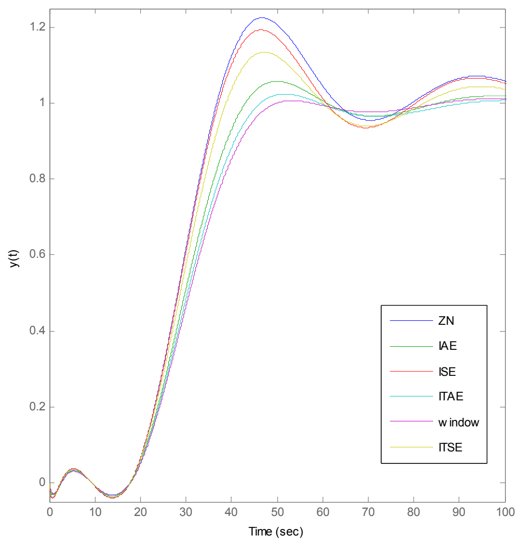
$$G_4(s) = \frac{1-5s}{(1+10s)(1+20s)}$$



Cost Function Comparison

Step response of output

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



The following cost functions were minimized using ES:

$$ISE = \frac{1}{T}\int_0^T e(\theta_k)^2\,dt$$

$$ITSE = \frac{1}{T}\int_0^T te(\theta_k)^2\,dt$$

$$IAE = \frac{1}{T}\int_0^T |e(\theta_k)|\,dt$$

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

$$Window = \frac{1}{T-t_0}\int_{t_0}^T e(\theta_k)^2\,dt$$

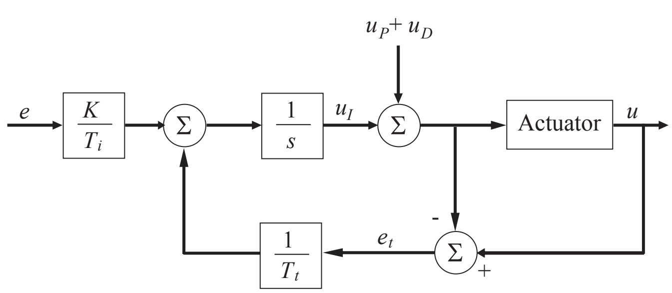
Control Saturation

- Saturation of 1.6 applied to control signal for plant *G*1

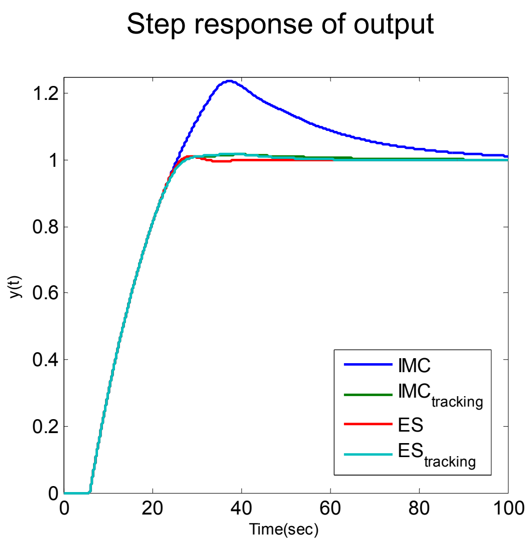
$$G_1(s) = \frac{1}{1+20s} e^{-5s}$$

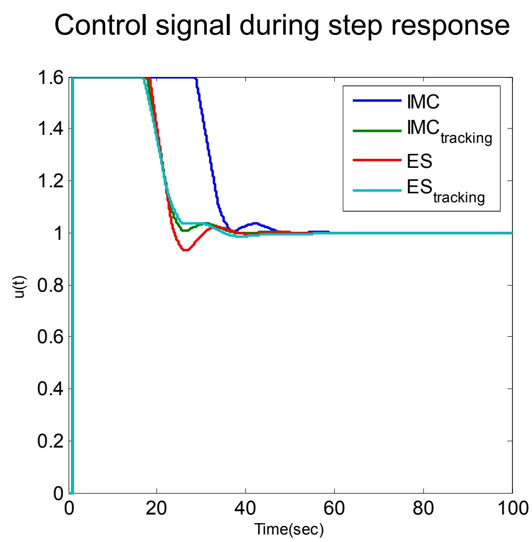
- ES and IMC compared with and without the addition of an anti windup scheme

Tracking anti-windup scheme



Control Saturation





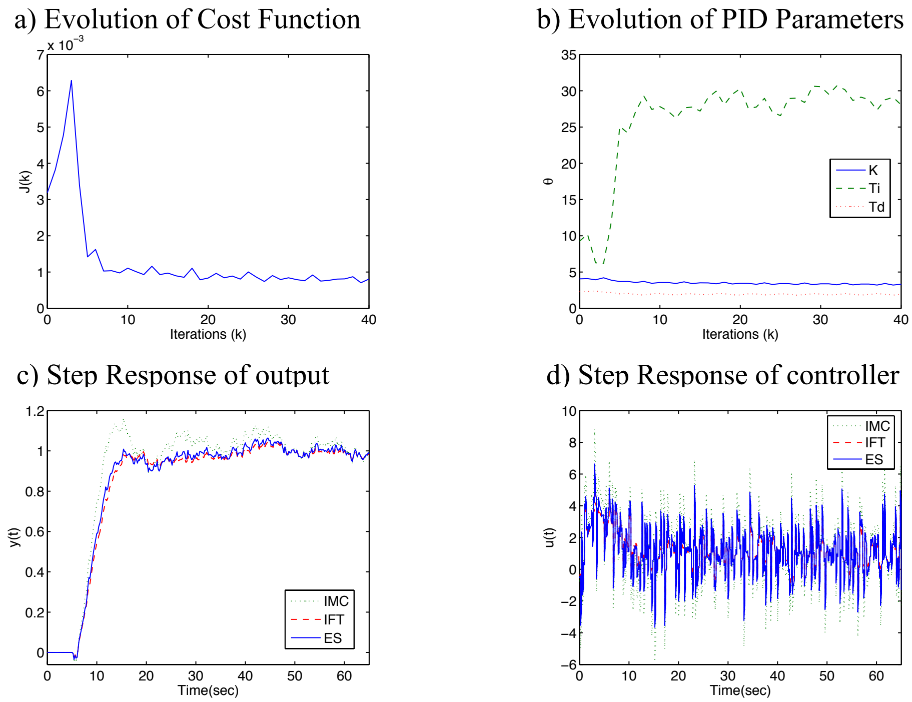
Effects of Noise

- Band-limited white noise has been added to output
- Power spectral density = 0.0025
- Correlation time = 0.2
- Independent noise signal for each iteration

- Simulations on plant G_1

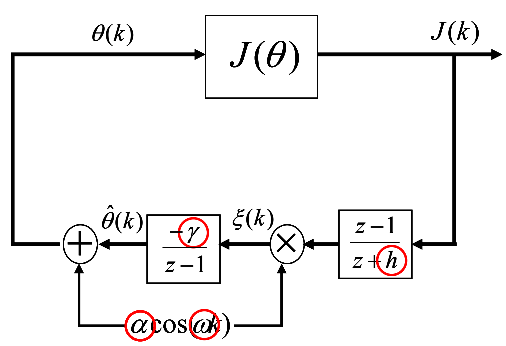
$$G_1(s) = \frac{1}{1+20s}e^{-5s}$$

Effects of Noise



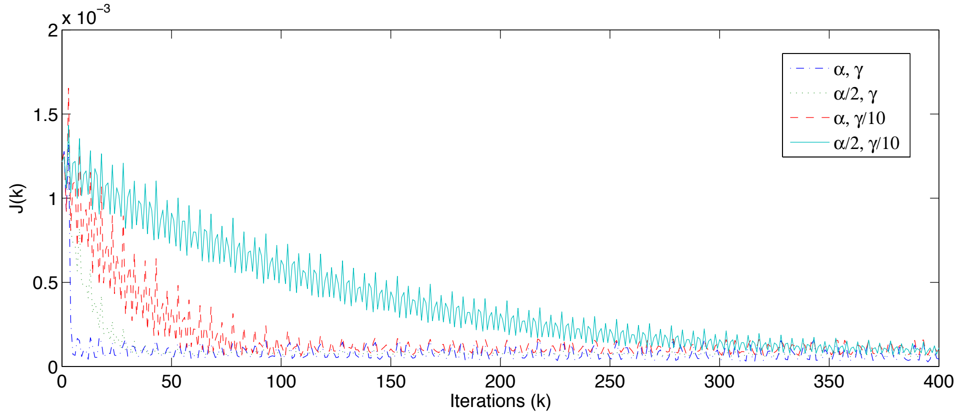
Selecting Parameters of ES Scheme

- Must select
  - α, perturbation step size
  - γ, adaptation gain
  - ω, perturbation frequency
  - h, high-pass filter cut-off frequency



- Note we have more parameters to pick than we started out with
- However, ES tuning is less sensitive to parameters than PID controller

Selecting Parameters of ES Scheme



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

$$\alpha = [0.06, 0.30, 0.20]^T$$

$$\gamma = [2500, 2500, 2500]^T$$

$$\omega_i = 0.8^i \pi$$

$$h = 0.5$$

ES Tuning Parameters	K	T_i	T_d
$\alpha\ ,\ \gamma$	1.01	31.5	7.16
$\alpha/2\ ,\ \gamma$	1.00	31.1	7.6
$\alpha\ ,\ \gamma/10$	1.01	31.3	7.54
$\alpha/2\ ,\ \gamma/10$	1.01	31.0	7.65

Conclusions

- ES provides an effective and efficient method to tune PID controllers that minimize a cost function which characterizes some desired behavior.
- Es is able to deal with nonlinearities and noise.
  - has advantages over model based methods in real world applications which exhibit these behaviors.
- Initial PID parameters are needed to start the algorithm.
- The cost function can be designed to emphasize specific performance attributes.