

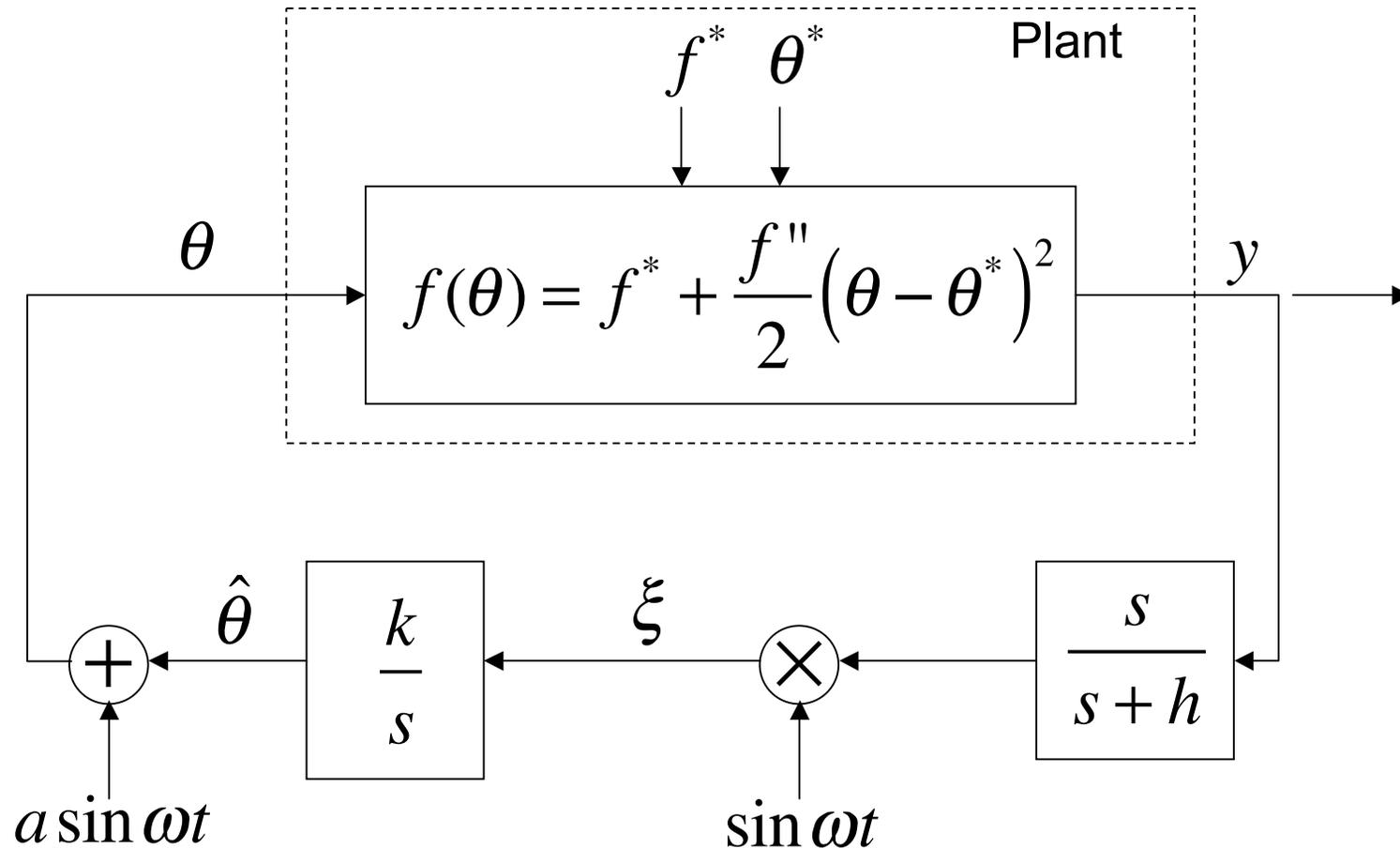
Extremum Seeking Control for Real-Time Optimization

Miroslav Krstic

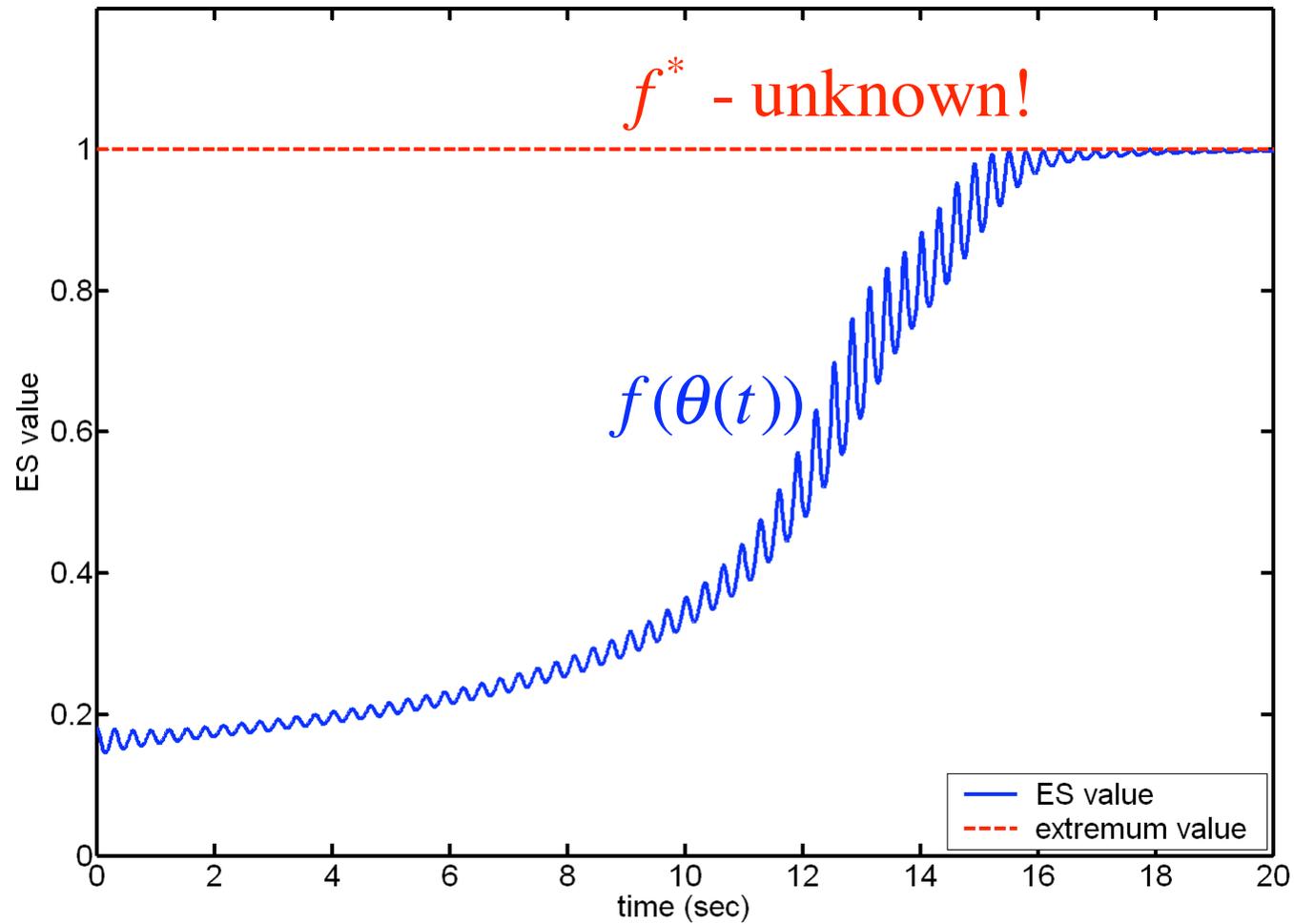
UC San Diego

**IEEE Advanced Process Control Applications for Industry Workshop
Vancouver, 2007**

Example of Single-Parameter Maximum Seeking



Example of Single-Parameter Maximum Seeking



Topics - Theory

- History
- **Single** parameter ES, how it works, and stability analysis by averaging
- **Multi-parameter** ES
- **ES** in discrete time
- **ES** with plant dynamics and compensators for performance improvement
- Internal model principle for tracking parameter changes
- **Slope** seeking
- **Limit cycle** minimization via ES

Topics - Applications

- PID tuning
- Internal combustion (HCCI) engine fuel consumption minimization
- Compressor instabilities in jet engines
- Combustion instabilities
- Formation flight
- Fusion reflected RF power
- Thermoacoustic coolers
- Beam matching in particle accelerators
- Flow separation control in diffusers
- Autonomous vehicles without position sensing

History

- **Leblanc (1922)** - electric railways
- **Early Russian literature (1940's)** - many papers
- **Drapper and Li (1951)** - application to IC engine spark timing tuning
- **Tsien (1954)** - a chapter in his book on Engineering Cybernetics
- **Feldbaum (1959)** - book *Computers in Automatic Control Systems*
- **Blackman (1962 chap. in book by Westcott)** - nice intuitive presentation of ES
- **Wilde (1964)** - a book
- **Chinaev (1969)** - a handbook on self-tuning systems
- Papers by [Morosanov], [Ostrovskii], [Pervozvanskii], [Kazakevich], [Frey, Deem, and Altpeter], [Jacobs and Shering], [Korovin and Utkin] - late 50s - early 70's
- **Meerkov (1967, 1968)** - papers with averaging analysis
- **Sternby (1980)** - survey
- **Astrom and Wittenmark (1995 book)** - rates ES as one of the most promising areas for adaptive control

Recent Developments

- **Krstic and Wang (2000, *Automatica*)** - stability proof for single-parameter general dynamic nonlinear plants
- **Choi, Ariyur, Wang, Krstic** - discrete-time, limit cycle minimization, IMC for parameter tracking, etc.
- **Rotea; Walsh; Ariyur** - multi-parameter ES
- **Ariyur** - slope seeking
- **Tan, Nesic, Mareels (2005)** - semi-global stability of ES
- Other approaches: **Guay, Dochain, Titica**, and coworkers; **Zak, Ozguner**, and coworkers; **Banavar, Chichka, Speyer; Popovic, Teel**; etc.
- **Applications** not presented in this workshop:
 - Electromechanical valve actuator (Peterson and Stephanopoulou)
 - Artificial heart (Antaki and Paden)
 - Exercise machine (Zhang and Dawson)
 - Shape optimization for magnetic head in hard disk drives (UCSD)
 - Shape optimization of airfoils and automotive vehicles (King, UT Berlin)

ES Book

An up-close look at the theory behind and application of extremum seeking

Originally developed as a method of adaptive control for hard-to-model systems, extremum seeking solves some of the same problems as today's neural network techniques, but in a more rigorous and practical way. Following the resurgence in popularity of extremum-seeking control in aerospace and automotive engineering, *Real-Time Optimization by Extremum-Seeking Control* presents the theoretical foundations and selected applications of this method of real-time optimization.

Written by authorities in the field and pioneers in adaptive nonlinear control systems, this book presents both significant theoretic value and important practical potential. Filled with in-depth insight and expert advice, *Real-Time Optimization by Extremum-Seeking Control*:

- Develops optimization theory from the points of dynamic feedback and adaptation
- Builds a solid bridge between the classical optimization theory and modern feedback and adaptation techniques
- Provides a collection of useful tools for problems in this complex area
- Presents numerous applications of this powerful methodology
- Demonstrates the immense potential of this methodology for future theory development and applications

Real-Time Optimization by Extremum-Seeking Control is an important resource for both students and professionals in all areas of engineering—electrical, mechanical, aerospace, chemical, biomedical—and is also a valuable reference for practicing control engineers.

KARTIK B. ARIYUR is a research scientist at Honeywell Aerospace Electronic Systems in Minneapolis, Minnesota.

MIROSLAV KRSTIĆ is Professor of Mechanical and Aerospace Engineering at the University of California at San Diego.

Subscribe to our free Electrical Engineering eNewsletter at www.wiley.com/enewsletters

Visit www.wiley.com/electrical

WILEY-INTERSCIENCE
wiley.com



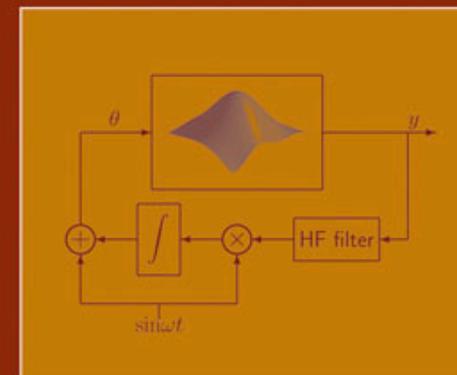
ARIYUR
KRSTIĆ

Real-Time Optimization by Extremum-Seeking Control



WILEY

Real-Time Optimization by Extremum-Seeking Control

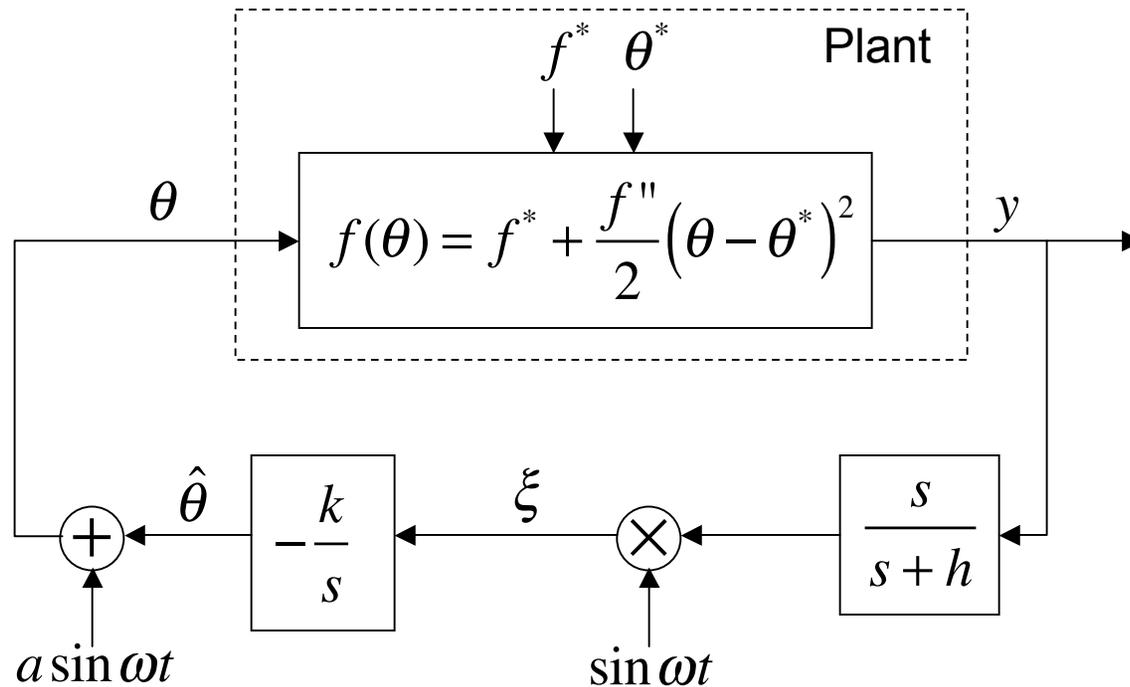


KARTIK B. ARIYUR
MIROSLAV KRSTIĆ

Tutorial Topics **Covered** in the Book

- Introduction, history, single-parameter stability analysis
- Plant dynamics, compensators, and IMC for tracking parameter changes
- Limit cycle minimization via ES
- Multi-parameter ES
- ES in discrete time
- Slope seeking
- Compressor instabilities in jet engines
- Combustion instabilities
- Formation flight
- Anti-skid braking
- Bioreactor
- Thermoacoustic coolers
- Internal combustion engines
- Flow separation control in diffusers
- Beam matching in particle accelerators
- PID tuning
- Autonomous vehicles without position sensing

Basic Extremum Seeking - Static Map



y = output to be minimized

f^* = minimum of the map

f'' = second derivative (positive - $f(\theta)$ has a min.)

θ^* = unknown parameter

$\hat{\theta}$ = estimate of θ^*

k = adaptation gain (positive) of the integrator $\frac{1}{s}$

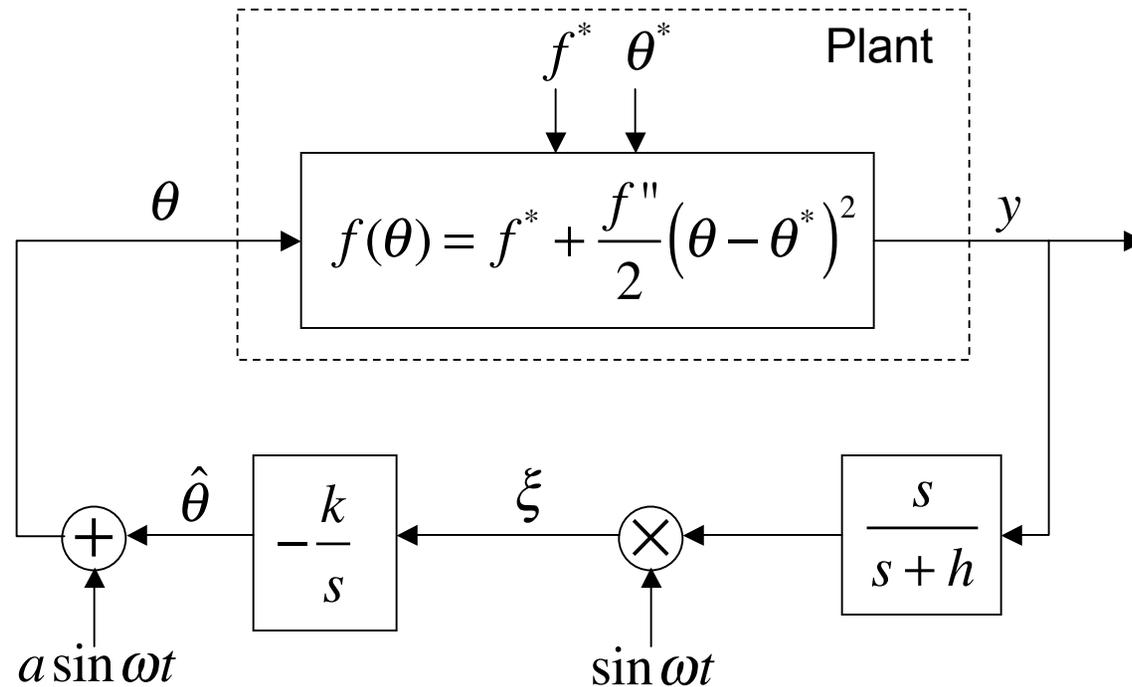
a = amplitude of the probing signal

ω = frequency of the probing signal

h = cut-off frequency of the "washout filter" $\frac{s}{s+h}$

+/ \times = modulation/demodulation

How Does It Work?



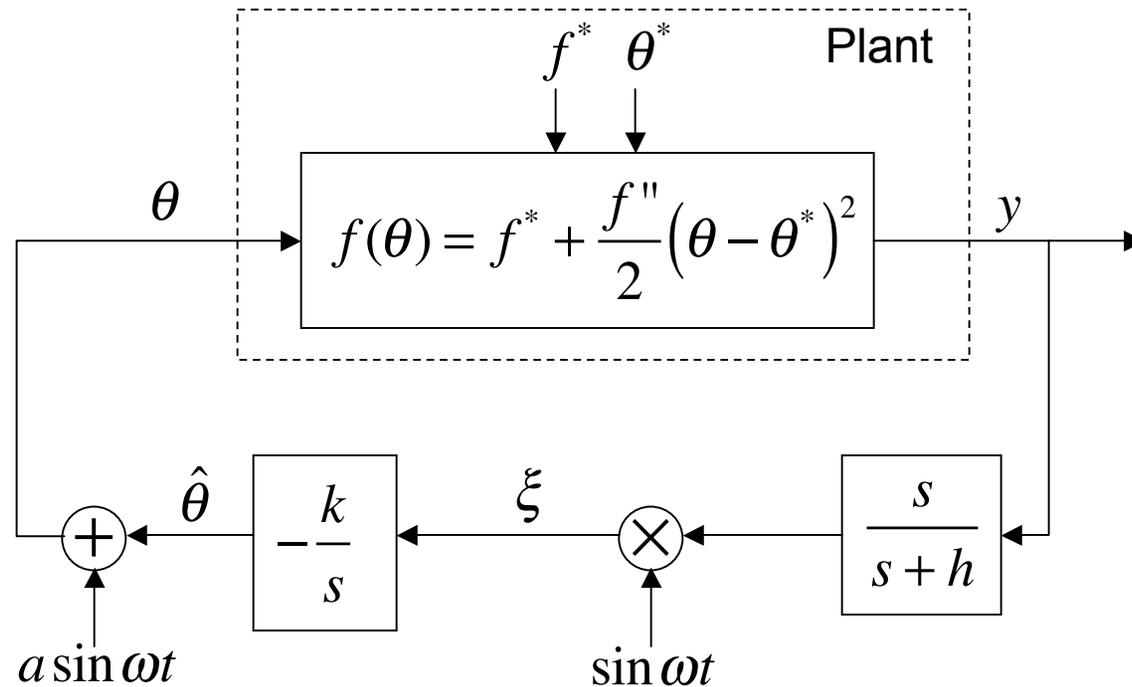
Estimation error: $\tilde{\theta} = \theta^* - \hat{\theta}$

$$y = f^* + \frac{a^2 f''}{4} + \frac{f''}{2} \tilde{\theta}^2 - a f'' \tilde{\theta} \sin \omega t + \frac{a^2 f''}{4} \cos 2\omega t$$

Loc. Analysis - neglect quadratic terms:

$$y \approx f^* + \frac{a^2 f''}{4} - a f'' \tilde{\theta} \sin \omega t + \frac{a^2 f''}{4} \cos 2\omega t$$

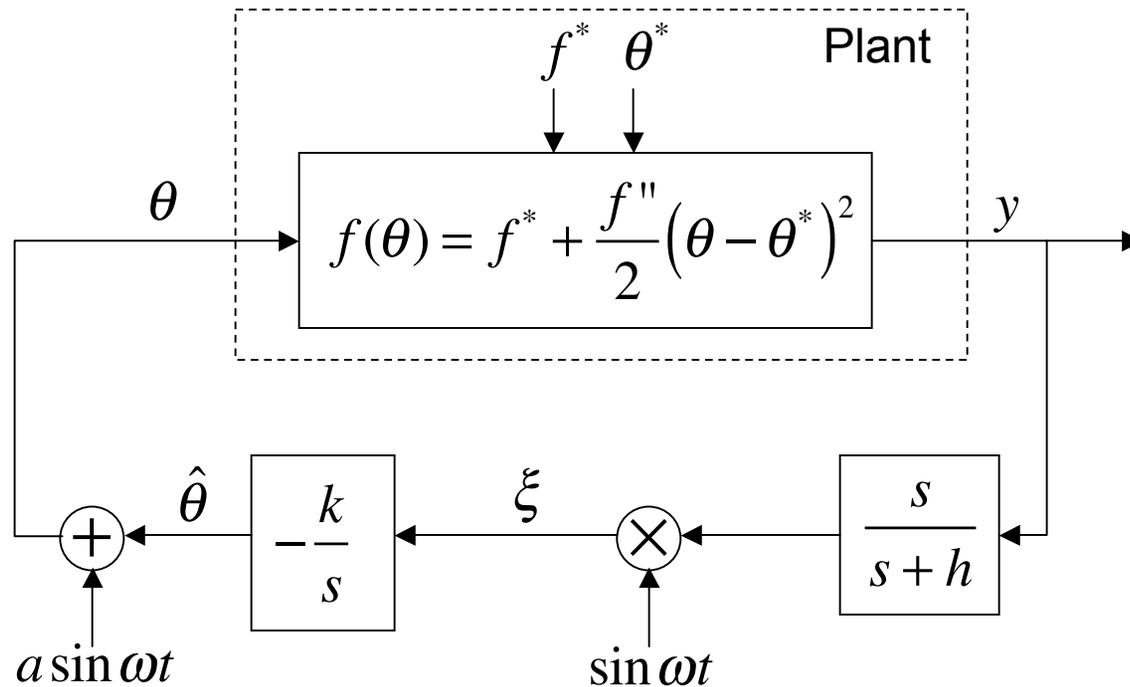
How Does It Work?



$$y \approx f^* + \frac{a^2 f''}{4} - af'' \tilde{\theta} \sin \omega t + \frac{a^2 f''}{4} \cos 2\omega t$$

$$\frac{s}{s+h}[y] \approx -af'' \tilde{\theta} \sin \omega t + \frac{a^2 f''}{4} \cos 2\omega t$$

How Does It Work?

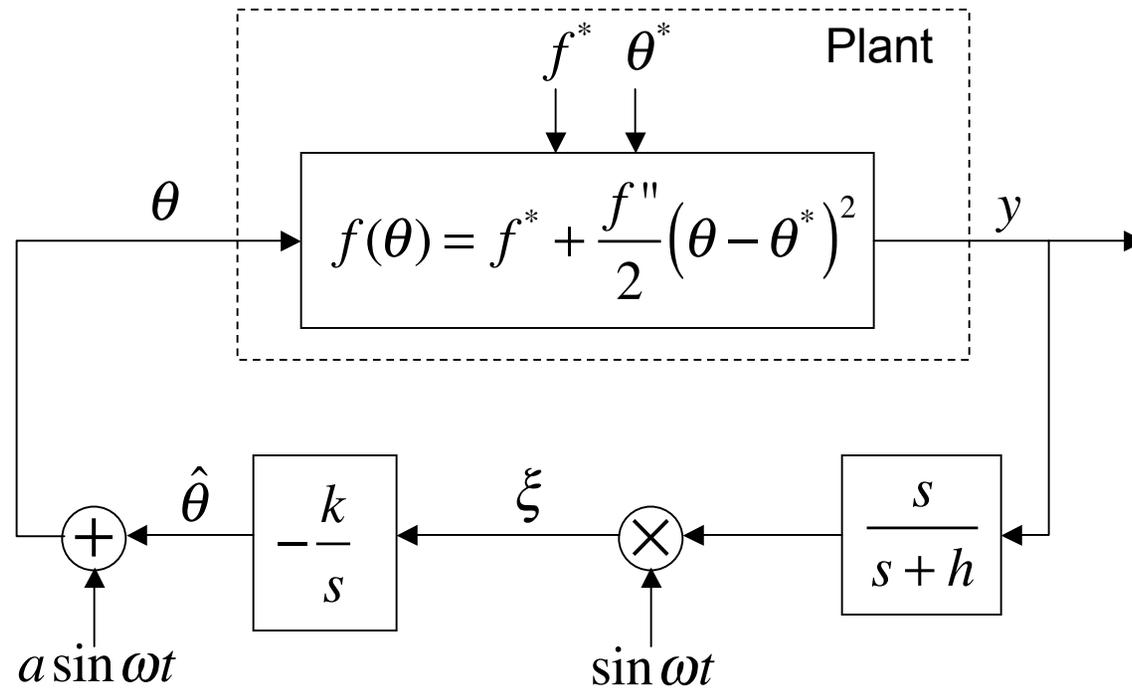


Demodulation:

$$\xi = \sin \omega t \frac{s}{s+h} [y] \approx -af'' \tilde{\theta} \sin^2 \omega t + \frac{a^2 f''}{4} \cos 2\omega t \sin \omega t$$

$$\xi \approx -\frac{a^2 f''}{4} \tilde{\theta} + \frac{a^2 f''}{4} \tilde{\theta} \cos 2\omega t + \frac{a^2 f''}{8} (\sin \omega t - \sin 3\omega t)$$

How Does It Work?



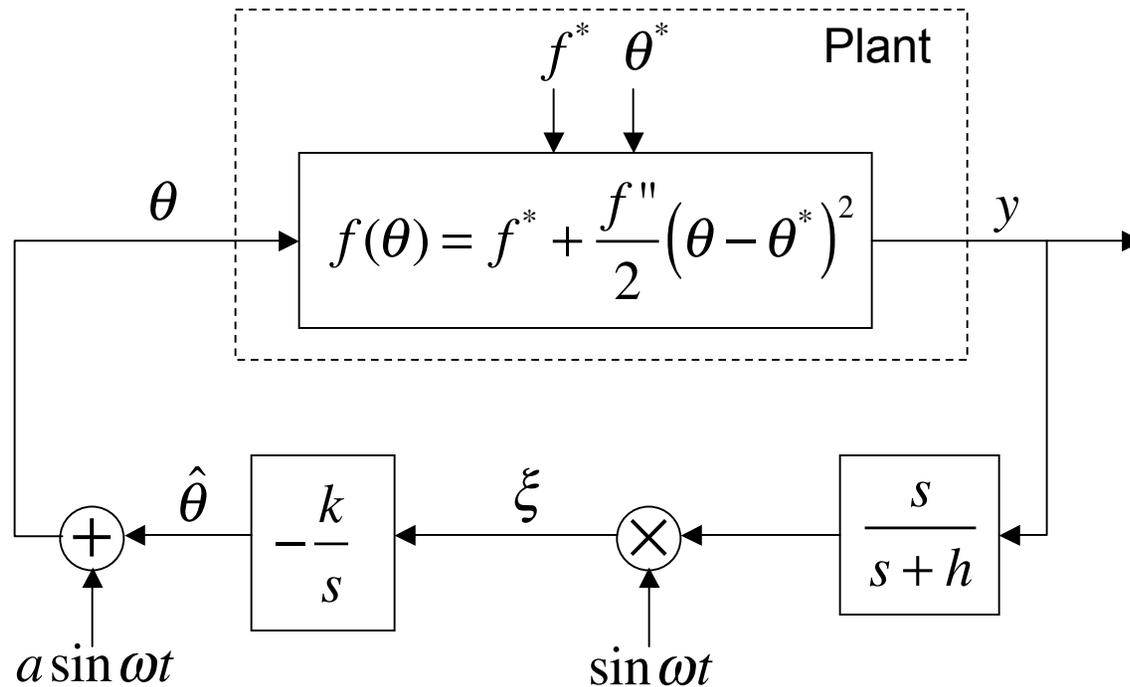
Since

$$\tilde{\theta} = \theta^* - \hat{\theta}$$

then

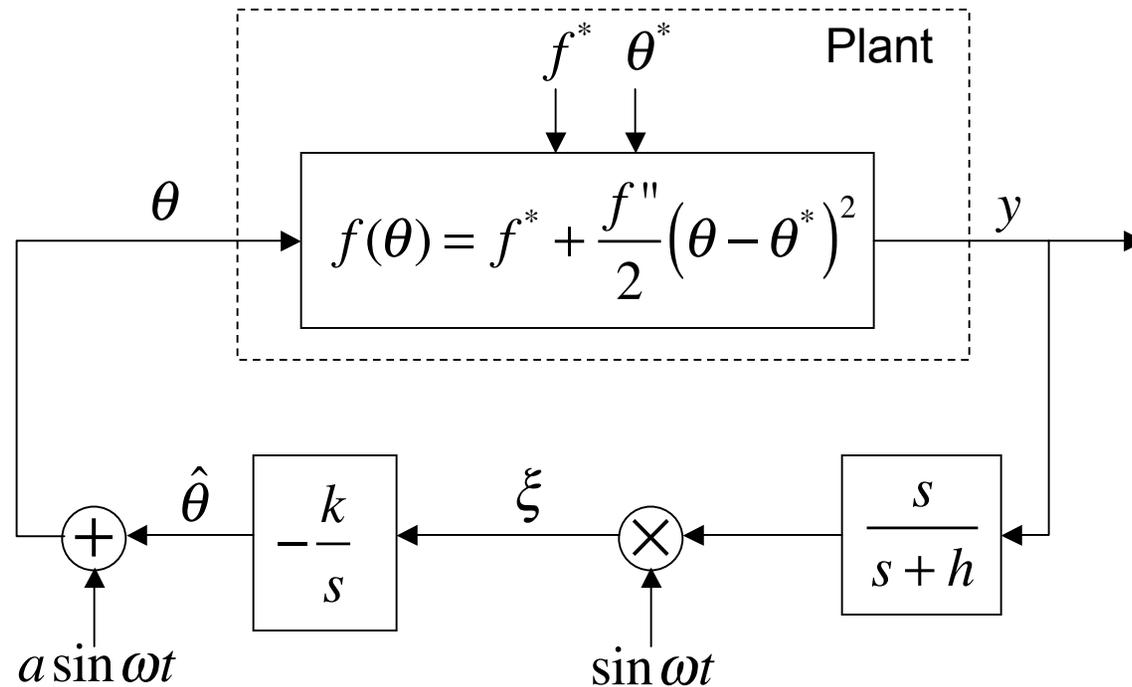
$$\dot{\tilde{\theta}} = -\dot{\hat{\theta}}$$

How Does It Work?



$$\tilde{\theta} \approx \frac{k}{s} \left[-\frac{a^2 f''}{4} \tilde{\theta} + \underbrace{\frac{a^2 f''}{4} \tilde{\theta} \cos 2\omega t + \frac{a^2 f''}{8} (\sin \omega t - \sin 3\omega t)}_{\text{high frequency terms - attenuated by integrator}} \right]$$

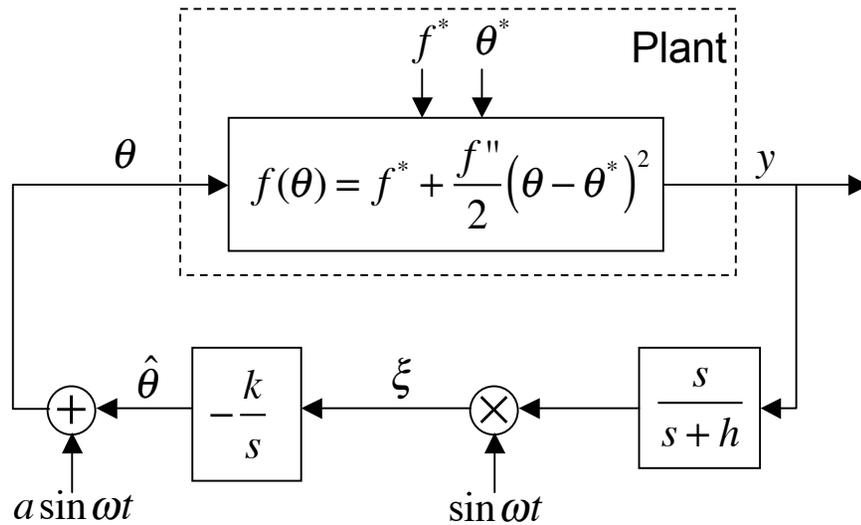
How Does It Work?



$$\dot{\tilde{\theta}} \approx -\frac{ka^2 f''}{4} \tilde{\theta}$$

Stable because $k, a, f'' > 0$

Stability Proof by Averaging



$$\tilde{\theta} = \theta^* - \hat{\theta}$$

$$e = f^* - \frac{h}{s+h}[y]$$

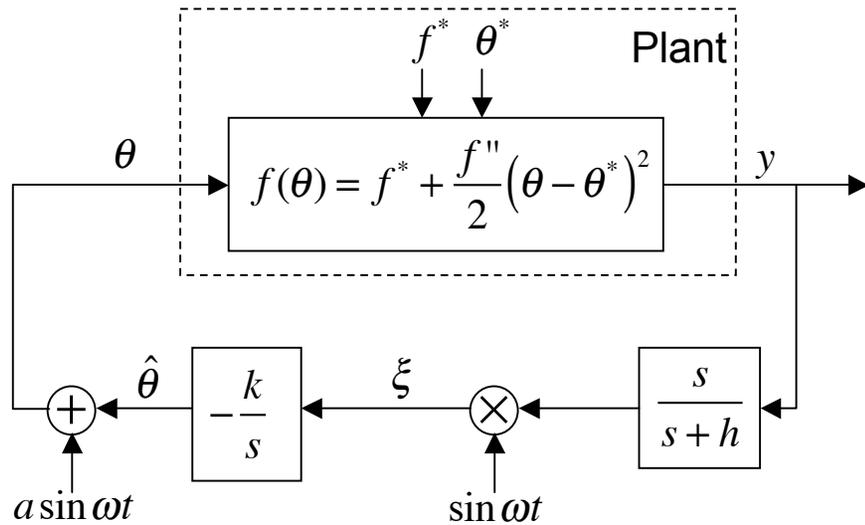
$$\tau = \omega t$$

Full nonlinear time-varying model:

$$\frac{d}{d\tau} \tilde{\theta} = \frac{k}{\omega} \left(\frac{f''}{2} (\tilde{\theta} - a \sin \tau)^2 - e \right) \sin \tau$$

$$\frac{d}{d\tau} e = \frac{h}{\omega} \left(-e - \frac{f''}{2} (\tilde{\theta} - a \sin \tau)^2 \right)$$

Stability Proof by Averaging



$$\tilde{\theta} = \theta^* - \hat{\theta}$$

$$e = f^* - \frac{h}{s+h}[y]$$

$$\tau = \omega t$$

Average system:

$$\frac{d}{d\tau} \tilde{\theta}_{\text{av}} = -\frac{kaf''}{2\omega} \tilde{\theta}_{\text{av}}$$

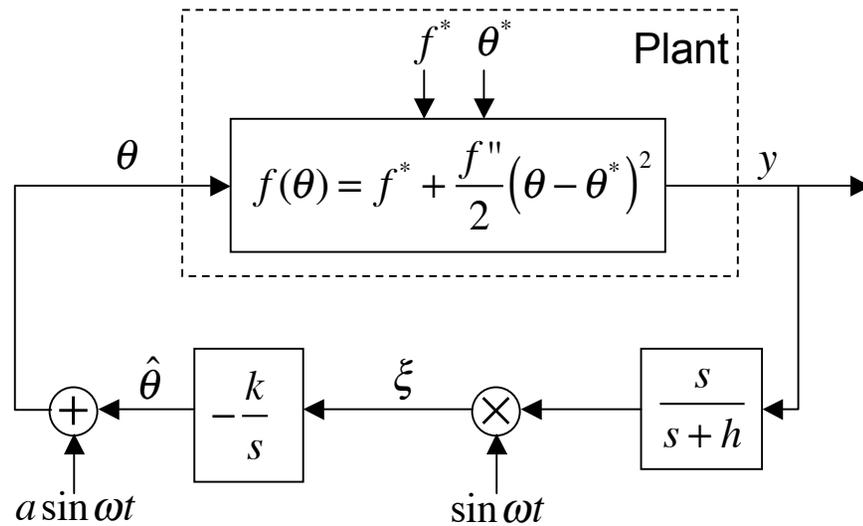
$$\frac{d}{d\tau} e_{\text{av}} = \frac{h}{\omega} \left(-e_{\text{av}} - \frac{f''}{2} \left(\tilde{\theta}_{\text{av}}^2 + \frac{a^2}{2} \right) \right)$$

Average equilibrium:

$$\tilde{\theta}_{\text{av}} = 0$$

$$e_{\text{av}} = -\frac{a^2 f''}{4}$$

Stability Proof by Averaging



$$\tilde{\theta} = \theta^* - \hat{\theta}$$

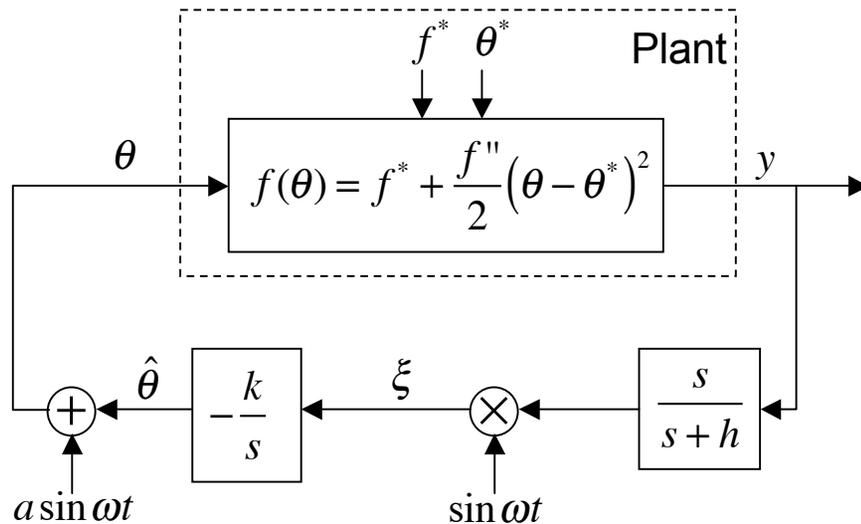
$$e = f^* - \frac{h}{s+h}[y]$$

$$\tau = \omega t$$

Jacobian of the average system:

$$J_{av} = \begin{bmatrix} -\frac{kaf''}{2\omega} & 0 \\ 0 & -\frac{h}{\omega} \end{bmatrix}$$

Stability Proof by Averaging



$$\tilde{\theta} = \theta^* - \hat{\theta}$$

$$e = f^* - \frac{h}{s+h}[y]$$

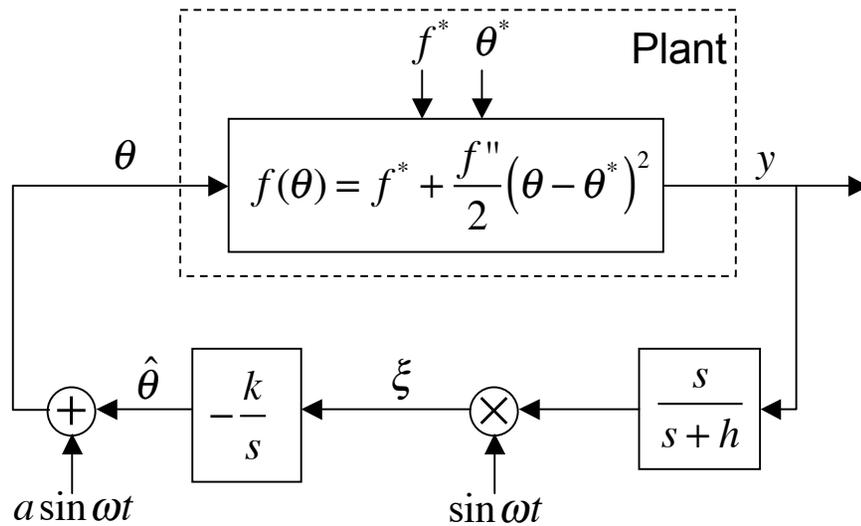
$$\tau = \omega t$$

Theorem. For sufficiently large ω there exists a unique exponentially stable periodic solution of period $2\pi/\omega$ and it satisfies

$$\left| \tilde{\theta}_{2\pi/\omega}(t) \right| + \left| e_{2\pi/\omega}(t) - \frac{a^2 f''}{4} \right| \leq O\left(\frac{1}{\omega}\right), \quad \forall t \geq 0$$

Speed of convergence proportional to $1/\omega, a^2, k, f''$

Stability Proof by Averaging



$$\tilde{\theta} = \theta^* - \hat{\theta}$$

$$e = f^* - \frac{h}{s+h}[y]$$

$$\tau = \omega t$$

Output performance:

$$y - f^* \rightarrow f'' O\left(\frac{1}{\omega^2} + a^2\right)$$

PID Tuning Using ES

Based on contributions by: Nick Killingsworth

Background & Motivation

Proportional-Integral-Derivative (PID) Control

- Consists of the sum of three control terms

- Proportional term: $u_P(t) = Ke(t)$

- Integral term: $u_I(t) = \frac{K}{T_I} \int e(s) ds$

- Derivative term: $u_D(t) = KT_D \frac{de(t)}{dt}$

$$e(t) = r(t) - y(t)$$

$r(t)$ reference signal

$y(t)$ measured output

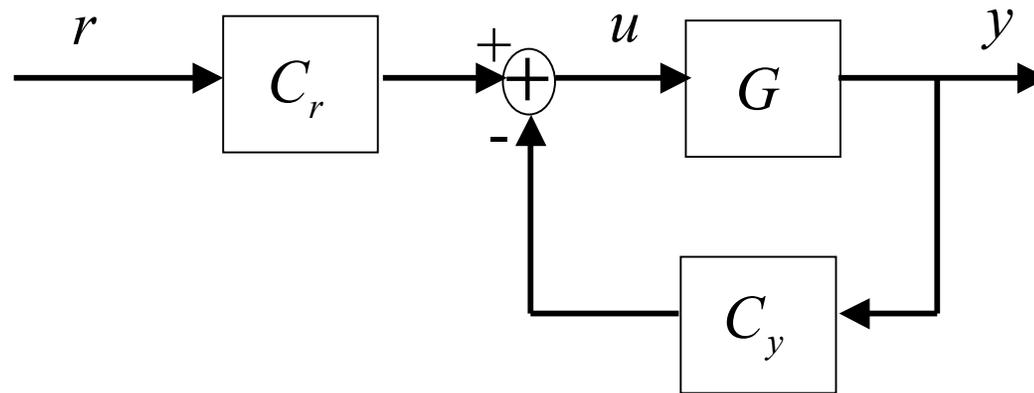
- Often poorly tuned (Astrom [1995], etc.)

Background – PID

We use a two degree of freedom controller

The derivative term only acts on $y(t)$

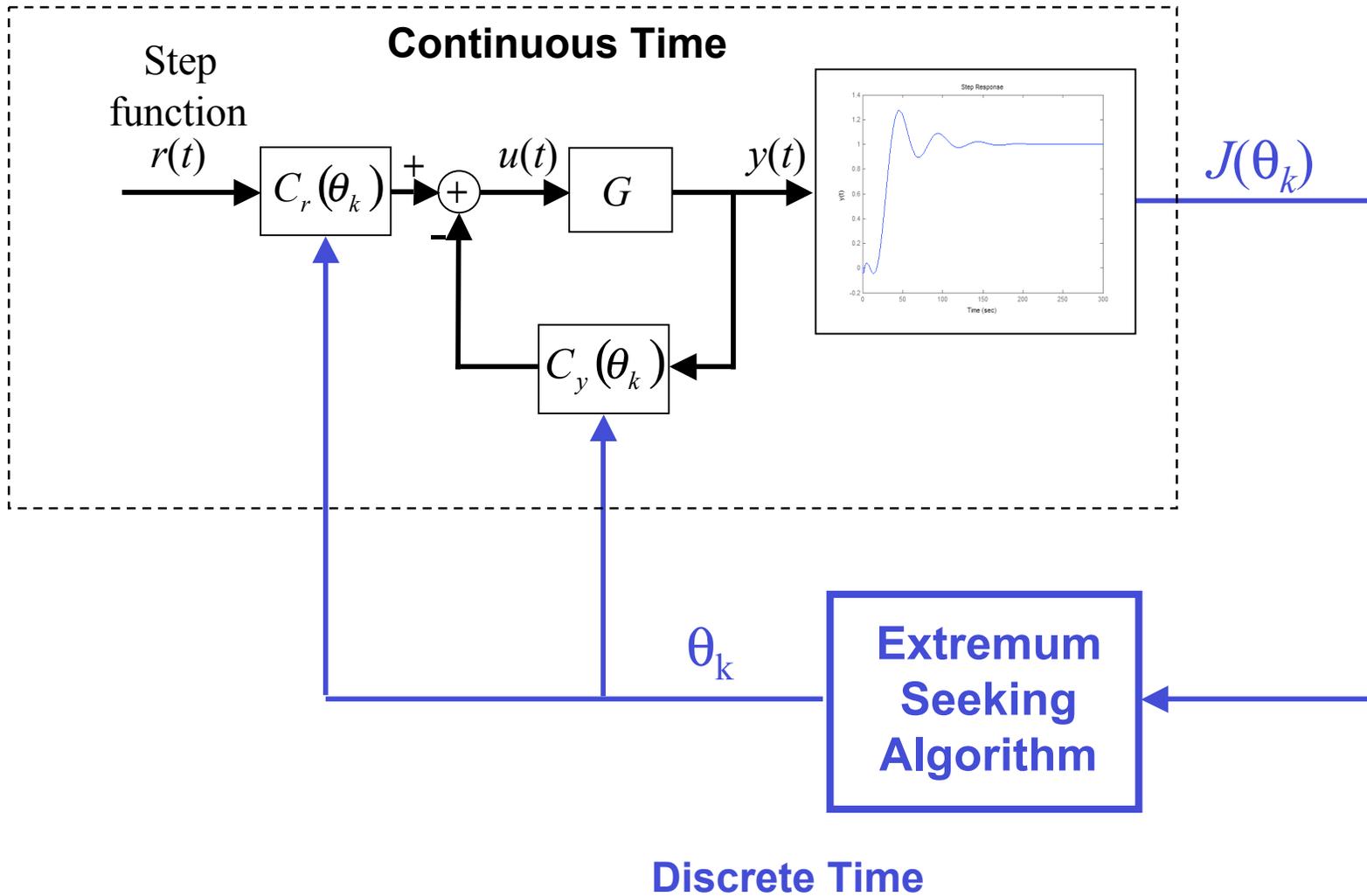
- This avoids large control effort when there is a step change in the reference signal



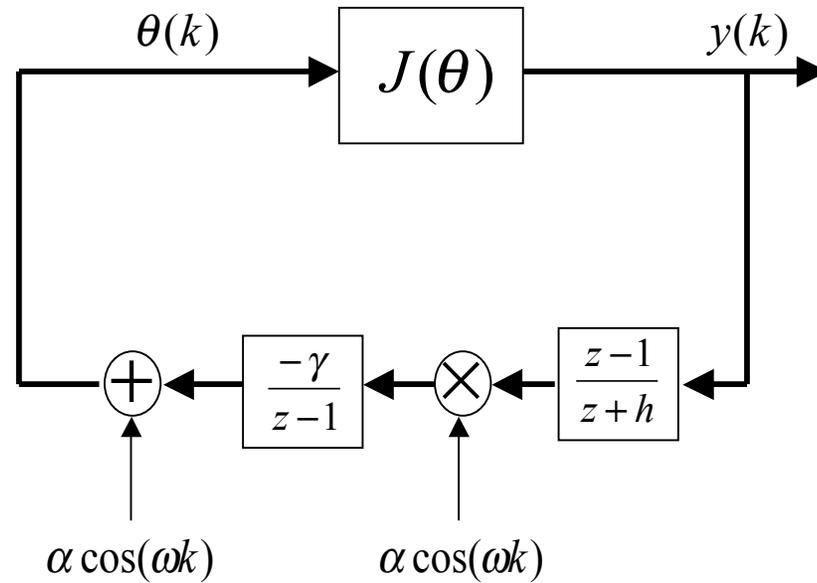
$$C_r = K \left(1 + \frac{1}{T_I s} \right)$$

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

Tuning Scheme



Extremum Seeking



Simple - three lines of code

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

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

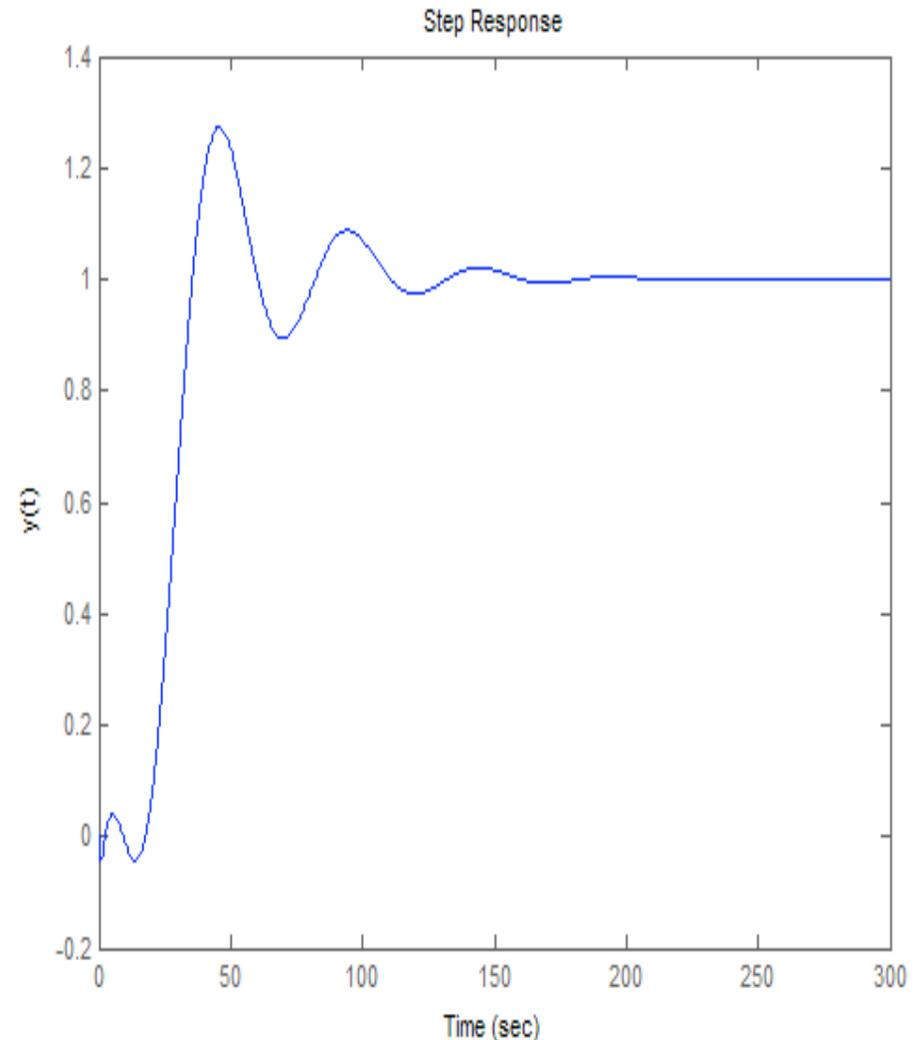
$$\hat{\theta}_i(k+1) = \hat{\theta}_i(k) - \gamma_i \alpha_i \cos(\omega_i k) [J(k) - (1+h)\xi(k)]$$

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

Extremum Seeking Tuning Scheme

Implementation

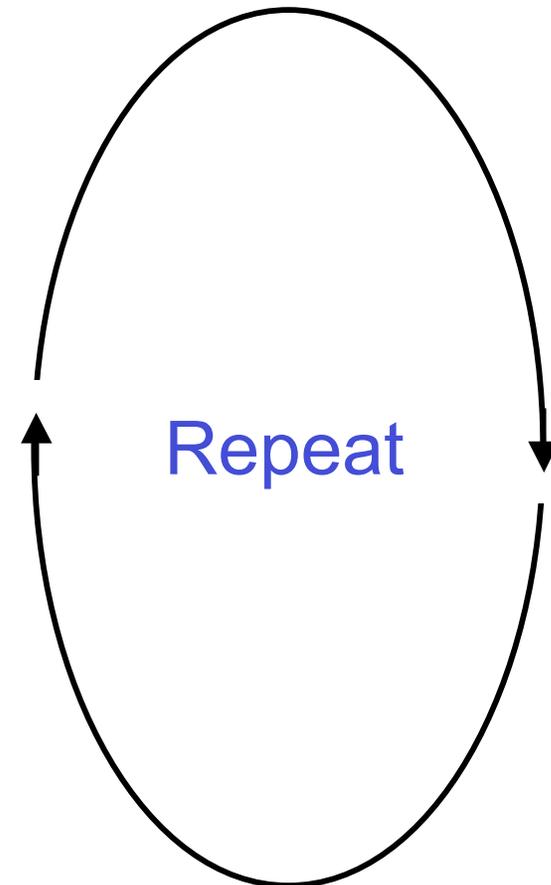
1. Run Step response experiment with ZN PID parameters
2. Calculate J
3. Calculate next set of PID parameters using discrete ES tuning method
4. Run another step response experiment with new PID parameters



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 J falls below a set value



Implementation – Cost Function

Cost Function $J(\theta_k)$

Used to quantify the controller's performance

Constructed from the output error of the plant and the control effort during a step response experiment

Has discrete values at the completion of each 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: $\theta = [K, T_I, T_D]$

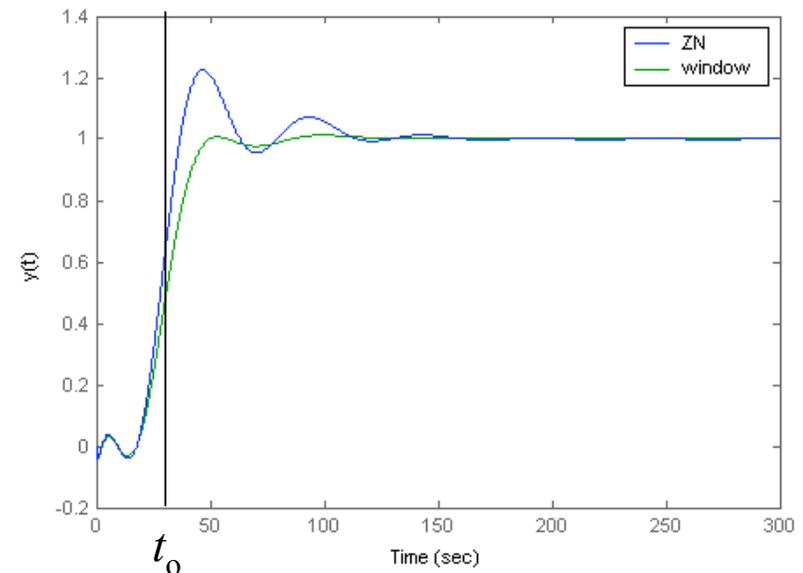
Implementation – Cost Function

Cost Function $J(\theta_k)$

t_0 is the time up until which zero weightings are placed on the error.

This shifts the emphasis of the PID controller from the transient phase of the response to that of minimizing the tracking error after the initial transient portion of the response

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



Example Plants

Four systems with dynamics typical of some industrial plants have been used to test the ES PID tuning method

1. Time delay

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

3. Single pole of order eight

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

2. Large time delay

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

4. Unstable zero

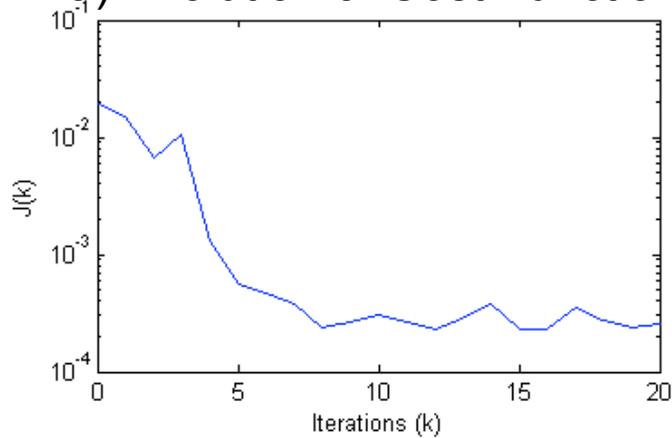
$$G_4(s) = \frac{1-5s}{(1+10s)(1+20s)}$$

Results

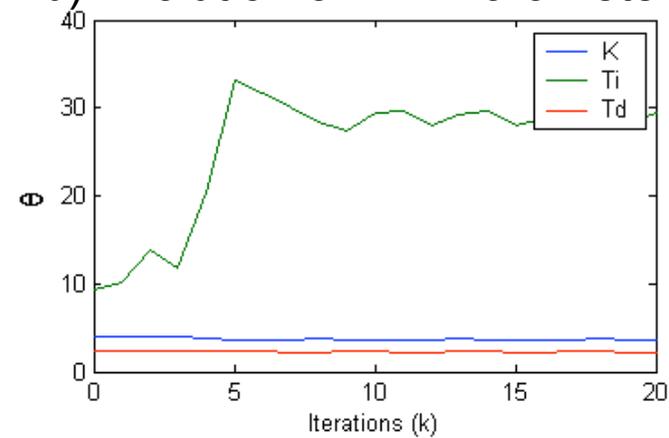
- Ziegler-Nichols values used as initial conditions in the ES tuning algorithm
- Results compared to three other popular PID tuning methods:
 - Ziegler-Nichols (ZN)
 - Internal model control (IMC)
 - Iterative feedback tuning (IFT, Gevers, '94, '98)

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

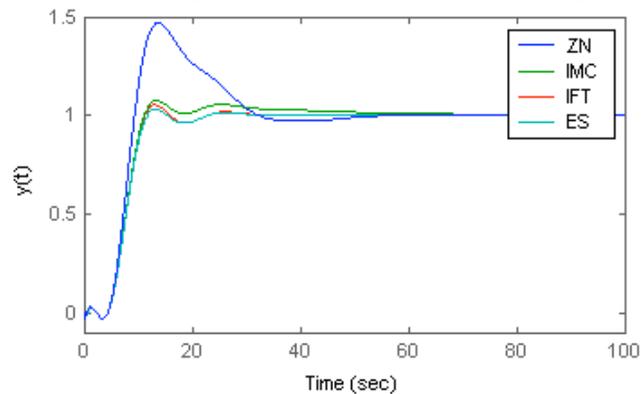
a) Evolution of Cost Function



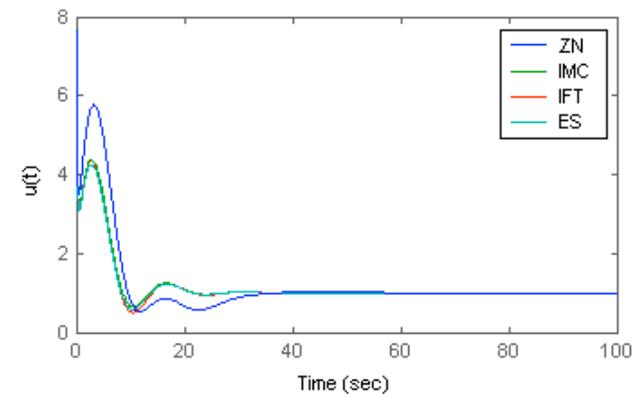
b) Evolution of PID Parameters



c) Step Response of output

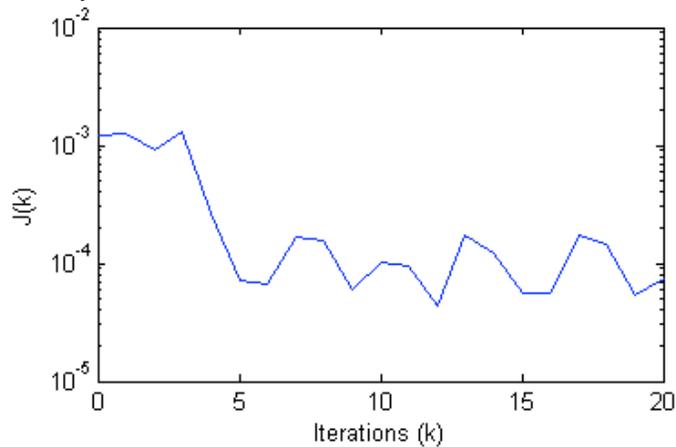


d) Step Response of controller

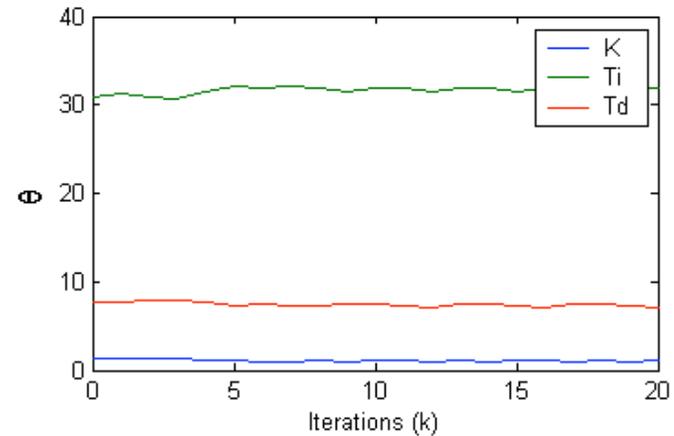


Results - $G_2(s) = \frac{1}{1+20s} e^{-20s}$

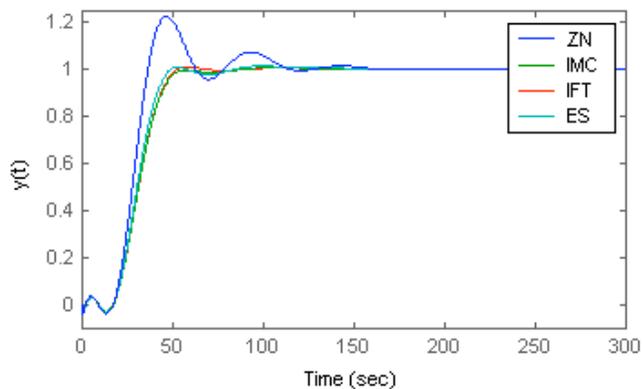
a) Evolution of Cost Function



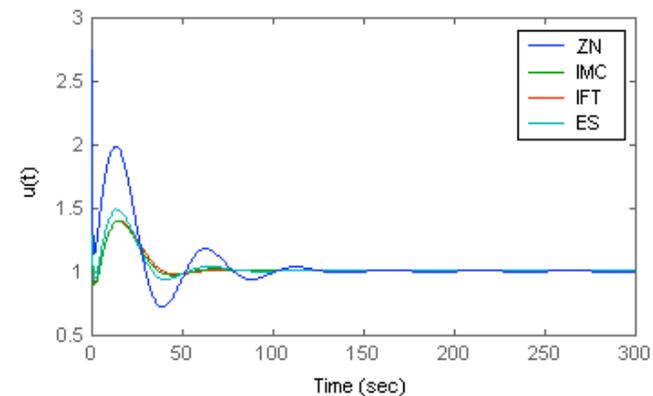
b) Evolution of PID Parameters



c) Step Response of output

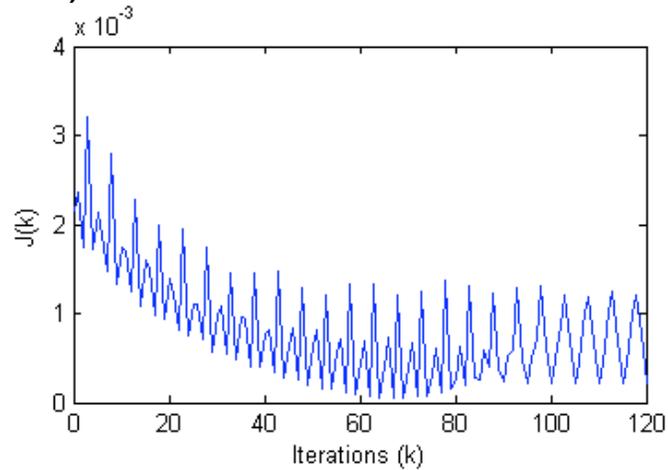


d) Step Response of controller

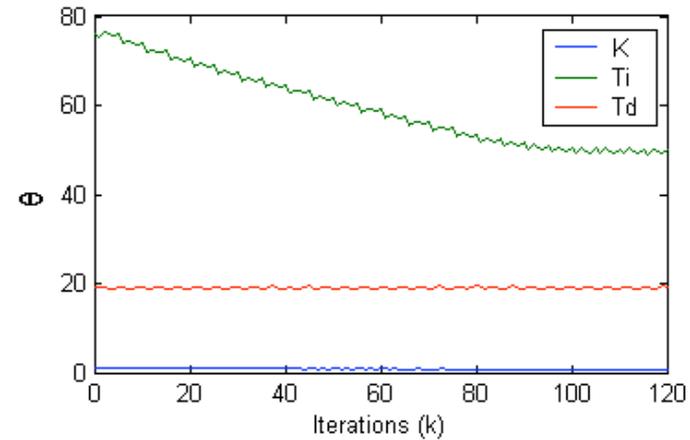


Results - $G_3(s) = \frac{1}{(1+10s)^8}$

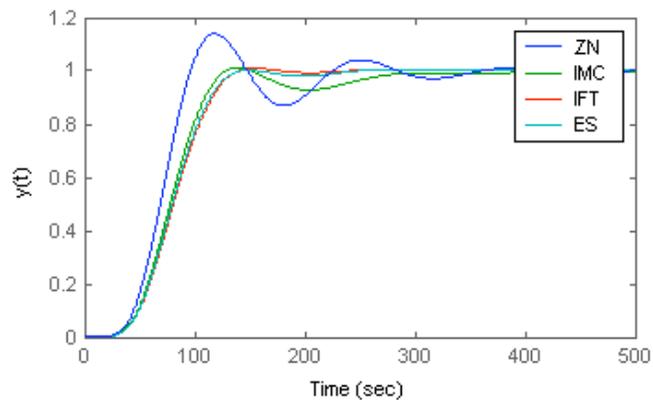
a) Evolution of Cost Function



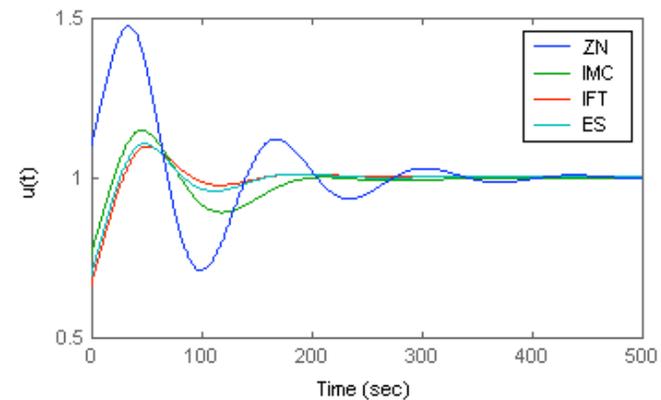
b) Evolution of PID Parameters



c) Step Response of output

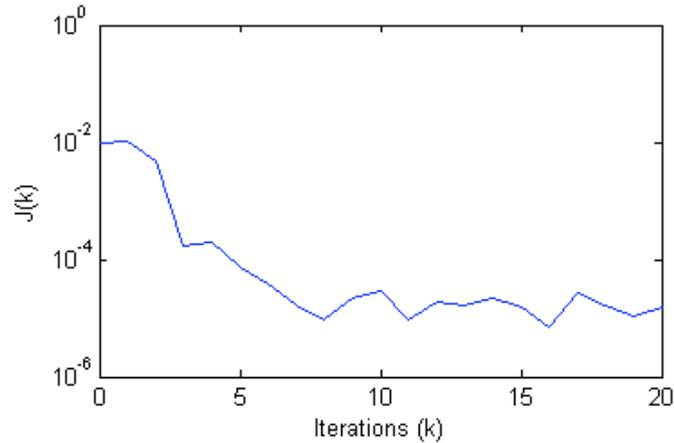


d) Step Response of controller

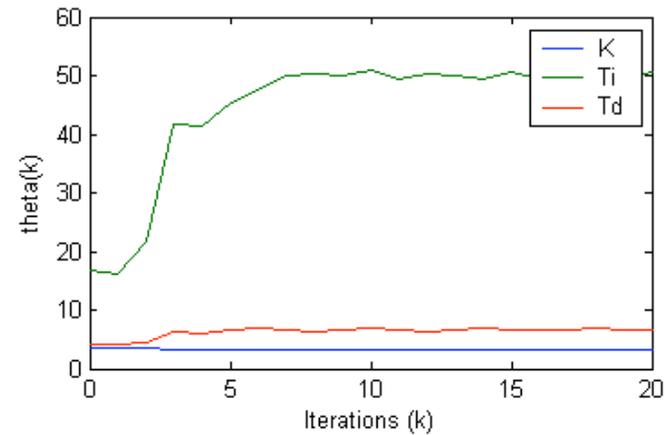


Results - $G_4(s) = \frac{1-5s}{(1+10s)(1+20s)}$

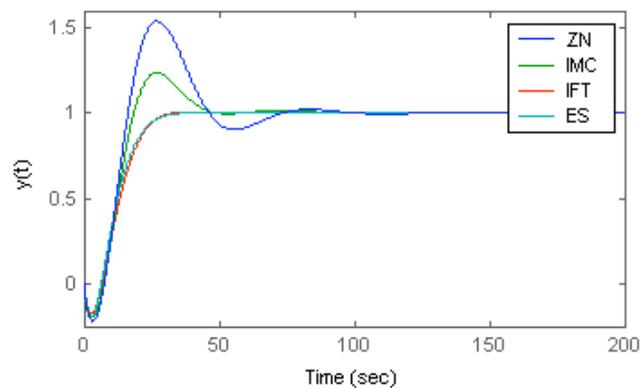
a) Evolution of Cost Function



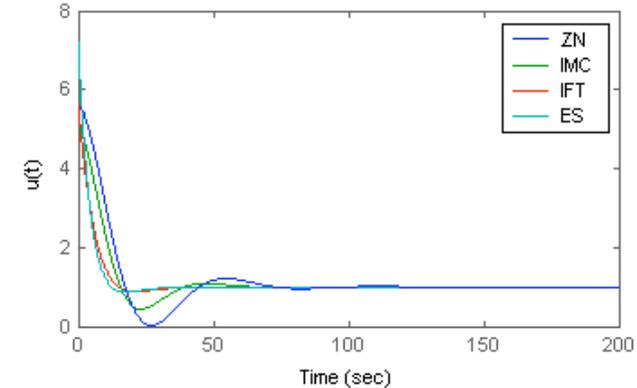
b) Evolution of PID Parameters



c) Step Response of output

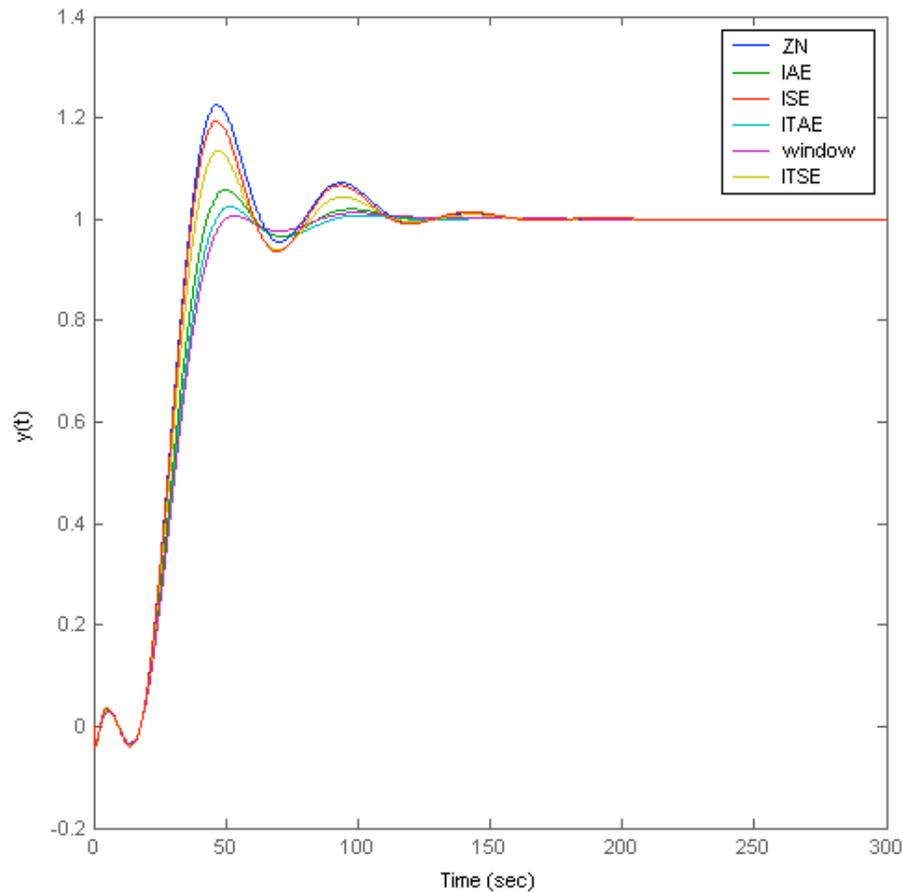


d) Step Response of controller



Results – Cost Function Comparison

Step Response of output

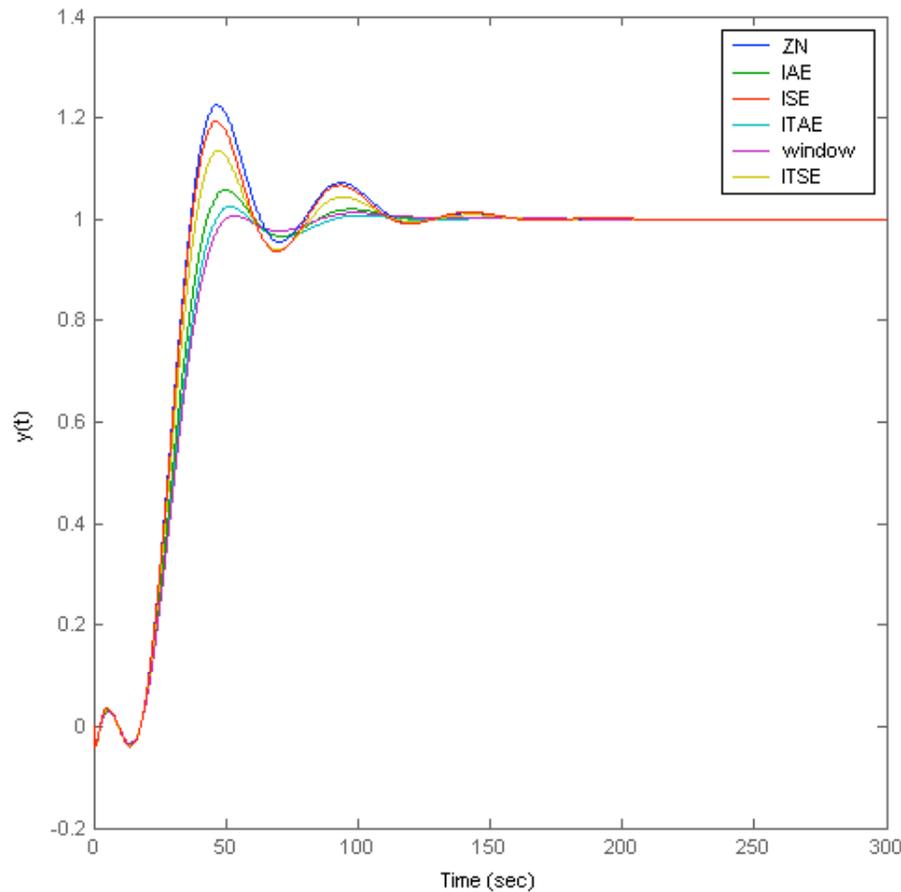


The following cost functions were minimized using ES:

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

Results – Cost Function Comparison

Step Response of output



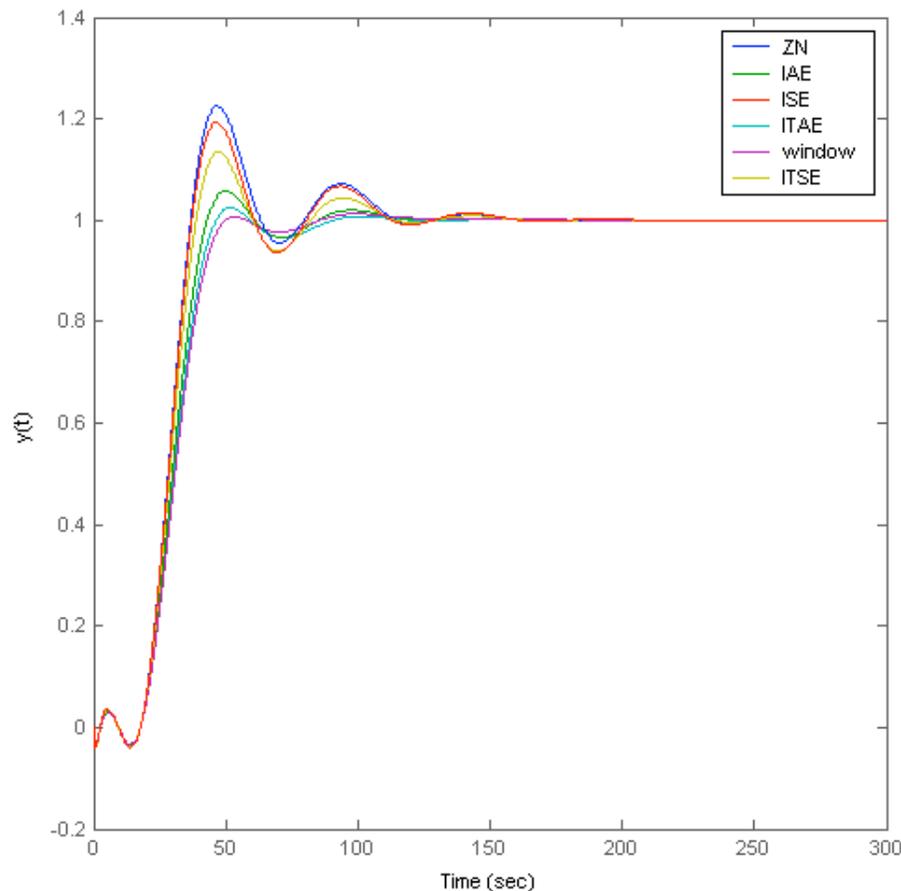
The following cost functions were minimized using ES:

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

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

Results – Cost Function Comparison

Step Response of output



The following cost functions were minimized using ES:

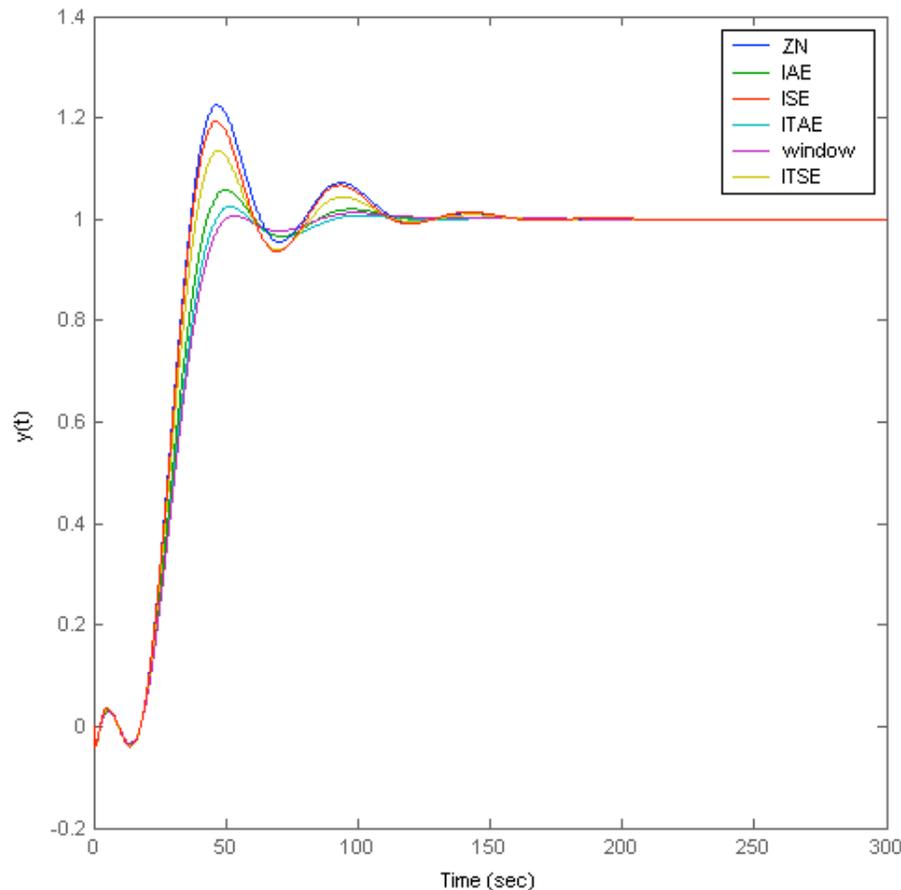
$$ISE = \frac{1}{T} \int_0^T e(\theta_k)^2 dt$$

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

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

Results – Cost Function Comparison

Step Response of output



The following cost functions were minimized using ES:

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

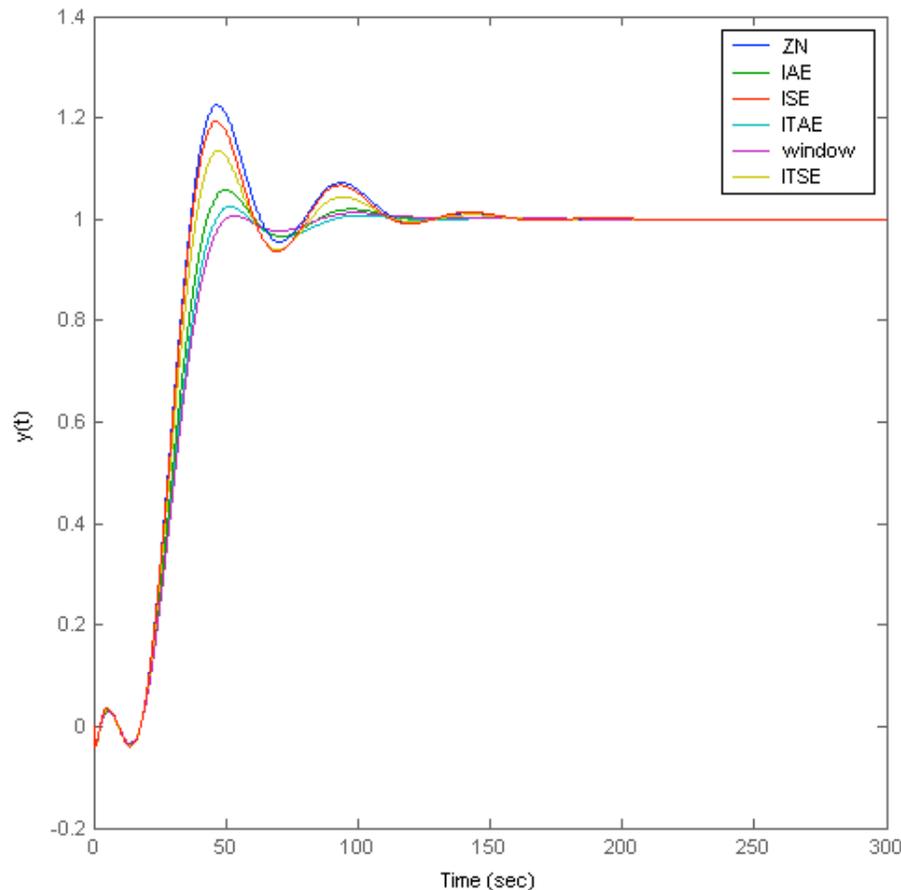
$$ITSE = \frac{1}{T} \int_0^T te(\theta_k)^2 dt$$

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

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

Results – Cost Function Comparison

Step Response of output



The following cost functions were minimized using ES:

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

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

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

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

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

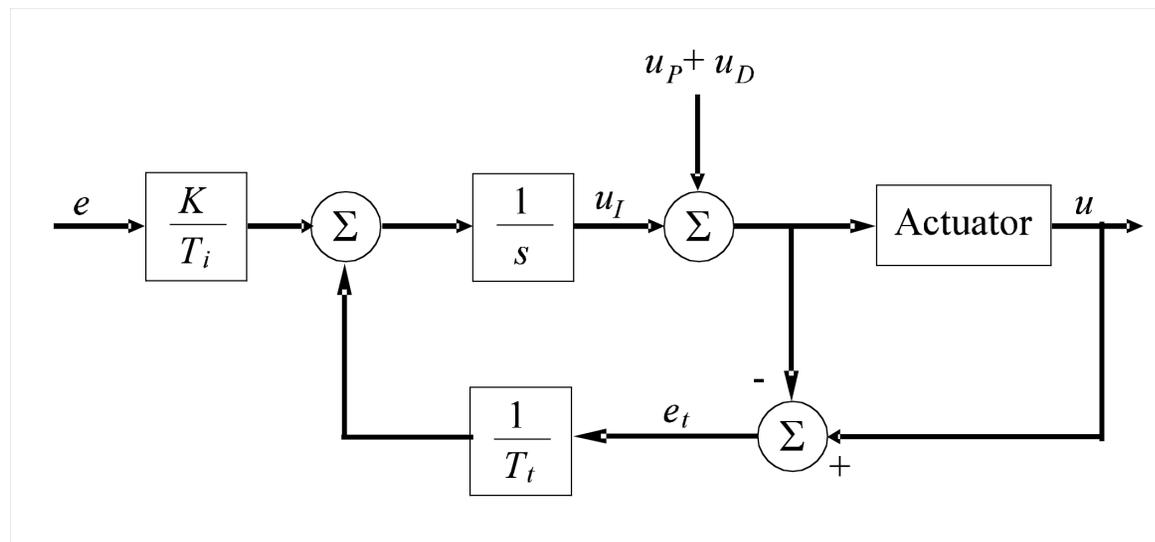
Actuator Saturation

Saturation of 1.6 applied to control signal for plant G_1

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

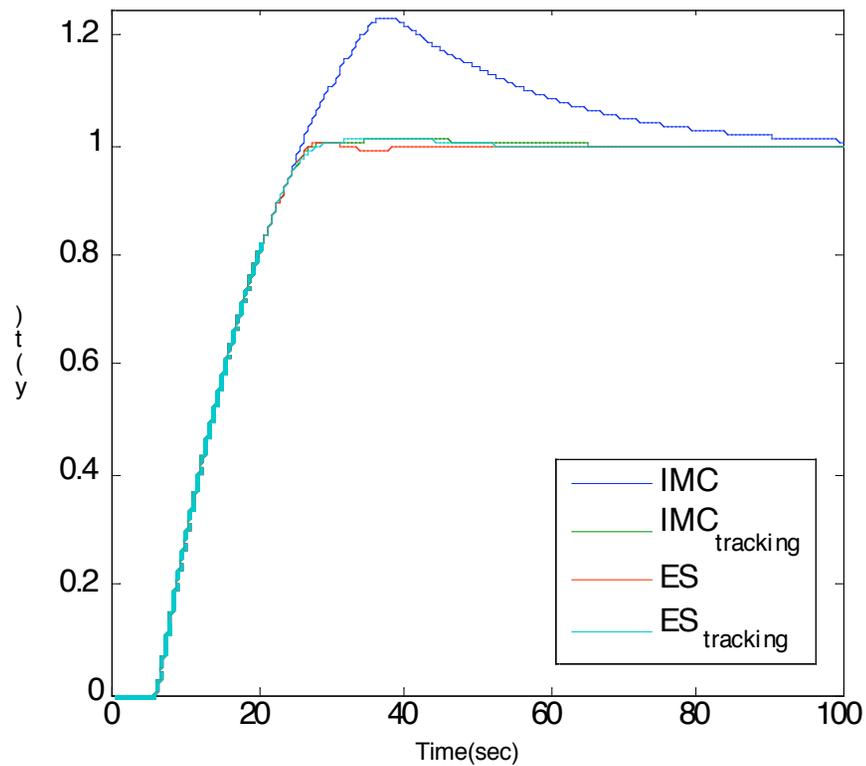
ES and IMC compared with and without the addition of an anti windup scheme

Tracking anti-windup scheme

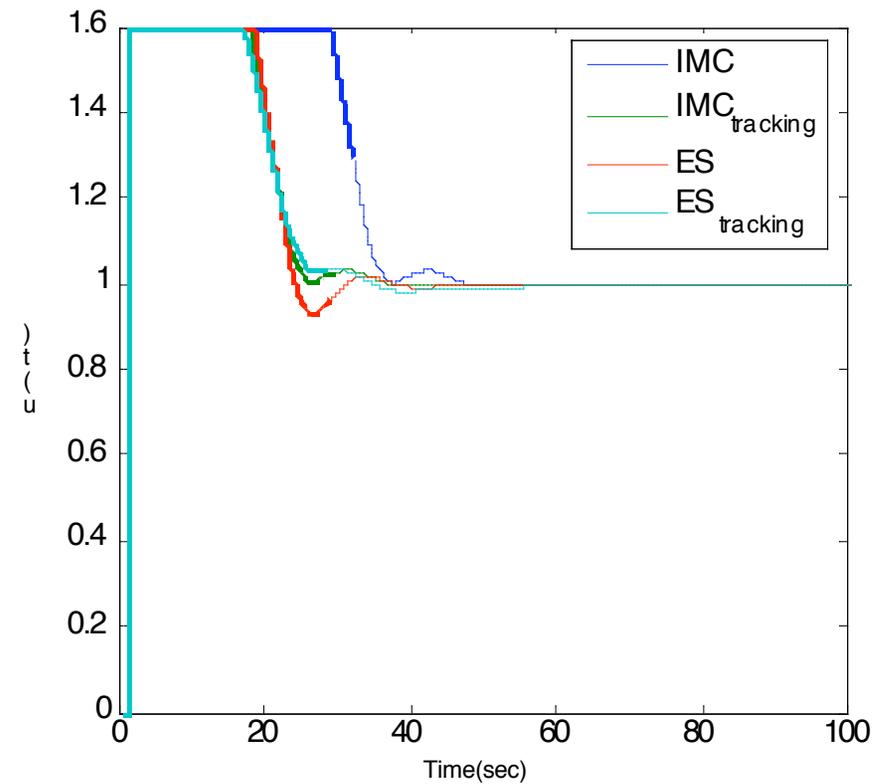


Actuator Saturation

Step response of output



Control signal during step response



Effects of Noise

Band-limited white noise has been added to output

Power spectral density = 0.0025

Correlation time = 0.2

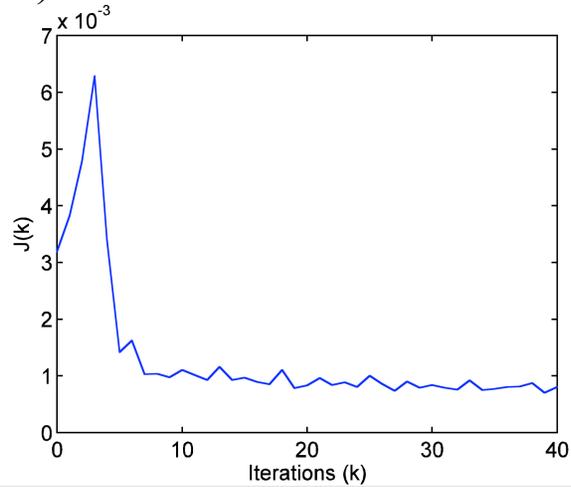
Independent noise signal for each iteration

Simulations on plant G_1

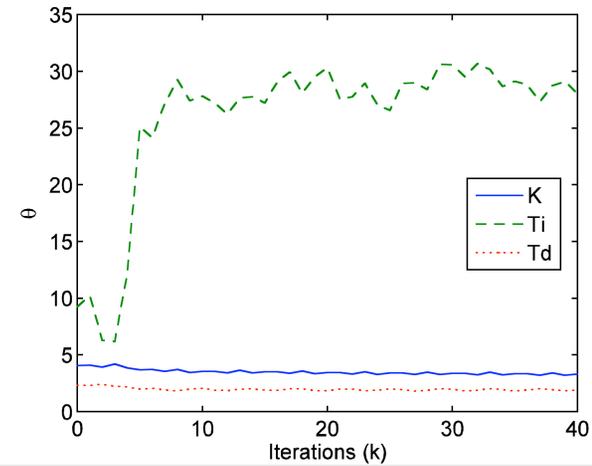
$$G_1(s) = \frac{1}{1 + 20s} e^{-5s}$$

Effects of Noise

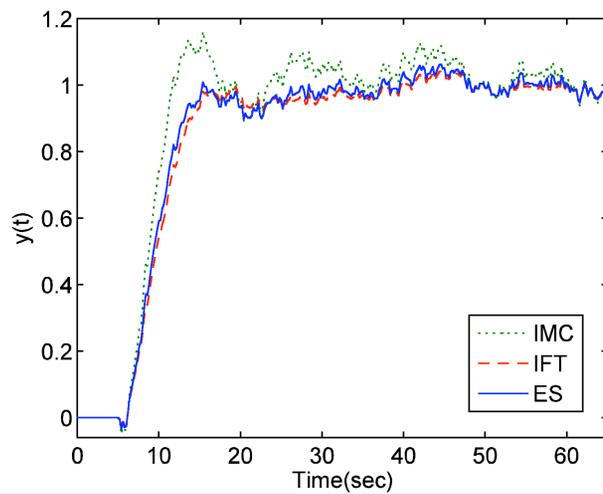
a) Evolution of Cost Function



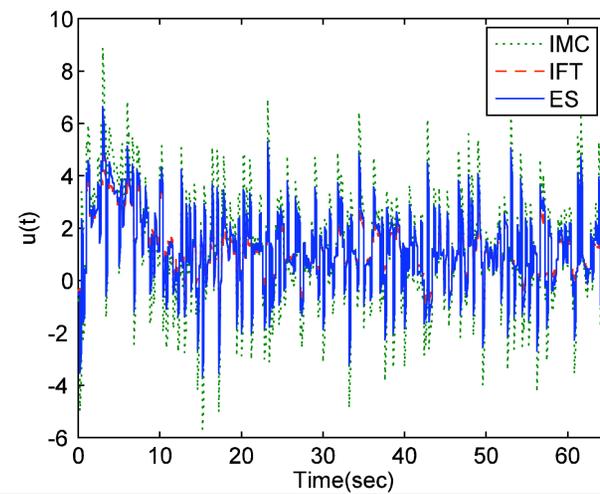
b) Evolution of PID Parameters



c) Step Response of output



d) Step Response of controller



Selecting Parameters of ES Scheme

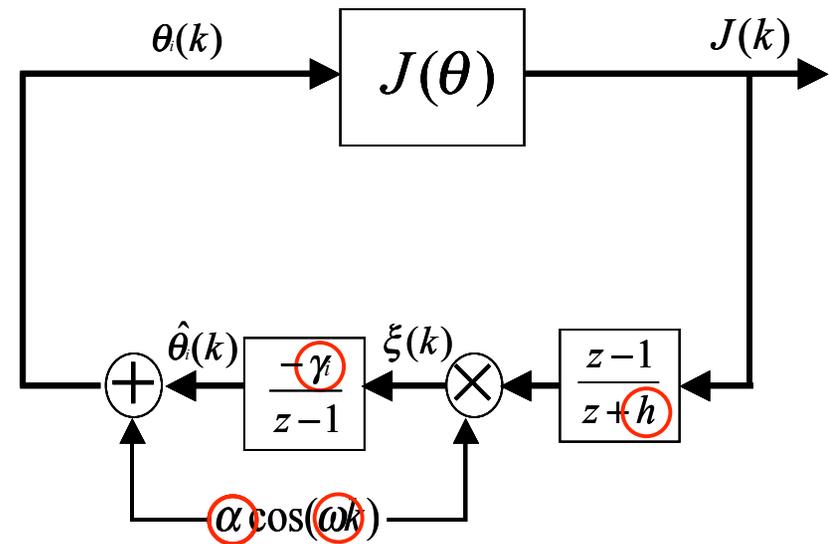
Must select

α , perturbation step size

γ , adaptation gain

ω , perturbation frequency

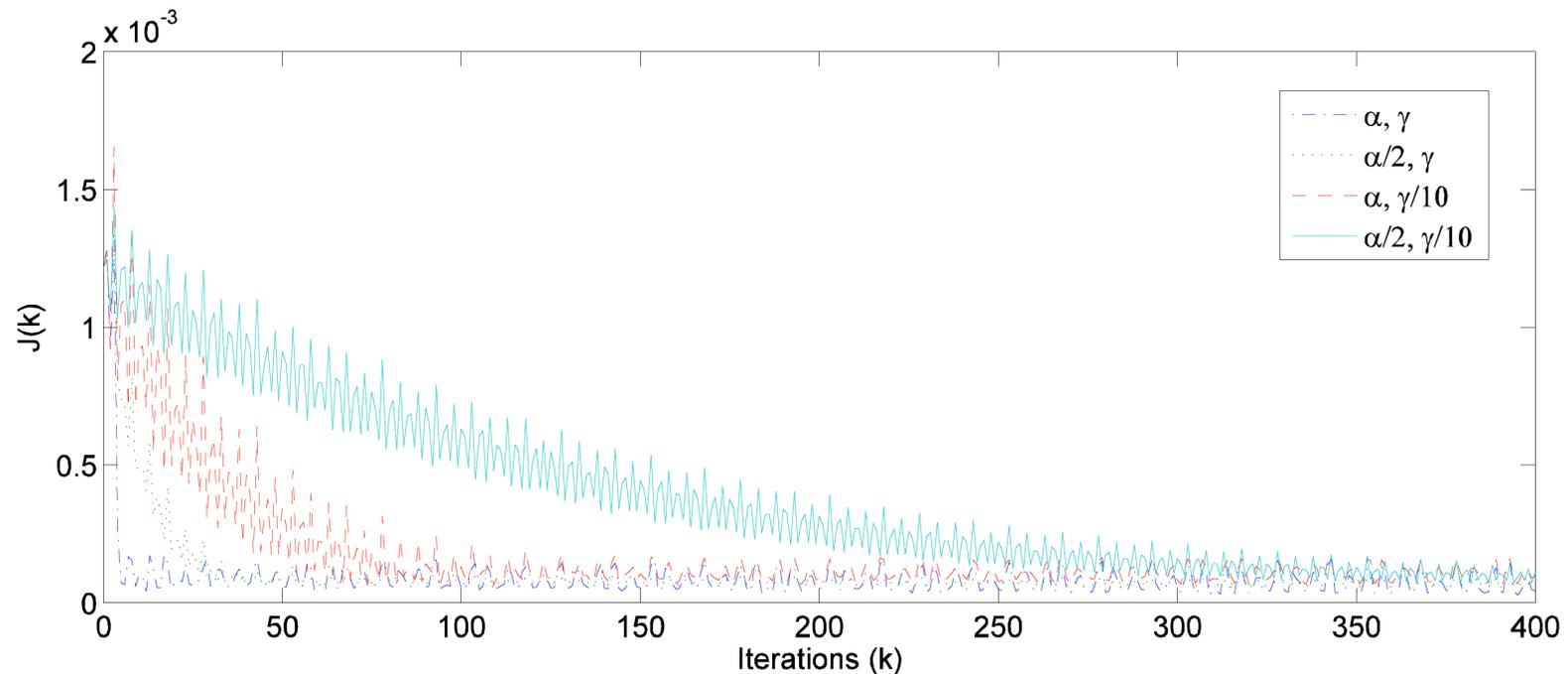
h , high-pass filter cut-off frequency



Looks like 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
α, γ	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

Selecting Parameters of ES Scheme

Need to select an adaptation gain γ and perturbation amplitude α for EACH parameter to be estimated

In the case of a PID controller, $\theta = [K, T_i, T_d]$, so we need three of each.

The modulation frequency is determined by:

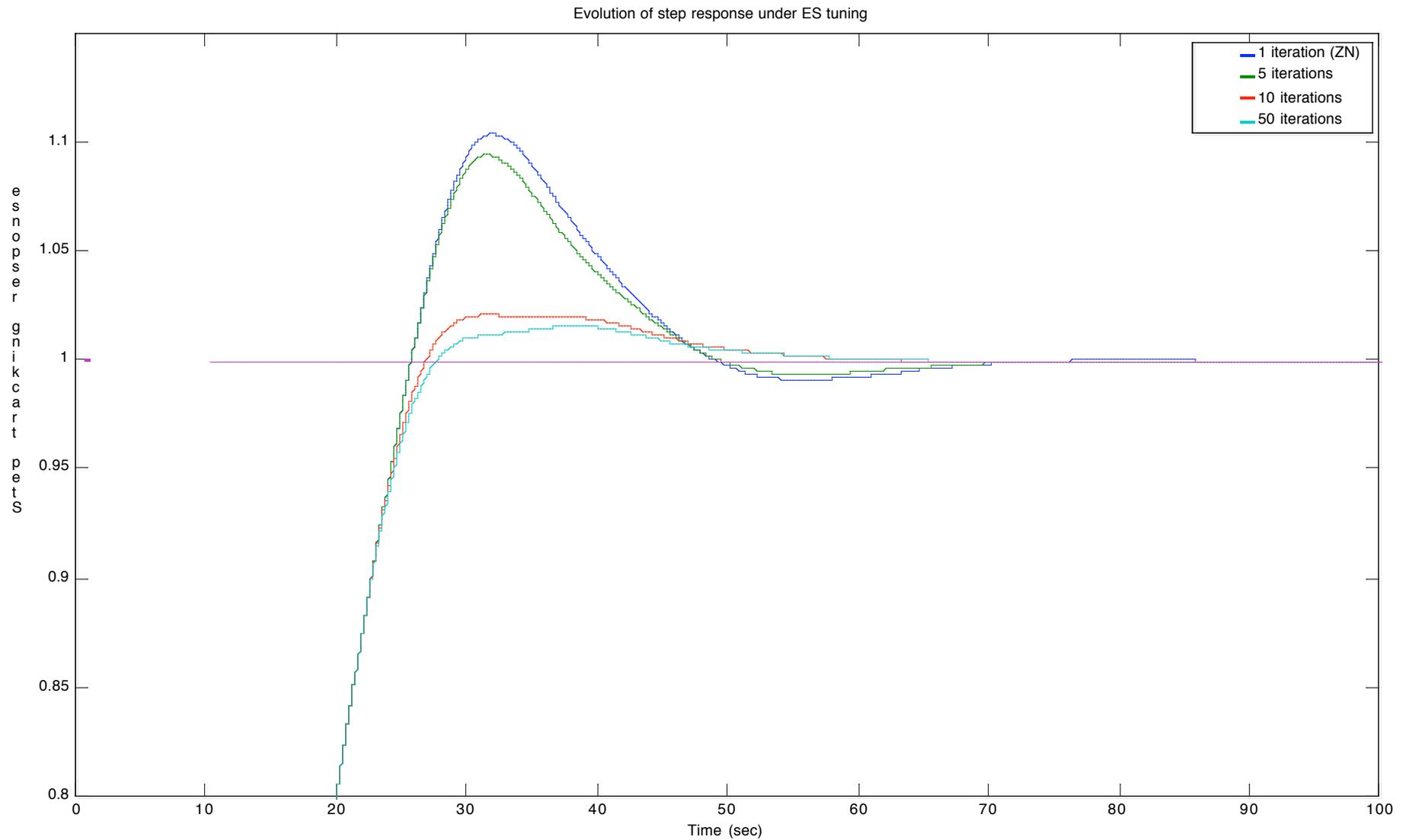
$$\omega_i = \alpha^i * \pi$$

where $0 < \alpha < 1$

The highpass filter $(z-1)/(z+h)$ is designed with $0 < h < 1$ with the cutoff frequency well below the modulation frequency ω_i .

Convergence rate is directly affected by choice of α and γ , as well as by cost function shape near minimizer.

Example of ES-PID tuner GUI



Punch Line

ES yields performance as good as the best of the other popular tuning methods

Can handle some nonlinearities and noise.

The cost function can be modified such that different performance attributes are emphasized

Control of HCCE Engines

Based on contributions by: Nick Killingsworth (UCSD),
Dan Flowers and Sal Aceves (Livermore Lab),
and Mrdjan Jankovic (Ford)

HCCI = ?

HCCI = Homogeneous Charge Compression Ignition

Low NOx emissions like **spark-ignition** engines

High efficiency like **Diesel** engines

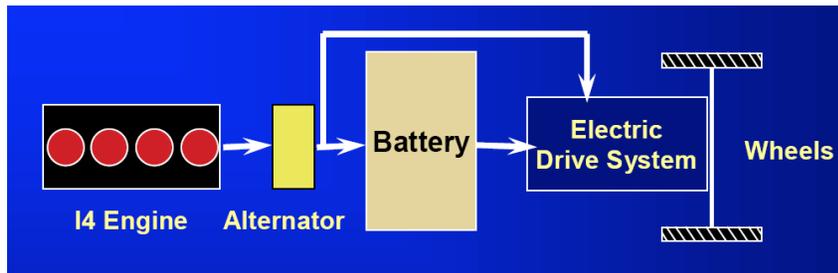
More promising in near term than fuel cell/hydrogen engines

HCCI Engine Applications

Distributed power generation



Automotive hybrid powertrain



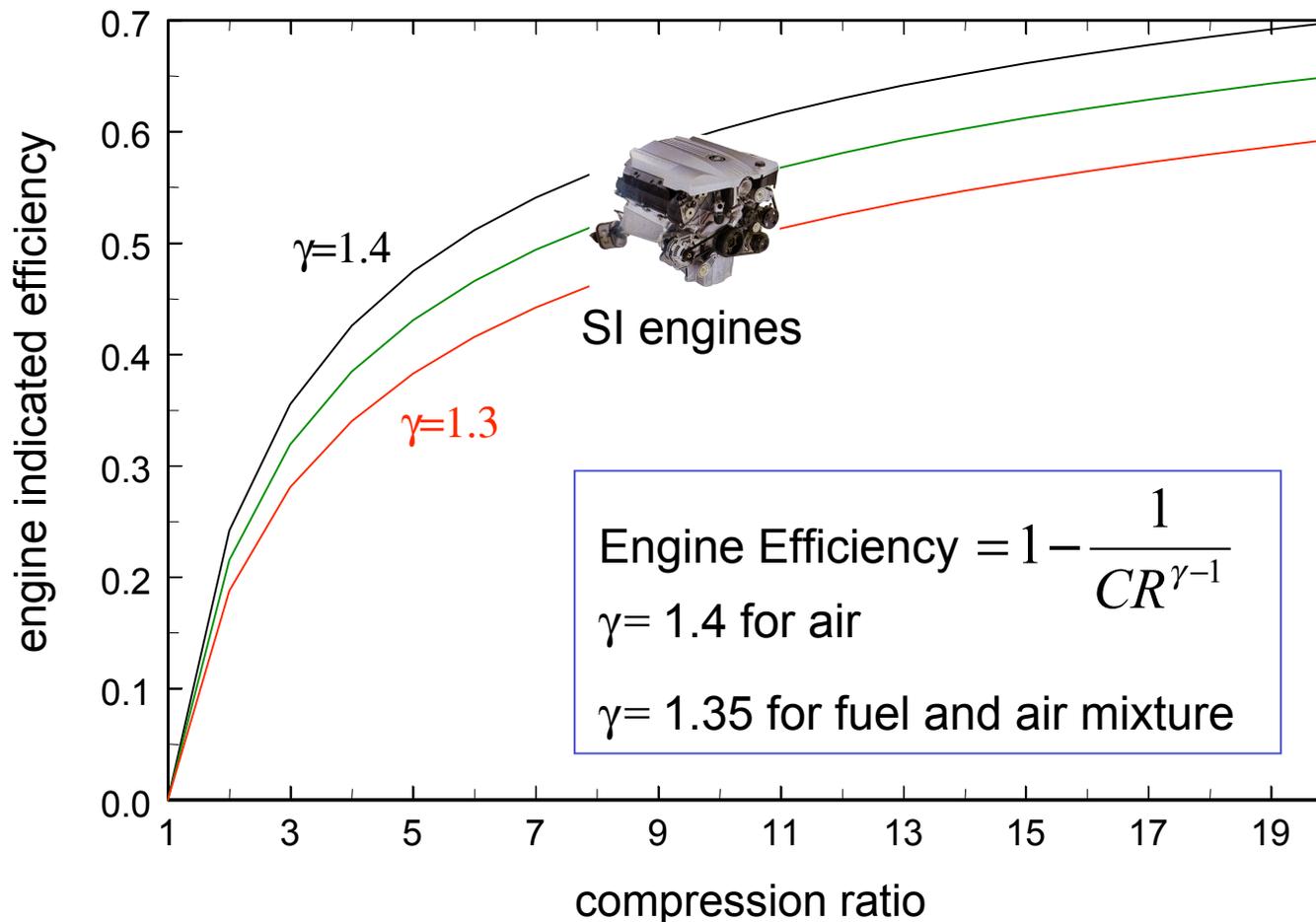
**What is the difference
between Spark Ignition,
Diesel, and HCCI engines?**

Categories of Engines

	Compression Ignition	Spark ignition
Homogeneous charge	HCCI	Spark ignition engine
Inhomogeneous charge	Diesel	Direct injection engine

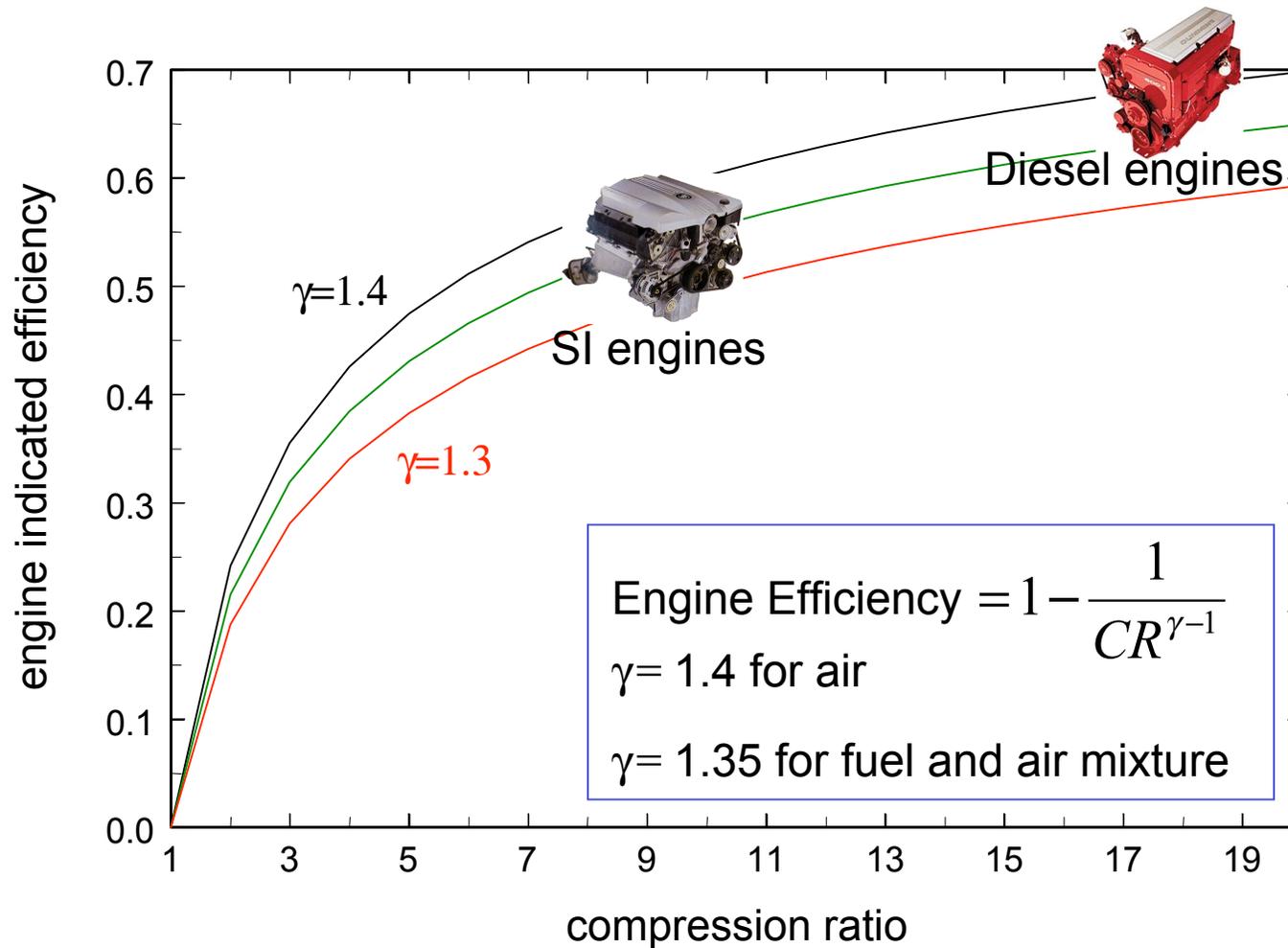
Spark Ignition Engine

Basic engine thermodynamics: engine **efficiency** increases as the compression ratio and $\gamma = c_p/c_v$ (ratio of specific heats) increase



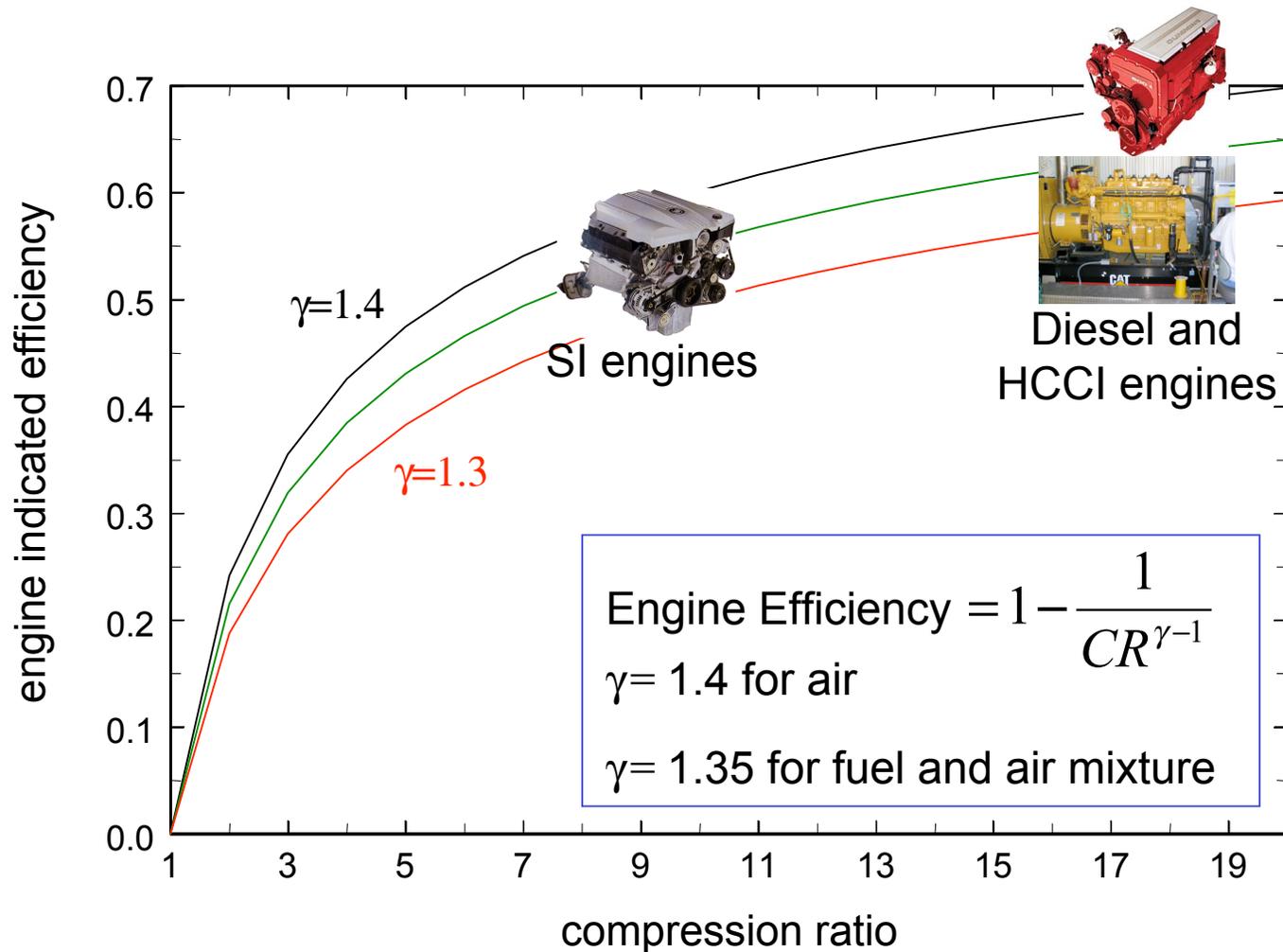
Diesel Engine

Highly **efficient** because they compress only air (γ is high) and are not restricted by knock (compression ratio is high)



HCCI Engine

Compression ratio not restricted by “knock” (autoignition of gas ahead of flame in spark ignition engines) \rightarrow efficiency comparable to Diesel



HCCI Engine

Potential for high efficiency (Diesel-like)

Low NO_x and PM (unlike Diesel)

BUT, no direct trigger for ignition - requires feedback to control the timing of ignition!

Experiment at Livermore Lab

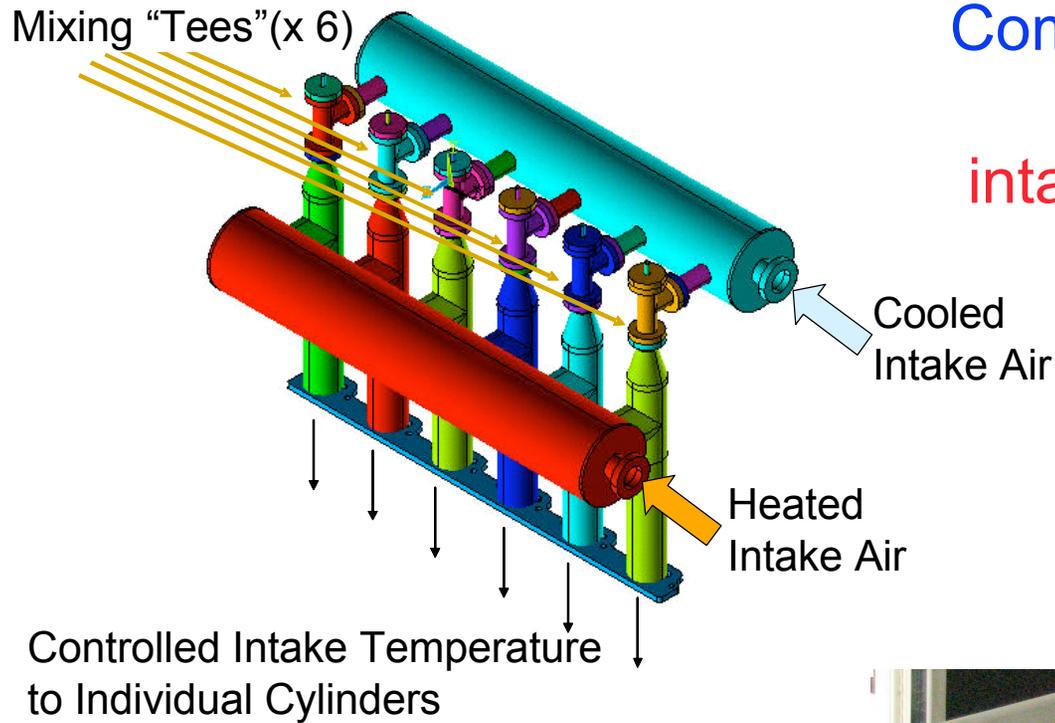
Caterpillar 3406 natural gas spark ignited engine converted to HCCI

Set up for stationary power generation (not automotive)



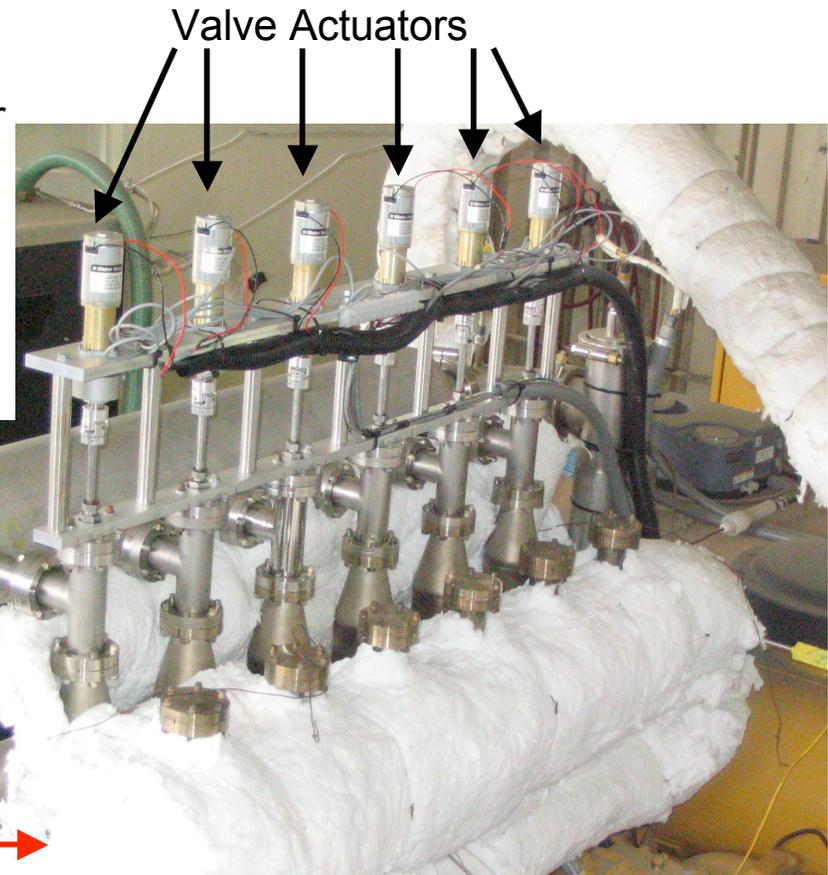
Actuators

Combustion timing (output)
is very sensitive to
intake temperature (input)



Cold Manifold

Hot Manifold



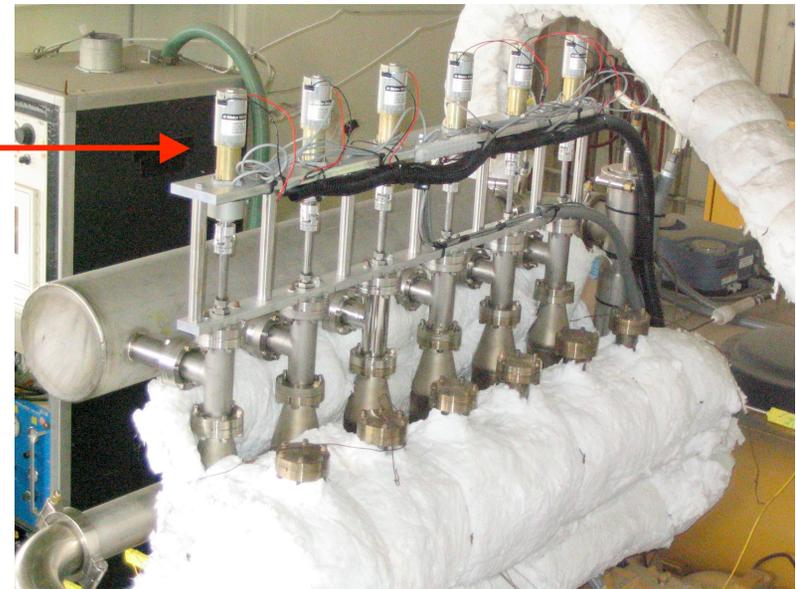
Overall Architecture: Sensors and Software

User interface



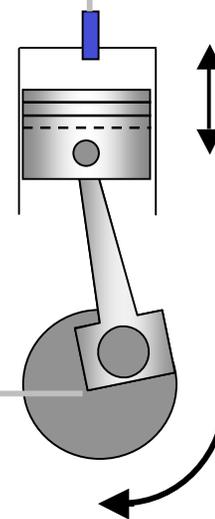
Valve Position

Real-Time Controller
PC running Labview RT OS

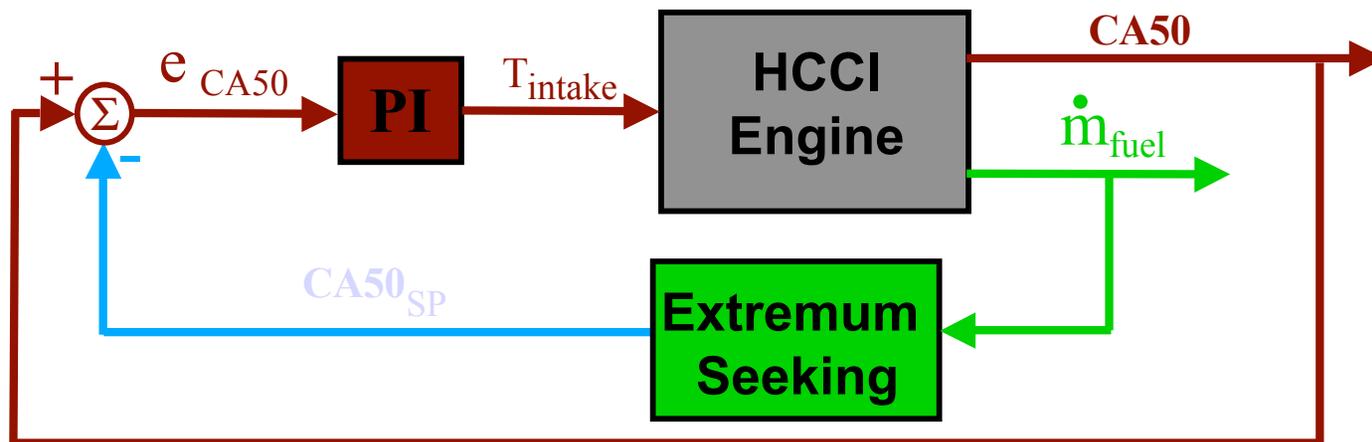


Cylinder Pressure

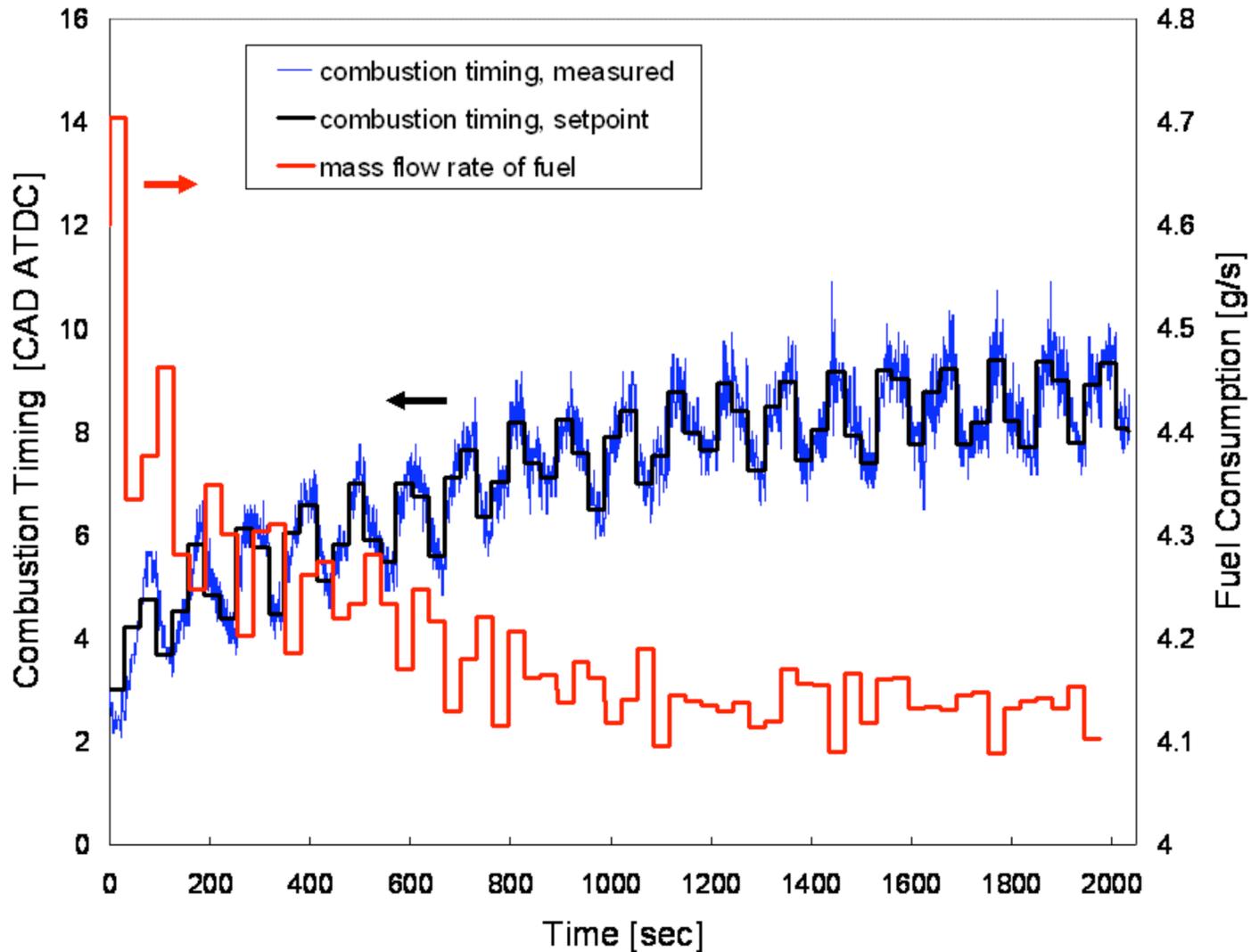
Crank Angle Position



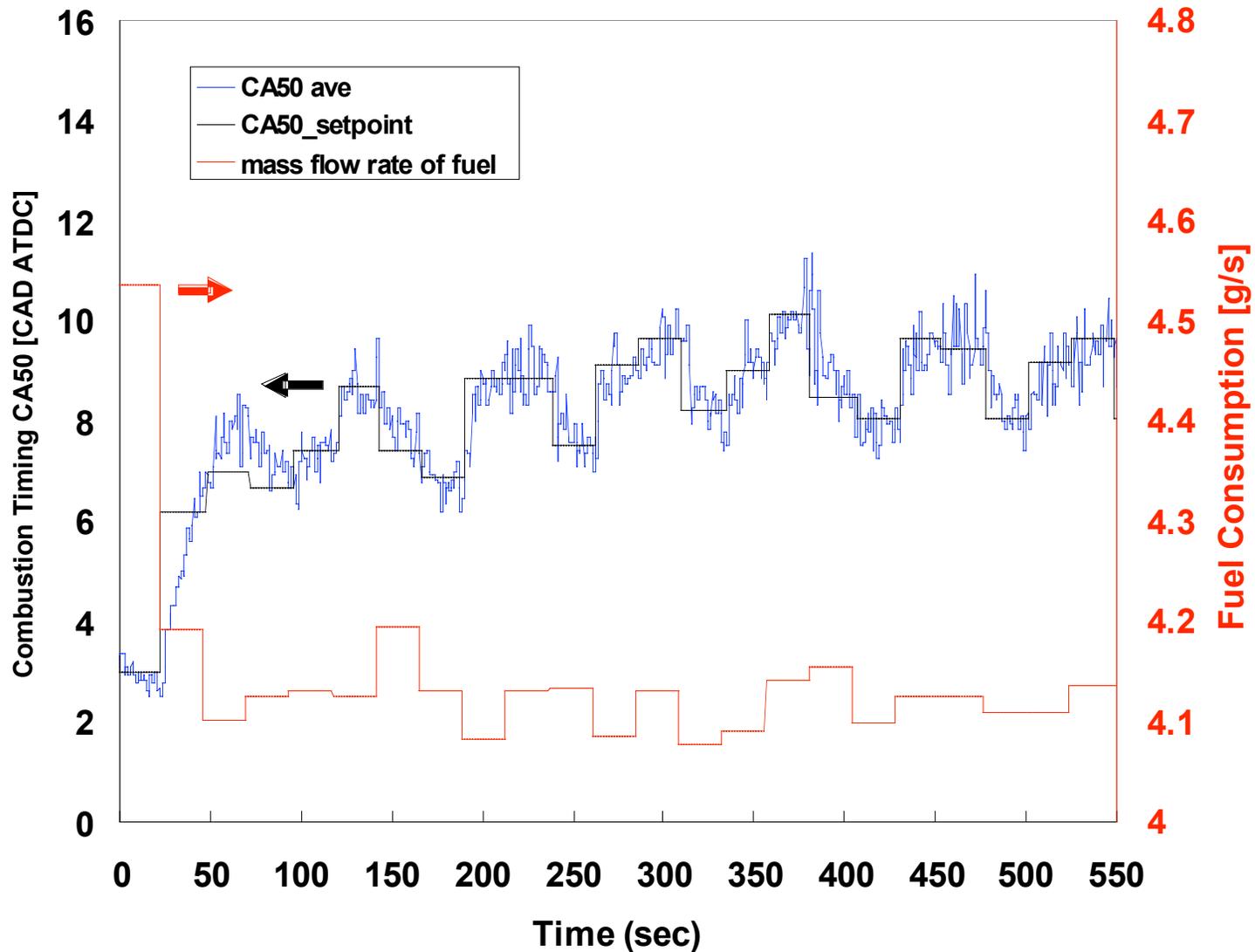
ES used to **MINIMIZE FUEL CONSUMPTION** of HCCI engine
by tuning combustion timing setpoint



ES delays the combustion timing 6 crank angle degrees, reducing fuel consumption by > 10%



Larger adaptive gain: ES finds same minimizer, but much more quickly



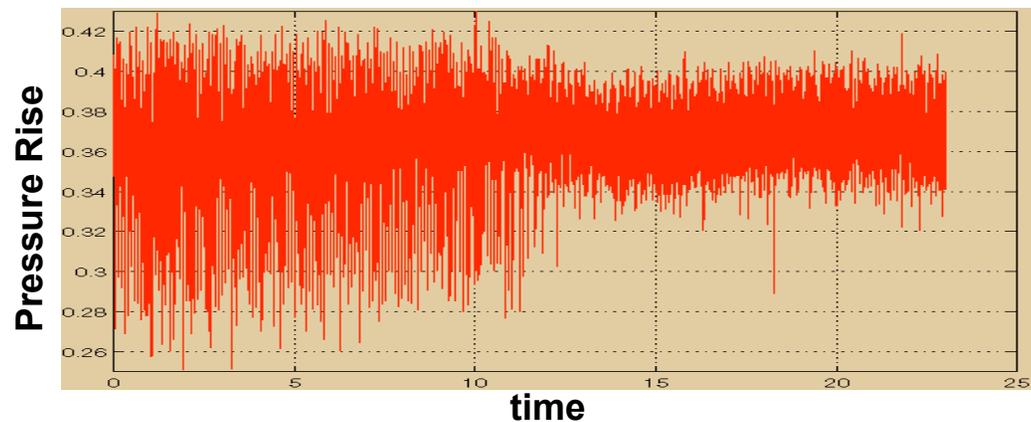
Axial Flow (Jet Engine-Like) Compressor Control

Problem Statement

- Active controls for **rotating stall** only reduce the stall oscillations but they do not bring them to zero nor do they maximize pressure rise.
- Extremum seeking to optimize compressor operating point.

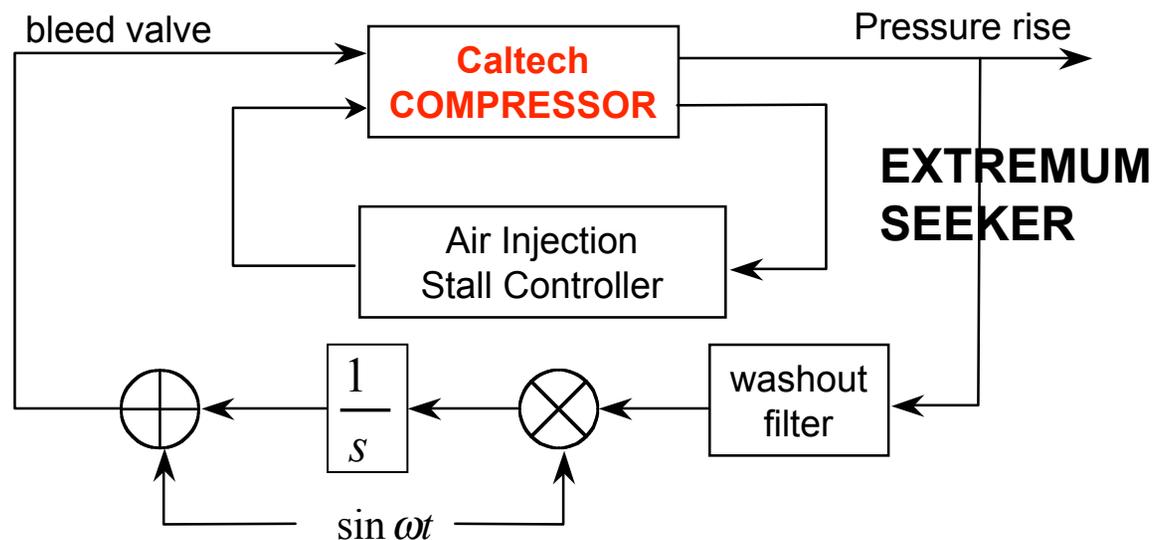
Experimental Results

Extremum seeking stabilizes the **maximum** pressure rise.



Motivation

Smaller, **lighter compressors**; higher payload in aircraft

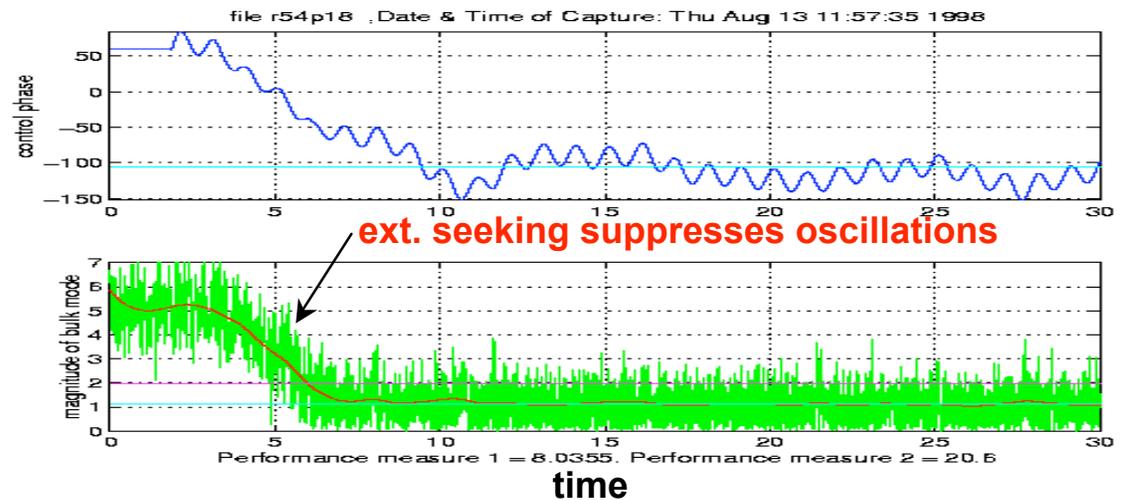


Combustion Instability Control

Problem Statement

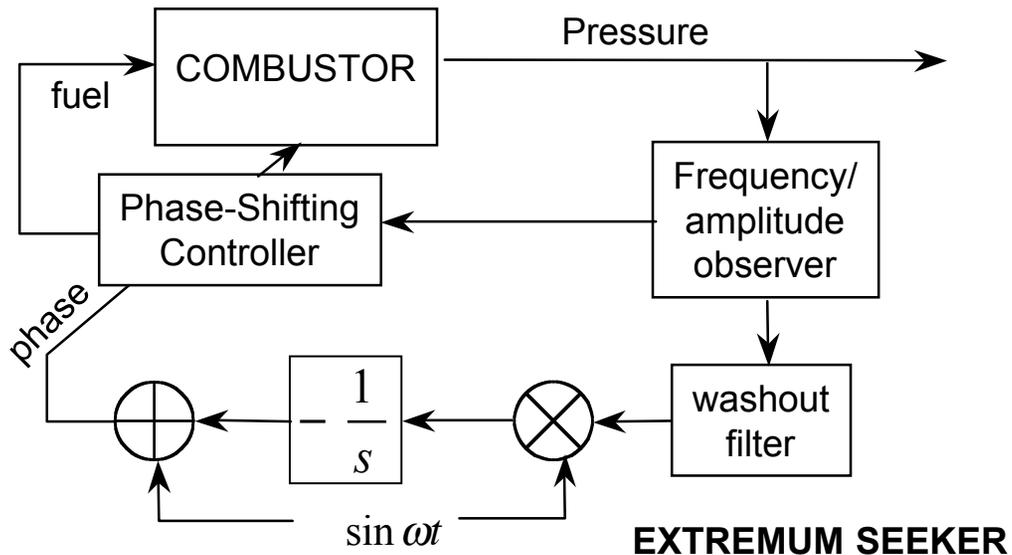
- Rayleigh criterion-based controllers, which use phase-shifted **pressure measurements** and **fuel modulation**, have emerged as prevalent
- The length of the phase needed varies with operating conditions. The **tuning** method must be non-model based.

Experiment on UTRC 4MW combustor

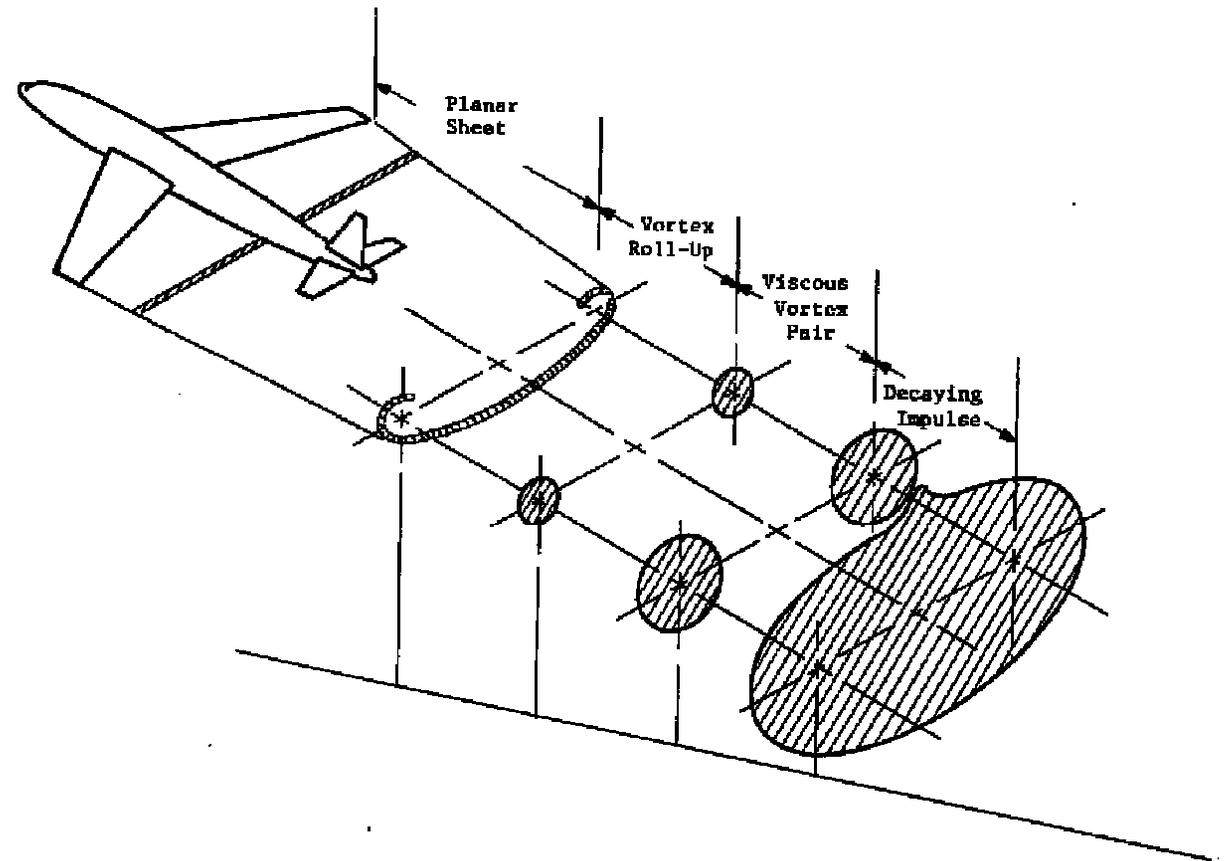


Motivation

- Tuning allows operation with **minimum oscillations** at **lean** conditions
- **Reduced** engine size, fuel consumption and NO_x **emissions**

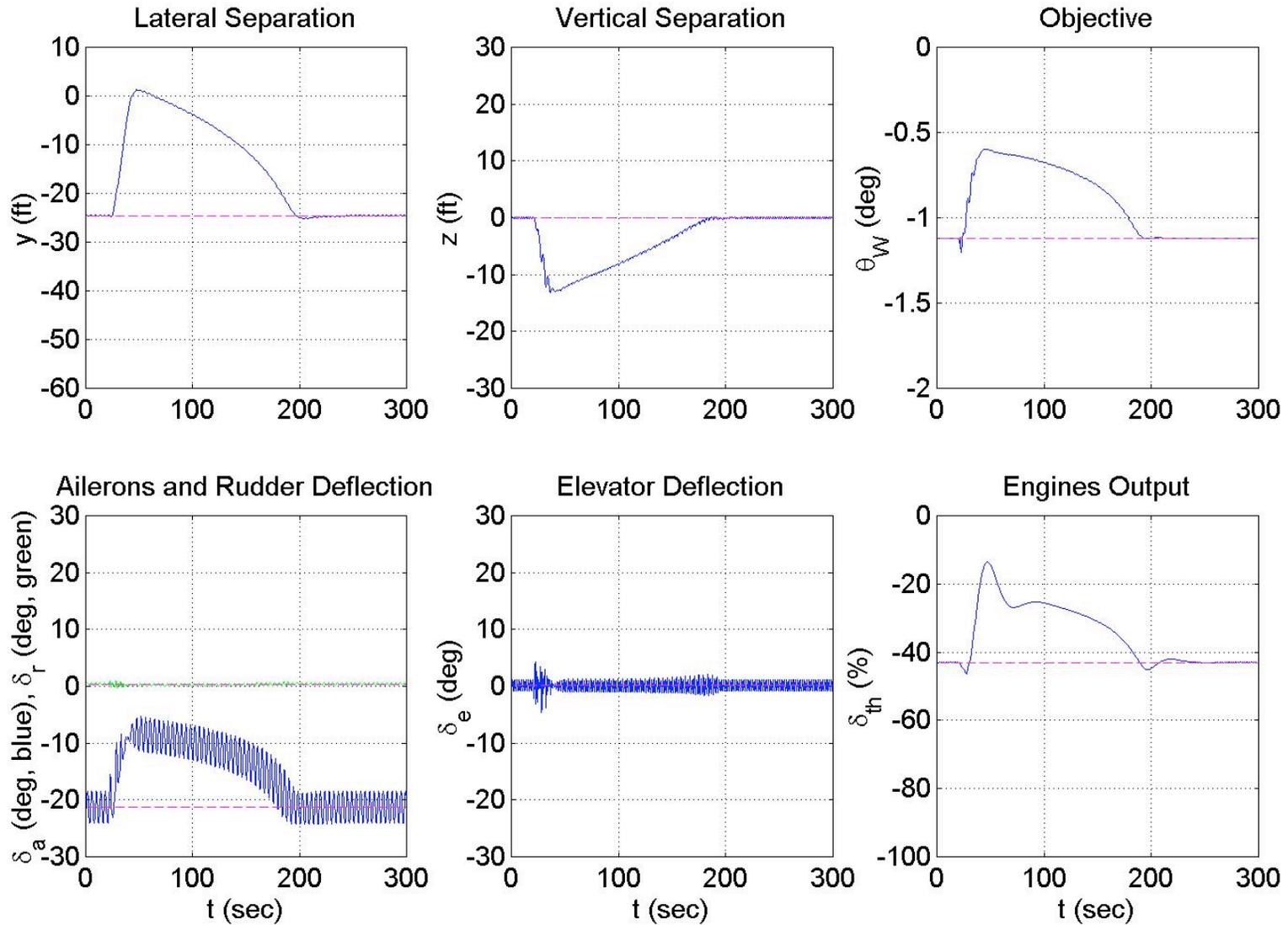


Formation Flight Engine Output Minimization

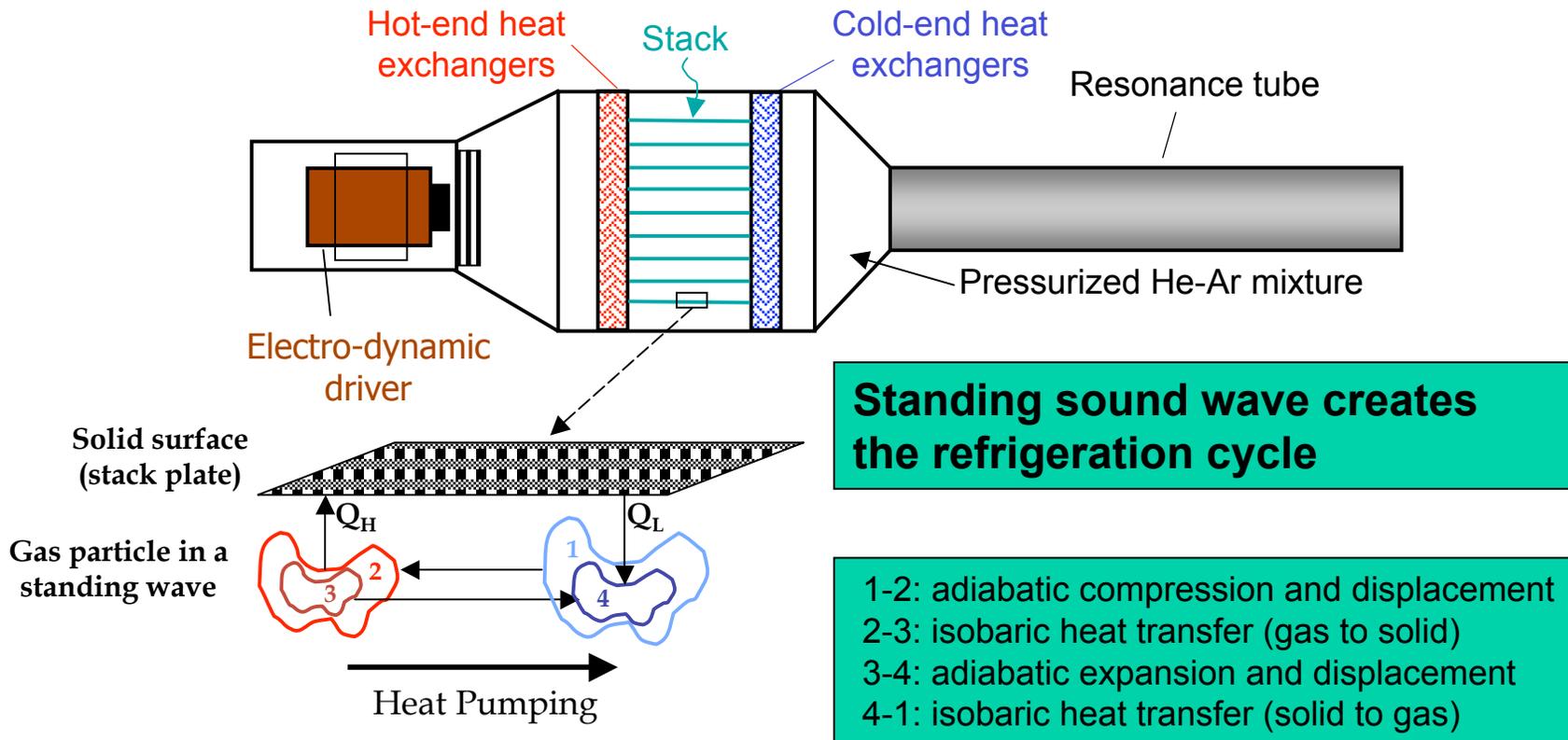


Tune reference inputs y_{ref} and z_{ref} to the autopilot of the **wingman** to maximize its downward pitch angle or to **minimize its engine output**

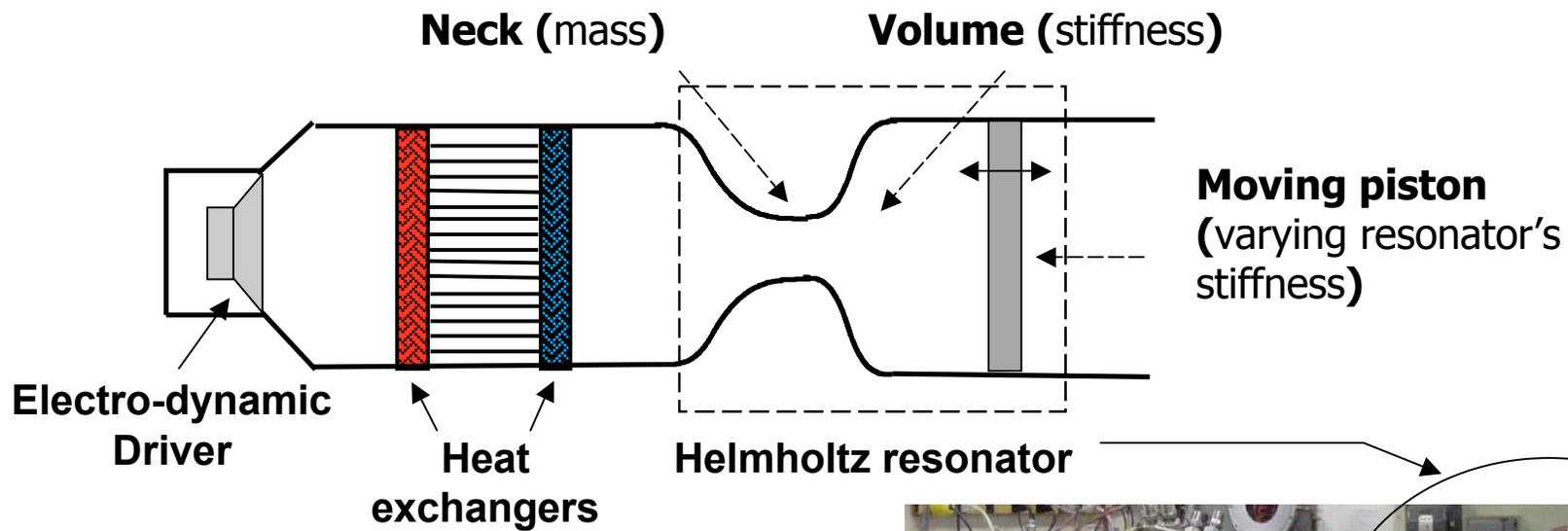
Simulation of C-5 Galaxy transport airplane for a brief encounter of “clear air turbulence”



Thermoacoustic Cooler (M. Rotea)

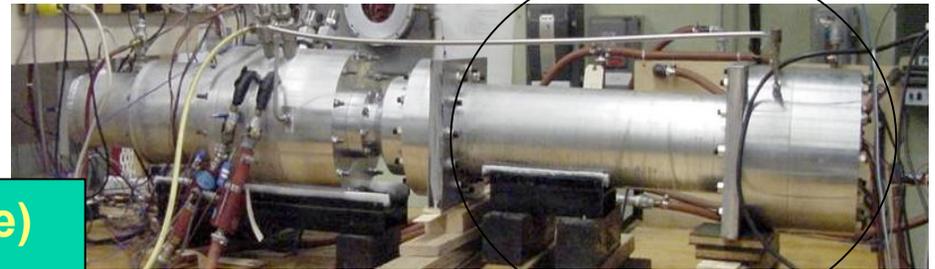


Thermoacoustic Cooler

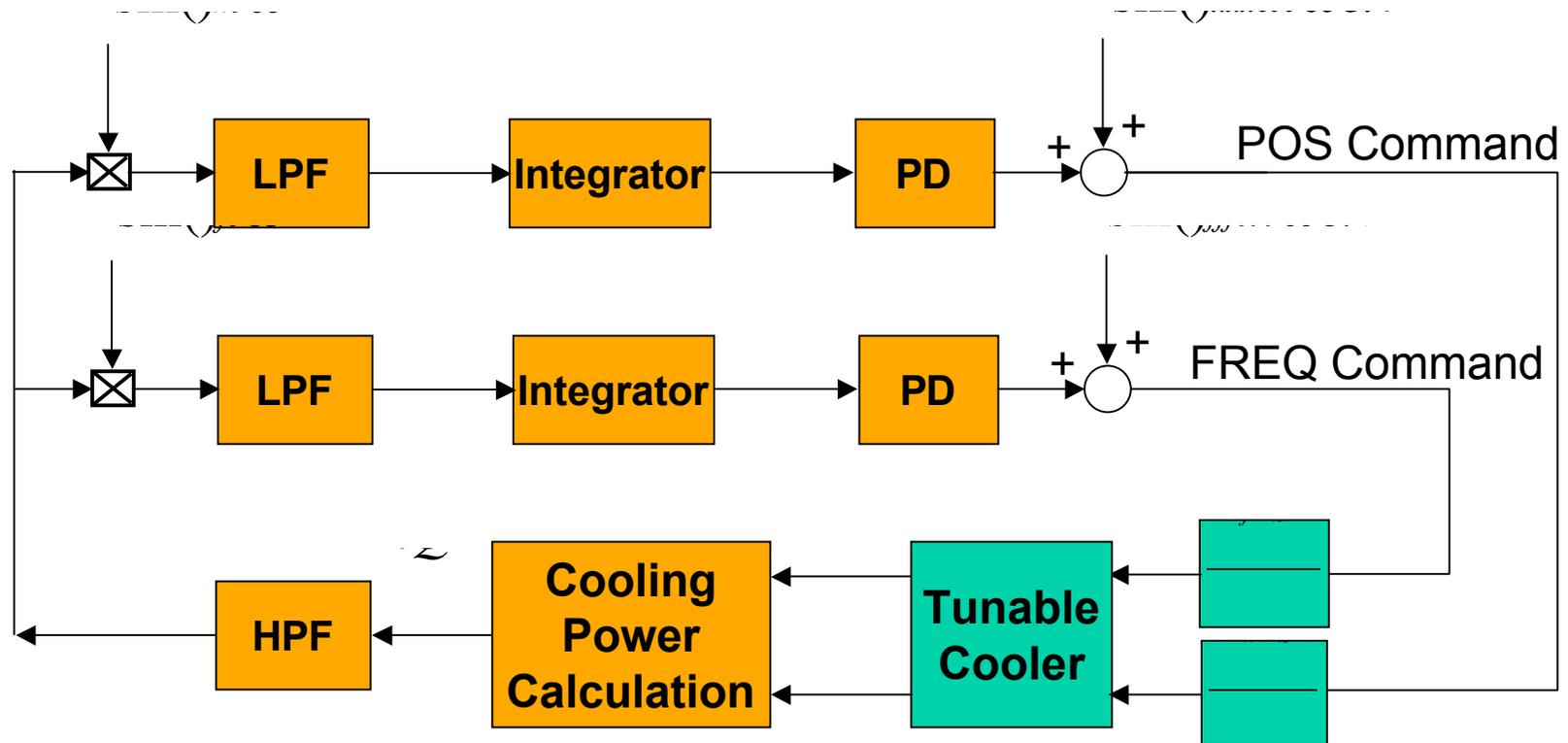


Tuning Variables

- Piston position (acoustic impedance)
- Driver frequency



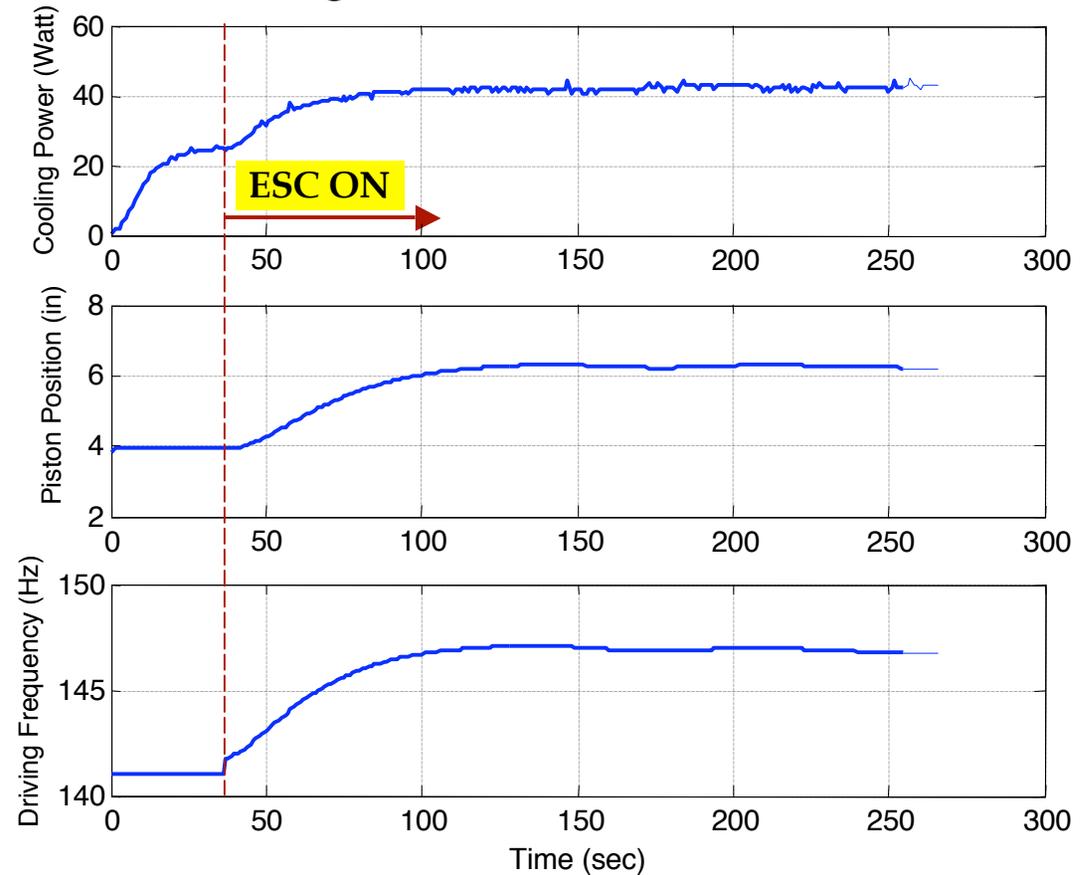
ES with PD compensator



Experiment – Fixed Operating Condition

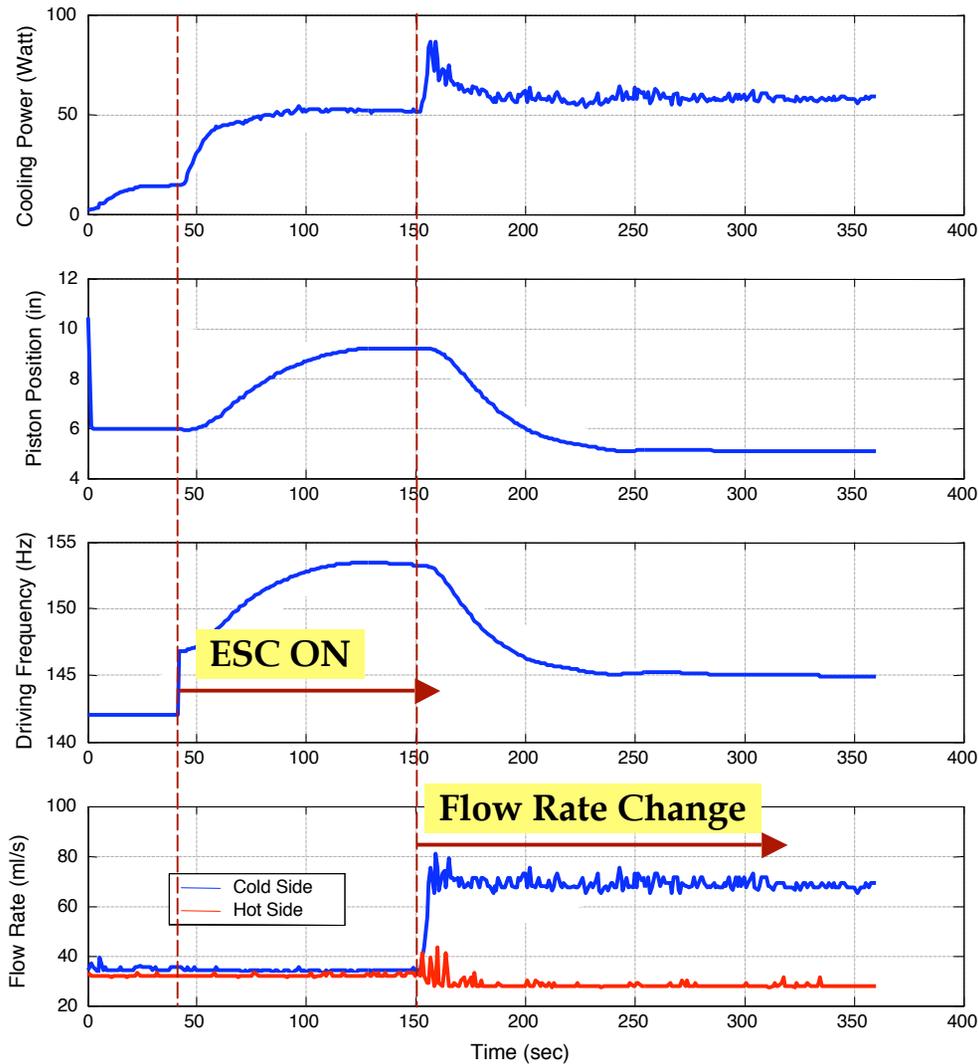
POS in.	FREQ Hz	POWER W
4	141	22.65
	142	29.92
	143	35.67
	144	28.63
	145	21.25
5	142	15.89
	144	34.12
	145	39.68
	146	35.12
	148	19.34
6	140	4.95
	142	9.00
	144	18.55
	145	23.86
	146	35.99
	147	41.28
	148	38.00
	149	30.36
	150	19.36
7	146	16.34
	148	33.34
	149	41.21
	150	40.70
	151	34.69
8	153	19.63
	151	32.16
	152	35.60
	153	31.74

Cooling Performance with ESC



ESC quickly finds optimum operating point (41.3W, 147Hz, 6.2in)

Experiment – Varying Operating Condition



ESC tracks optimum after cold-side flow rate is increased

ES for the Plasma Control in the Frascati Fusion Reactor

Contribution by Luca Zaccarian (U. Rome, Tor Vergata)

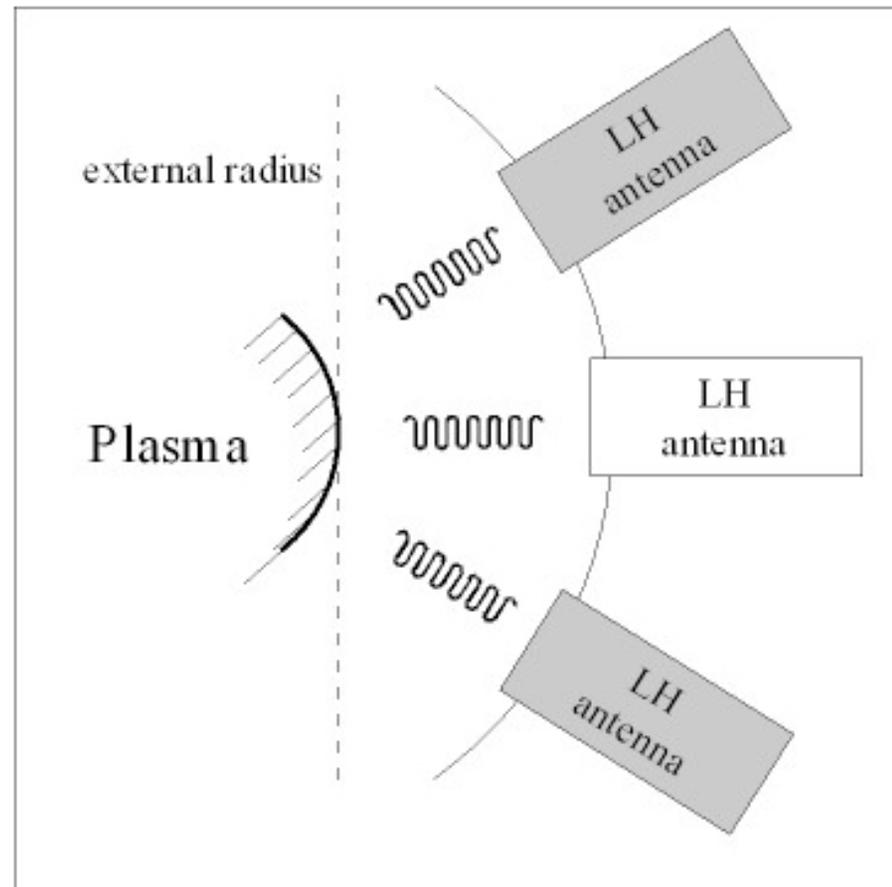
Optimization Objective

Framework:

Additional Radio Frequency heating injected in the plasma by way of Lower Hybrid (LH) antennas: plasma reflects some power

Goal:

Optimize coupling between the Lower Hybrid antenna and the plasma, during the LH pulse



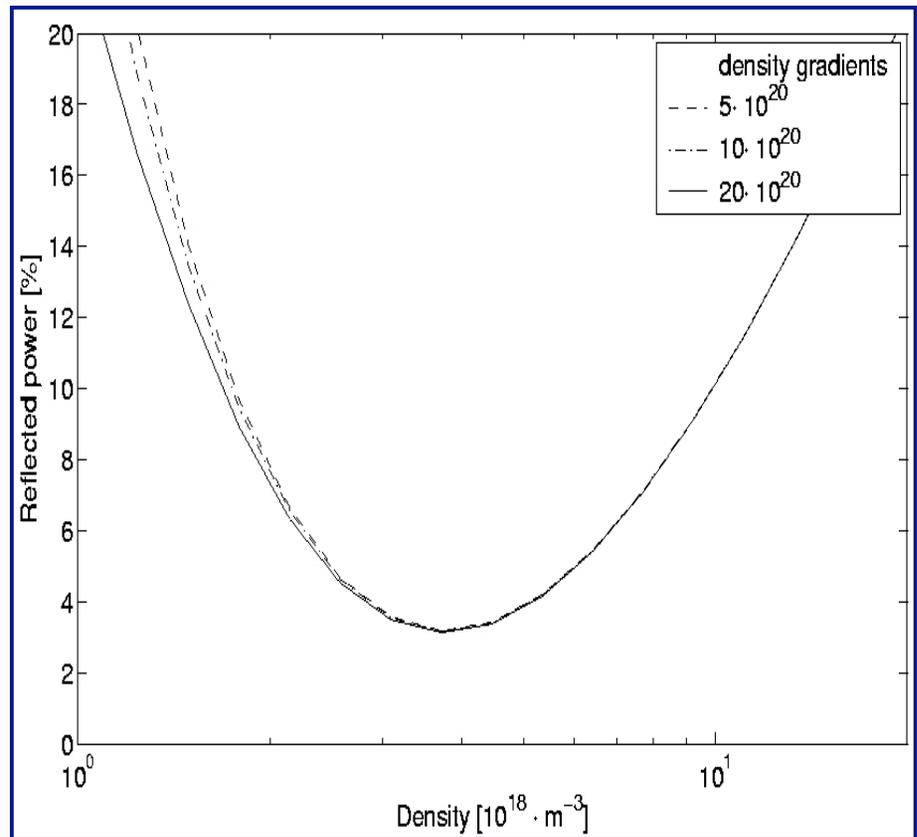
Reflected Power Map

Reflected power:

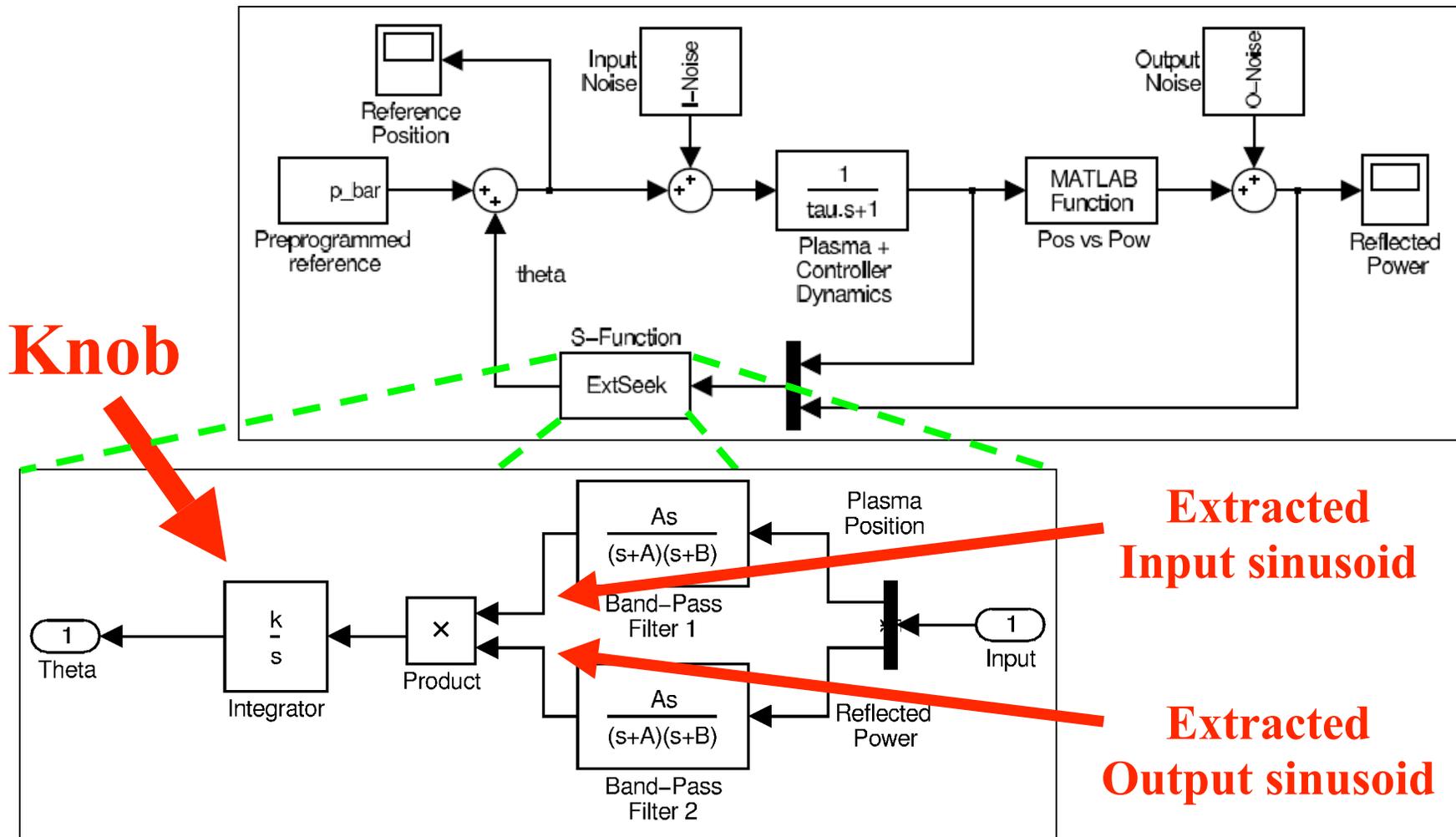
Convex fcn of edge density
Convex fcn of edge position

Possible approaches to optimize:

1. Move the antenna
(too slow!)
2. Move the plasma
(viable – adopted here)



Probing not Allowed - Modified ES Scheme

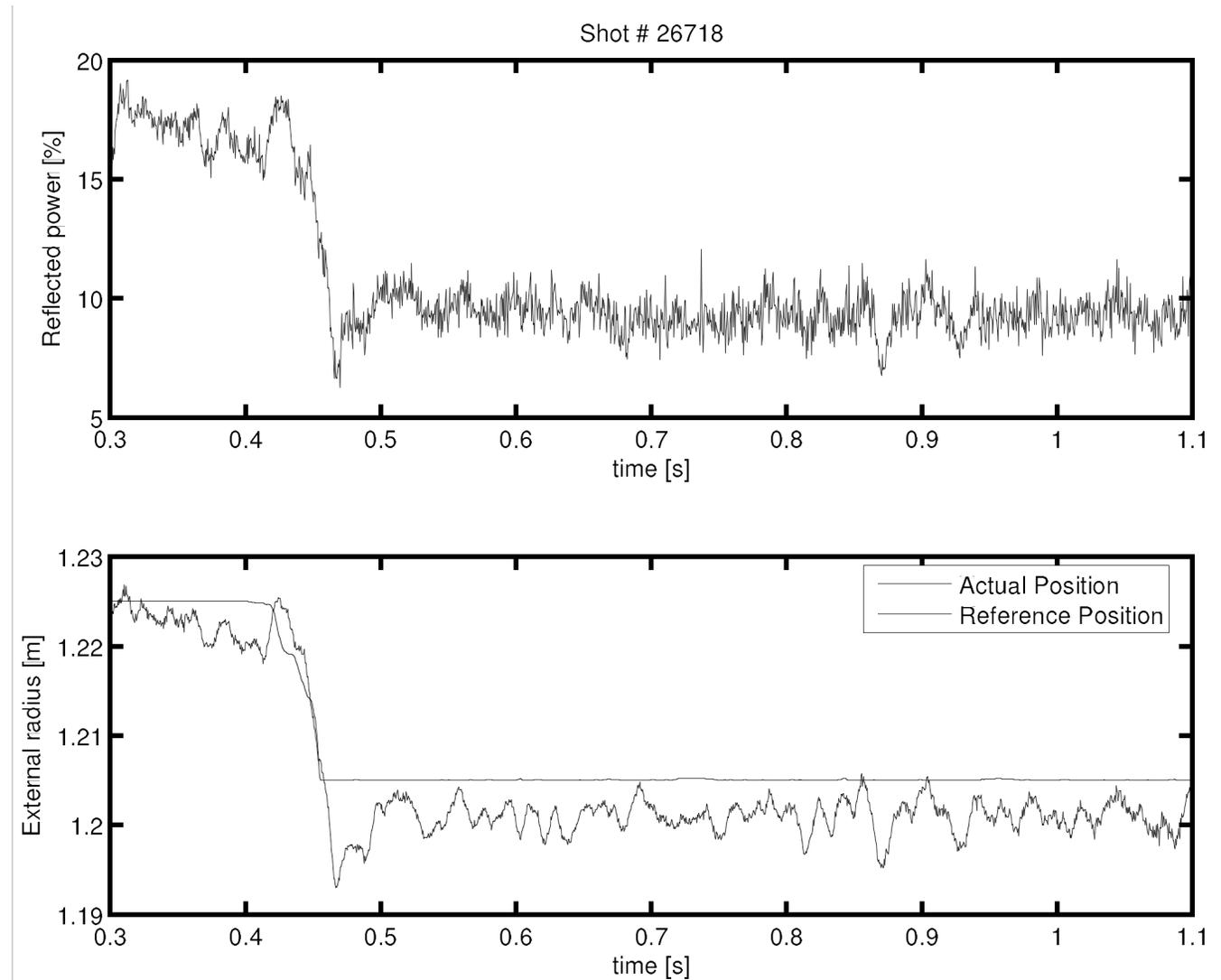


Experimental results with medium gain

$$K = 300$$

Safety
saturation
limits
performance

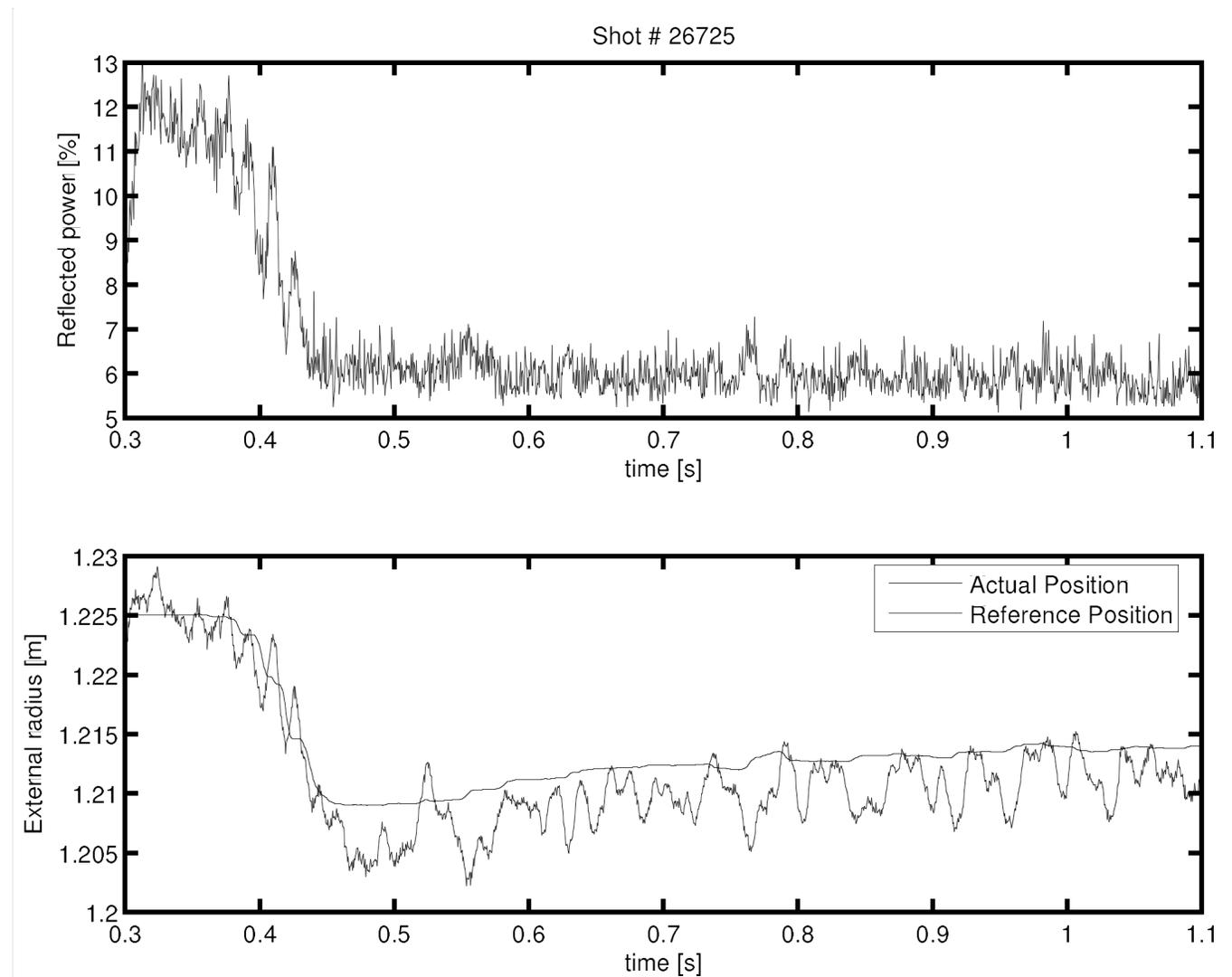
Control action
is quite
aggressive.



Experimental results with lower gain

$K = 200$
(Antenna has
been moved)

Graceful
convergence
to the
minimum
reflected
power

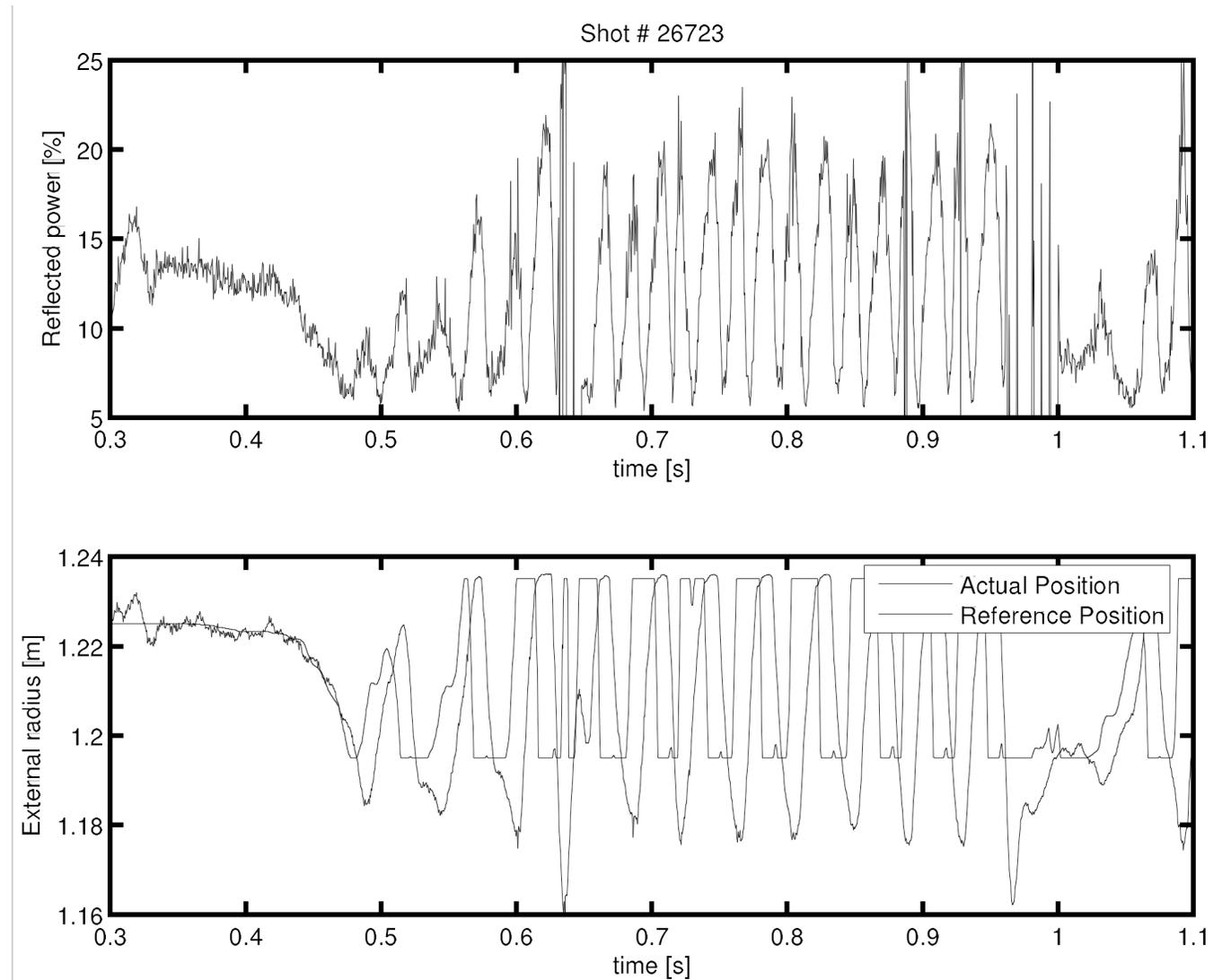


Gain too high - instability

$K = 350$

Instability

**Gain is
too large**



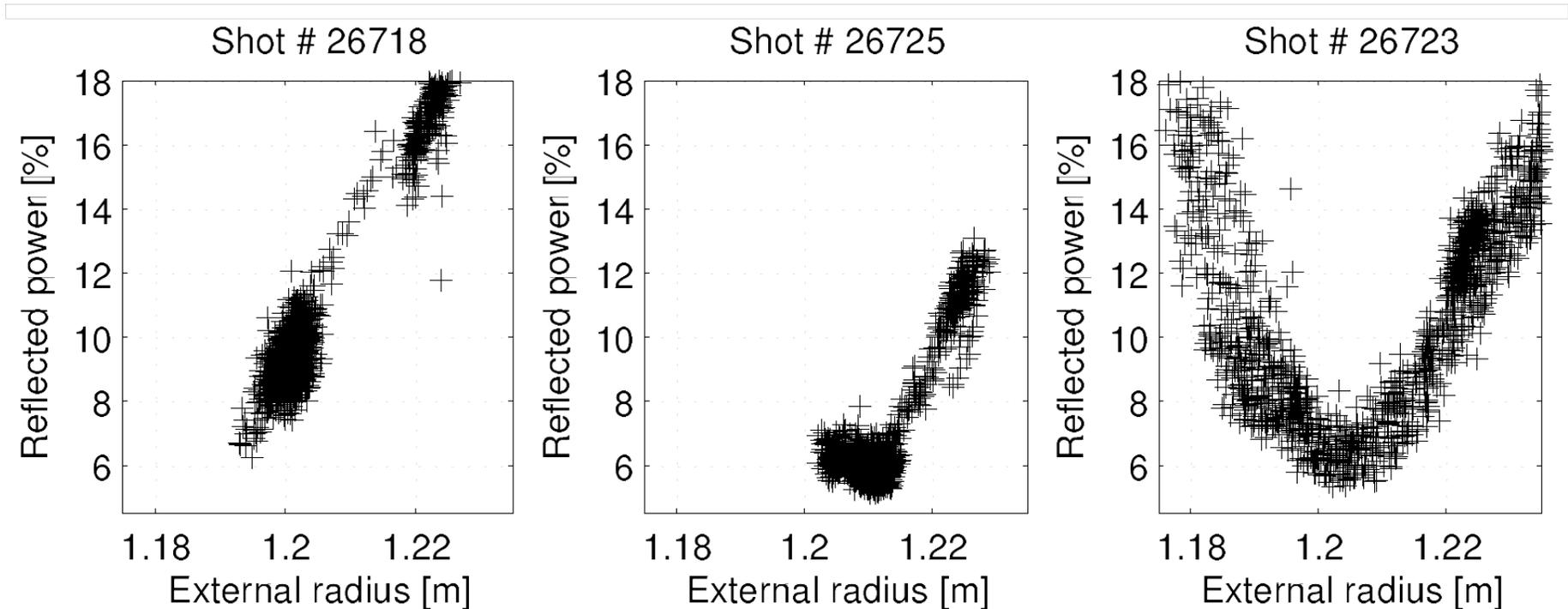
Experiments - Summary

Input/output plane representation:

$K = 300$: saturation prevents reaching the minimum

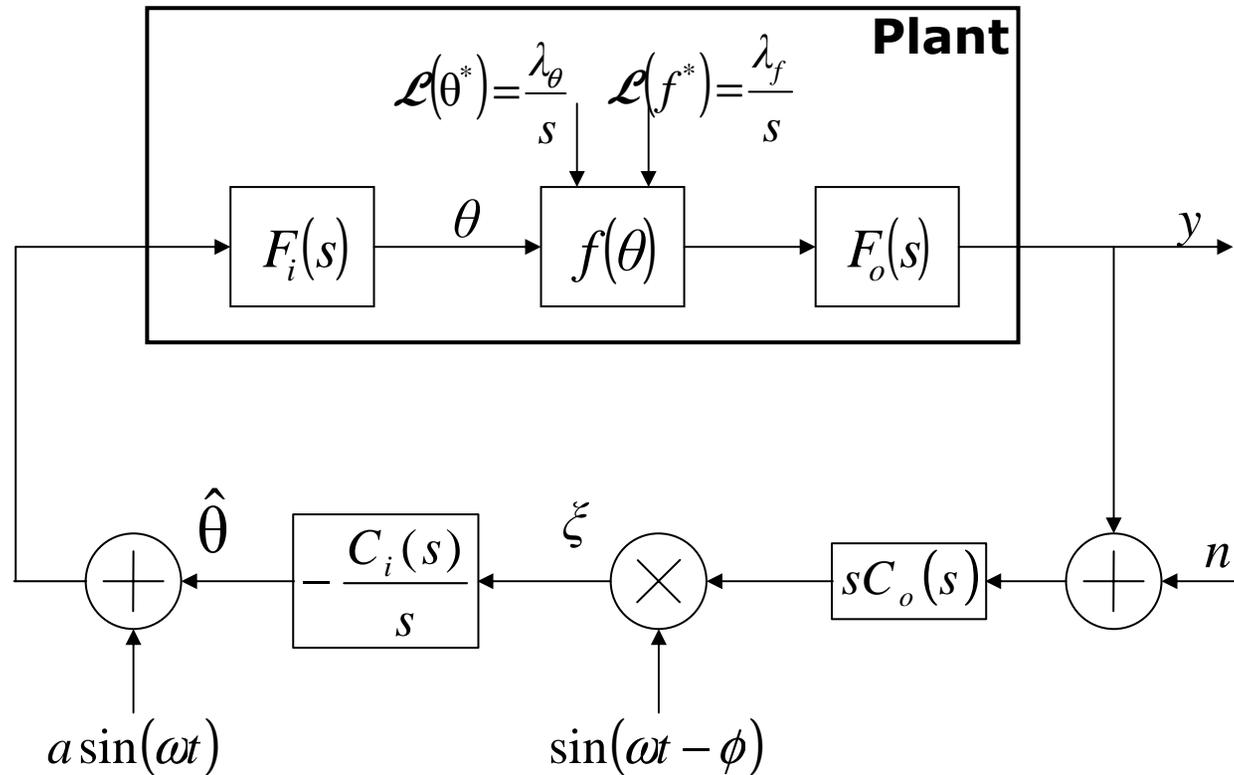
$K = 200$: graceful convergence to minimum (slight overshoot)

$K = 350$: gain too high – all the curve is explored



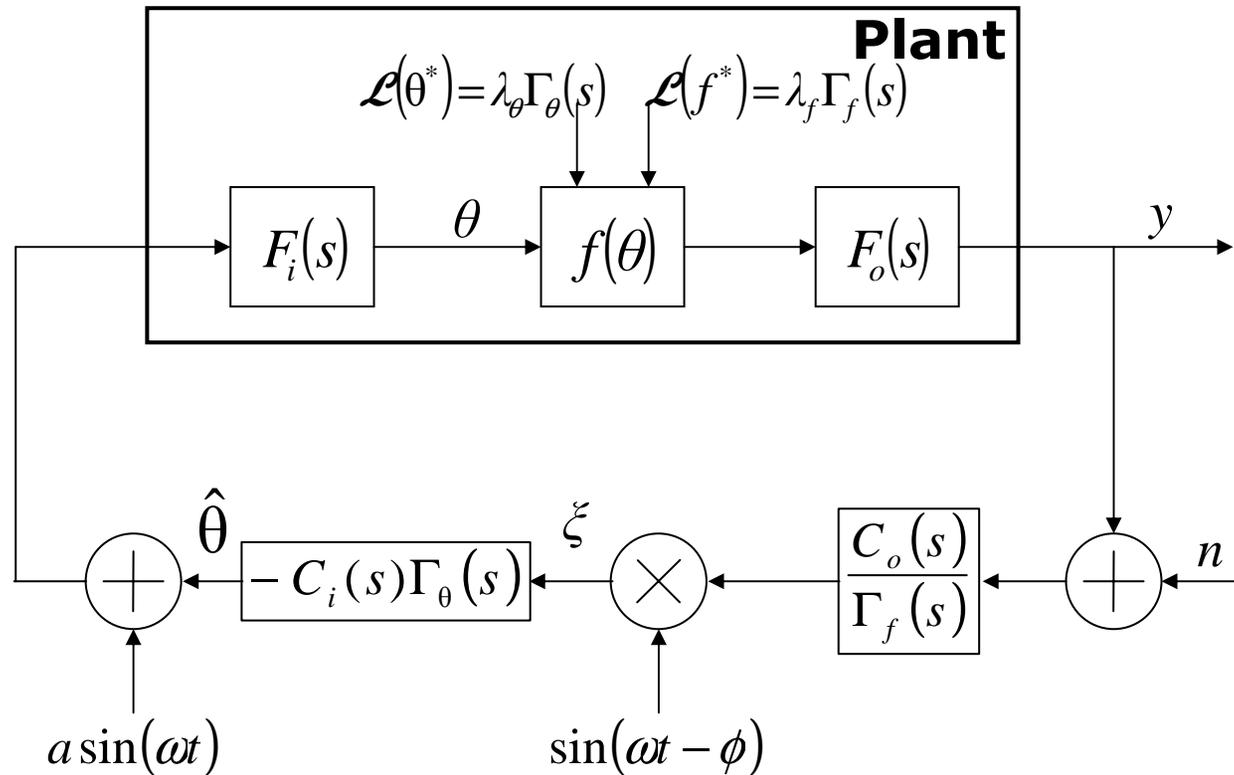
Extremum Seeking with Plant Dynamics and Parameter Tracking

Plant with Dynamics



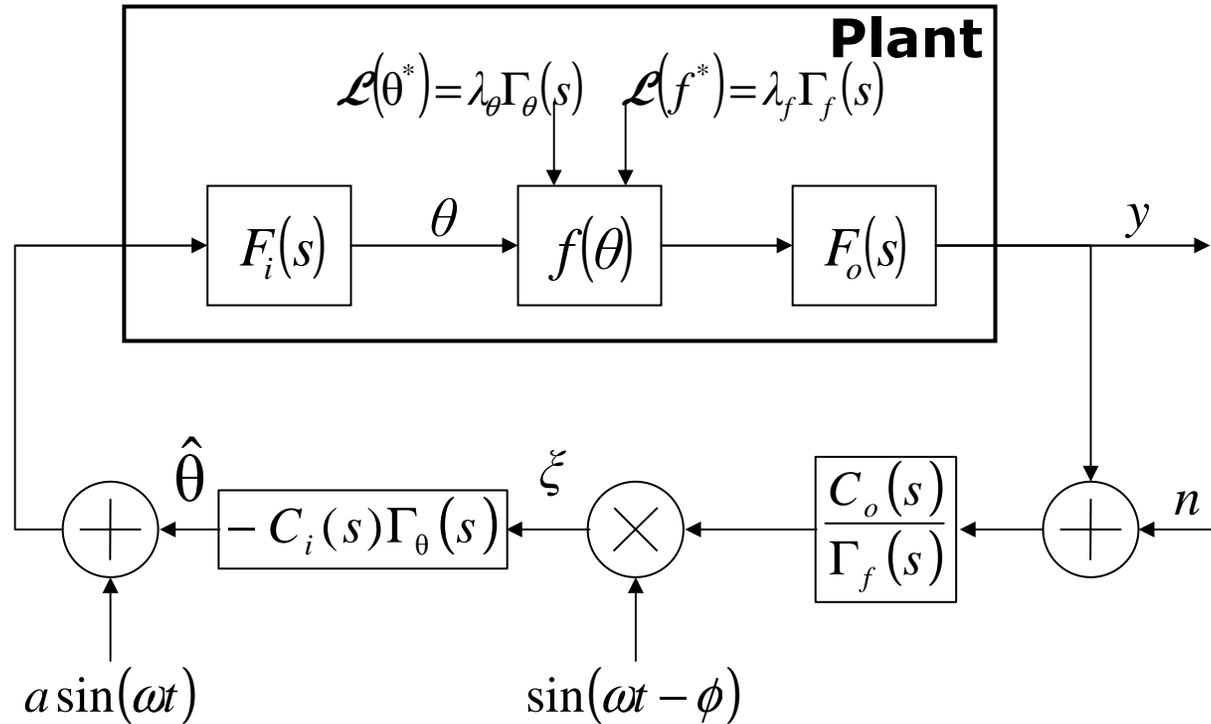
- $f(\theta) = f^* + \frac{f''}{2} (\theta - \theta^*)^2$
- $F_i(s)$ and $F_o(s)$ are asymptotically stable and proper
- $sC_o(s)$ and $\frac{C_i(s)}{s}$ are proper

Parameter Tracking



1. $f(\theta) = f^*(t) + \frac{f''}{2} (\theta - \theta^*(t))^2$
2. $\Gamma_\theta(s)$ and $\Gamma_f(s)$ are strictly proper rational functions
3. $\frac{C_o(s)}{\Gamma_f(s)}$ and $C_i(s)\Gamma_\theta(s)$ are proper

Example



$$\text{Plant : } F_i(s) = \frac{s-1}{s^2+3s+2}, F_o(s) = \frac{1}{s+1}, f(\theta) = f^* + (\theta - \theta^*)^2$$

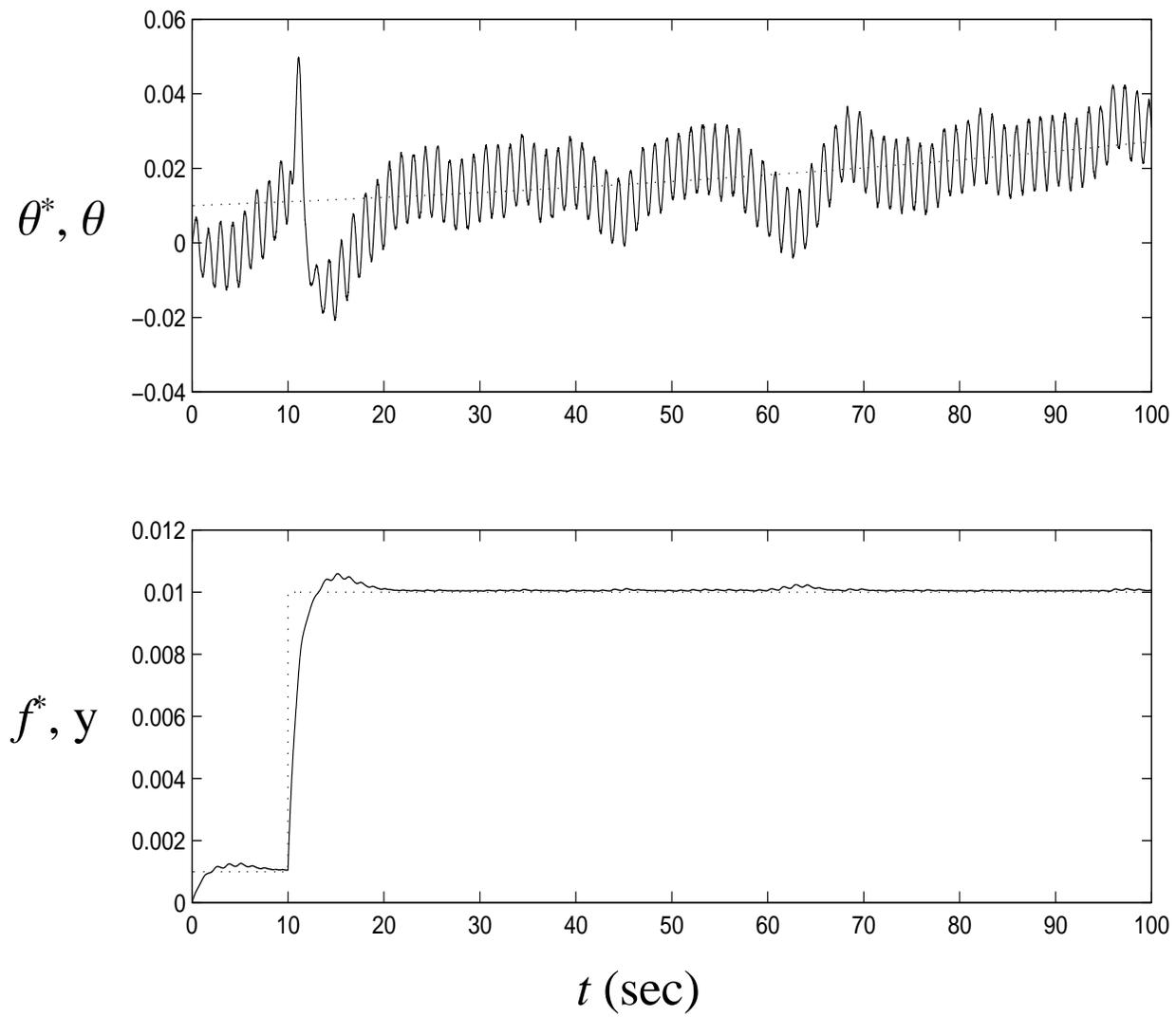
$$f^* = 0.01u(t - \tau), \theta^* = 0.01e^{0.01t}, \tau = 10 \text{ sec}$$

Design :

$$\omega = 5 \text{ rad/ sec}, a = 0.05, C_o(s) = \frac{1}{s+5}, \phi = 0.7955, C_i(s) = s - 4$$

$$k = 107.7$$

Simulation Results

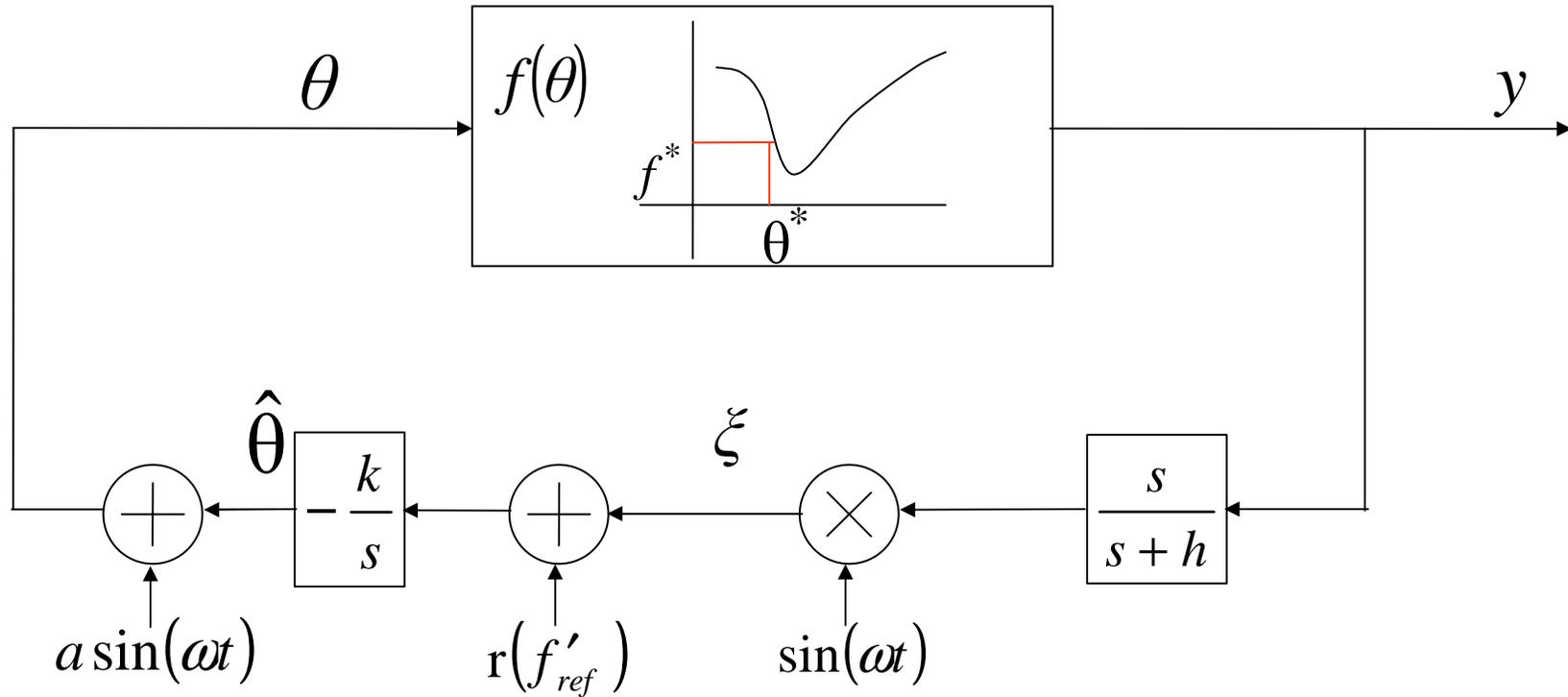


Slope Seeking

Why Slope Seeking?

- Extremum of plant reference-to-output map susceptible to destabilization:
 - Compressor instability
 - Antiskid Braking
 - Formation flight
- Need to operate at a particular slope of plant operating characteristic
 - Nuclear fusion

Slope Seeking on a Static Map

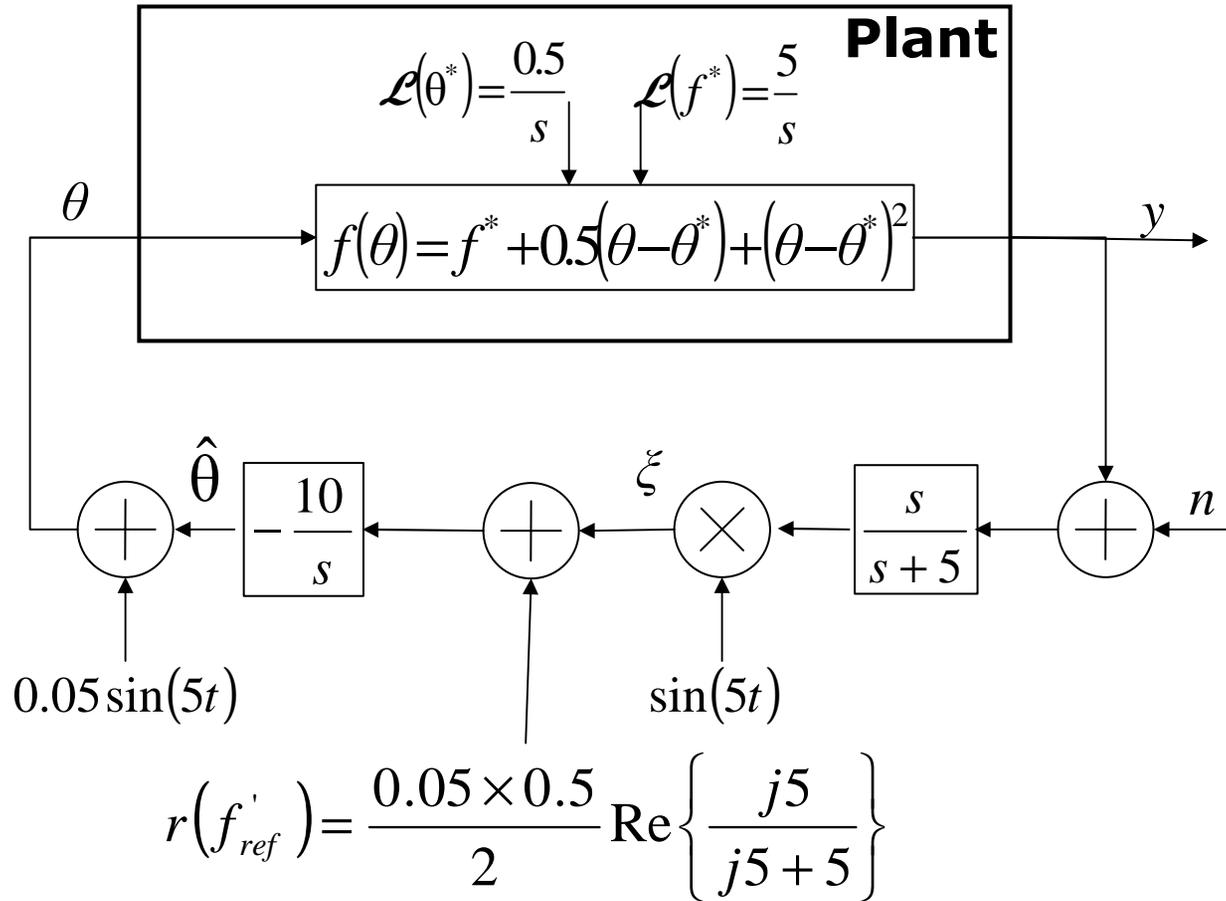


Stability Test:

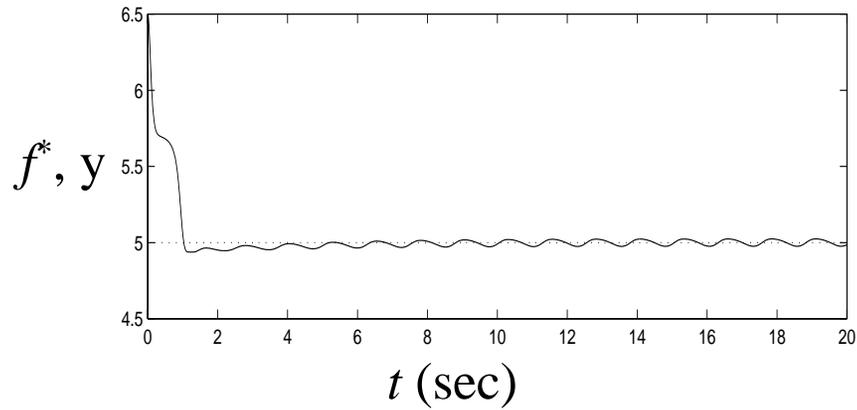
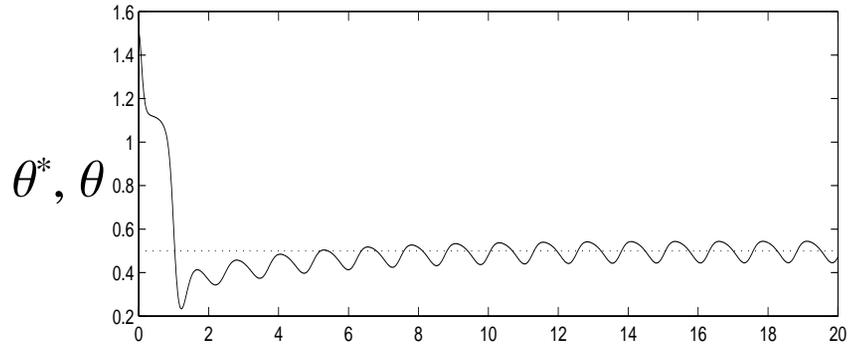
y converges to an $f^* + O(a + 1/\omega)$ if $\frac{1}{1+L(s)}$ is a.s.,

$$L(s) = \frac{kaf''}{2s}, \text{ and } r(f'_{ref}) = -\frac{af'_{ref}}{2} \operatorname{Re} \left\{ \frac{j\omega}{j\omega + h} \right\}$$

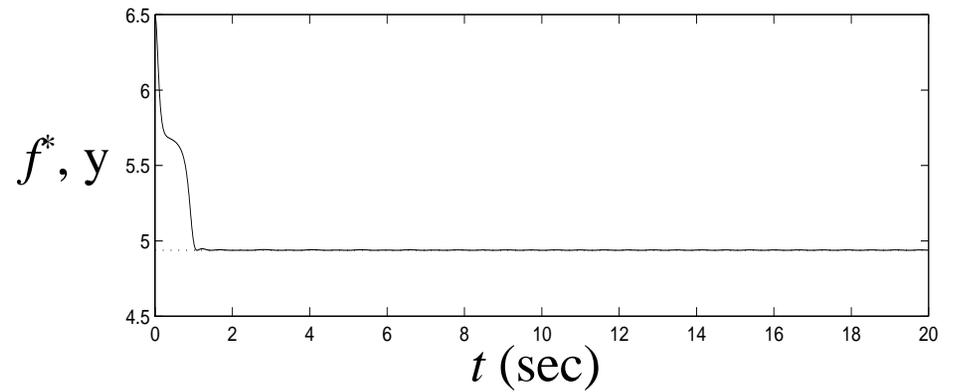
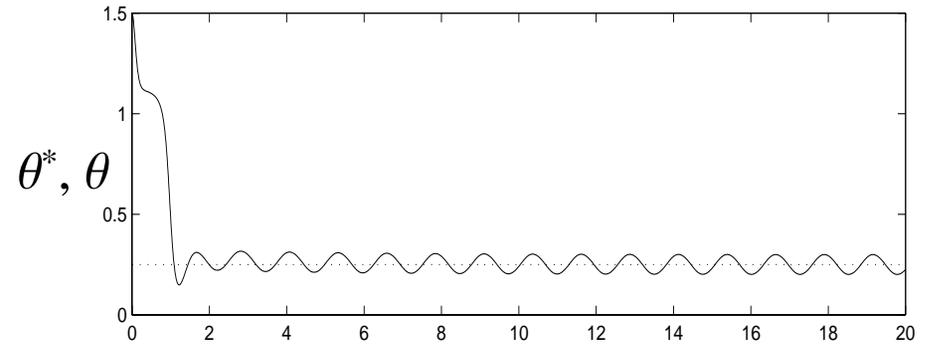
Example



Simulation

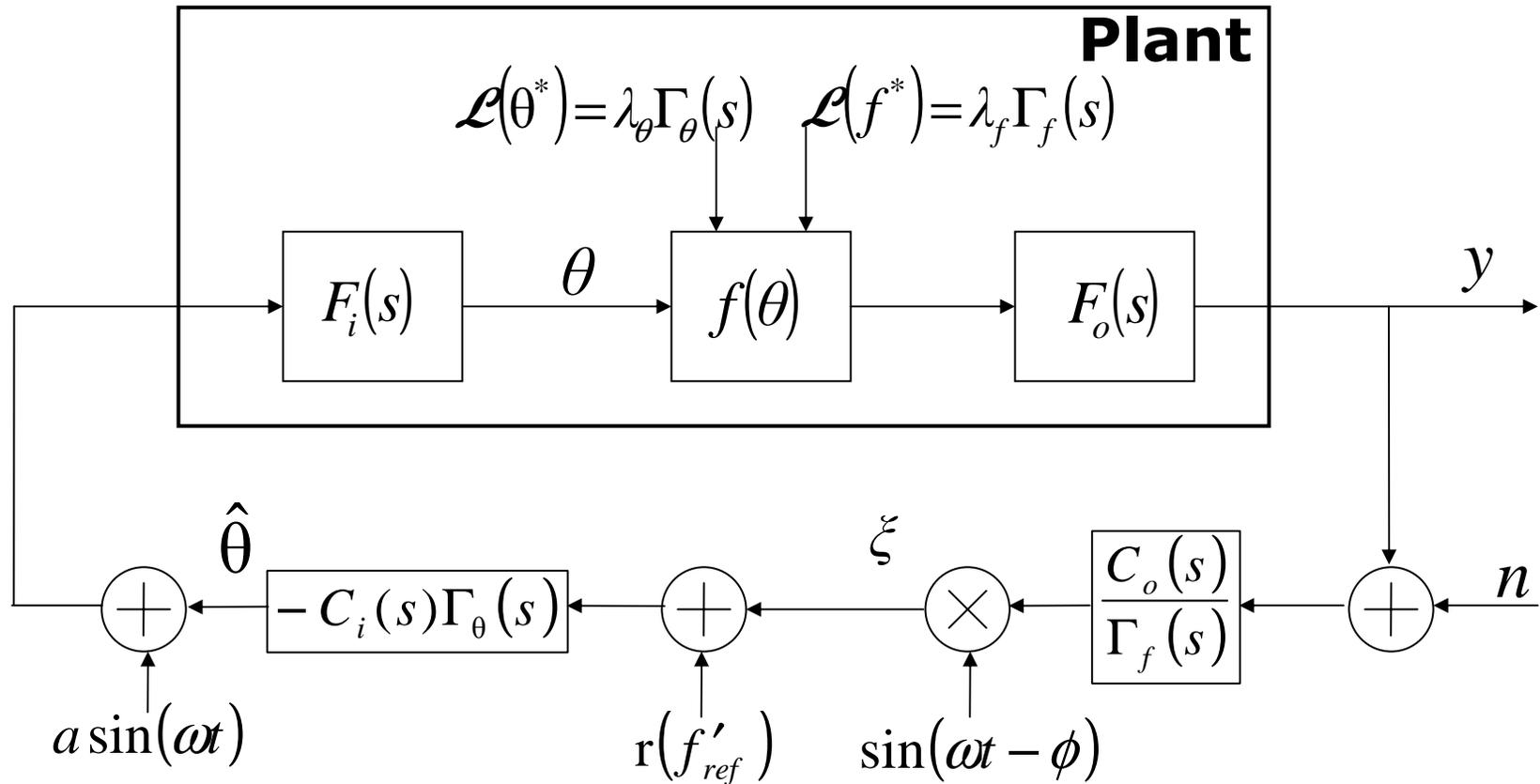


Slope seeking: $r(f'_{ref})=0.5$



Extremum seeking: $r(f'_{ref})=0$

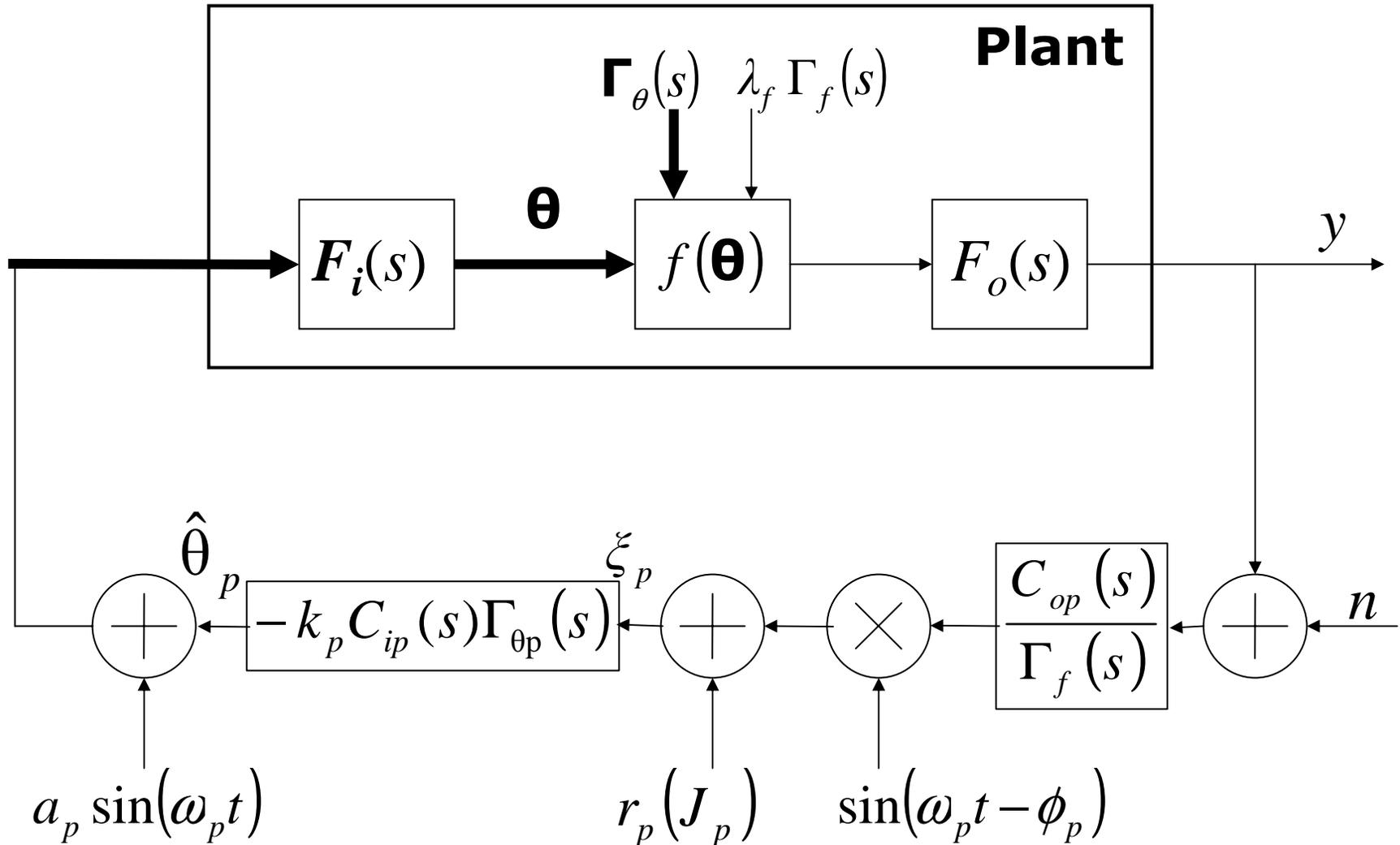
Generalized Slope Seeking



Slope setting :

$$r(f'_{ref}) = \frac{af'_{ref}}{2} \operatorname{Re} \left\{ e^{j\phi} H_o(j\omega) F_i(j\omega) \right\}$$

Gradient Seeking



Near-Optimal Compressor Operation via Slope Seeking

The Moore-Greitzer Model

$$\dot{R} = \sigma R \mathcal{F}(R, \Phi); \text{ where } \mathcal{F}(R, \Phi) = \frac{1}{3\pi\sqrt{R}} \int_0^{2\pi} \Psi_c(\Phi + 2\sqrt{R} \sin \theta) \sin \theta d\theta$$

$$\dot{\Phi} = -\Psi + \mathcal{G}(R, \Phi); \text{ where } \mathcal{G}(R, \Phi) = \frac{1}{2\pi} \int_0^{2\pi} \Psi_c(\Phi + 2\sqrt{R} \sin \theta) d\theta$$

$$\dot{\Psi} = \frac{1}{\beta^2} (\Phi - \Phi_T)$$

$\Psi_c(\Phi)$: compressor characteristic,

$$\Psi = \frac{1}{\gamma^2} (1 + \Phi_{c0} + \Phi_T)^2 : \text{throttle characteristic,}$$

γ : throttle opening

Notation in the Moore-Greitzer Model

$$\Phi = \hat{\Phi} / W - 1 - \Phi_{c0}$$

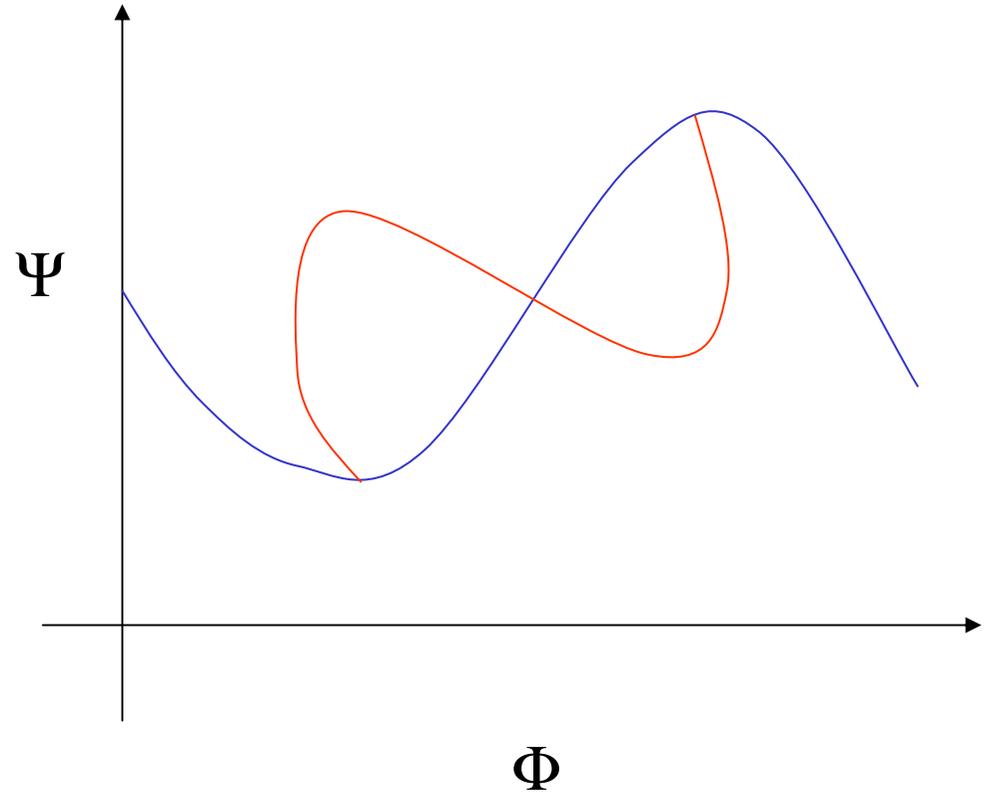
$$\Phi_T = \dot{m}_T / W - 1$$

$$\Psi = \hat{\Psi} / H$$

$$A = \hat{A} / W, R = (A/2)^2$$

$$\beta = \frac{2H}{W} B; \sigma = \frac{3l_c}{m + \mu}$$

$$t = \frac{H}{Wl_c} \hat{t}; \hat{t} = \Omega\tau$$



The ε -MG3 Parametrization

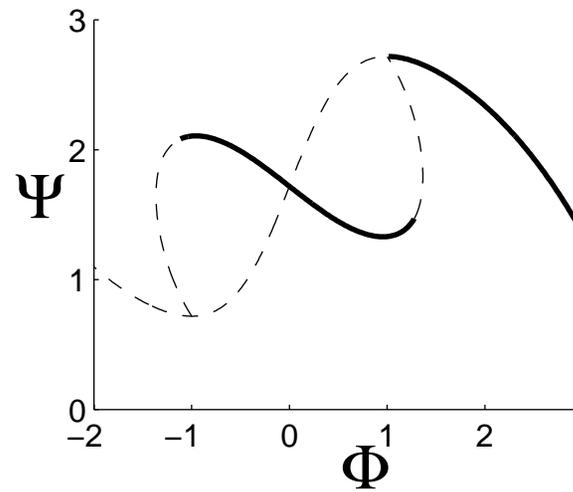
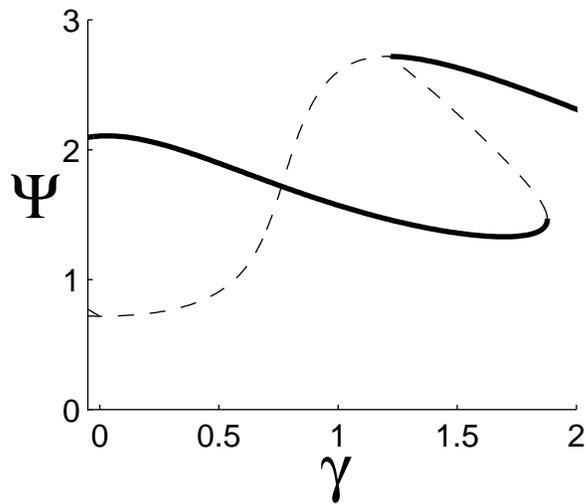
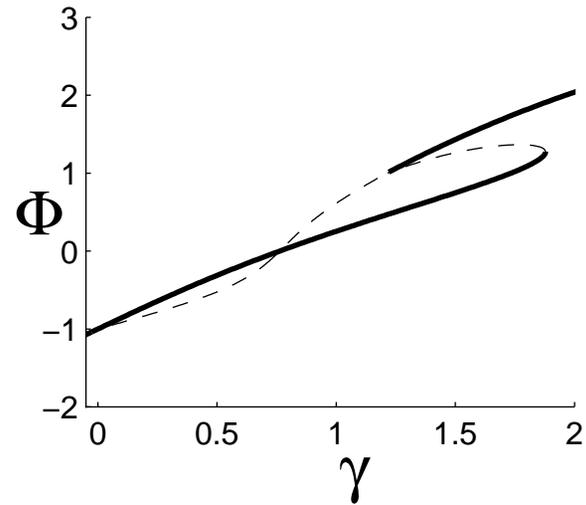
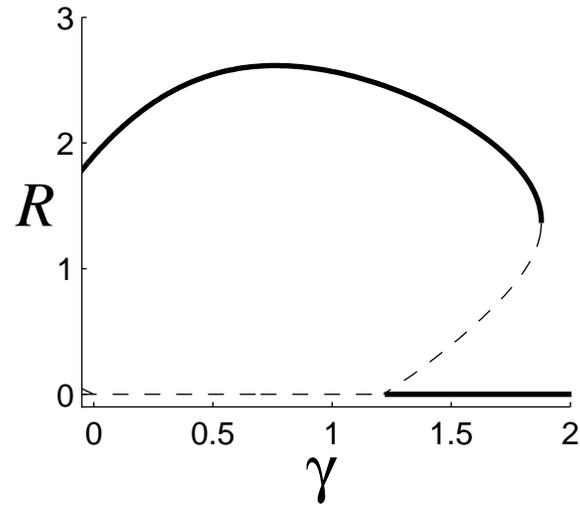
Using $\Psi_C(\Phi) = \Psi_{C0} + 1 + (1 - \varepsilon) \left(\frac{3}{2} \Phi - \frac{1}{2} \Phi^3 \right) + \varepsilon \frac{2\Phi}{1 + \Phi^2}$ gives

$$\dot{R} = \sigma \left\{ (1 - \varepsilon) R (1 - \Phi^2 - R) + \frac{2\varepsilon}{3} \left[1 - \frac{1}{\sqrt{2} \left[(\Phi^2 - 4R - 1)^2 + 4\Phi^2 \right]^{1/2}} \right. \right. \\ \left. \left. \times \left(\left((\Phi^2 - 1)(\Phi^2 - 4R - 1) + 4\Phi^2 \right)^2 + 64\Phi^2 R^2 \right)^{1/2} \right. \right. \\ \left. \left. + (\Phi^2 - 1)(\Phi^2 - 4R - 1) + 4\Phi^2 \right]^{1/2} \right\}$$

$$\dot{\Phi} = -\Psi + \Psi_{C0} + 1 + (1 - \varepsilon) \left(\frac{3}{2} \Phi - \frac{1}{2} \Phi^3 - 3\Phi R \right) \\ + \varepsilon \frac{\sqrt{2} \operatorname{sgn}(\Phi)}{\left[(\Phi^2 - 4R - 1)^2 + 4\Phi^2 \right]^{1/2}} \left\{ \left[(\Phi^2 - 4R - 1)^2 + 4\Phi^2 \right]^{1/2} + (\Phi^2 - 4R - 1) \right\}^{1/2}$$

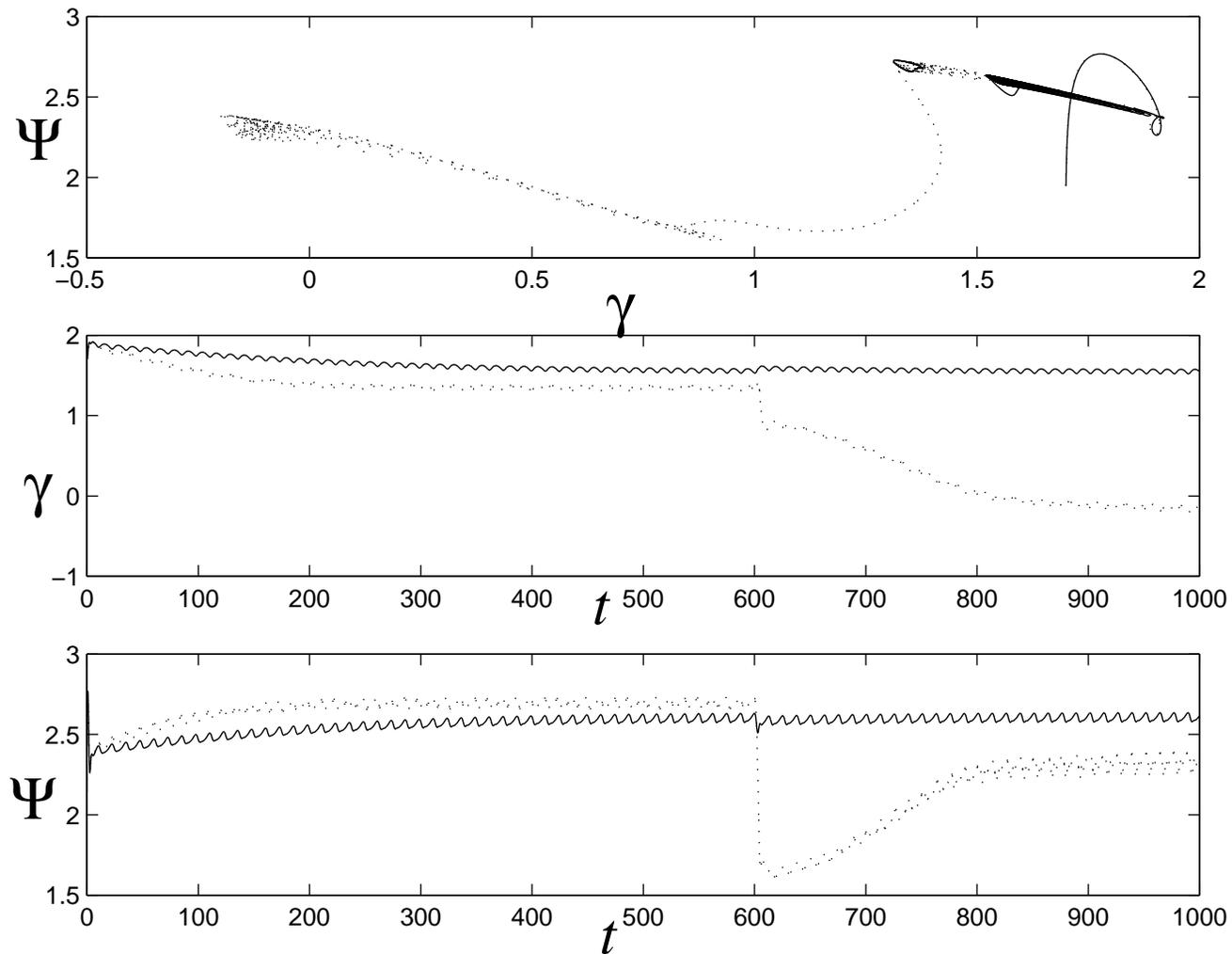
$$\dot{\Psi} = \frac{1}{\beta^2} (\Phi - \Phi_T)$$

Equilibria and Bifurcation Diagrams



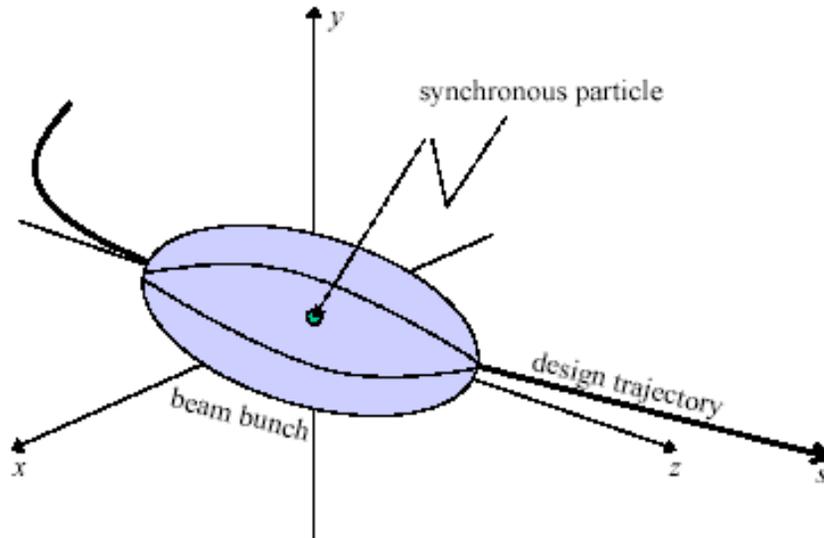
Deep hysteresis: $\varepsilon=0.9$

Compressor Simulation



Deep hysteresis: $\varepsilon=0.9$

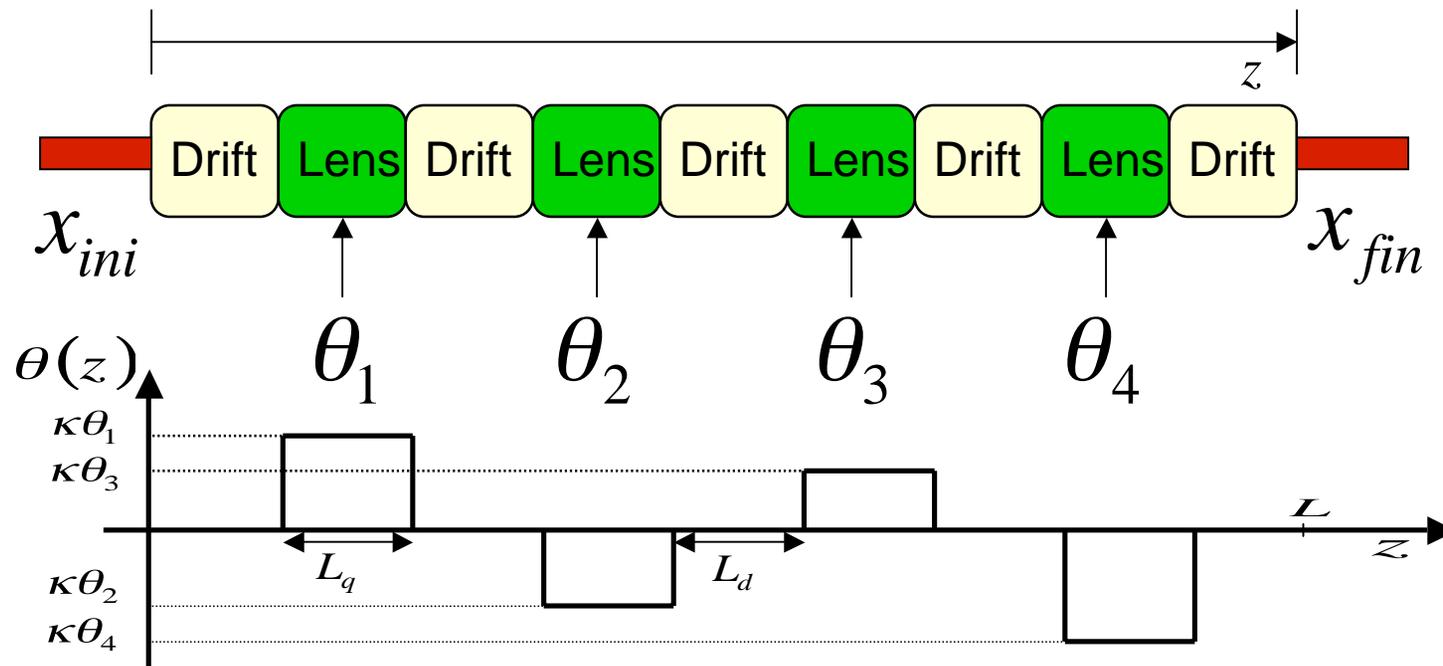
PROBLEM - BEAM MATCHING CHANNEL



$$x_{ini} = \begin{bmatrix} X_{ini} \\ X'_{ini} \\ Y_{ini} \\ Y'_{ini} \end{bmatrix}, x_{fin} = \begin{bmatrix} X_{fin} \\ X'_{fin} \\ Y_{fin} \\ Y'_{fin} \end{bmatrix}, x_{tar} = \begin{bmatrix} X_{tar} \\ X'_{tar} \\ Y_{tar} \\ Y'_{tar} \end{bmatrix}$$

Goal: Tune θ in order to minimize matching error J

Constraint: No model is available
Real time minimization



SOLUTION – EXTREMUM SEEKING

$$X'' - \theta(z)X - \frac{2K}{X+Y} - \frac{\varepsilon_X^2}{X^3} = 0$$

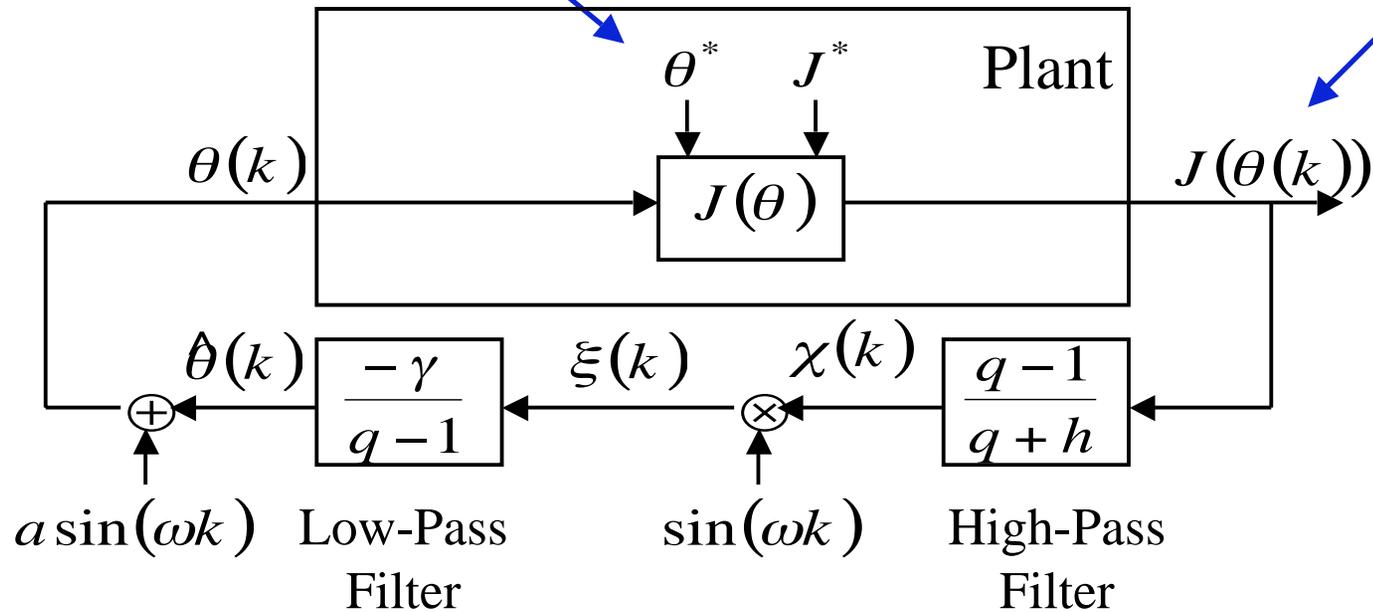
$$Y'' + \theta(z)Y - \frac{2K}{X+Y} - \frac{\varepsilon_Y^2}{Y^3} = 0$$

$$J = k_1 J_1 + k_2 J_2 + k_3 J_3$$

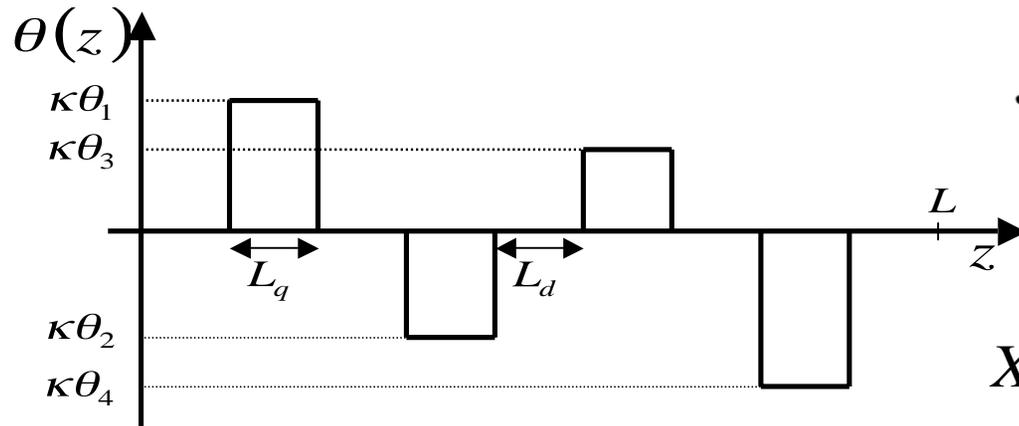
$$J_1 = K_X (X_{fin} - X_{tar})^2 + K_Y (Y_{fin} - Y_{tar})^2$$

$$J_2 = K_{dX} (X'_{fin} - X'_{tar})^2 + K_{dY} (Y'_{fin} - Y'_{tar})^2$$

$$J_3 = \int_0^L w(z) \left[K_{iX} (X(z) - X_{des}(z))^2 + K_{iY} (Y(z) - Y_{des}(z))^2 \right] dz$$

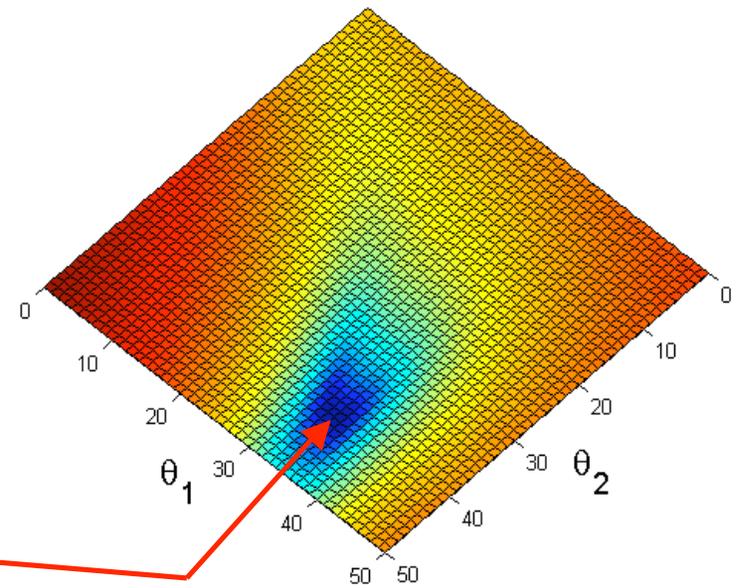
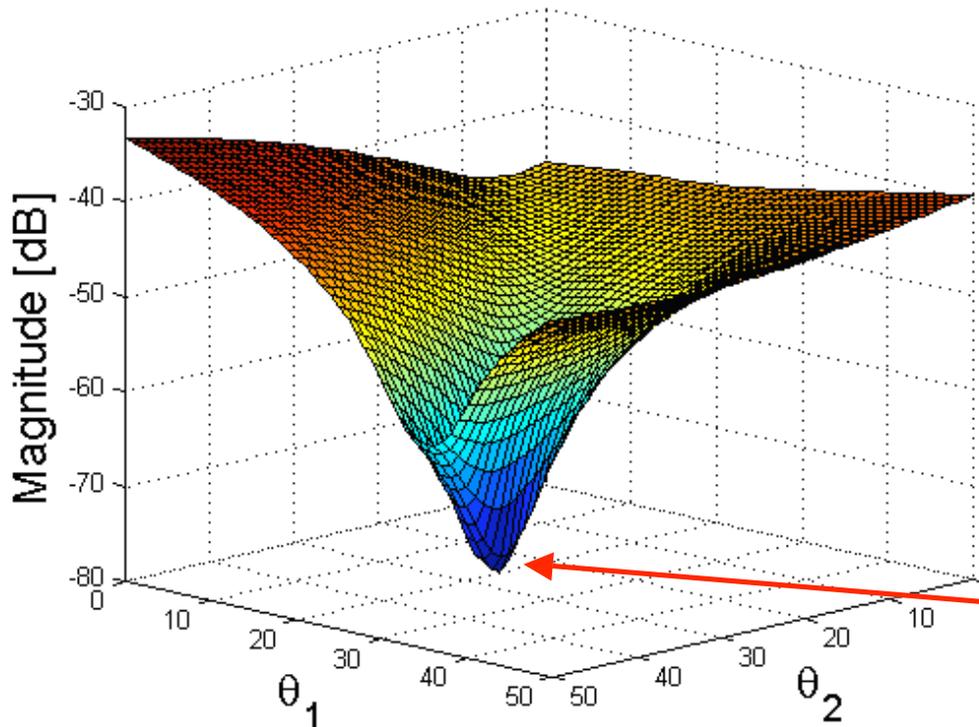


BEAM MATCHING OPTIMIZATION – 2D



$$J = \sqrt{(X_{fin} - X_{tar})^2 + (Y_{fin} - Y_{tar})^2}$$

$$X_{tar} = 0.001092, Y_{tar} = 0.002055$$



$$\theta_1 = 34, \theta_2 = 40$$

Extremum-Seeking Control of Flow Separation in a Planar Diffuser

Andrzej Banaszuk

United Technologies Research Center, E. Hartford, CT, U.S.A.

Acknowledgements:

Satish Narayanan, *UTRC: experiment*

Youping Zhang, *Numerical Technologies: algorithm*

Workshop on Real-Time Optimization by Extremum-Seeking Control, ACC 05

Partially sponsored by AFOSR



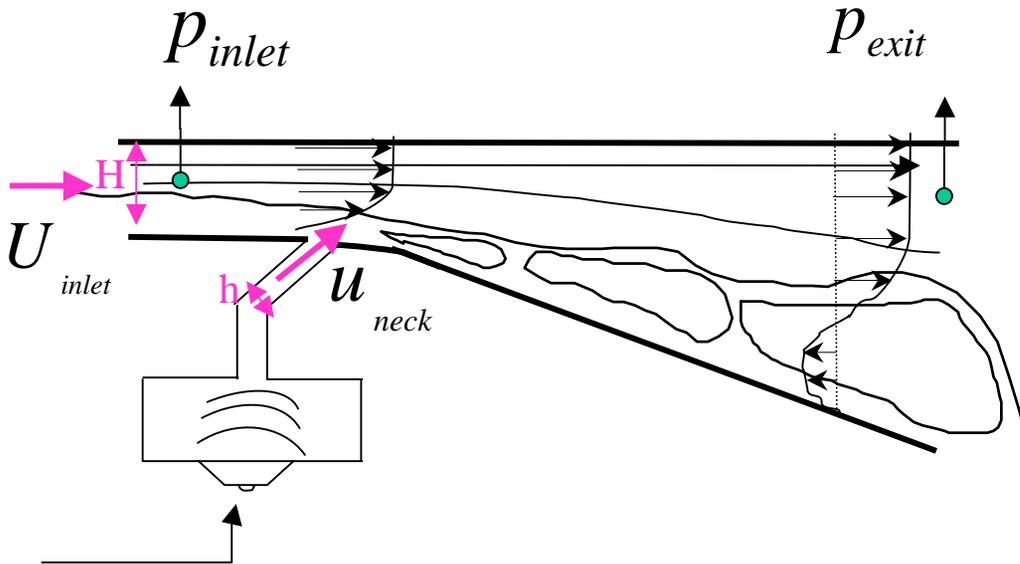
Objective of pressure recovery control

Performance

$$C_p(t) = \frac{P_{exit} - P_{inlet}}{\frac{1}{2} \rho U_{inlet}^2}$$

Control effort

$$C_\mu(t) = \frac{u_{neck}^2 h}{U_{inlet}^2 H}$$

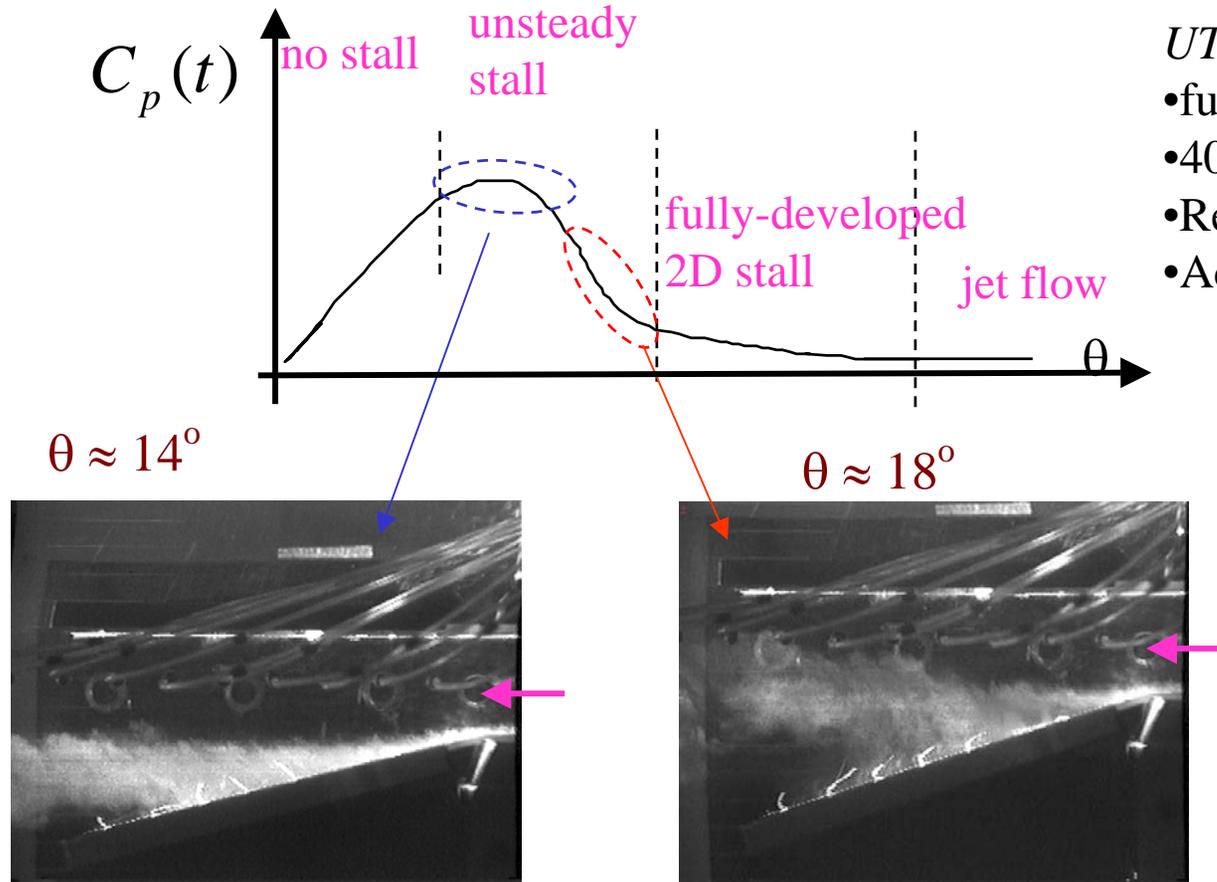


Speaker command

$$\frac{\overline{C_p}}{\overline{C_\mu}} = \frac{\text{bang}}{\text{buck}}$$

Experimental Setup

Pressure recovery as function of diffuser angle (no control)



UTRC diffuser rig

- fully turbulent BL
- $40,000 < Re_H < 140,000$
- $Re_{\theta e} > 300$; $M < 0.1$
- Actuation: $C_\mu \sim 0.001$

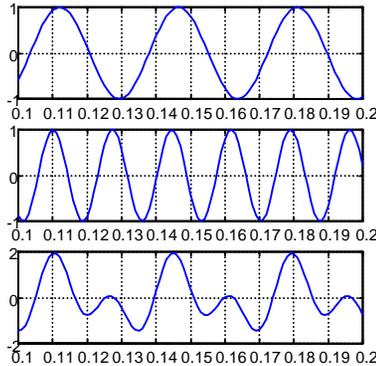
- Optimum uncontrolled performance
- Insignificant improvement with control

- Poor uncontrolled performance
- Significant improvement with control

Two frequency control creates “beneficial” vortex interaction

Control signal is $U(t) = A_1 \sin(2\pi f t) + A_2 \sin(2\pi 2f t - \theta)$

Construction of control waveform



$$A_1 \sin(2\pi f t)$$

+

$$A_2 \sin(2\pi 2f t - \theta)$$



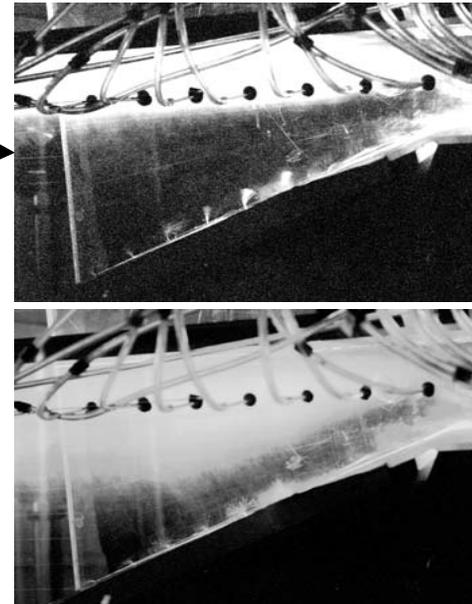
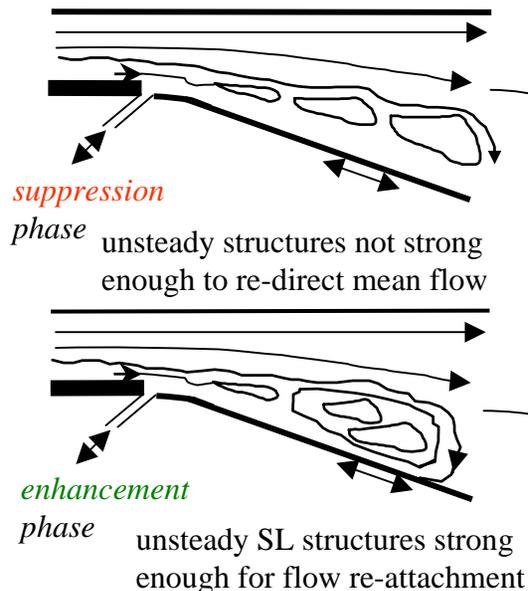
||

$$U(t)$$

$A_1 = A_2 = \text{const} \Rightarrow$ constant “power”

Adjustable parameters: f & θ

with appropriate choice of control phase one can **suppress** or **enhance** vortex interaction

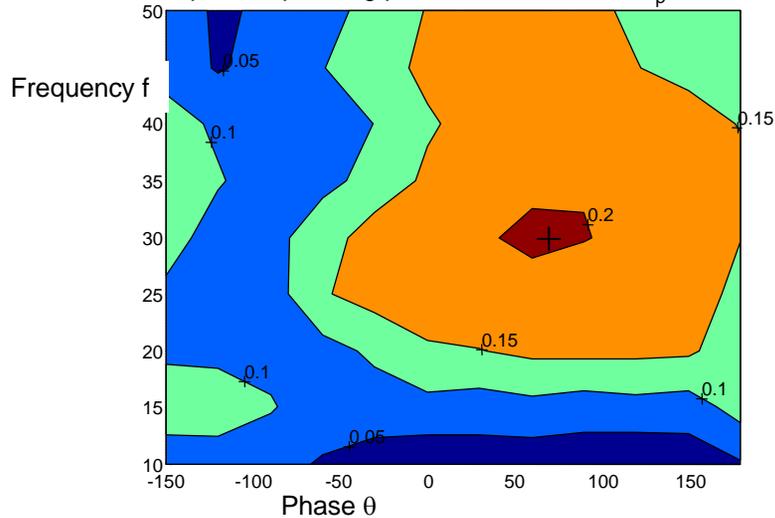


Need: control algorithm to optimize performance

Two frequency control law: $U(t) = A_1 * (\sin(2 * \pi * f * t) + \sin(2 * \pi * 2f * t - \theta))$

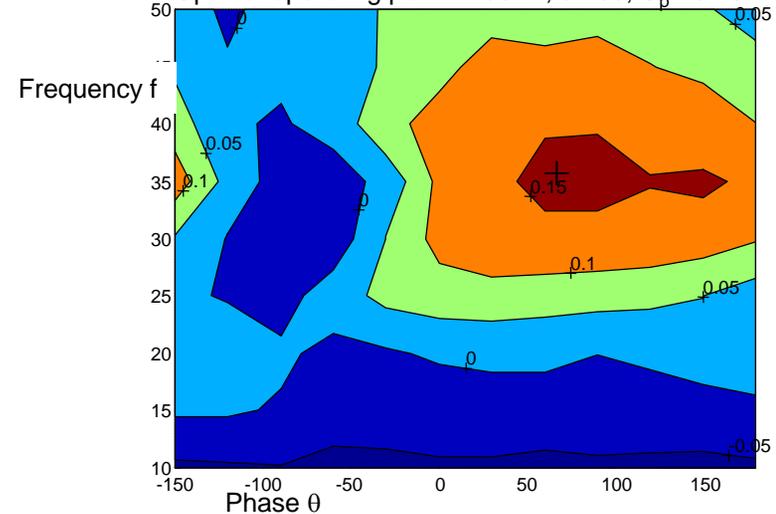
U = 20m/s

Optimal operating point: $f = 31\text{Hz}$, $\theta = 60$, $C_p = 0.21$



U = 30m/s

Optimal operating point: $f = 36\text{Hz}$, $\theta = 60$, $C_p = 0.16$



Objective:

- Optimize performance without exhaustive search

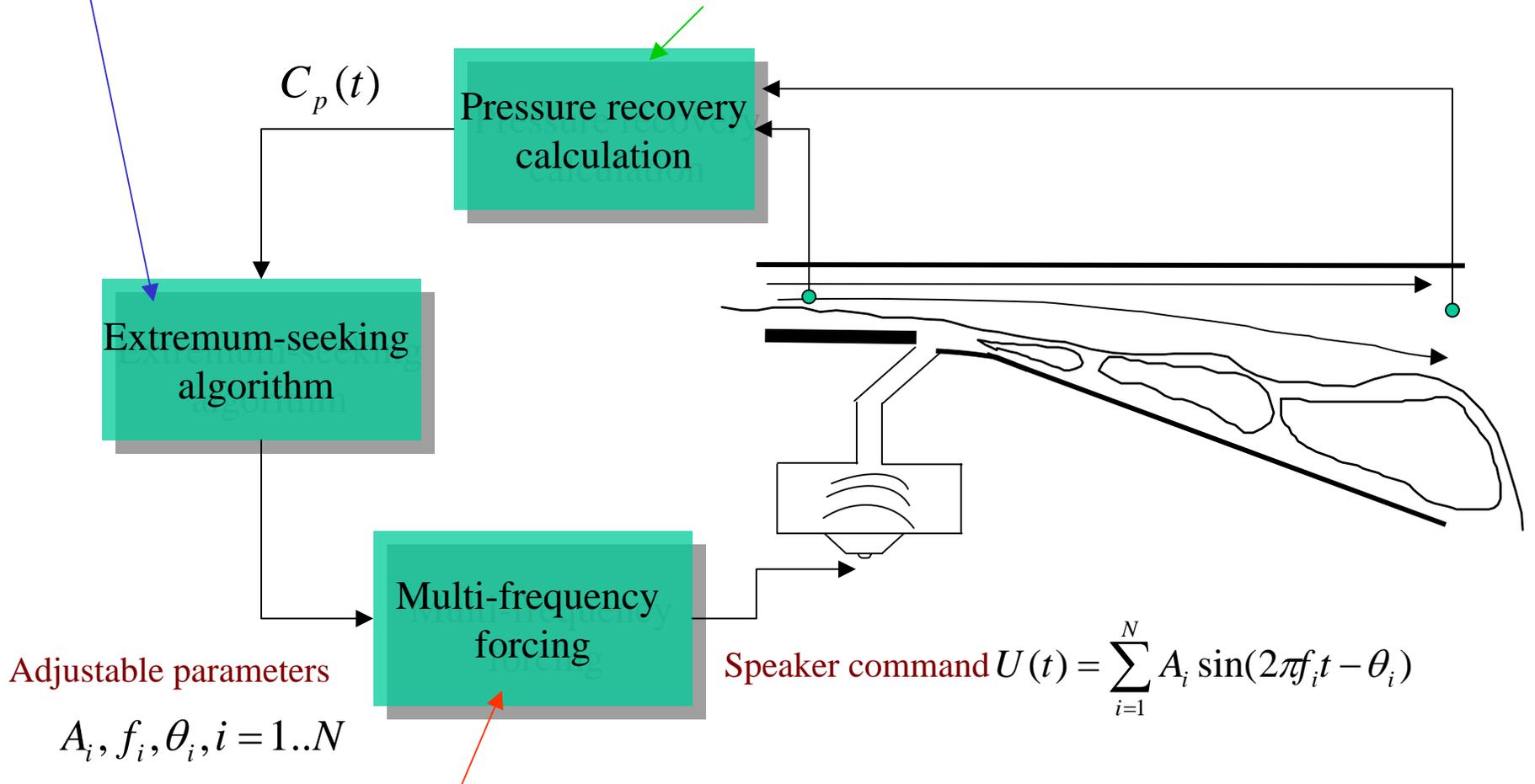
Challenges:

- Noisy measurement
- Flow transients
- Keeping up with operating condition change

Adaptive control used to optimize performance

Filter noise, wait for transient to settle, adapt parameters to improve performance

Measure performance

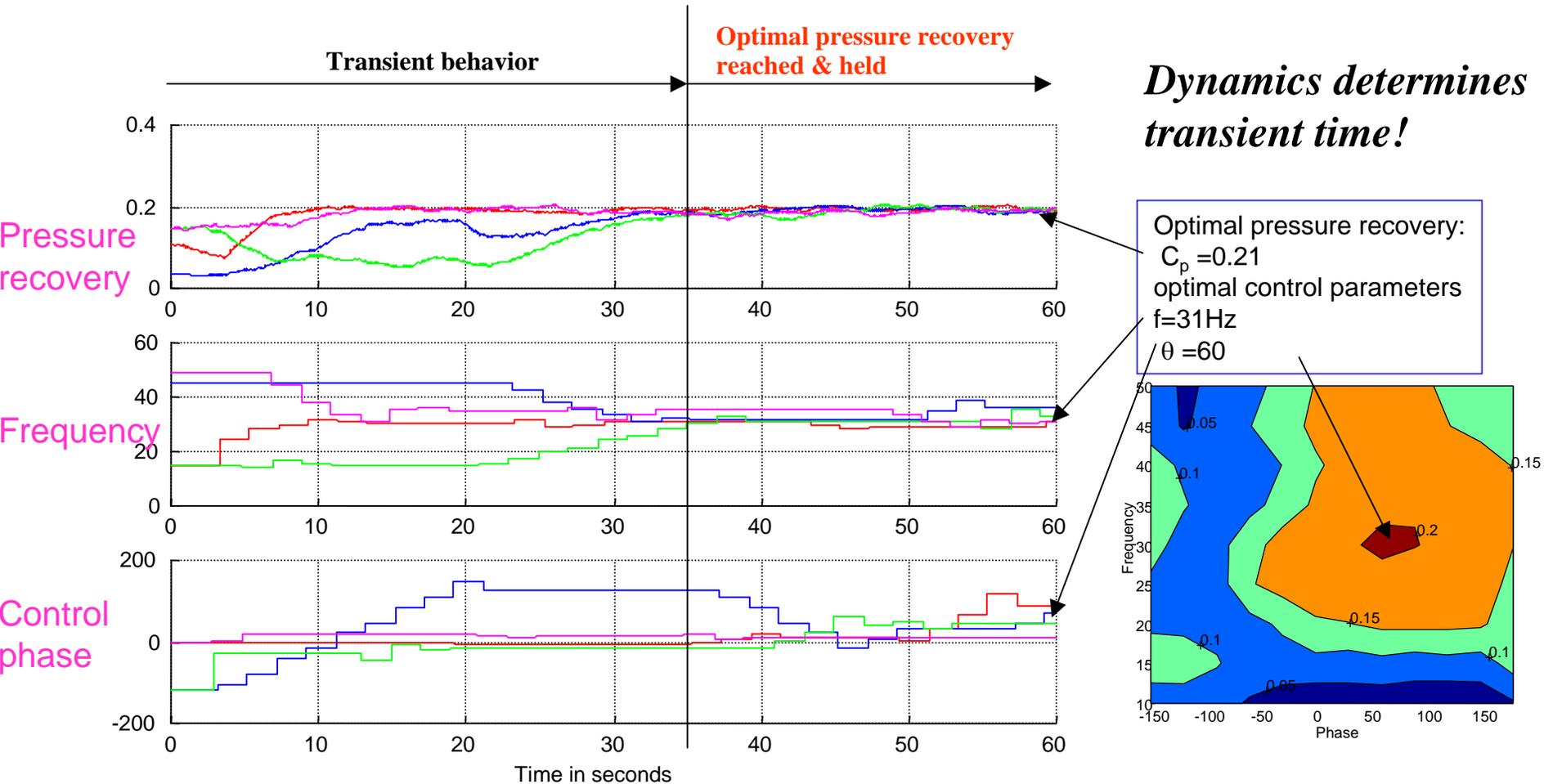


Excite multiple vortices, explore their interactions

$$C_\mu(t) = const$$

Automatic Control Parameter Tuning to Optimum Values

On-line optimization of pressure recovery using extremum-seeking algorithm demonstrated.



- Mean pressure recovery, control frequency, and phase in four independent adaptive control experiments.
- The control frequency and phase initialized away from the optimal values.

Automatic Parameter Tuning for Operating Condition Changes

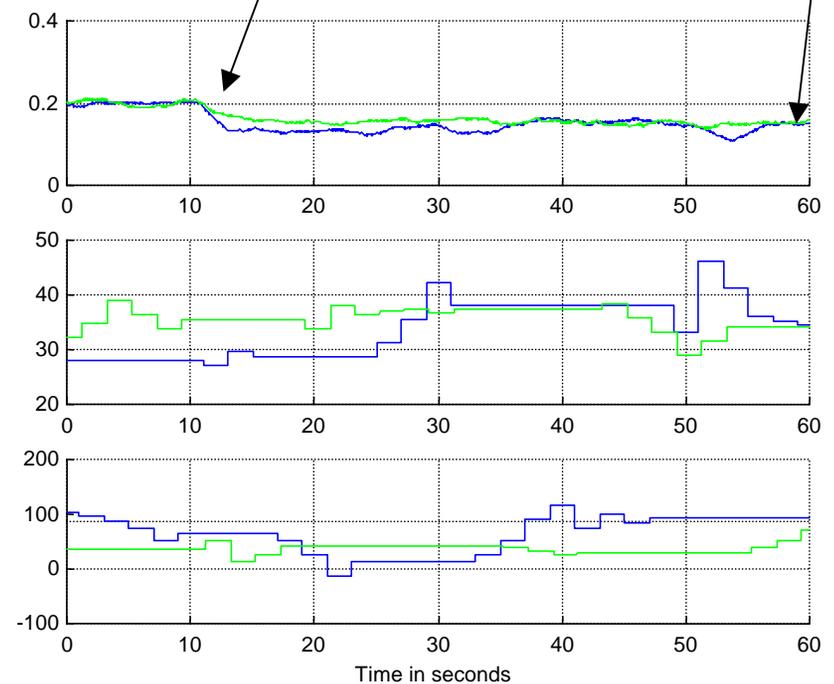
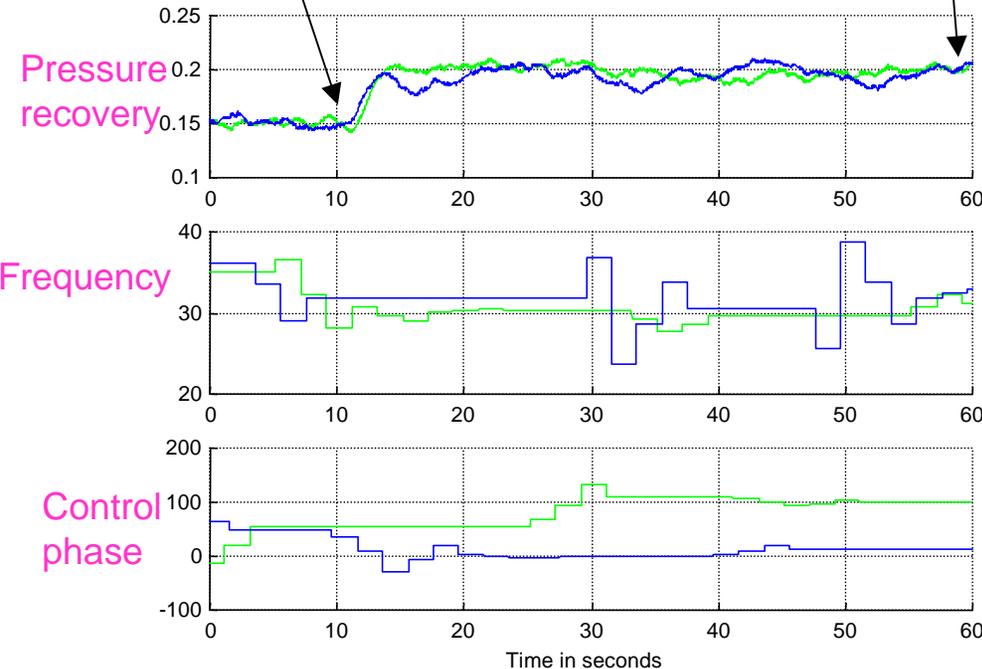
Adaptive algorithm tunes control frequency & phase during abrupt changes in operating conditions.

**Optimal pressure recovery
for lower velocity reached**

**Optimal pressure recovery
for higher velocity reached**

Air velocity decrease events
(overlay from two independent experiments)

Air velocity increase events
(overlay from two independent experiments)



Mean pressure recovery & control frequency & phase during abrupt changes in air velocity between 20m/sec & 30m/sec in two independent experiments.

Source Seeking Without Position Measurement

Based on contributions by: Jennie Cochran, Dan Arnold, Nima Ghods, Chunlei Zhang, Antranik Siranosian, and Chris Manzie

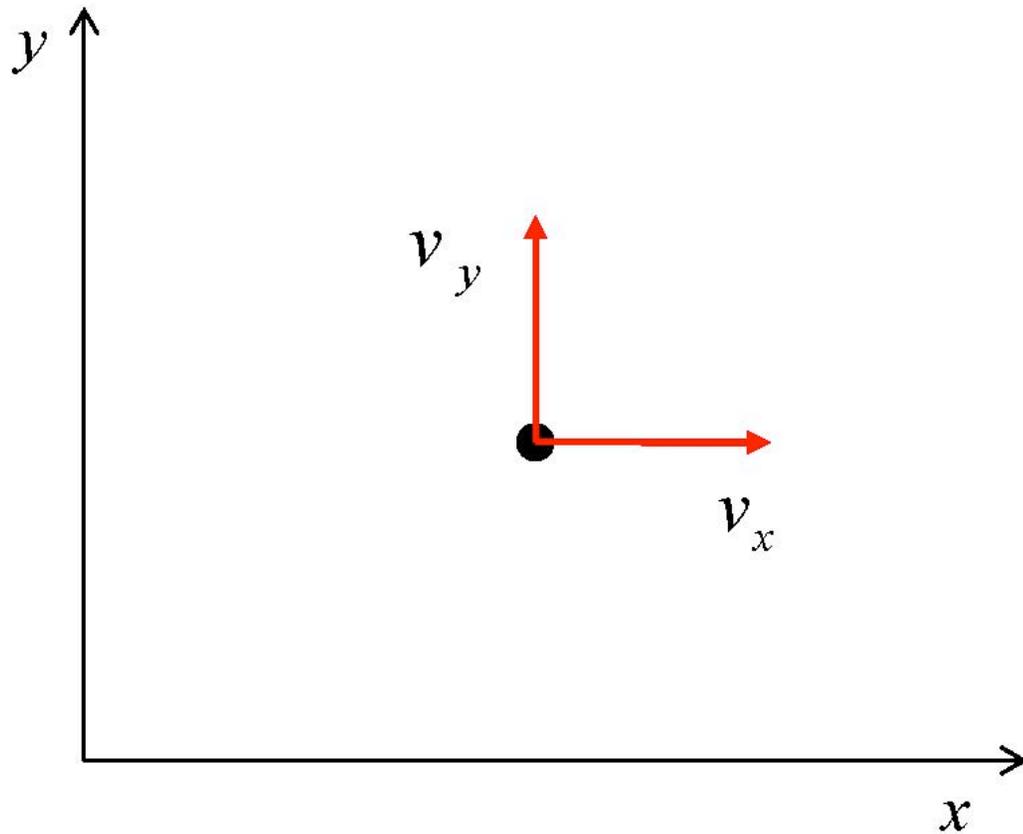
Introduction

- Motivation
 - Control a vehicle to locate the source of an unknown signal
- Previous Work
 - Porat and Nehorai - vehicle has position information
 - Ogren, Fiorelli and Leonard - “group” gradient estimation
 - Justh and Krishnaprasad - convergent vehicle formation
 - Klein and Morgansen - trajectory tracking
 - Marshal, Broucke and Francis - cyclic pursuit problem

Introductory Example: Point Mass

Model

Point Mass



Dynamics

$$\dot{x} = v_x$$

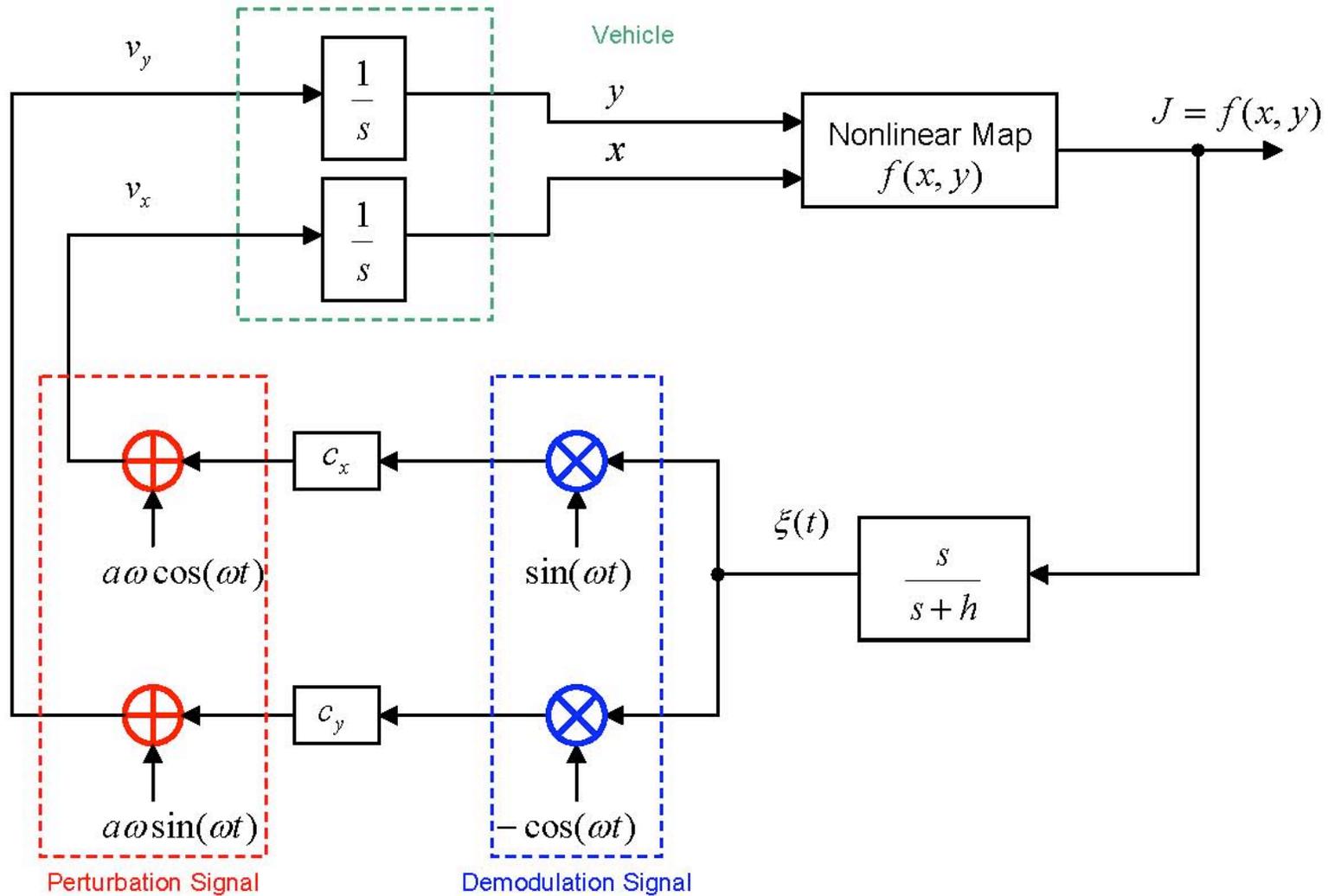
$$\dot{y} = v_y$$

Inputs

$$v_x, v_y$$

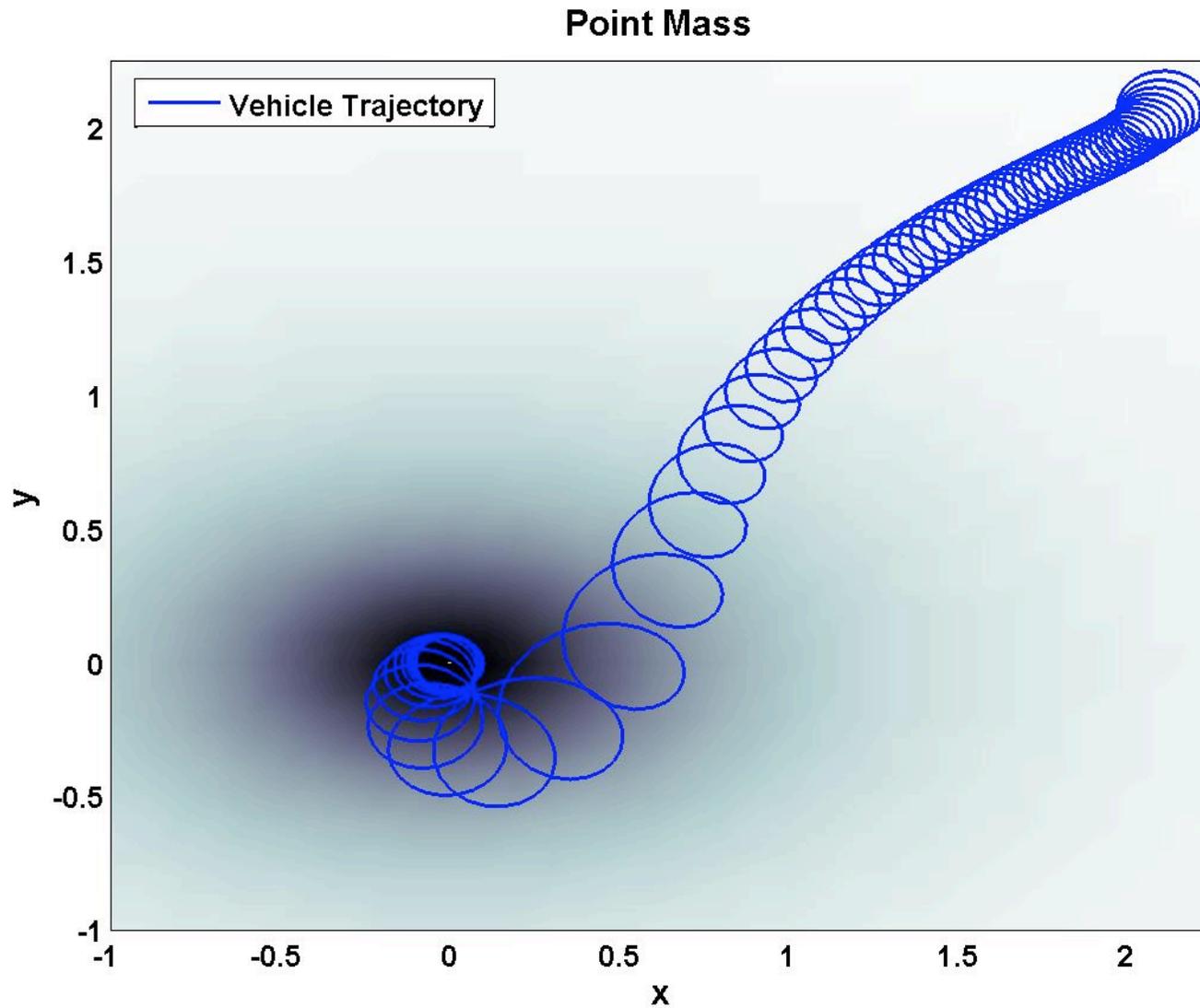
Block Diagram

Point Mass



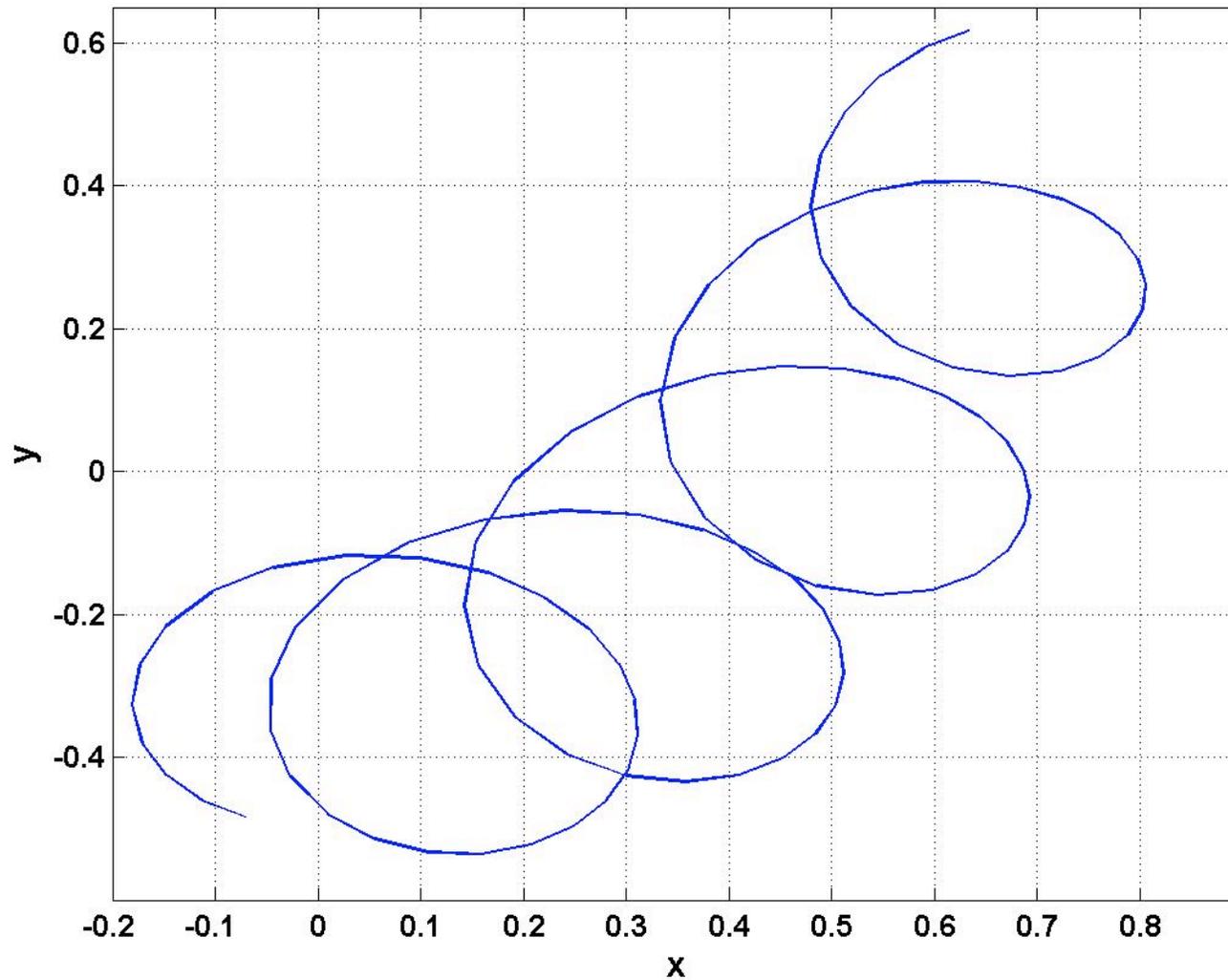
Simulation Results

Point Mass



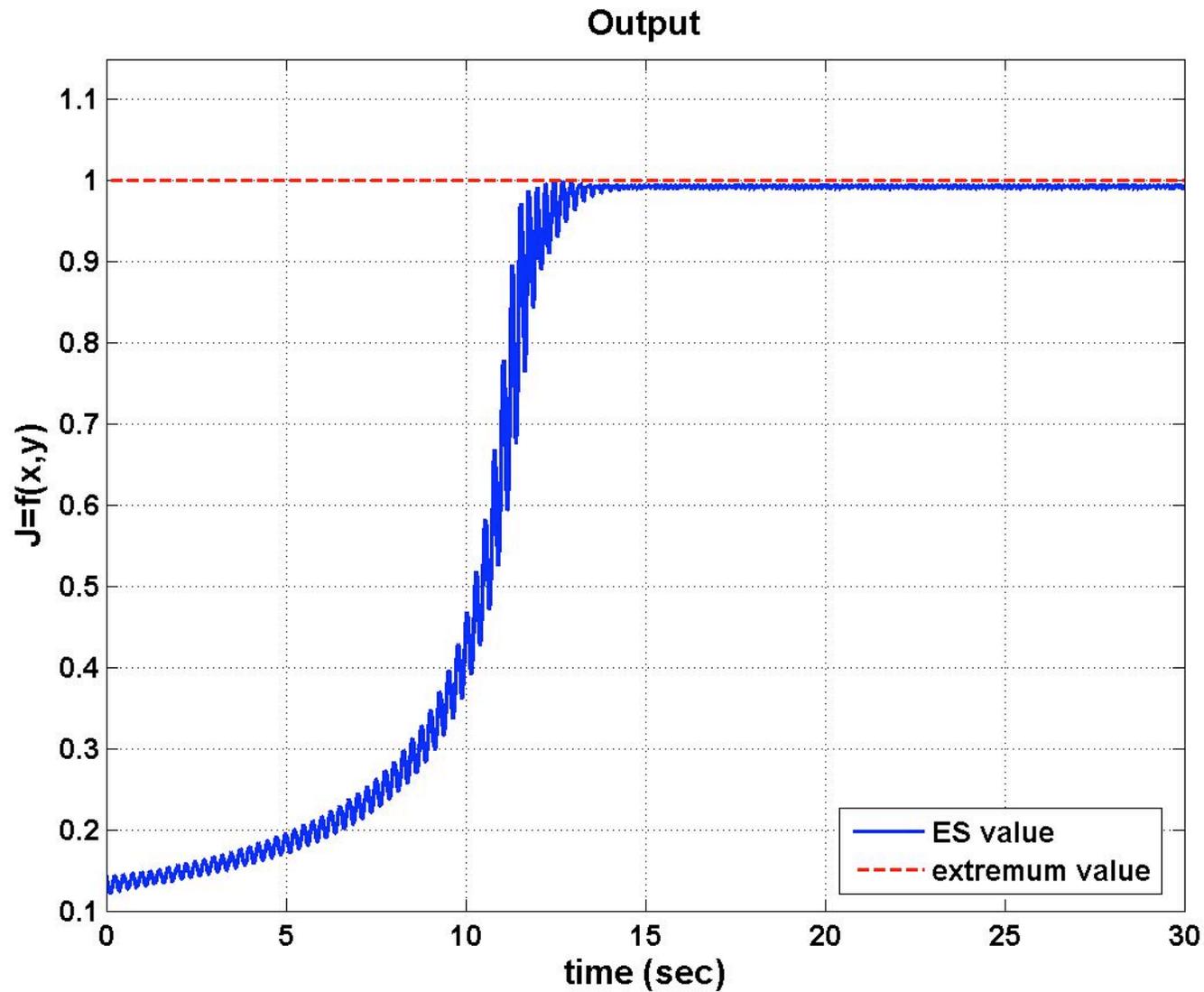
Simulation Results

Circular Pattern of Vehicle Movement

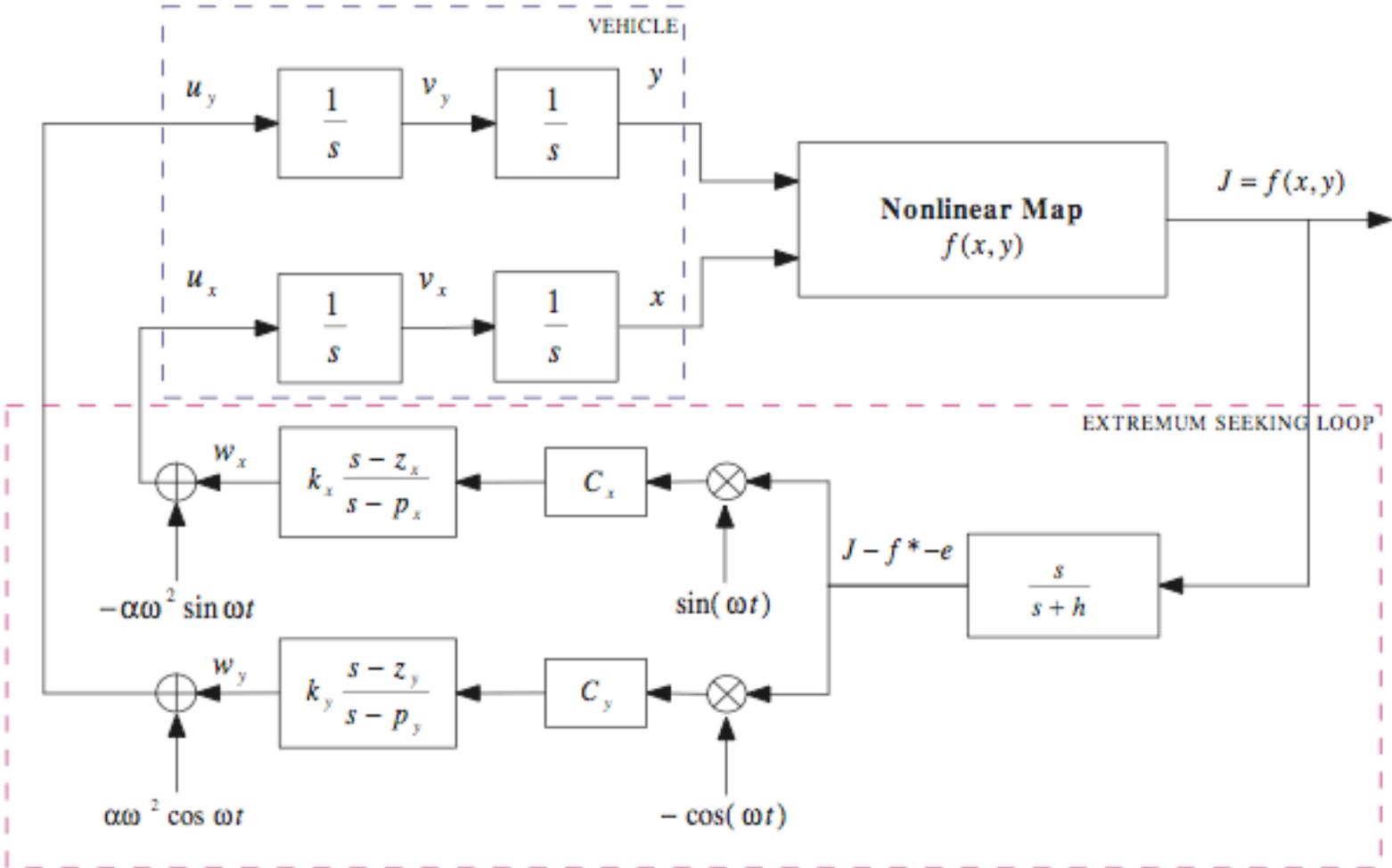


Simulation Results

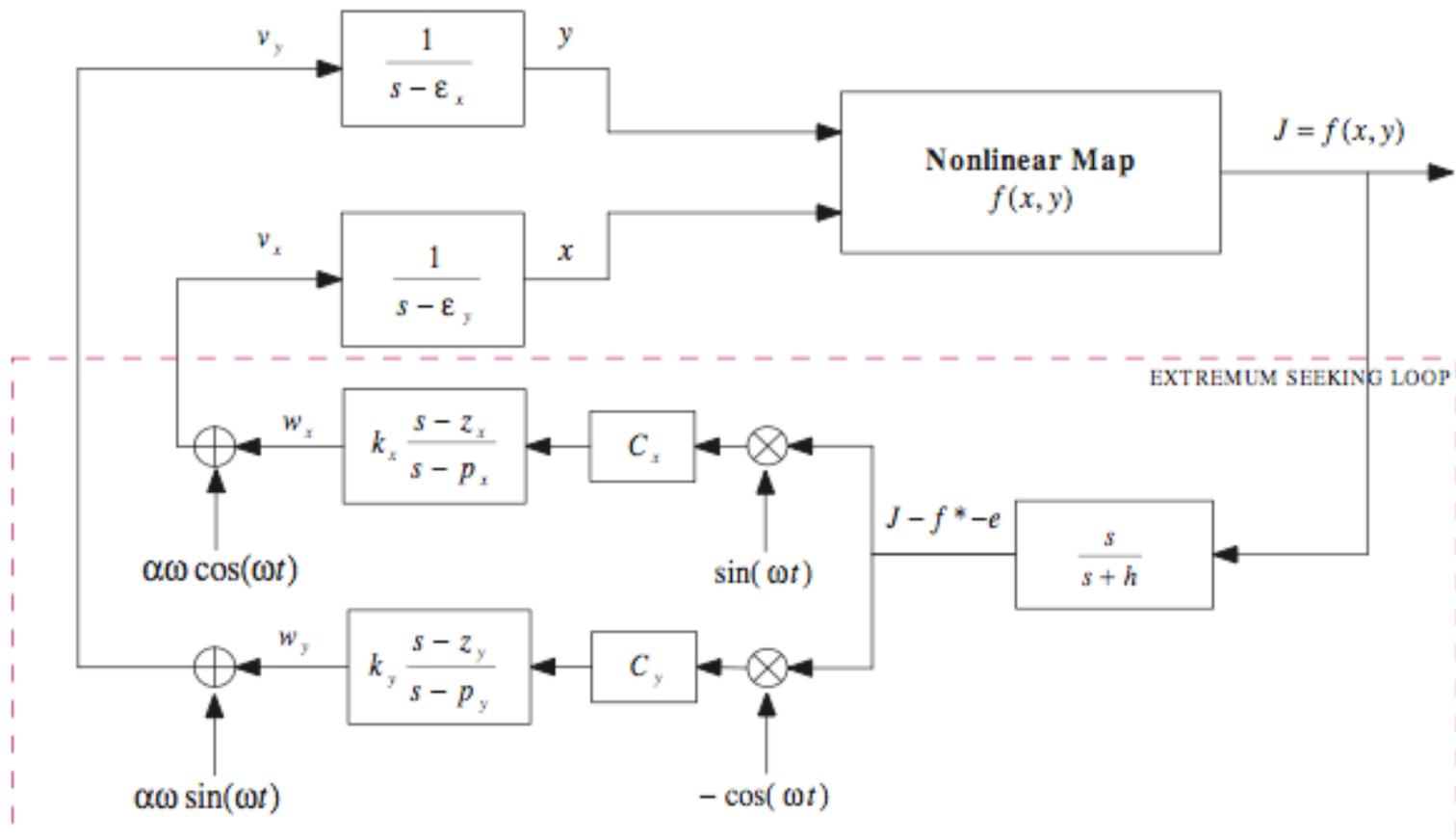
Point Mass



Double Integrators – Block Diagram



A Plant with Moderately Unstable Poles



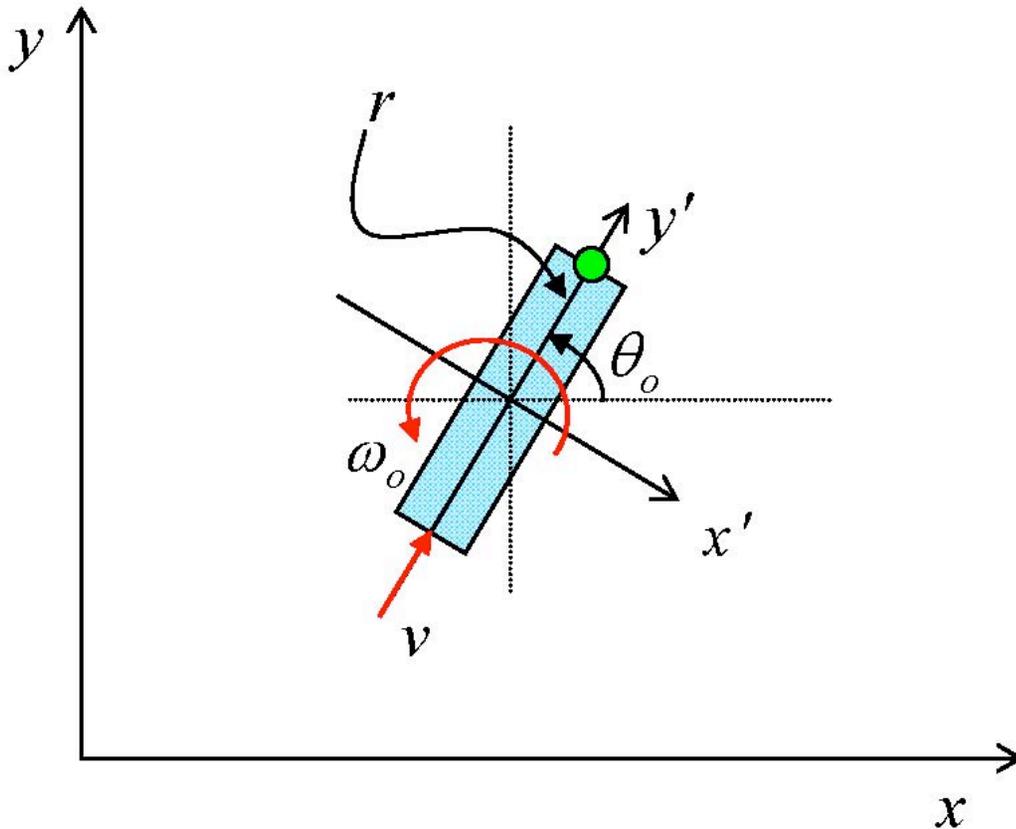
Nonholonomic Vehicle: Unicycle Model

2D Model - “Unicycle”

- Nonholonomic (kinematically constrained, underactuated) **mobile robot**
- Constant forward velocity, controlled angular velocity
- Or vice versa

Model

Unicycle: non-collocated



Sensor Dynamics

$$\dot{x}_s = v \cos \theta_o - r \dot{\theta}_o \sin \theta_o$$

$$\dot{y}_s = v \sin \theta_o + r \dot{\theta}_o \cos \theta_o$$

$$\dot{\theta}_o = \omega_o = \frac{d}{dt} \theta_o$$

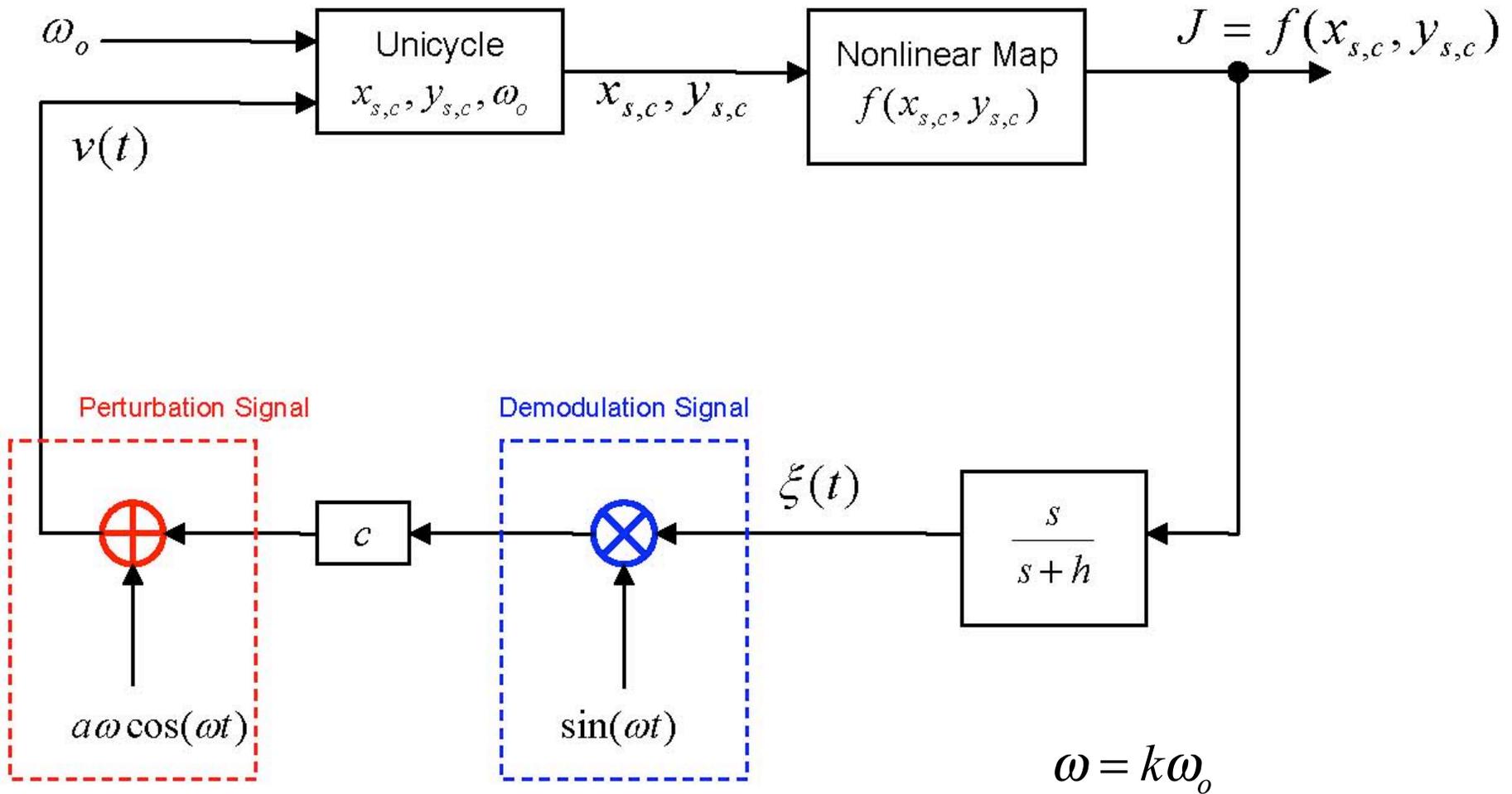
Inputs

$$v, \omega_o$$

System is linearly **uncontrollable** (from inputs v, ω_o)
and **unobservable** (from the output $f(x,y)$ at its peak)

Tuning the Forward Velocity

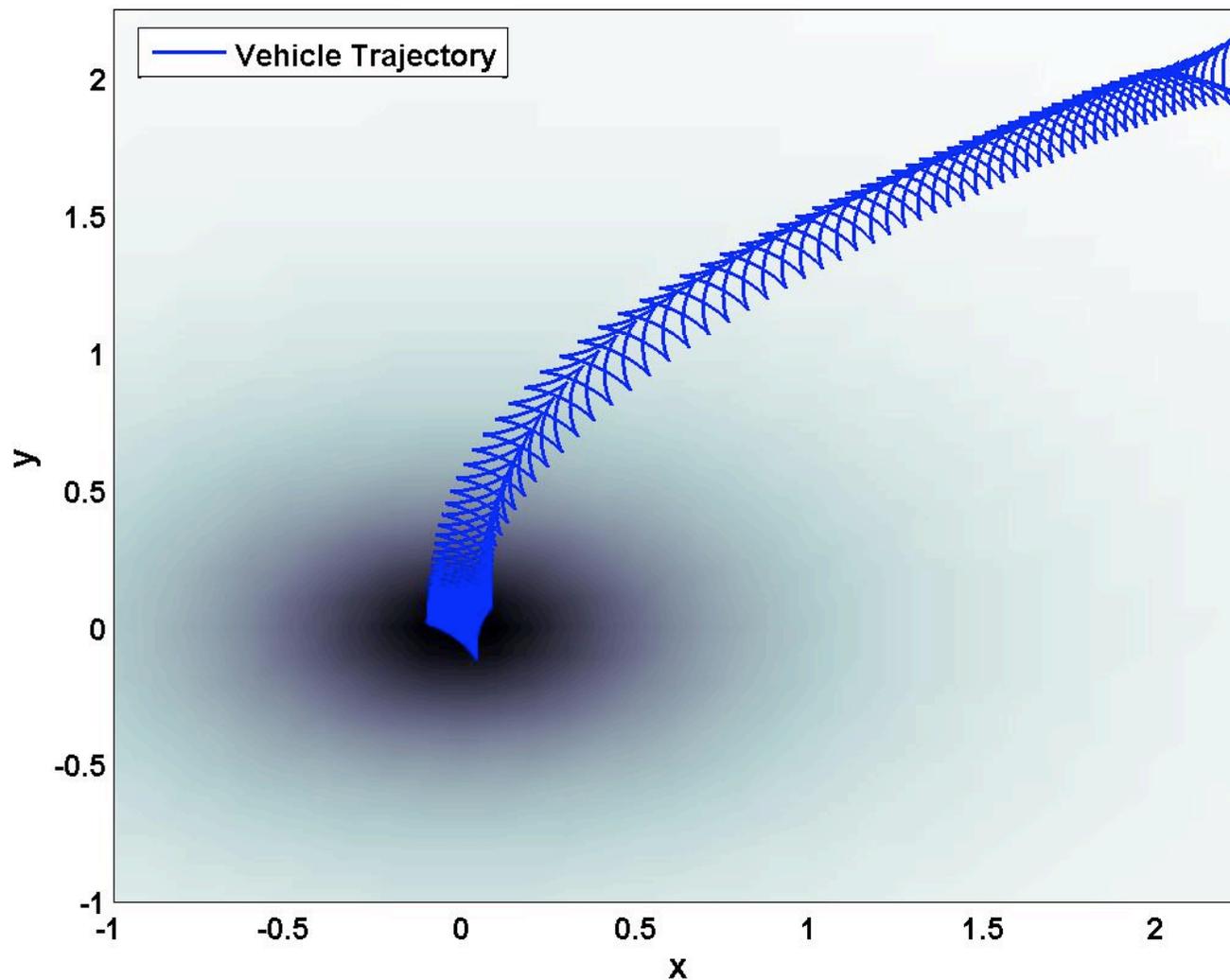
Unicycle: collocated



Simulation Results

Unicycle: collocated

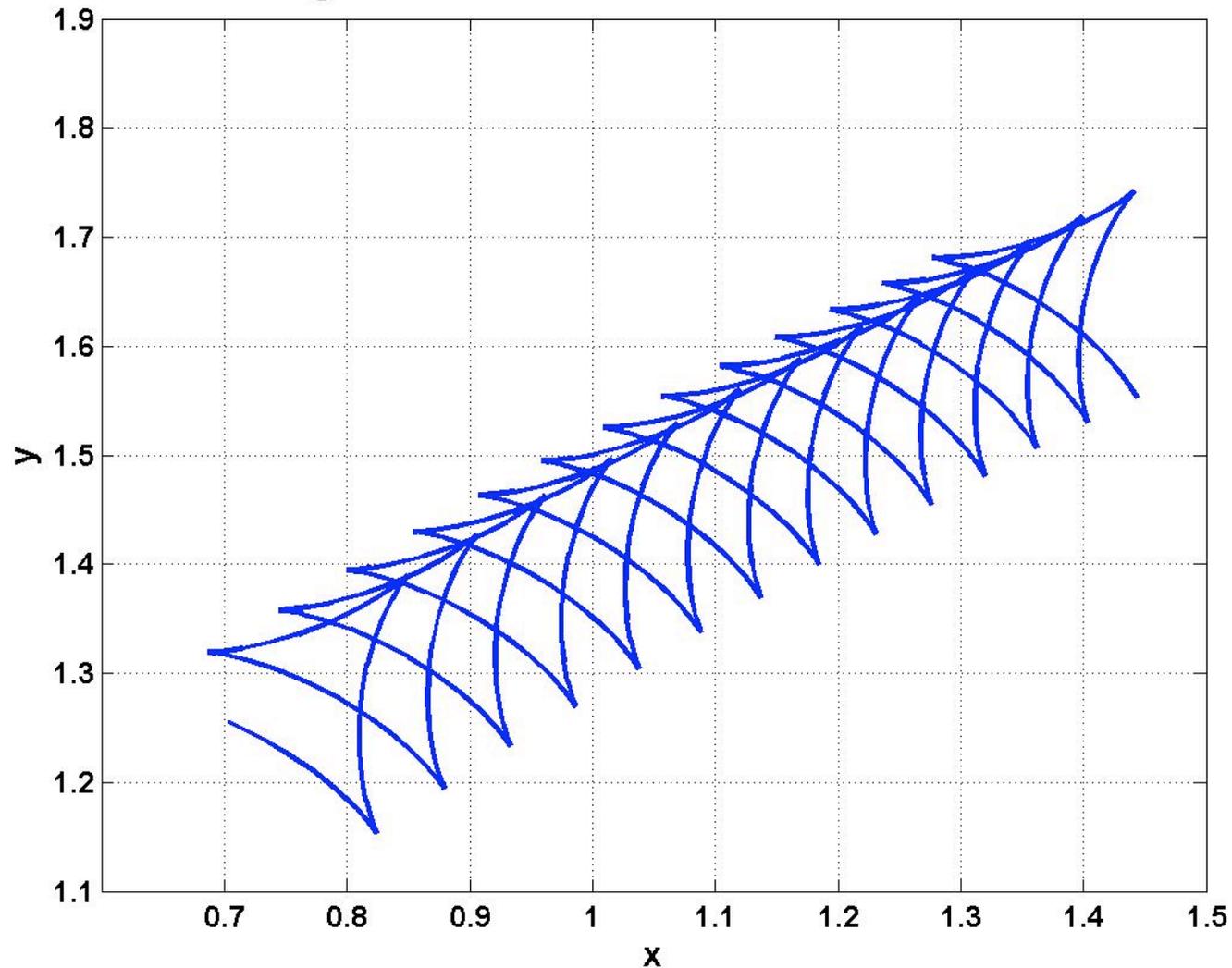
Tuning of Foward Velocity



Simulation Results

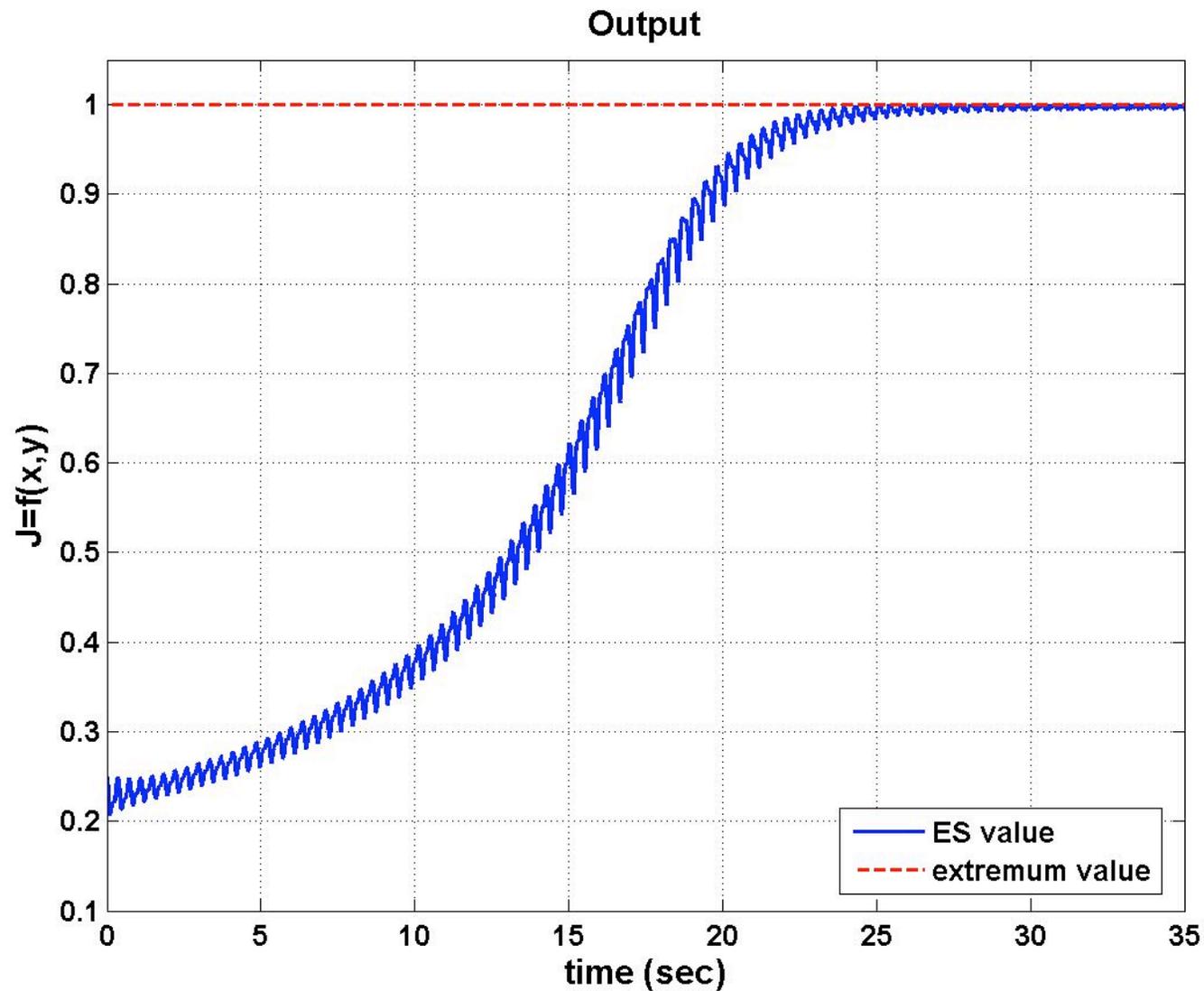
Unicycle: collocated

Triangular Pattern of the Vehicle Center Movement



Simulation Results

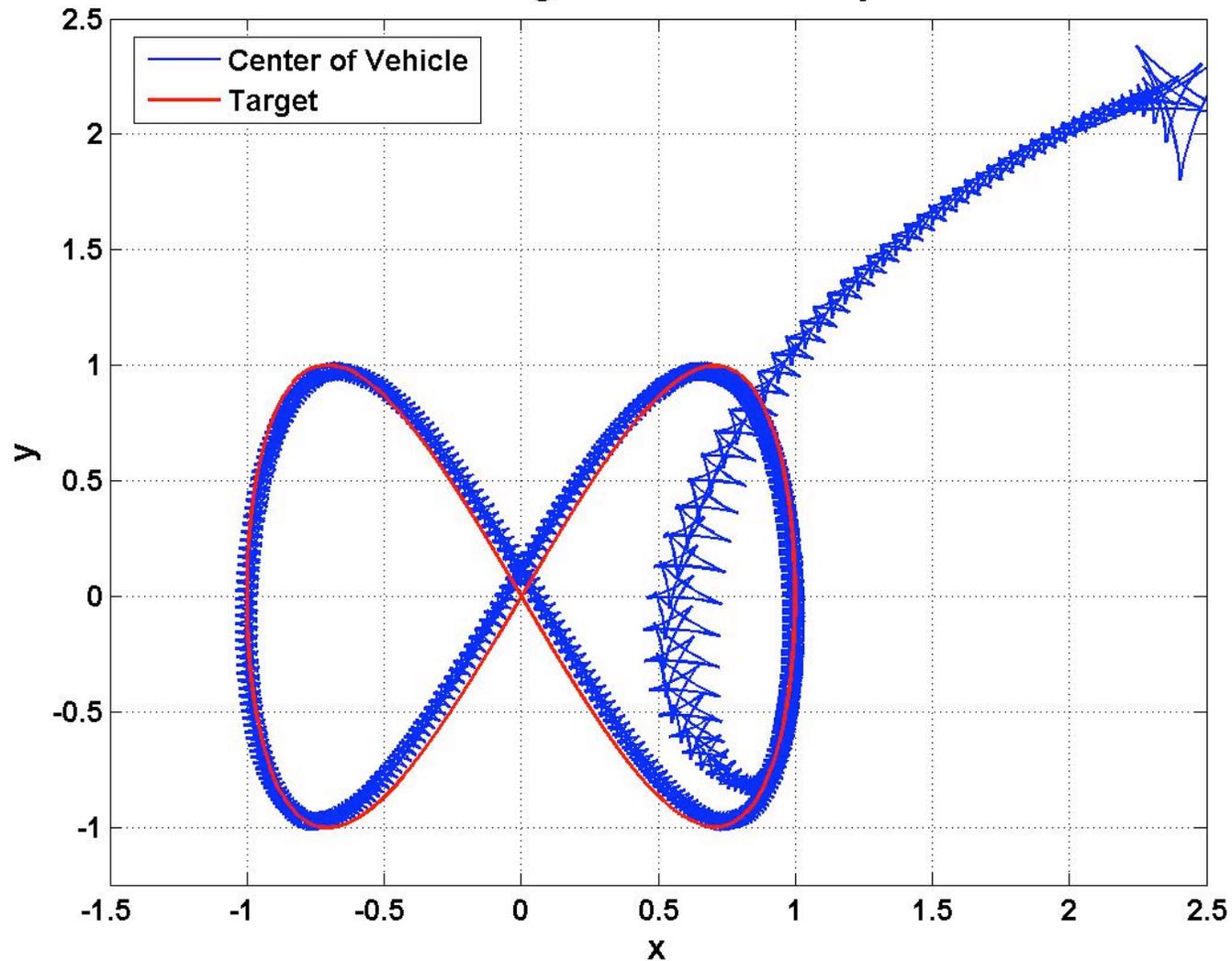
Unicycle: collocated



Simulation Results

Unicycle: collocated

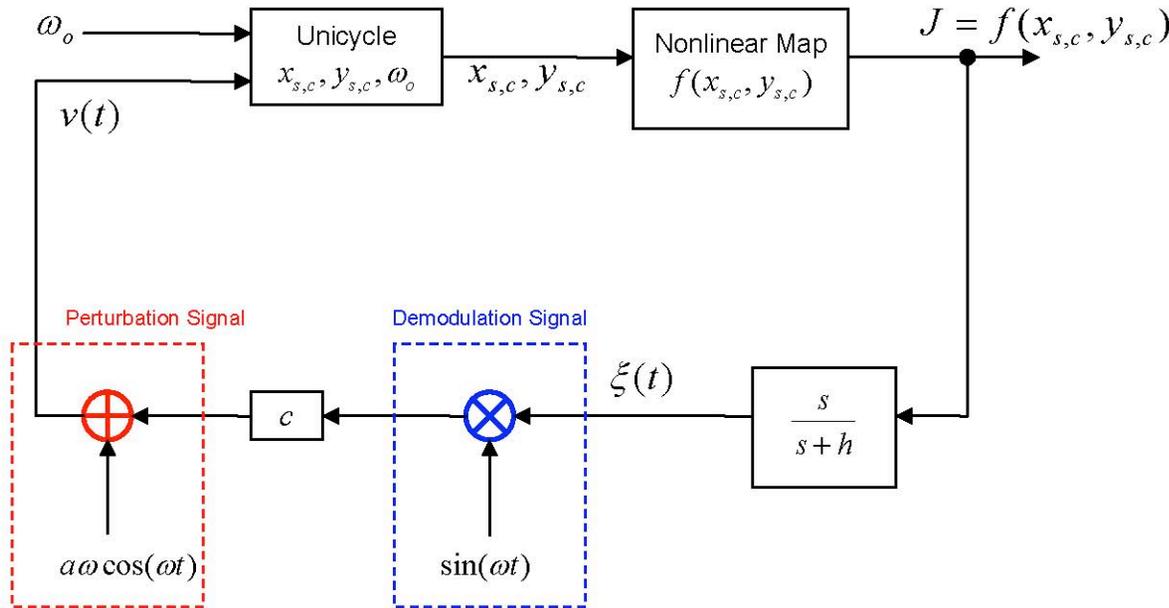
Tuning of Forward Velocity



Stability Proof by Averaging Tuning of Forward Velocity

Stability Proof by Averaging

Full Nonlinear Time-Varying Model



$$\tilde{x} = x_c - x^* - a \sin(\omega t) \cos(\omega_o t)$$

$$\tilde{y} = y_c - y^* - a \sin(\omega t) \sin(\omega_o t)$$

$$e = \frac{h}{s+h} [J] - f^*$$

$$\xi = \frac{s}{s+h} [J] = J - f^* - e$$

$$\tau = \omega t$$

$$\frac{d}{d\tau} \tilde{x} = \frac{1}{\omega} \left[c \sin \tau \cos \left(\frac{\tau}{k} \right) \xi + a \omega_o \sin \tau \sin \left(\frac{\tau}{k} \right) \right]$$

$$\frac{d}{d\tau} \tilde{y} = \frac{1}{\omega} \left[c \sin \tau \sin \left(\frac{\tau}{k} \right) \xi - a \omega_o \sin \tau \cos \left(\frac{\tau}{k} \right) \right]$$

$$\frac{d}{d\tau} e = \frac{h}{\omega} \xi$$

where

$$\xi = -q_x \left(\tilde{x} + a \sin \tau \cos \left(\frac{\tau}{k} \right) \right)^2 - q_y \left(\tilde{y} + a \sin \tau \sin \left(\frac{\tau}{k} \right) \right)^2 - e$$

Stability Proof by Averaging

Applying Averaging

$$\begin{aligned}
 \frac{d}{d\tau} \tilde{x}_{avg} &= \frac{1}{\omega} \frac{1}{2\pi} \int_0^{2\pi} [c \sin(k\tau) \cos \tau \xi + a\omega_o \sin(k\tau) \sin \tau] d\tau \\
 &= -\frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2\pi} c \sin(k\tau) \cos \tau (q_x \tilde{x}^2 + q_y \tilde{y}^2 + e) d\tau \\
 &\quad - \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2\pi} c \sin(k\tau) \cos \tau [2q_x \tilde{x} a \sin(k\tau) \cos \tau + 2q_y \tilde{y} a \sin(k\tau) \sin \tau] d\tau \\
 &\quad - \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2\pi} c \sin(k\tau) \cos \tau [q_x a^2 \sin^2(k\tau) \cos^2 \tau + 2q_y a^2 \sin^2(k\tau) \sin^2 \tau] d\tau \\
 &\quad + \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2\pi} a\omega_o \sin(k\tau) \sin \tau d\tau \\
 &= -\frac{1}{2\omega} acq_x \tilde{x}_{avg}
 \end{aligned}$$

Stability Proof by Averaging

Equilibrium of Average System

The equilibrium of the average model is:

$$\frac{d}{d\tau} \tilde{x}_{avg} = -\frac{1}{2\omega} acq_x \tilde{x}_{avg}$$

$$\tilde{x}_{avg}^e = 0,$$

$$\frac{d}{d\tau} \tilde{y}_{avg} = -\frac{1}{2\omega} acq_y \tilde{y}_{avg}$$

$$\tilde{y}_{avg}^e = 0,$$

$$\frac{d}{d\tau} e_{avg} = -\frac{h}{\omega} \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \left(\frac{a^2}{4} \right) (q_x + q_y) \right]$$

$$e_{avg}^e = -\frac{a^2}{4} (q_x + q_y)$$

$$J_{avg}|_{equil.} = \frac{1}{2\omega} \begin{bmatrix} -acq_x & 0 & 0 \\ 0 & -acq_y & 0 \\ 0 & 0 & -2h \end{bmatrix}$$

Stability Proof by Averaging

Theorem

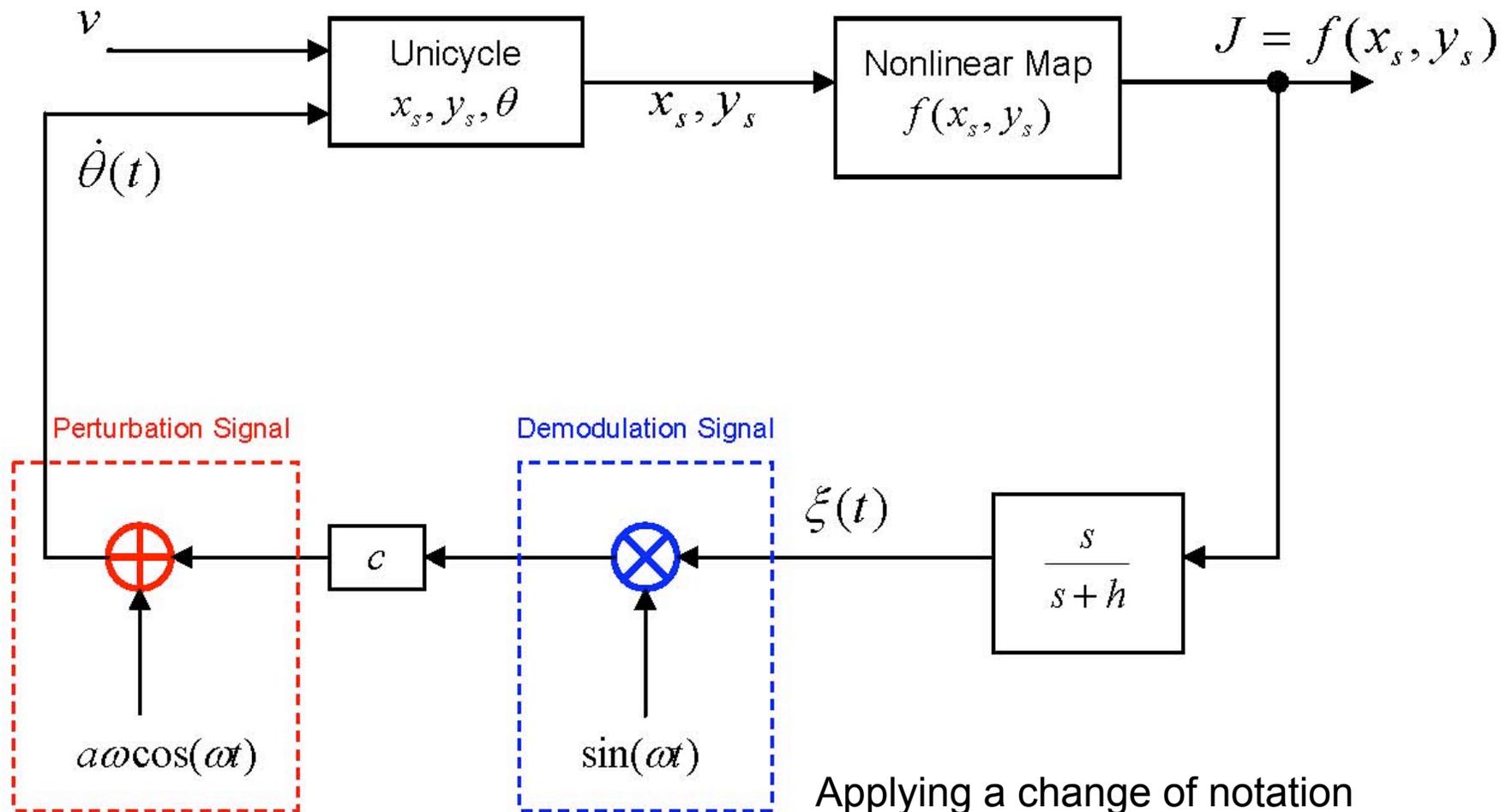
For sufficiently large ω there exists a unique exponentially stable periodic solution of period $2\pi/\omega$ and it satisfies

$$\left\| \begin{bmatrix} \tilde{x} \frac{2\pi}{\omega} \\ \tilde{y} \frac{2\pi}{\omega} \\ e^{\frac{2\pi}{\omega} t} + \frac{a^2}{4} (q_x + q_y) \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall t \geq 0$$

Speed of convergence proportional to $1/\omega$, a^2 , c , q_x , q_y

Tuning the Angular Velocity

Unicycle: non-collocated



Applying a change of notation

$$\theta = \theta_o$$

Extremum Seeking Algorithm

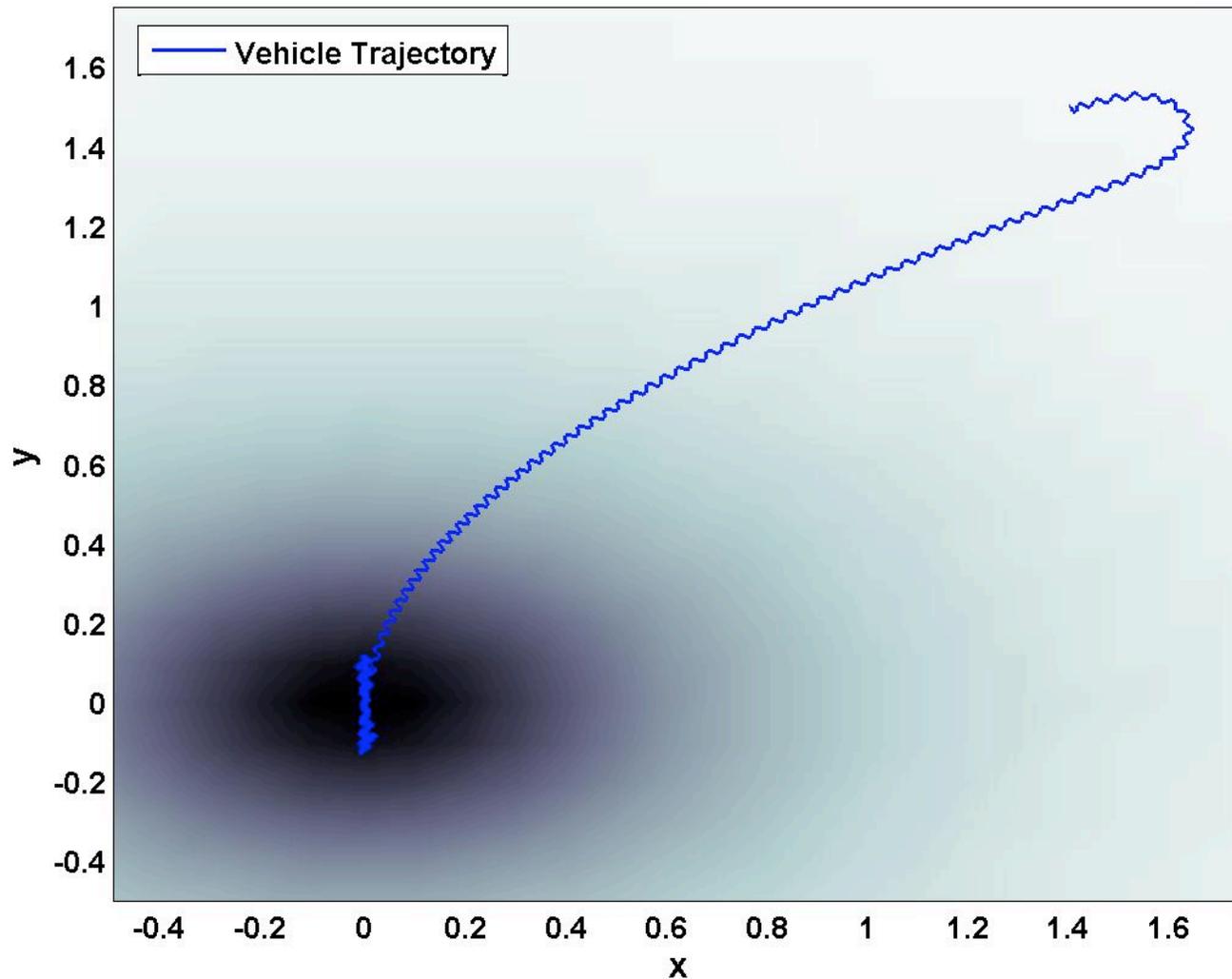
$$\Omega = a\omega \cos(\omega t) + c \sin(\omega t) \frac{s}{s + h} [J]$$

- Linear combination of cosine and sine but non-constant coefficients!
- Two actions:
 - continuous periodic perturbation
 - estimate of optimal input

Simulation Results

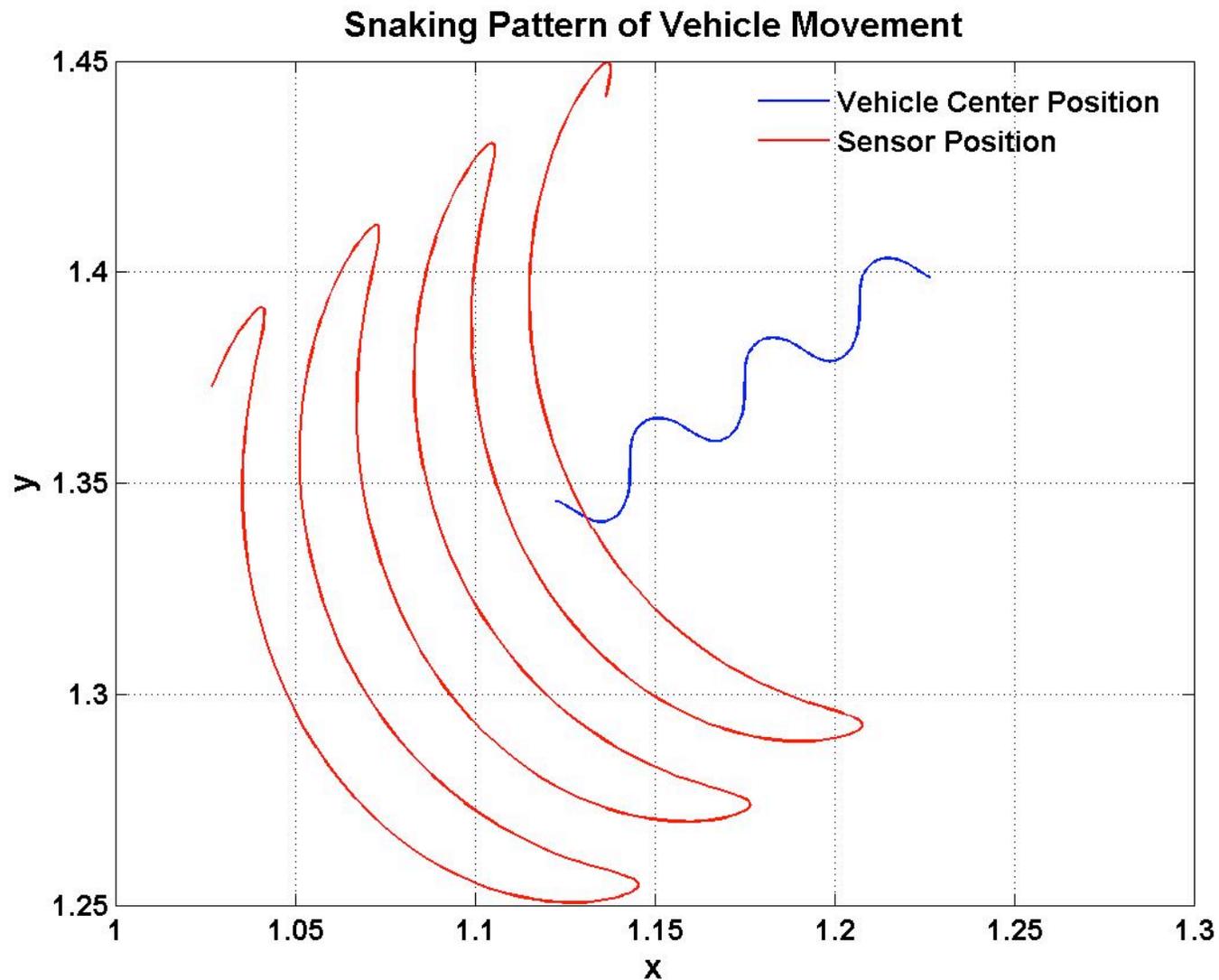
Unicycle: non-collocated

Tuning Angular Velocity

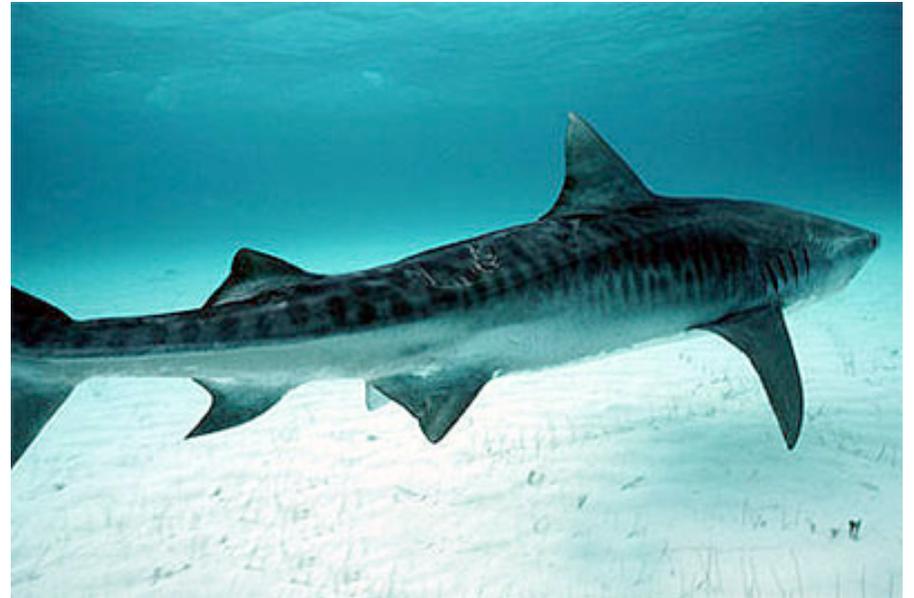


Simulation Results

Unicycle: non-collocated

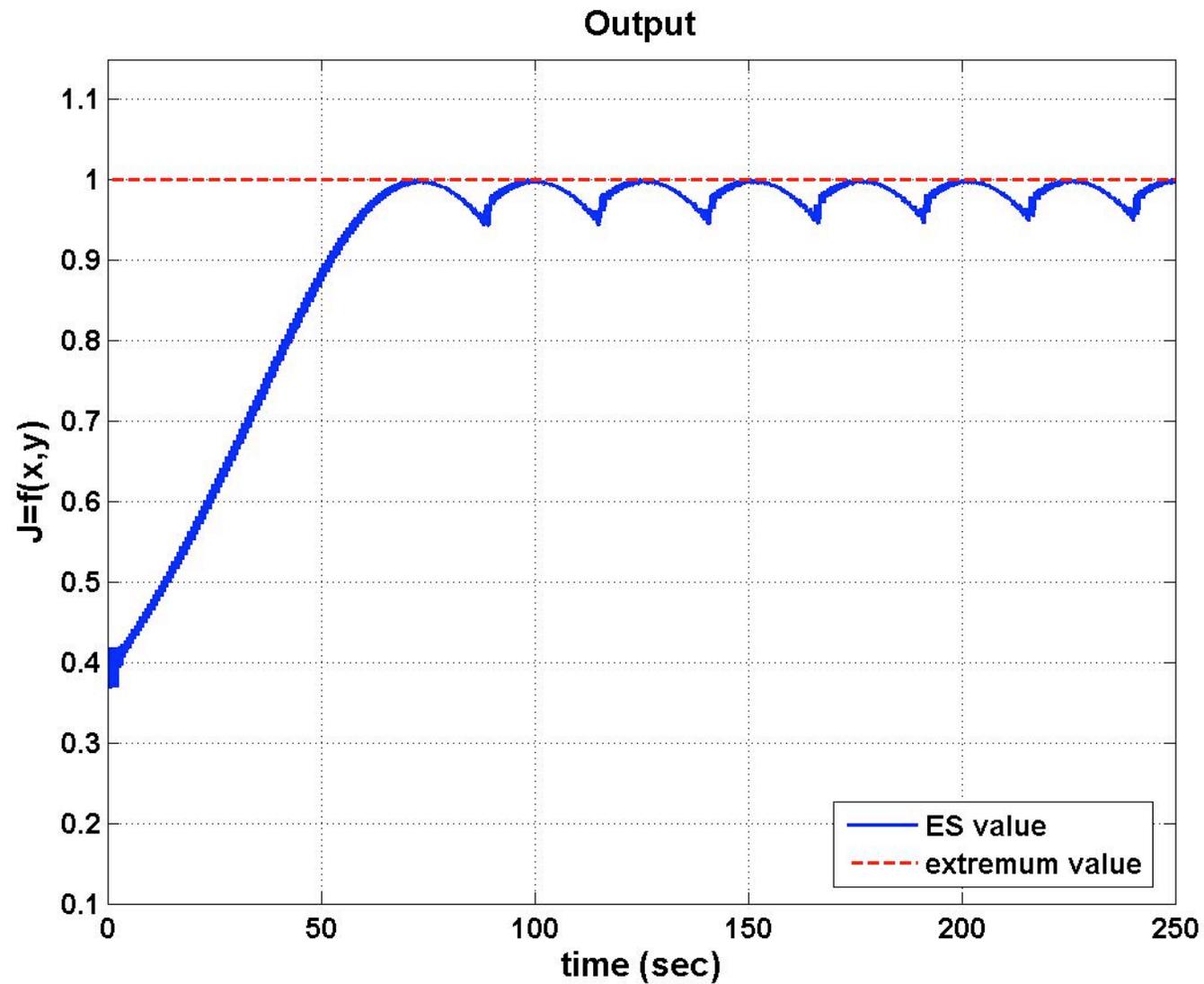


Biologically Inspired

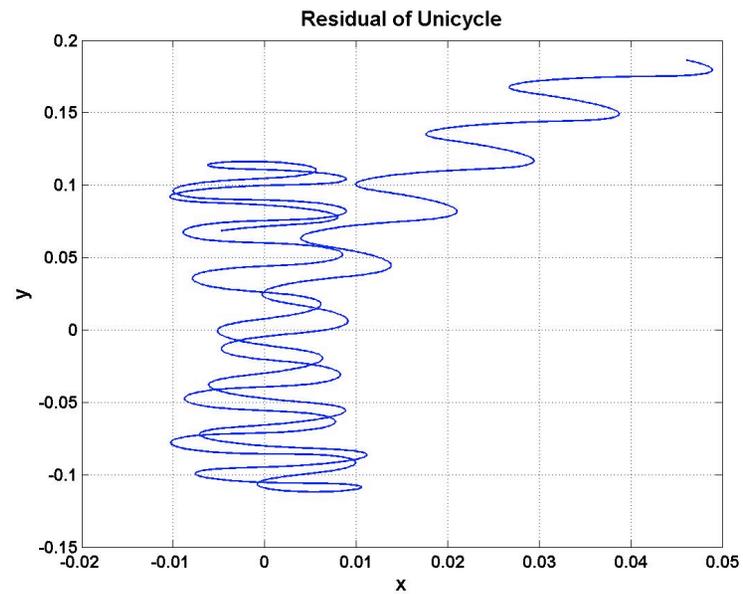
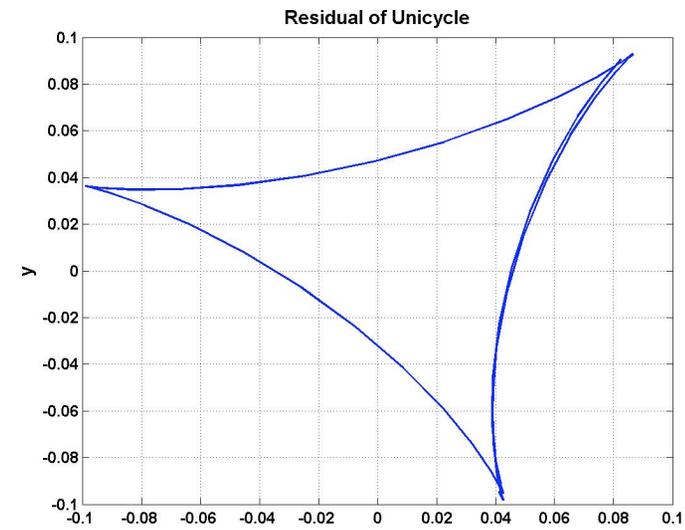
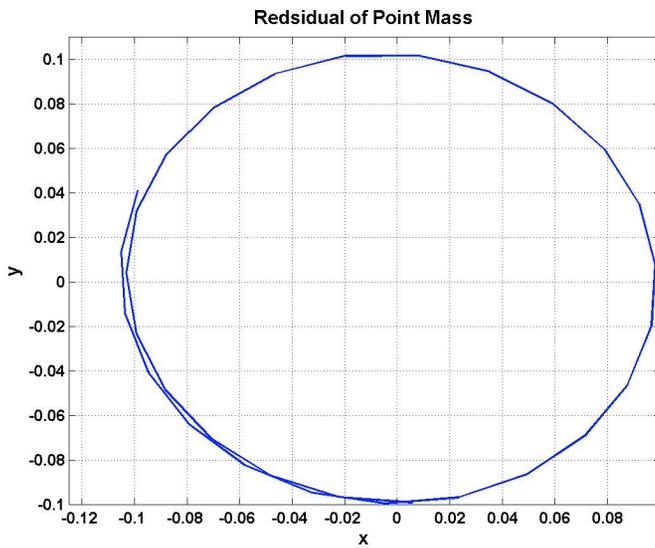


Simulation Results

Unicycle: non-collocated



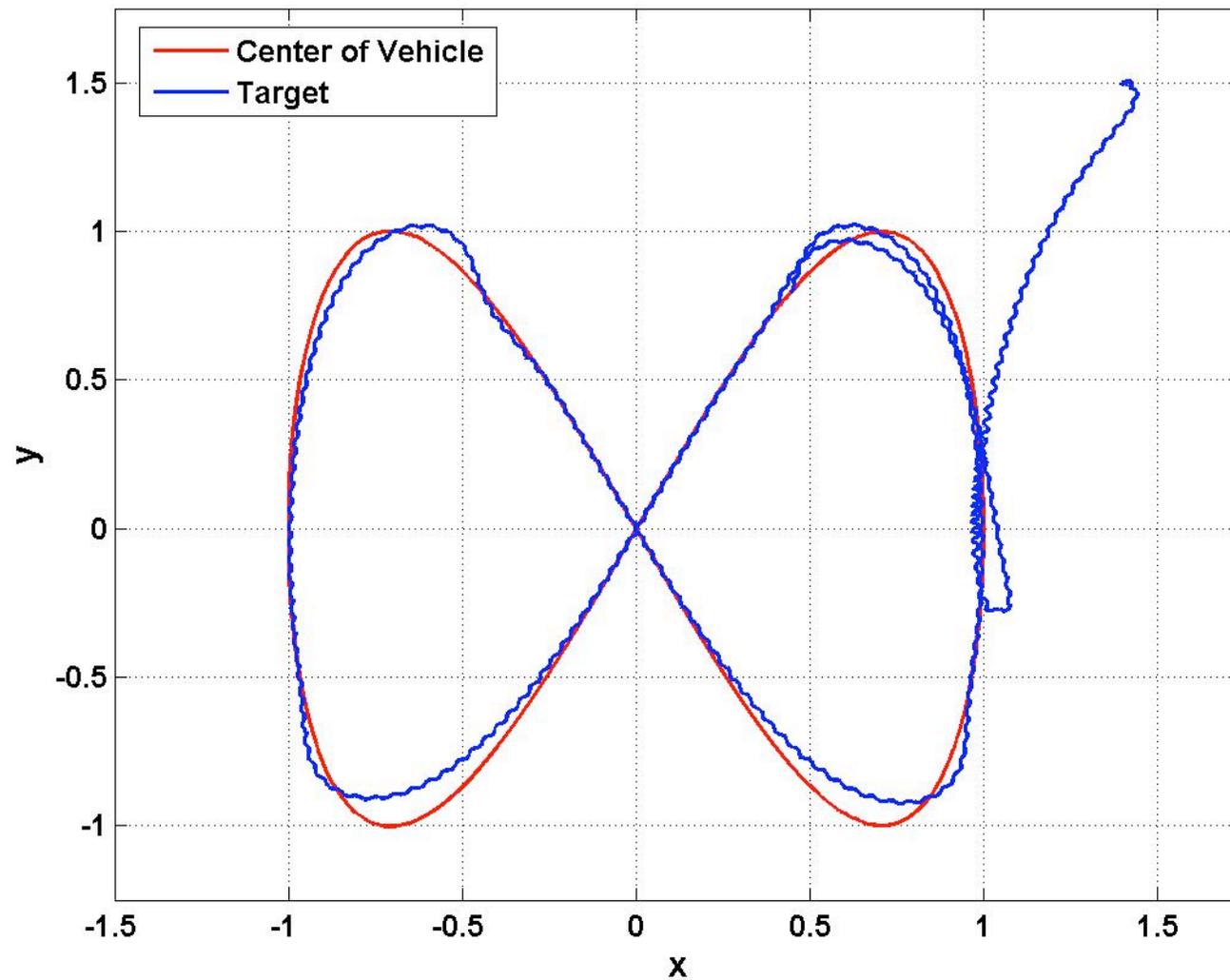
Summary



Simulation Results

Unicycle: non-collocated

Tuning of Angular Velocity



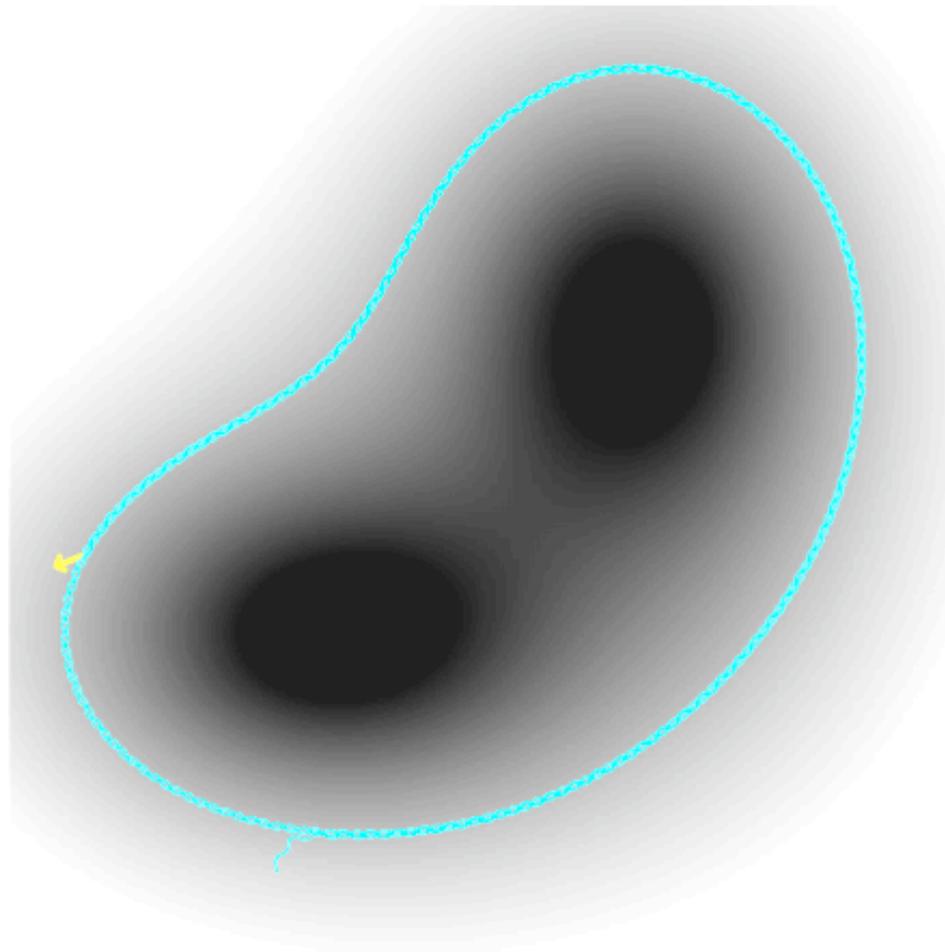
Track a Diffusive Source



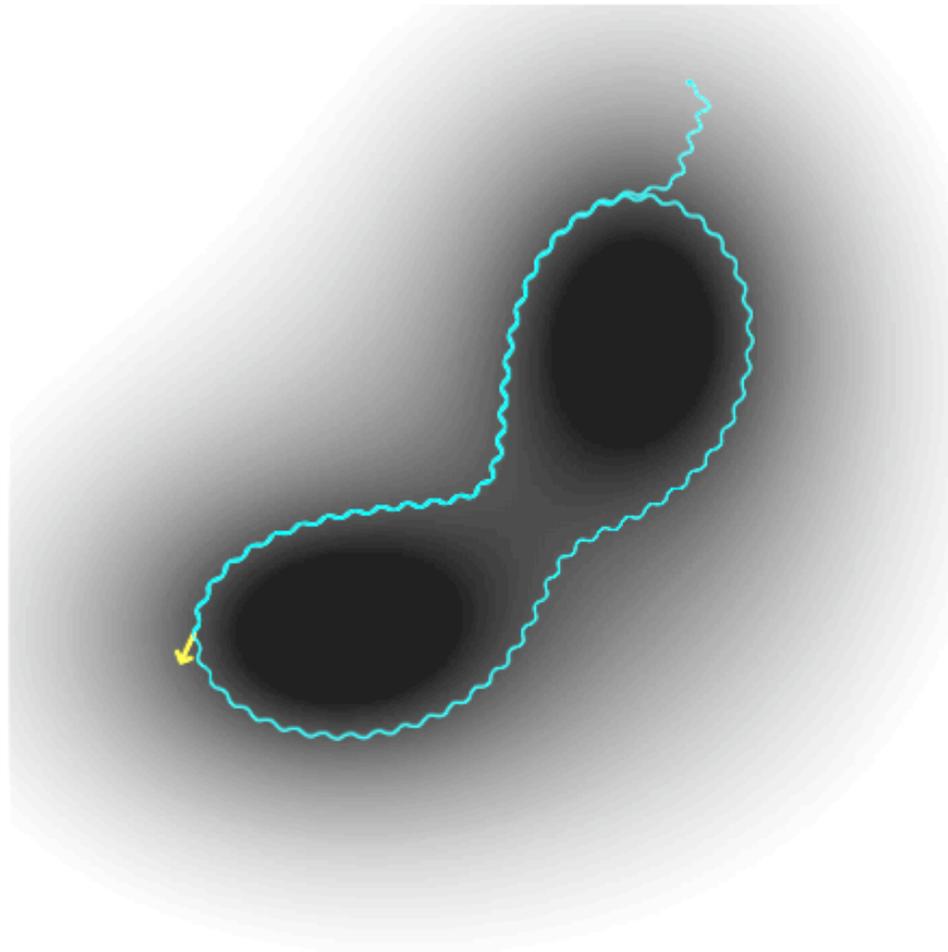
Level Sets

- Several methods using **multiple AUVs** have been proposed
 - Kalantar & Zimmer
 - N. Leonard, Fiorelli, Ogren
 - Bertozzi
 - Burian, Singh
 - Bennett, J. Leonard
- Neutrally buoyant drifters

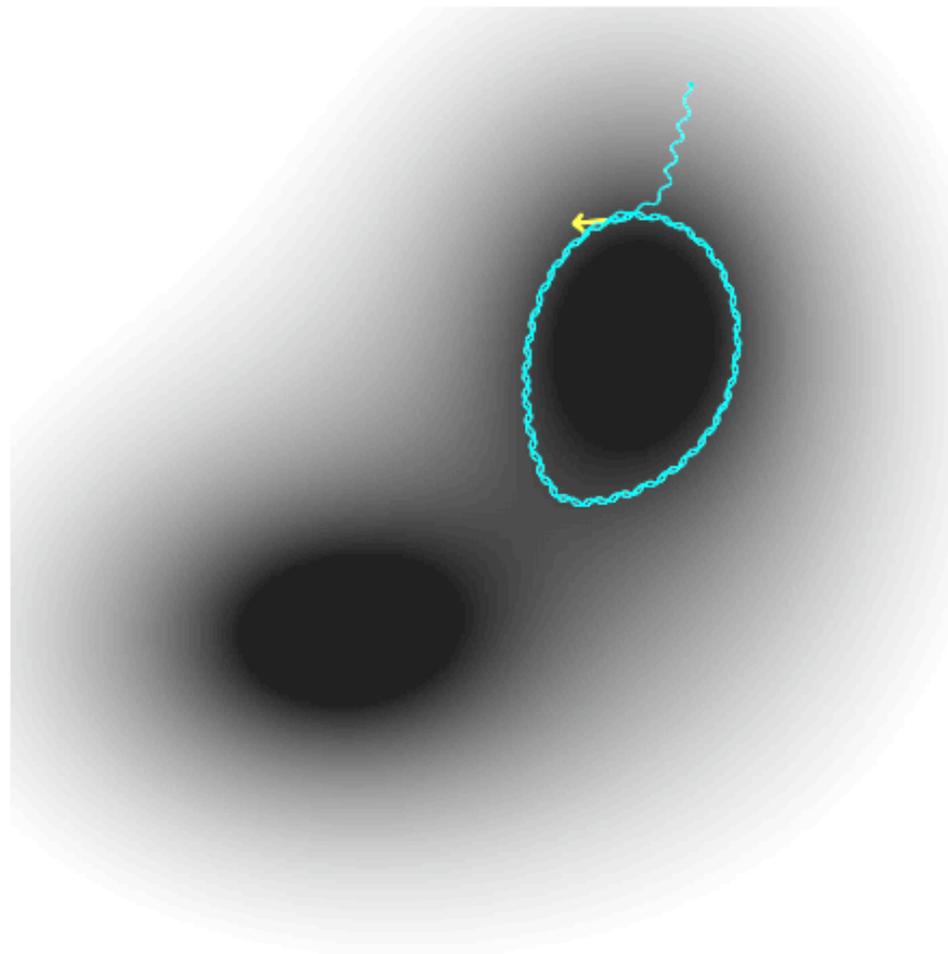
Level Sets



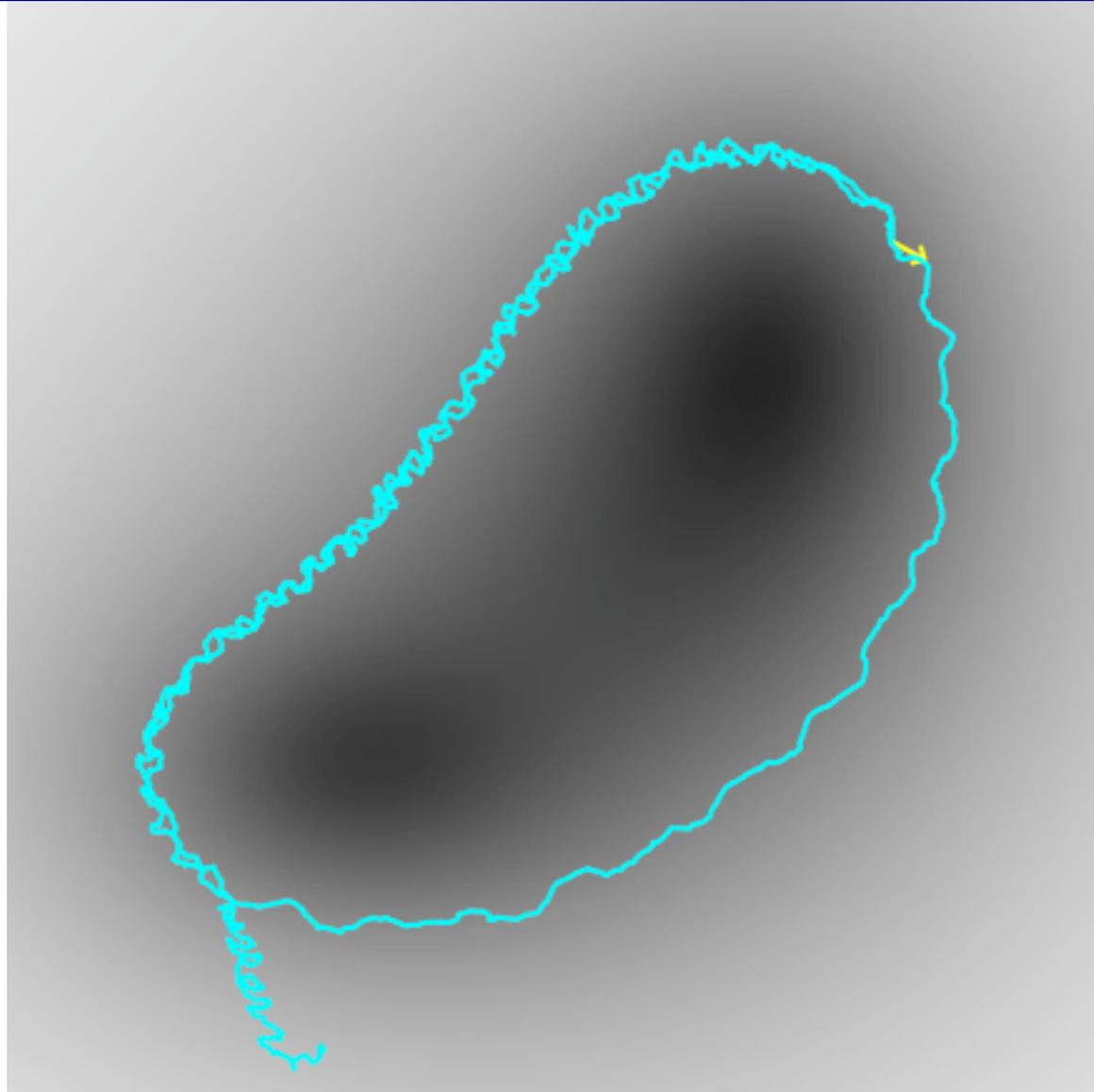
Level Sets



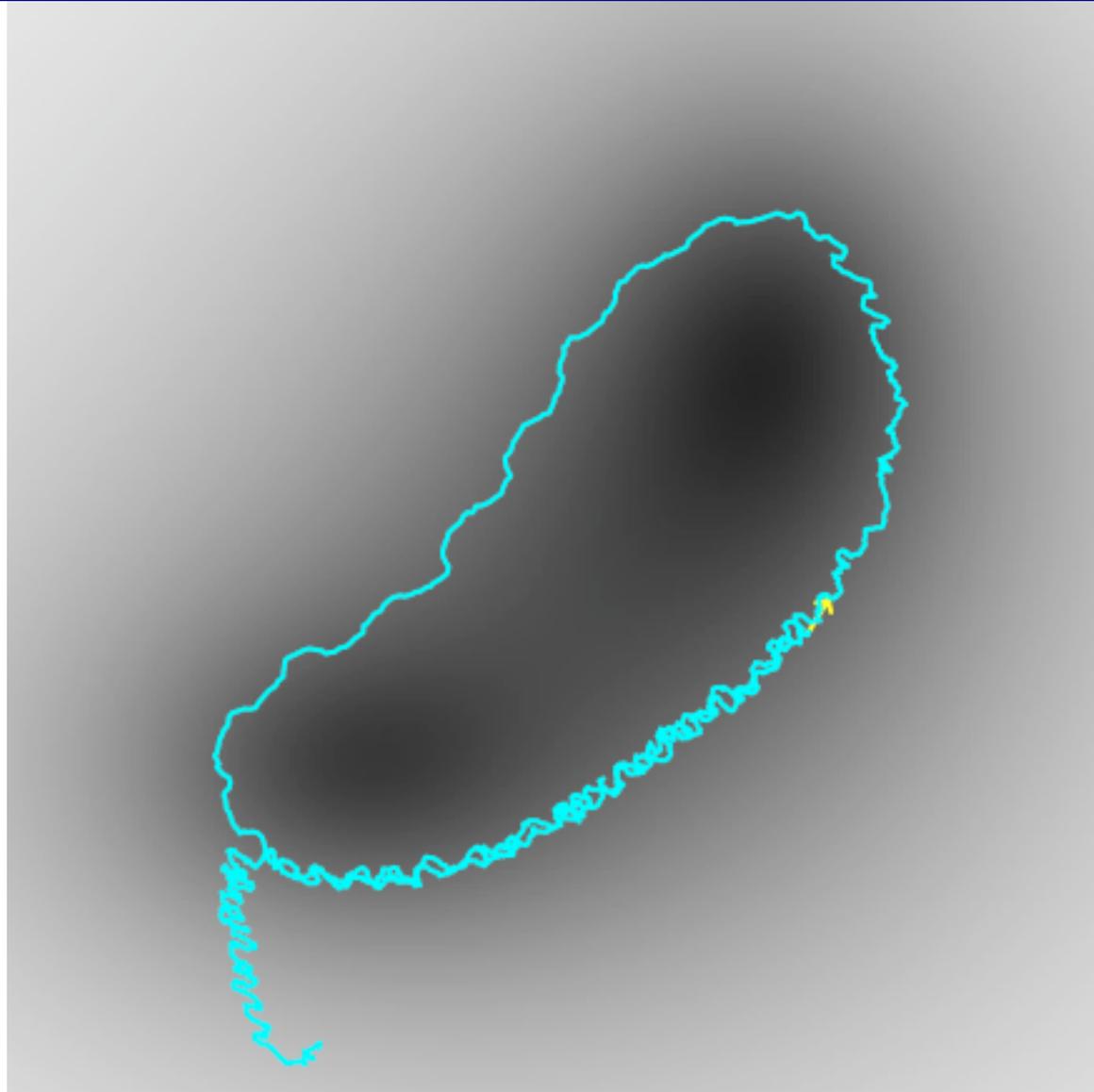
Level Sets



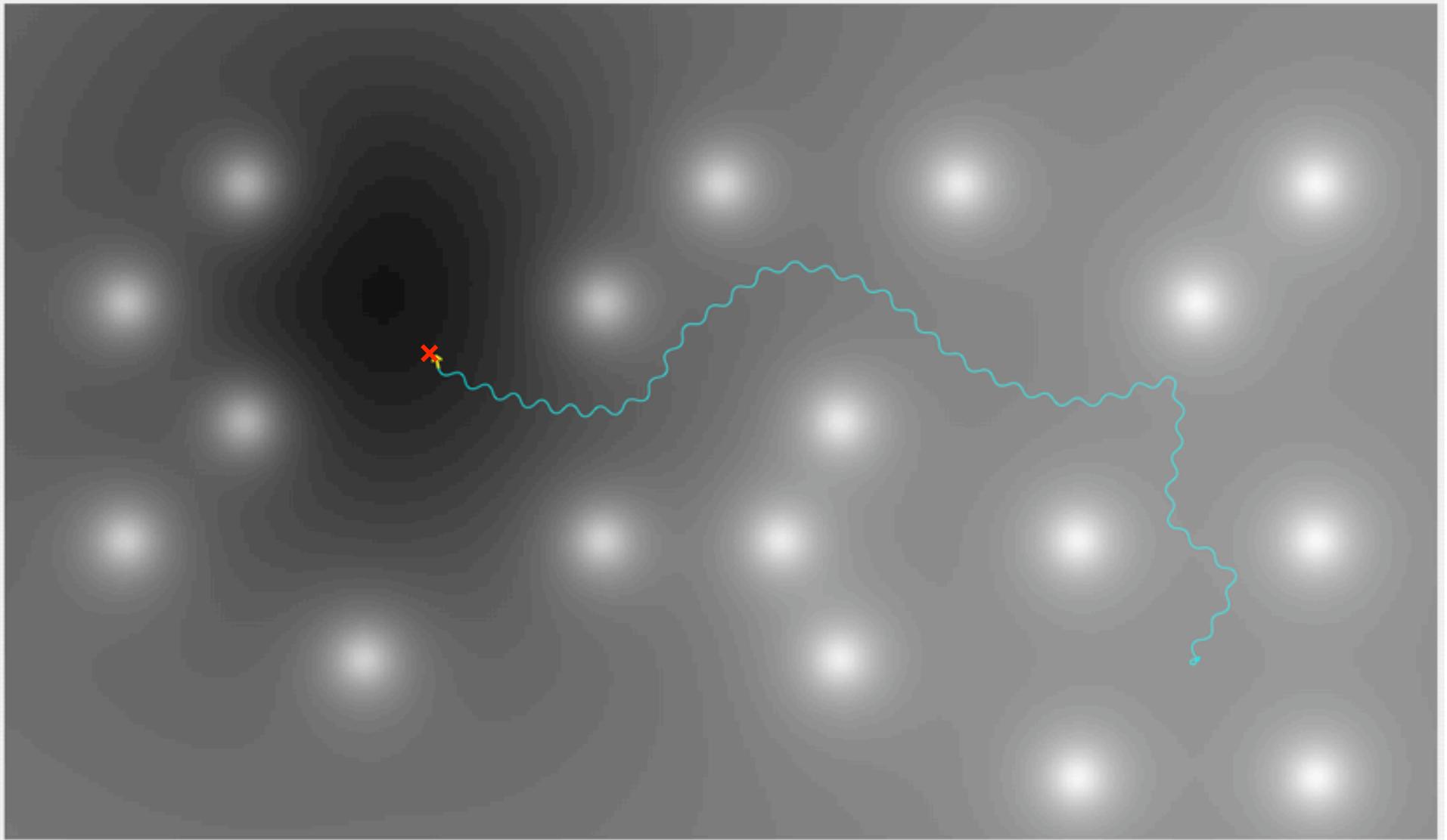
Level Sets



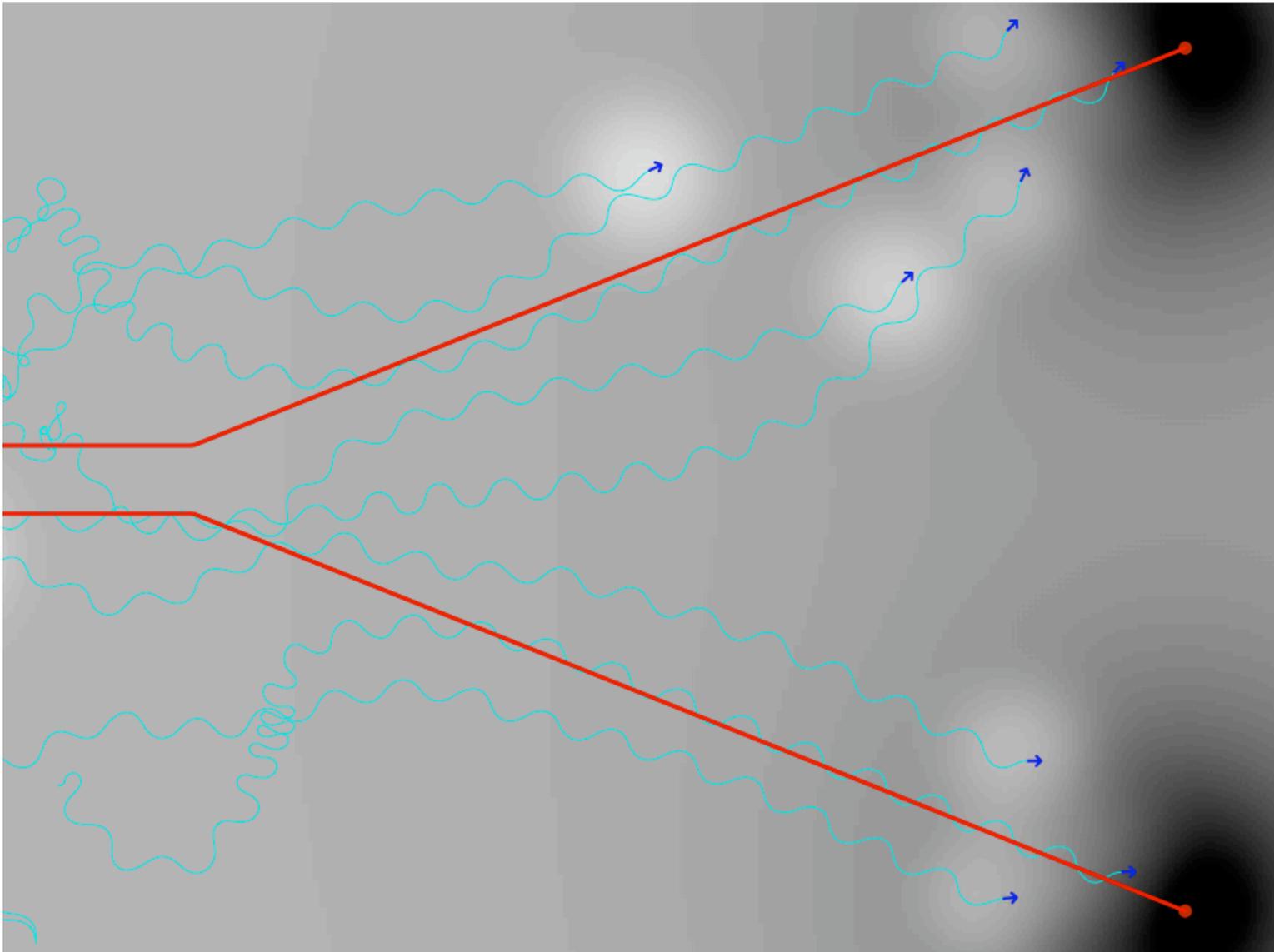
Level Sets



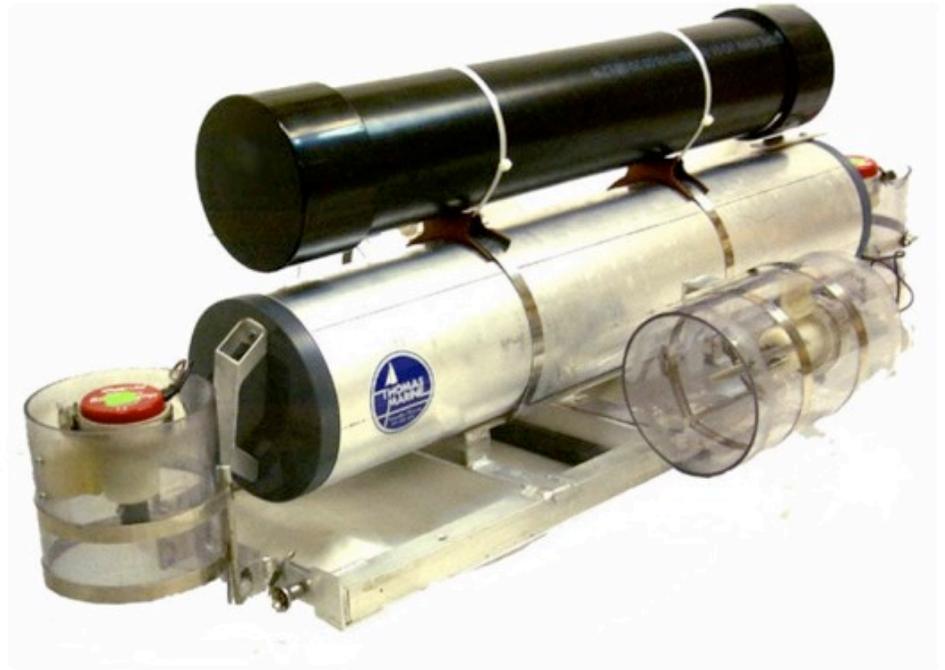
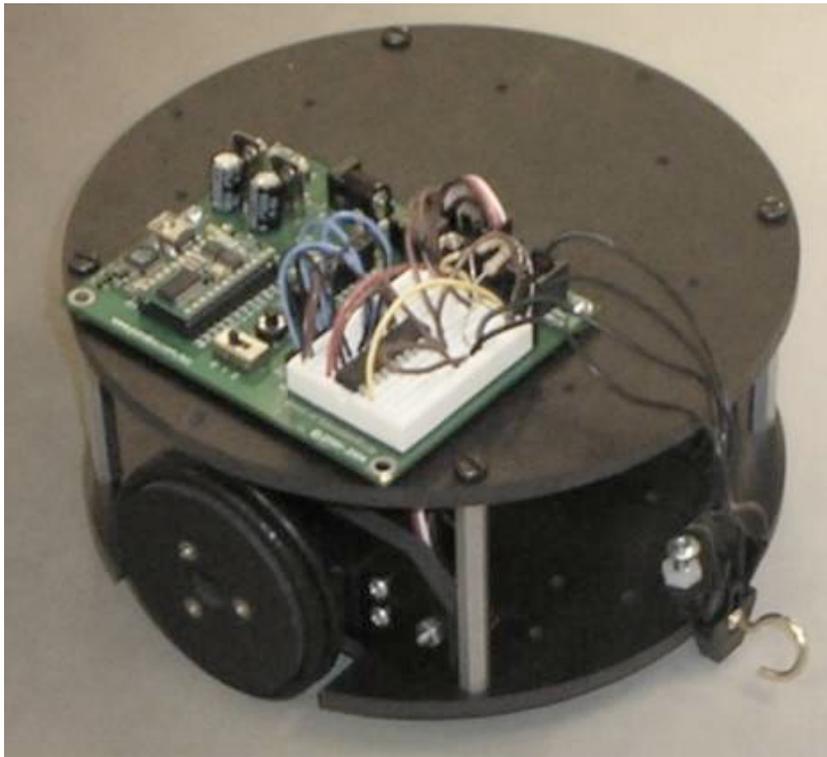
Navigation Through a Minefield



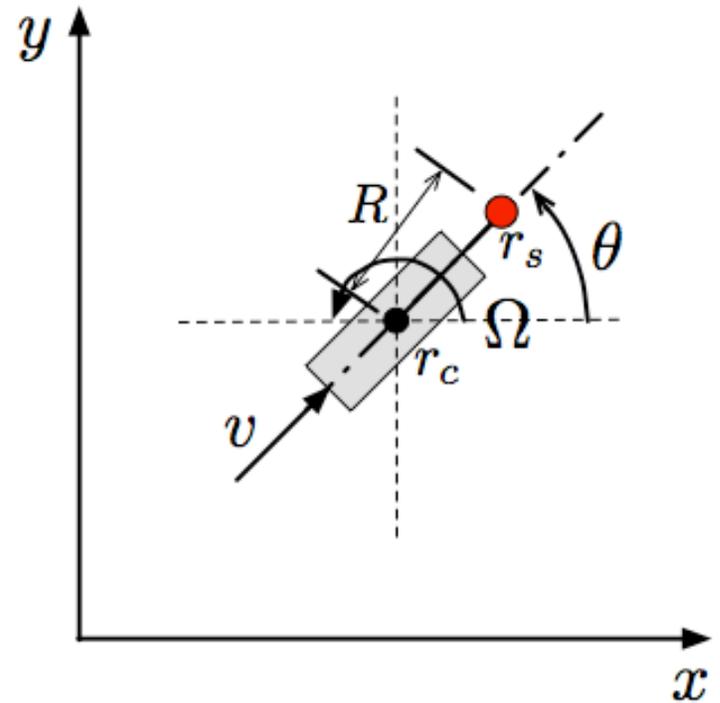
Multi-Vehicle Pursuit



Experimental Results



2D- Stability and Convergence

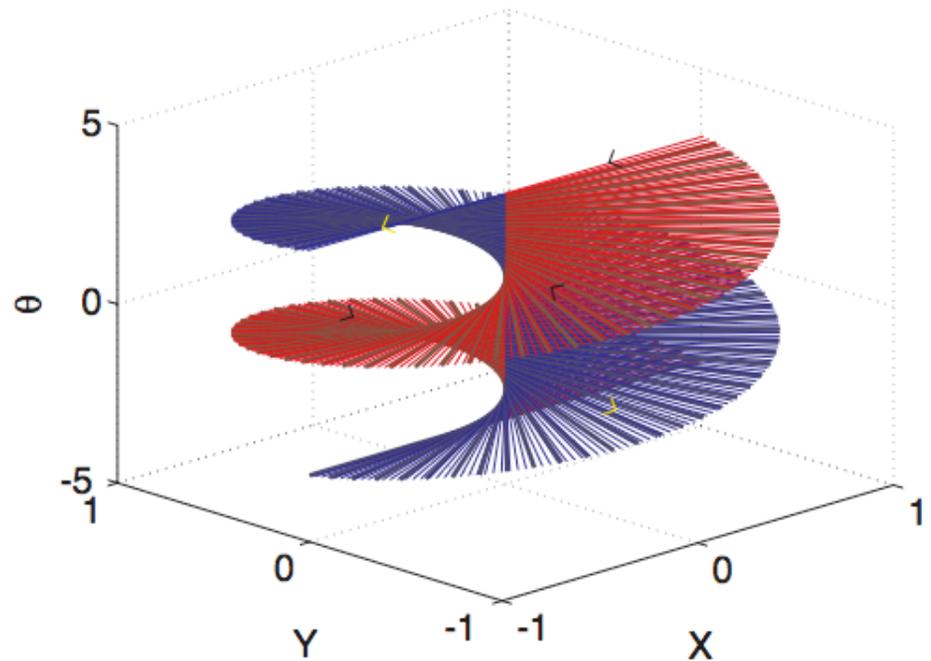


$$\dot{r}_c = V_c e^{j\theta}$$

$$\dot{\theta} = a\omega \cos(\omega t) + c \sin(\omega t) \frac{s}{s+h} [J]$$

The Optimal Heading Manifold

Unstable Solutions



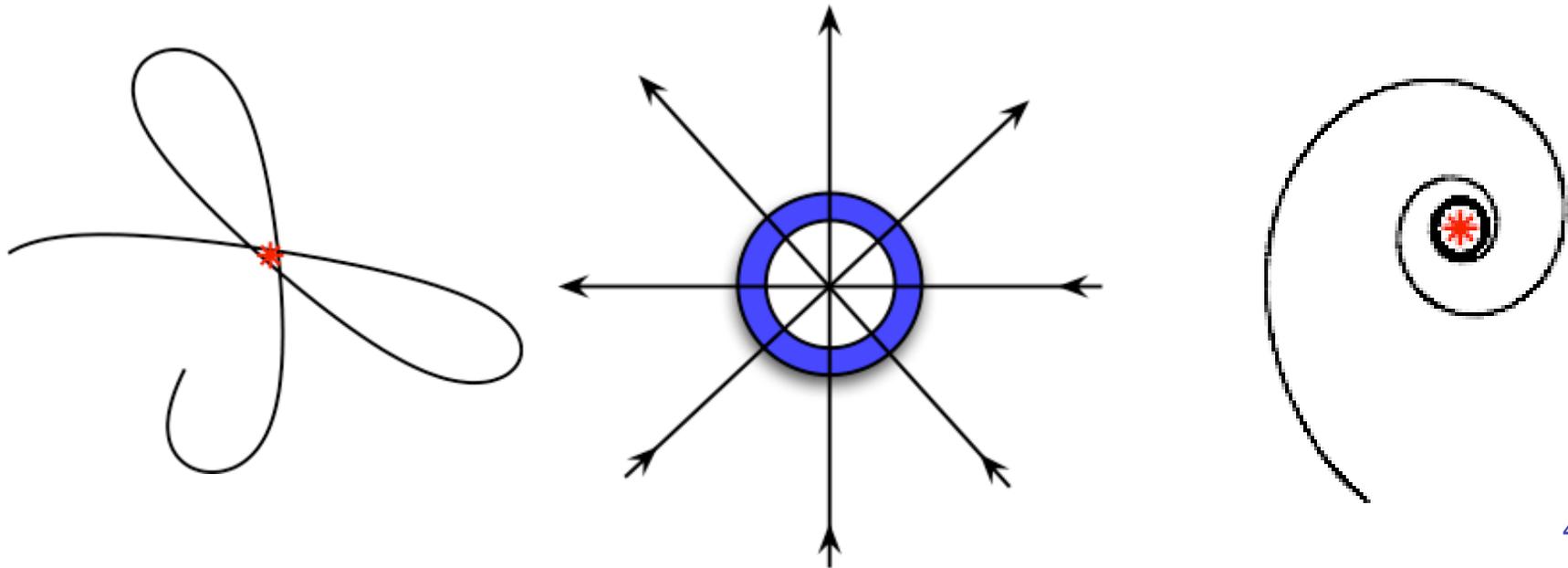
$$\theta^* = \arg(r_c)$$

$$\tilde{\theta} = \theta - \theta^* - a \sin(\omega t)$$

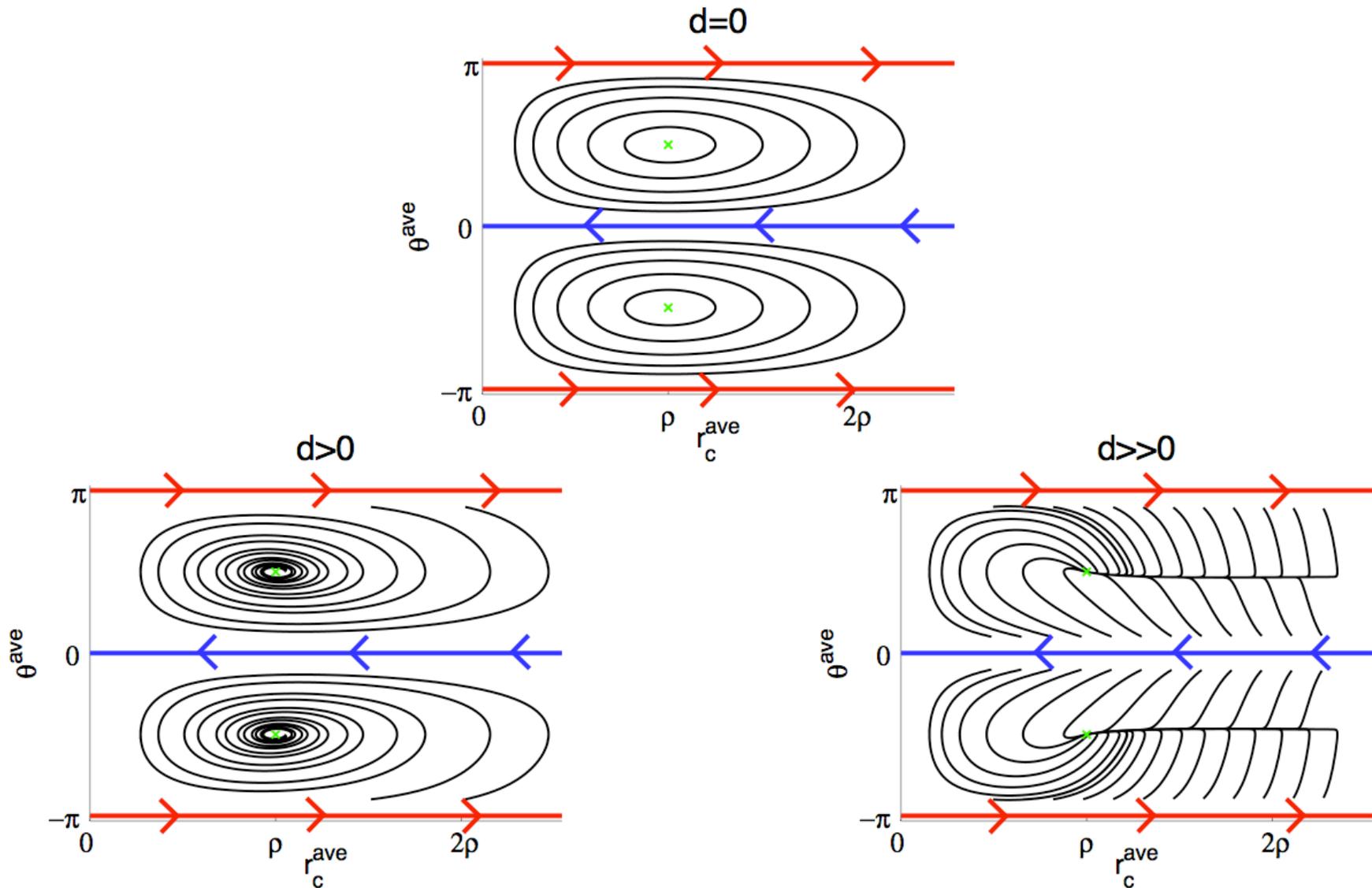
The “Average System”

$$\overbrace{|r_c|}^{\dot{\cdot}}{}^{ave} = -\frac{V_c}{\omega} \cos(\tilde{\theta}^{ave})$$

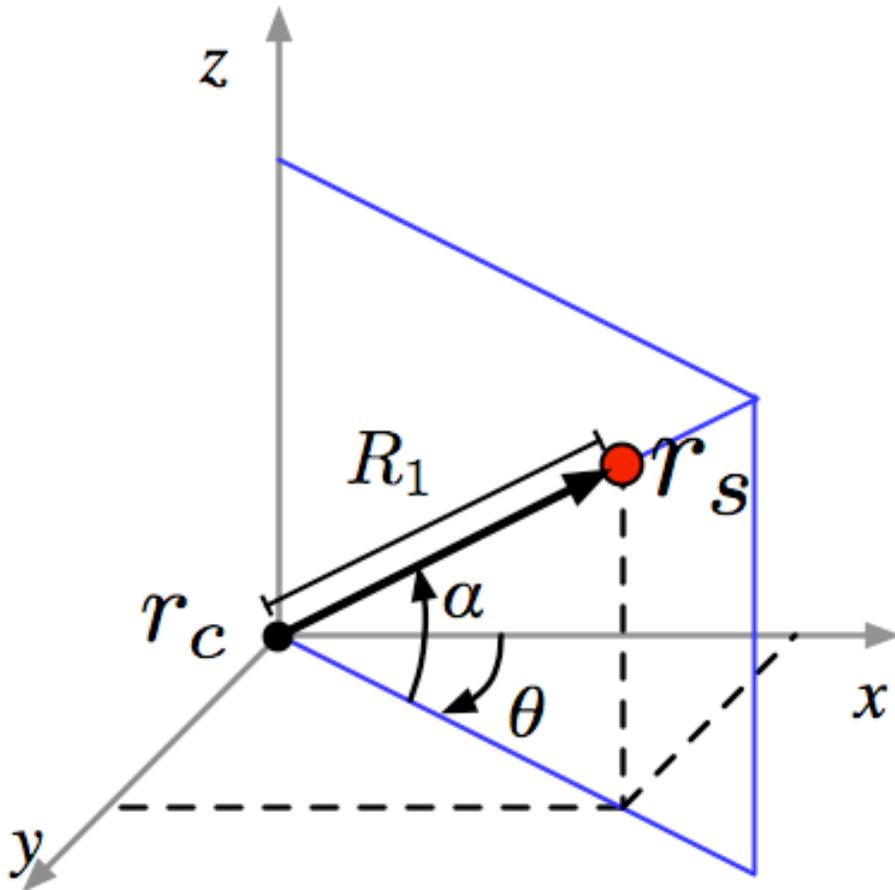
$$\dot{\tilde{\theta}}^{ave} = \frac{1}{\omega} \frac{\sin(\tilde{\theta}^{ave})}{|r_c|^{ave}} \left(V_c - \gamma |r_c|^{ave^2} \right)$$



The “Average Dynamics”



3D - UUV or UAV



$$\dot{y}_c = V_c \cos(\alpha) \sin(\theta)$$

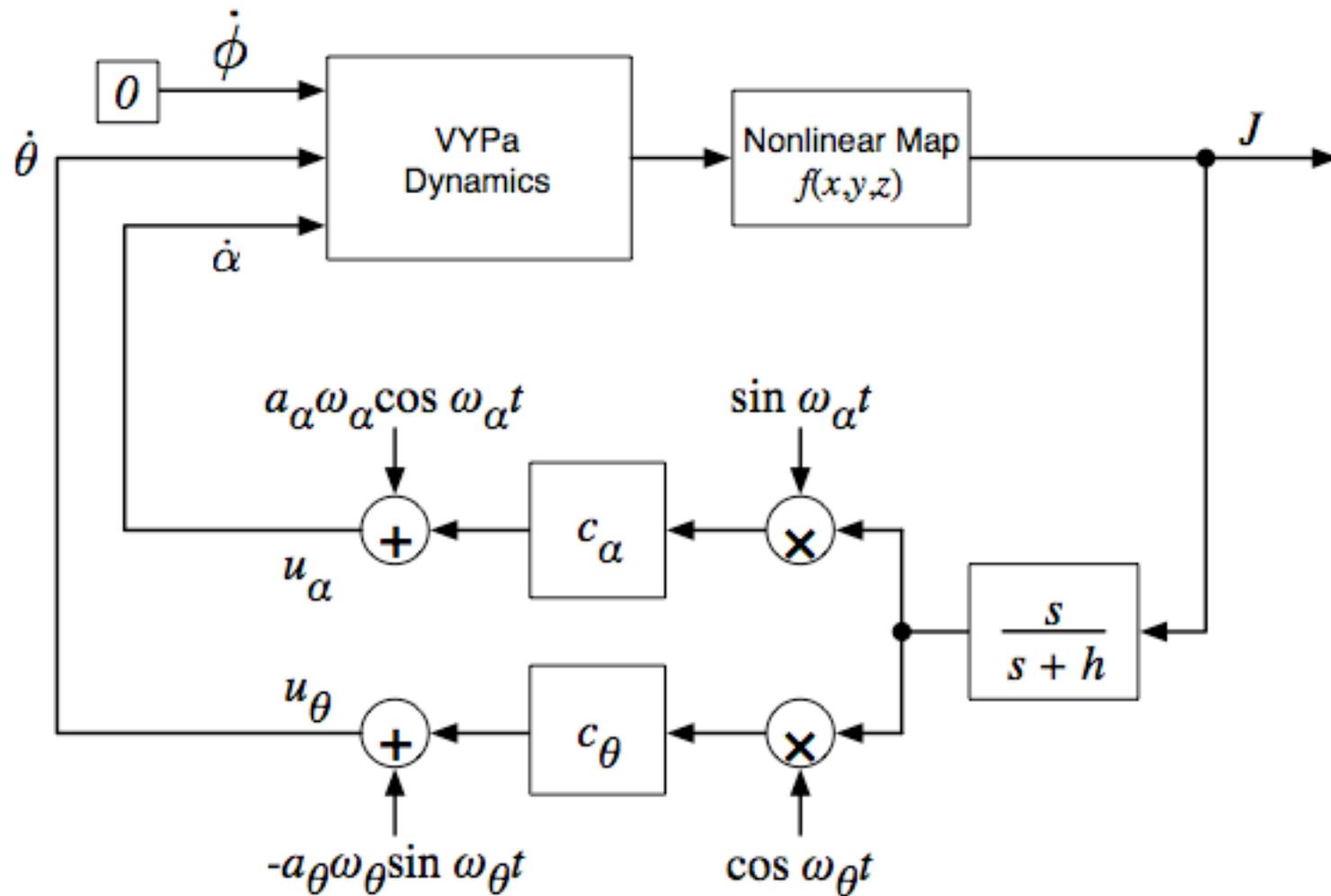
$$\dot{x}_c = V_c \cos(\alpha) \cos(\theta)$$

$$\dot{z}_c = V_c \sin(\alpha)$$

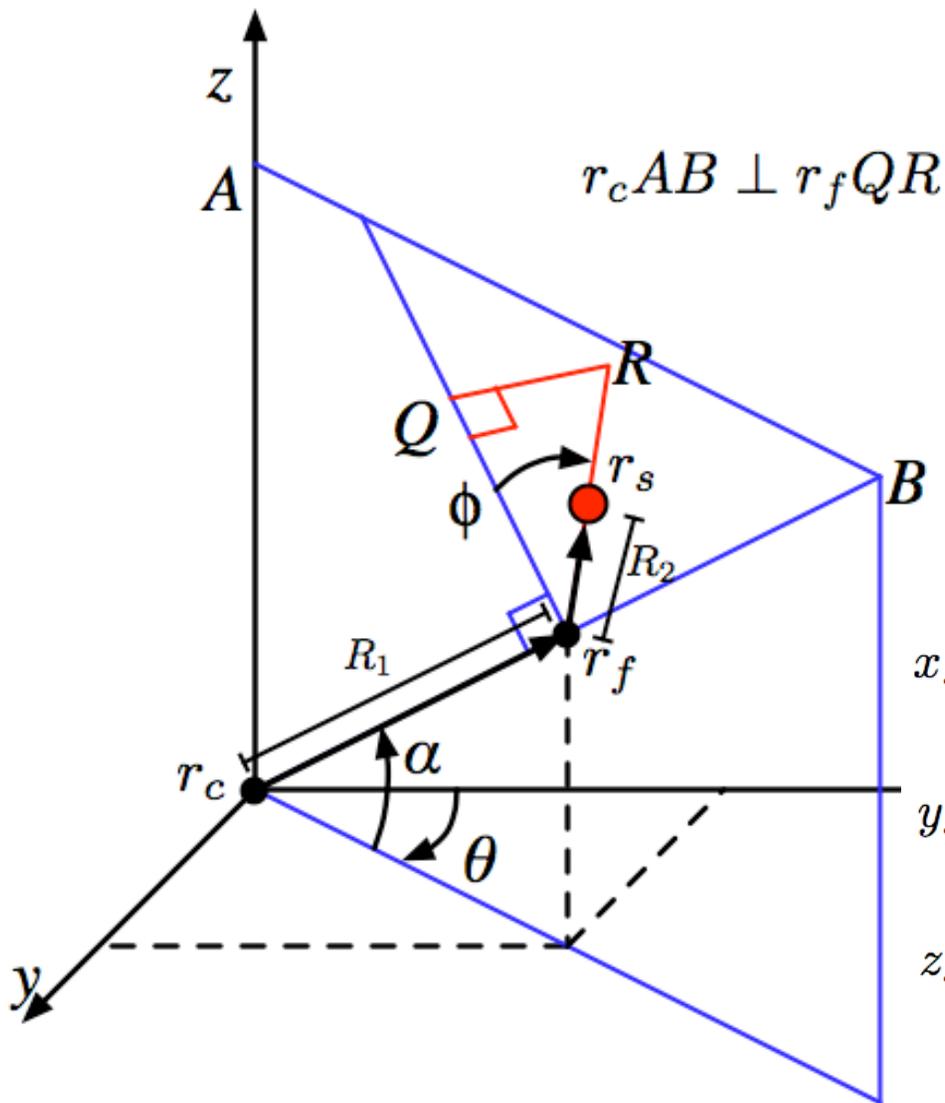
$$\dot{\theta} = \Omega_2$$

$$\dot{\alpha} = \Omega_1$$

Yaw and Pitch Actuated



Vehicle with Const Fwd Velocity and Const Pitch Up Velocity, Sensor Off the Vehicle



$$r_cAB \perp r_fQR$$

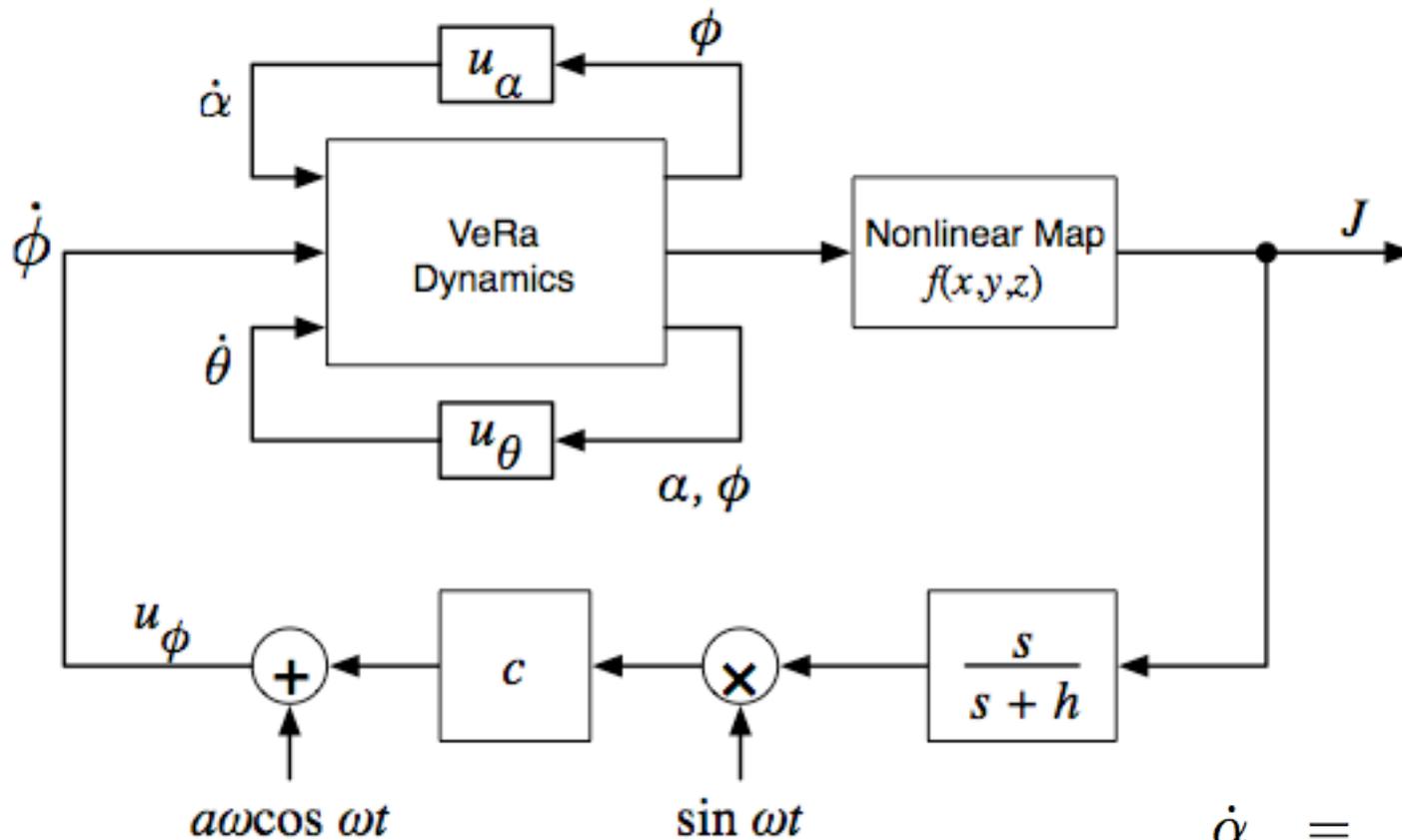
$$\dot{y}_c = V_c \cos(\alpha) \sin(\theta)$$

$$\dot{x}_c = V_c \cos(\alpha) \cos(\theta)$$

$$\dot{z}_c = V_c \sin(\alpha)$$

$$\begin{aligned} x_s &= x_c + R_1 \cos \alpha \cos \theta \\ &\quad + R_2 (-\cos \phi \sin \alpha \cos \theta + \sin \phi \sin \theta) \\ y_s &= y_c + R_1 \cos \alpha \sin \theta \\ &\quad + R_2 (-\cos \phi \sin \alpha \sin \theta - \sin \phi \cos \theta) \\ z_s &= z_c + R_1 \sin \alpha + R_2 \cos \phi \cos \alpha, \end{aligned}$$

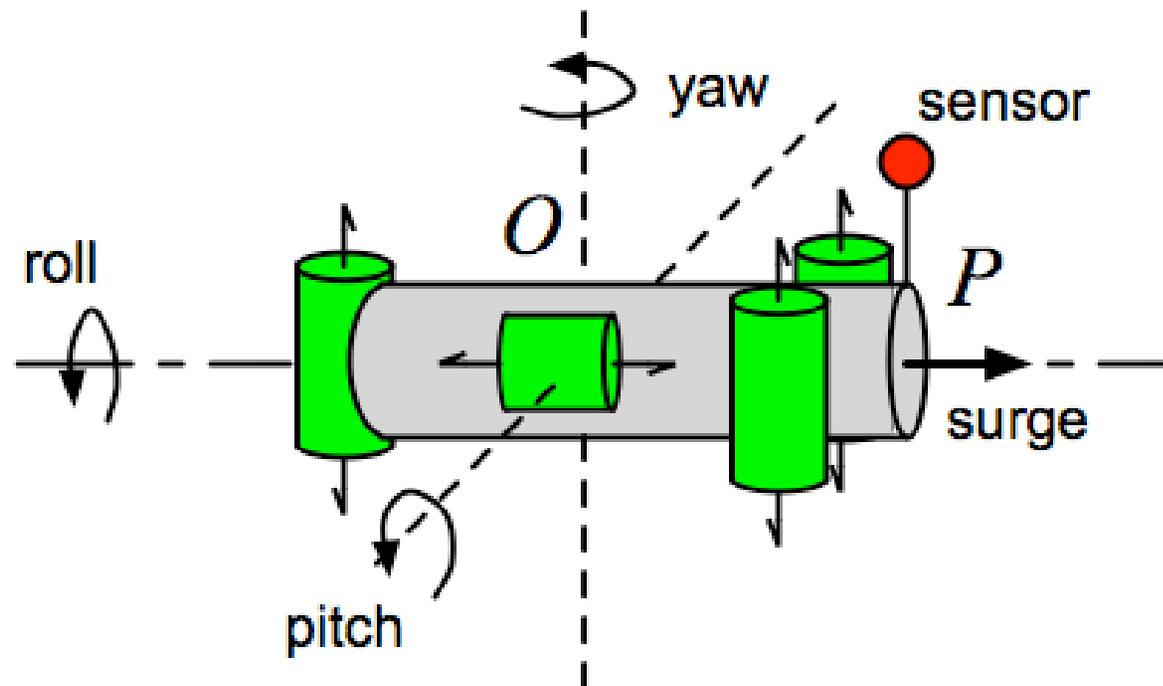
Roll Actuated



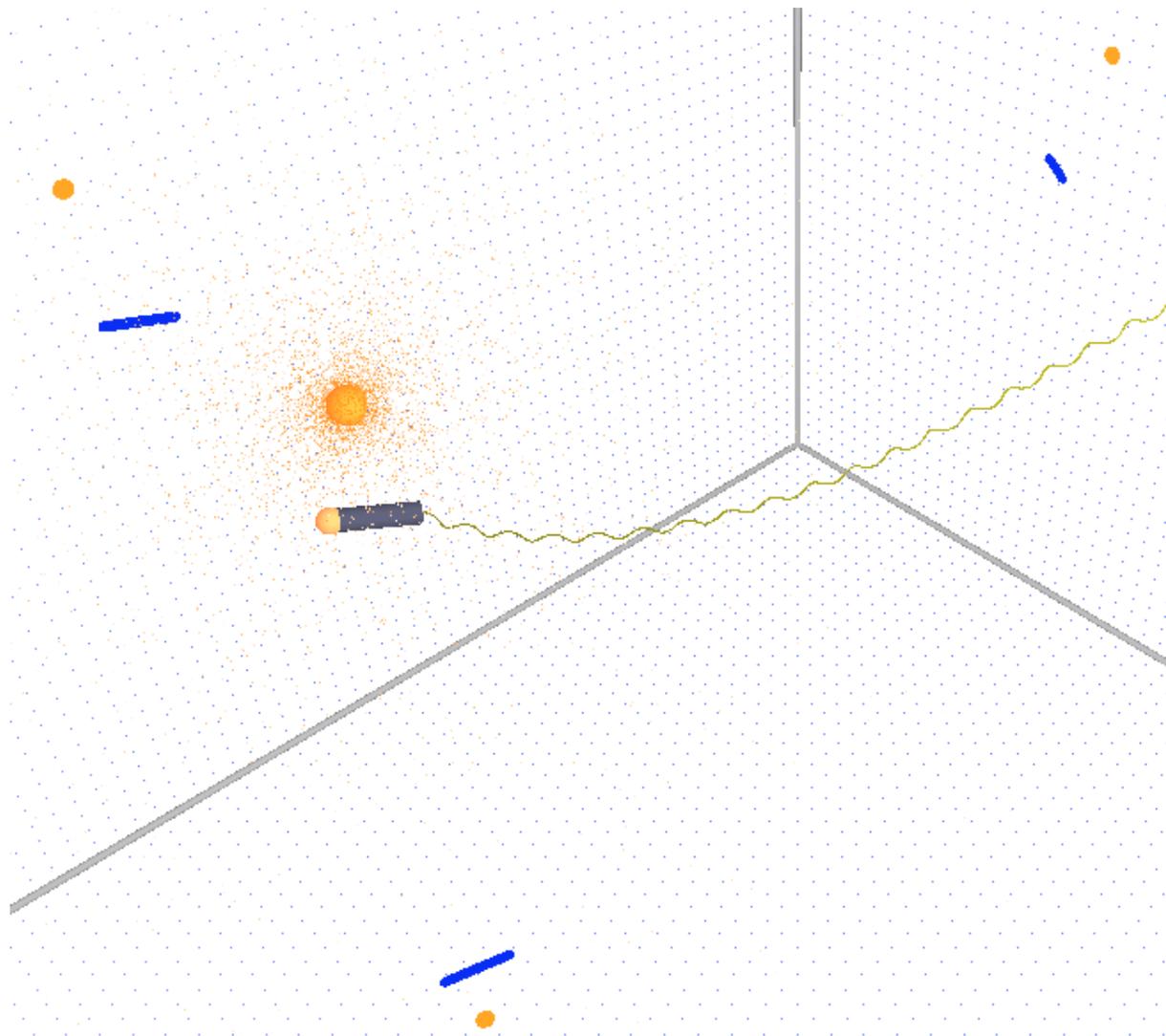
$$\dot{\alpha} = \frac{v_2}{r_1} \cos \phi$$

$$\dot{\theta} = -\frac{v_2 \sin \phi}{r_1 \cos \alpha}$$

3D Vehicle Design

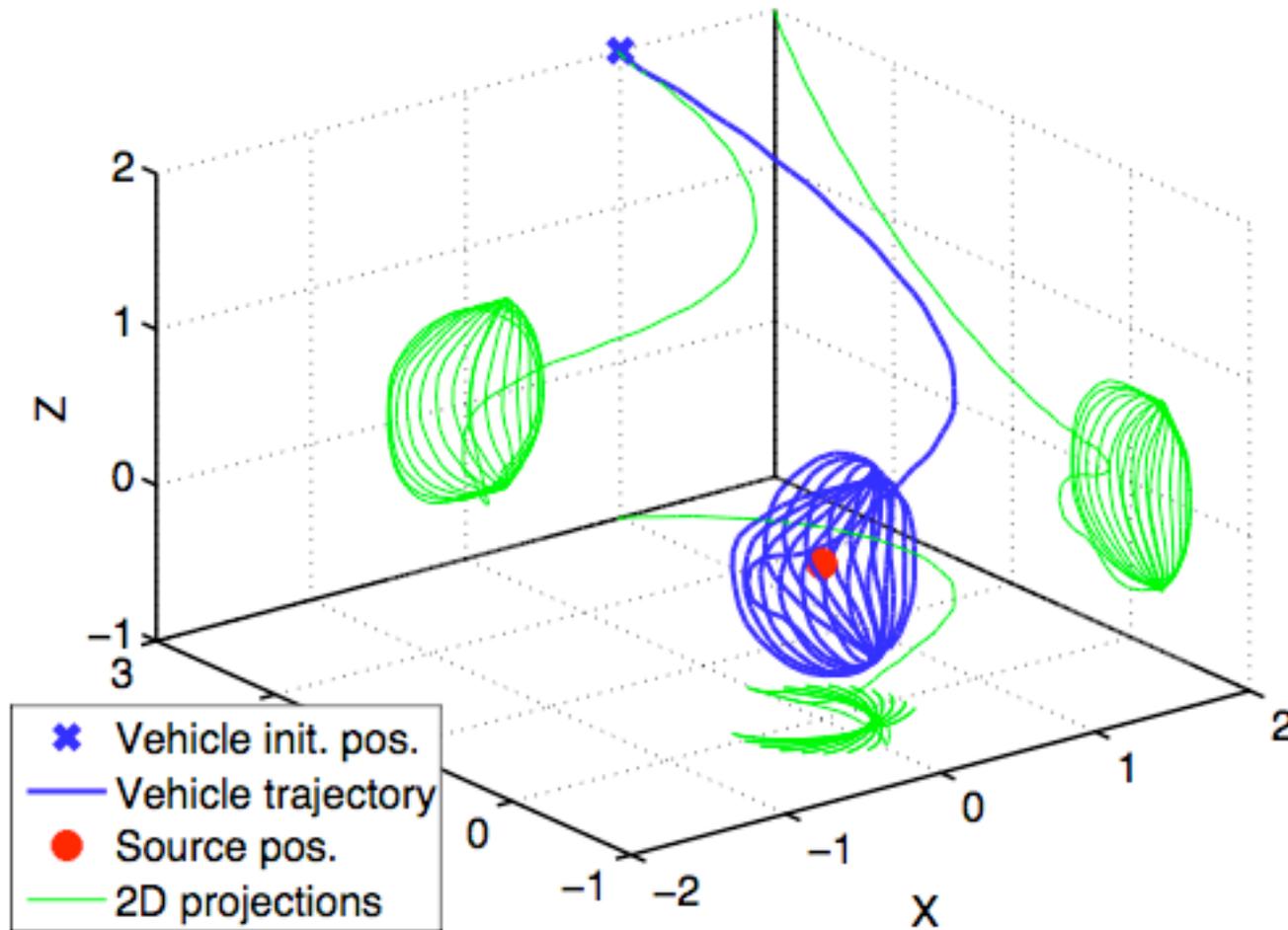


3D Movies



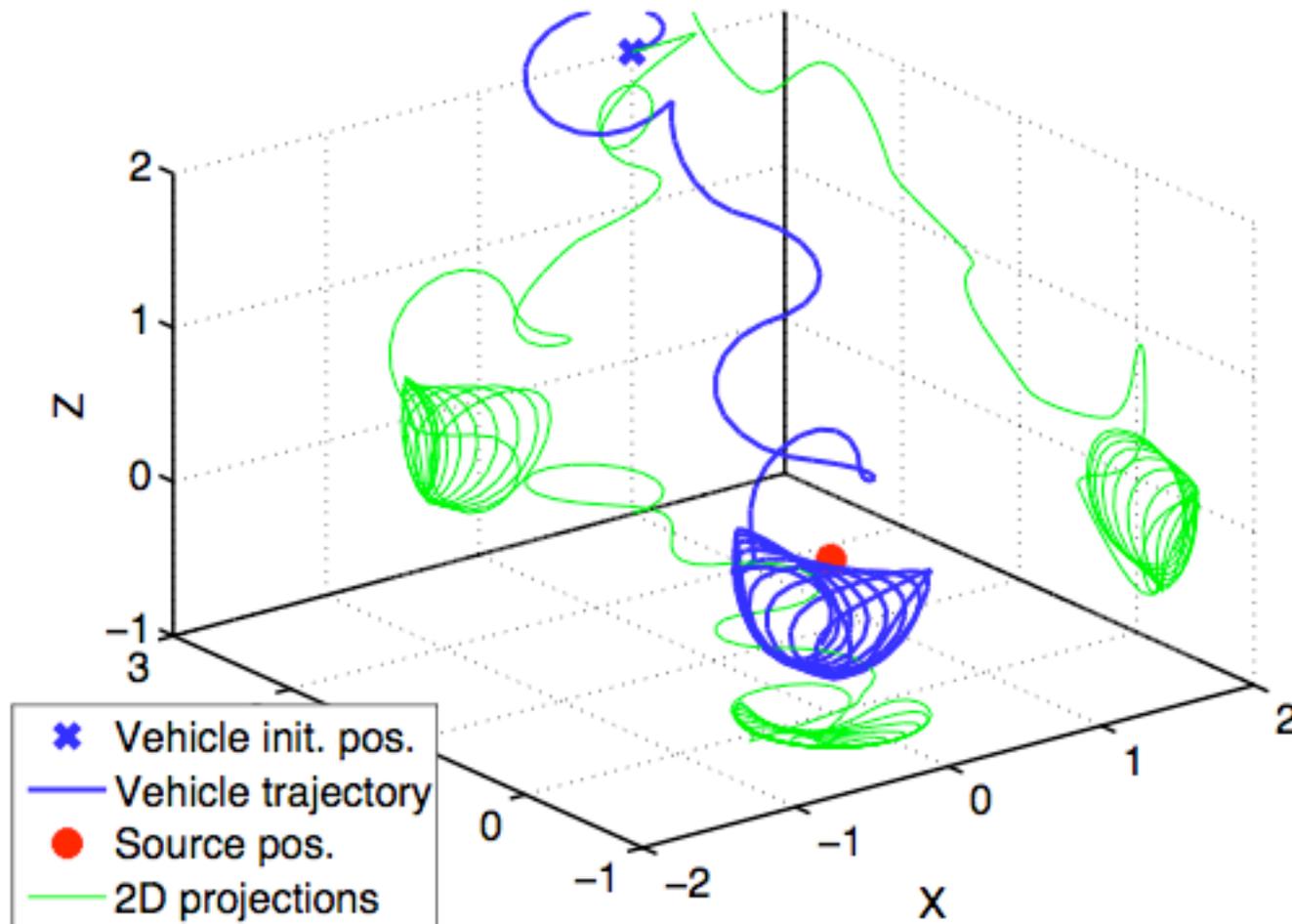
3D Boundary Tracing: Yaw+Pitch Actuation

Torpedo Tracing a Level Set

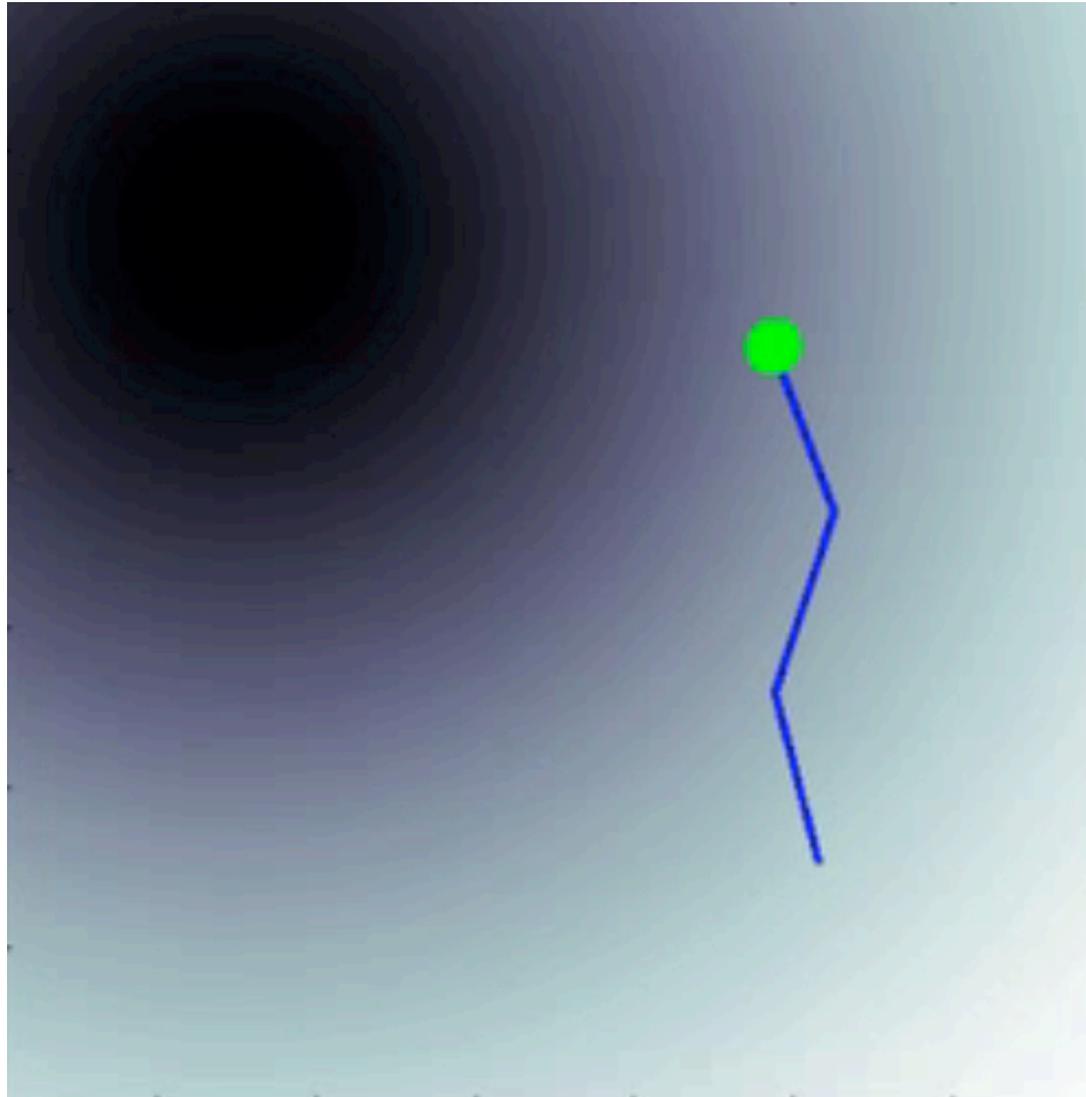


3D Boundary Tracing: Roll Actuation

Satellite Tracking a Level Set



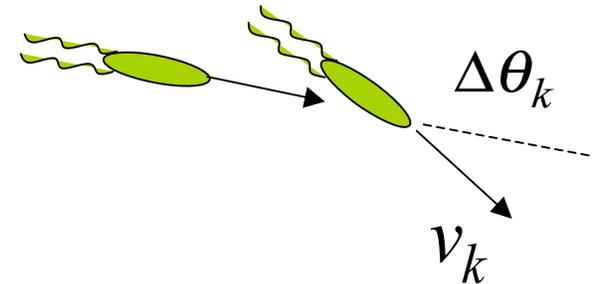
Food-Seeking “Fish”





Mimicking *E Coli*

- *E Coli* motility has two phases, **run** and **tumble**
 - During **run** phase all flagella spin counter clockwise and particle moves forward
 - During **tumble** phase some flagella spin clockwise and bacterium changes orientation
- *E Coli* reacts to spatial gradients
- Change in direction during tumbles is biased in forward direction
 - mean direction change is $\pm 68^\circ$ (Berg and Brown, 1972)

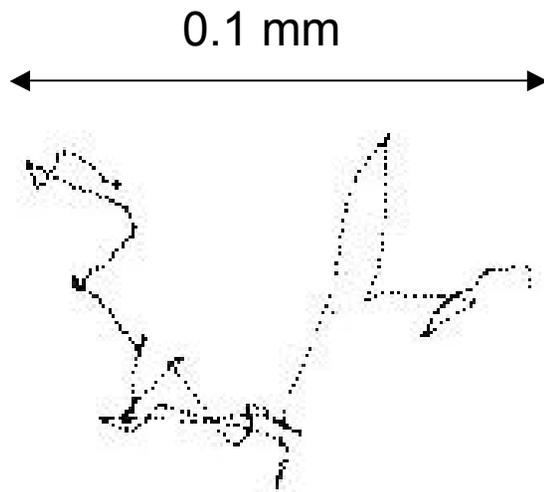


- SIMULATION

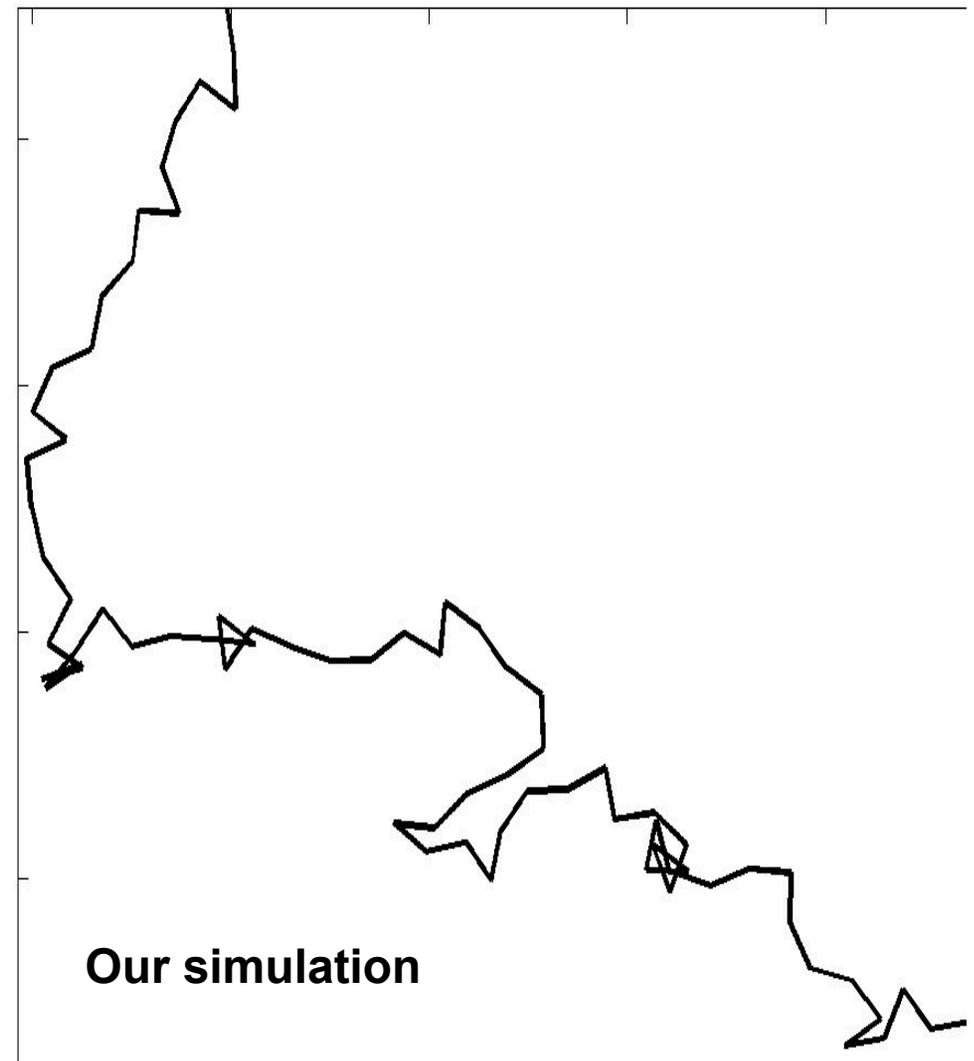
- Run durations constant, tumble angle θ controlled by discrete-time ES
- Measurement of local nutrient concentration at end of run

$$\theta_{k+1} = \theta_k + w_k + \gamma w_{k-1} \frac{z-1}{z+h} \left[J(x(\theta_k), y(\theta_k)) \right]$$

Zooming In



- 30 sec trace for individual bacterium, illustrating run and tumble phases
- *Berg (2000)*



3D *E Coli*

