UTARI UNIVERSITY OF TEXAS AT ARLINGTON RESEARCH INSTITUTE

The University of Texas ARLINGTON

F.L. Lewis, NAI

Moncrief-O'Donnell Chair, UTA Research Institute (UTARI)
The University of Texas at Arlington, USA
and

Foreign Consulting Professor,
Chongqing University, China

Adaptive Control, Robust Control & Neural Networks for Control of Nonlinear Processes and Systems

Importance of Feedback Control

Darwin 1850- FB and natural selection
Vito Volterra 1890- FB and fish population balance
Adam Smith 1760- FB and international economy
James Watt 1780- FB and the steam engine
FB and cell homeostasis

The resources available to most species for their survival are meager and limited

Nature uses Optimal control

Feedback Control Systems

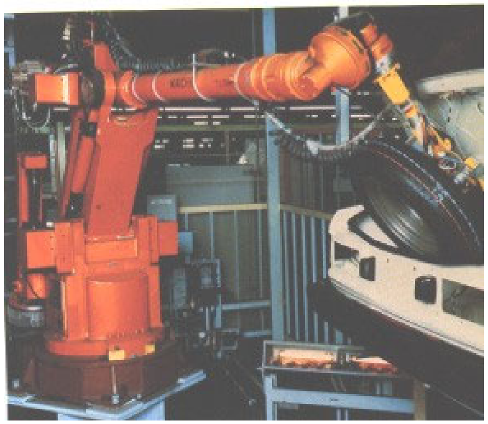
Aircraft autopilots
Car engine controls
Ship controllers
Compute Hard disk drive controllers
Industry process control – chemical, manufacturing
Robot control

Industrial Revolution –
  Windmill control, British millwrights - 1600s
  Steam engine and prime movers
    James Watt 1769
Steamship
Steam Locomotive boiler control

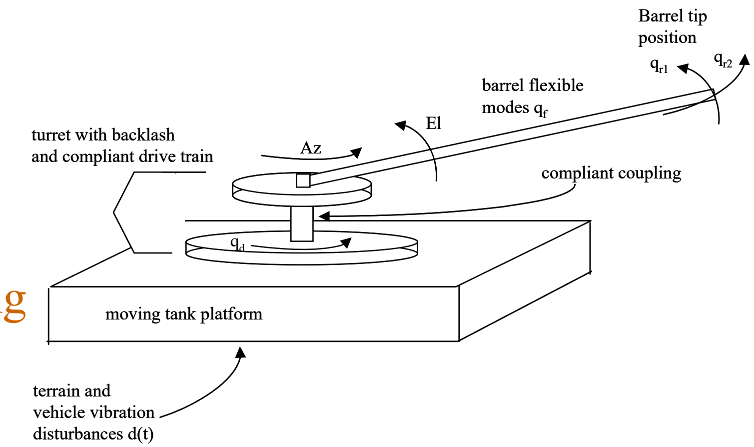
Sputnik 1957
Aerospace systems

Relevance- Machine Feedback Control

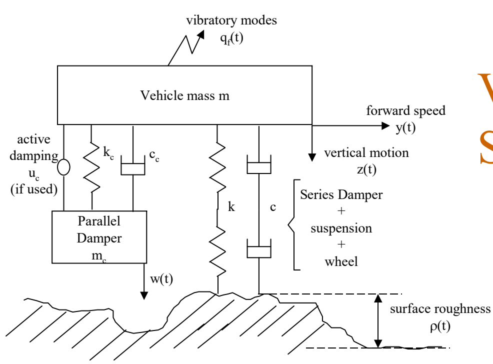
High-Speed Precision Motion Control with unmodeled dynamics, vibration suppression, disturbance rejection, friction compensation, deadzone/backlash control



Industrial Machines



Satellite pointing
Land Systems



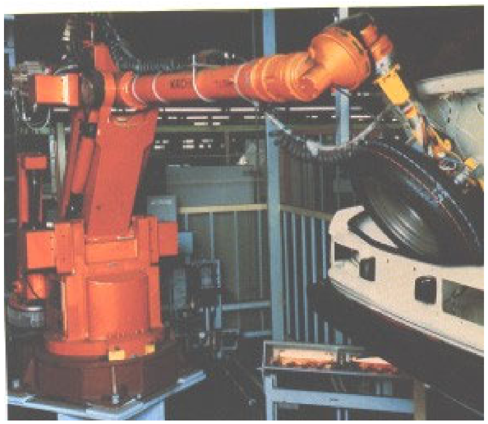
Single-Wheel/Terrain System with Nonlinearities

Vehicle Suspension

Aerospace

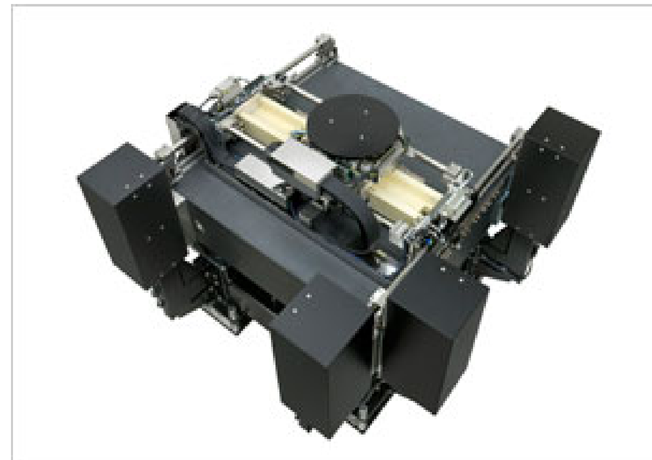
Relevance- Industrial Process Control

Precision Process Control with unmodeled dynamics, disturbance rejection, time-varying parameters, deadzone/backlash control



Industrial Machines

XY Table



Chemical Vapor Deposition

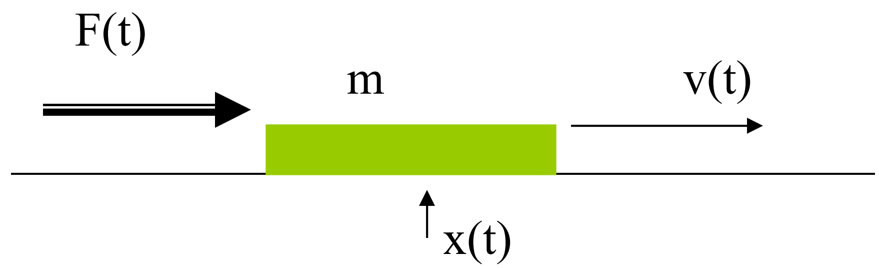
Autoclave

Lagrange Dynamical systems

Newton's Law

$$F = ma = m\ddot{x}$$

$$\ddot{x} = \frac{F(t)}{m} \equiv u(t)$$



Lagrange's Eqs. Of Motion $\Longrightarrow$

Industrial Process and Motion Systems (Vehicles, Robots)

Control Input

$$M(\dot{q})\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = B(q)\tau$$

inertia

Coriolis/centripetal force

gravity

friction

disturbances

Actuator problems

Dynamical System Models

Continuous-Time Systems | **Discrete-Time Systems**

Nonlinear system

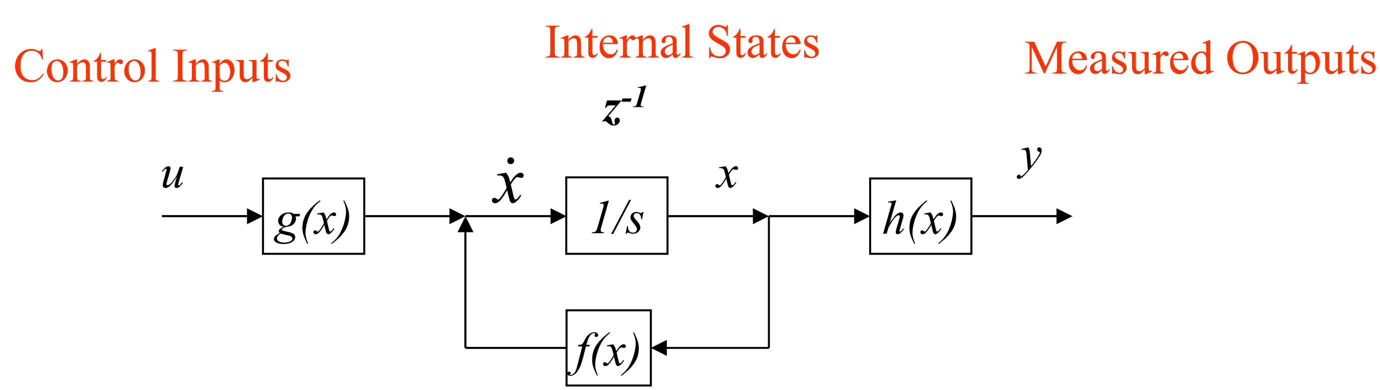
$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

$$x_{k+1} = f(x_k) + g(x_k)u_k$$
$$y_k = h(x_k)$$

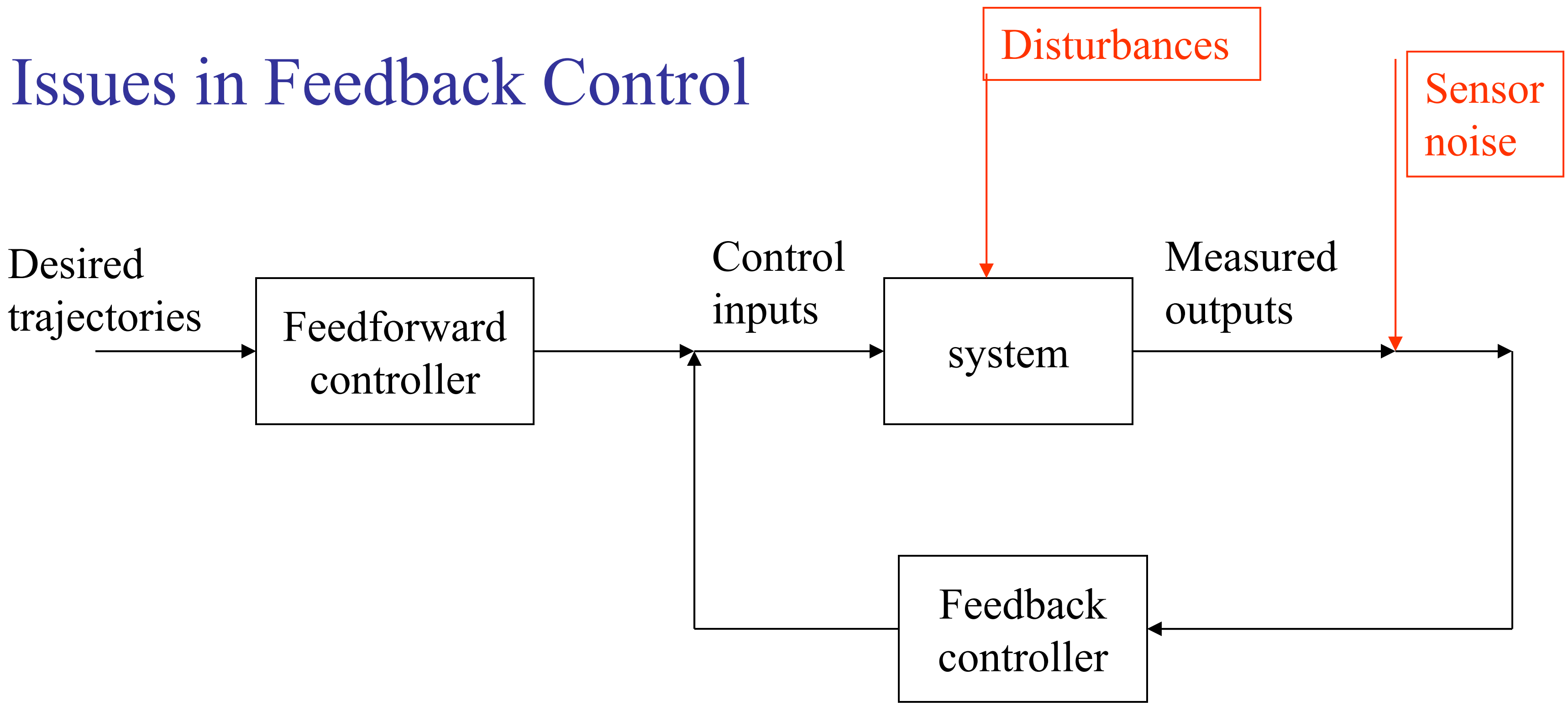
Linear system

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

$$x_{k+1} = Ax_k + B_k$$
$$y_k = Cx_k$$



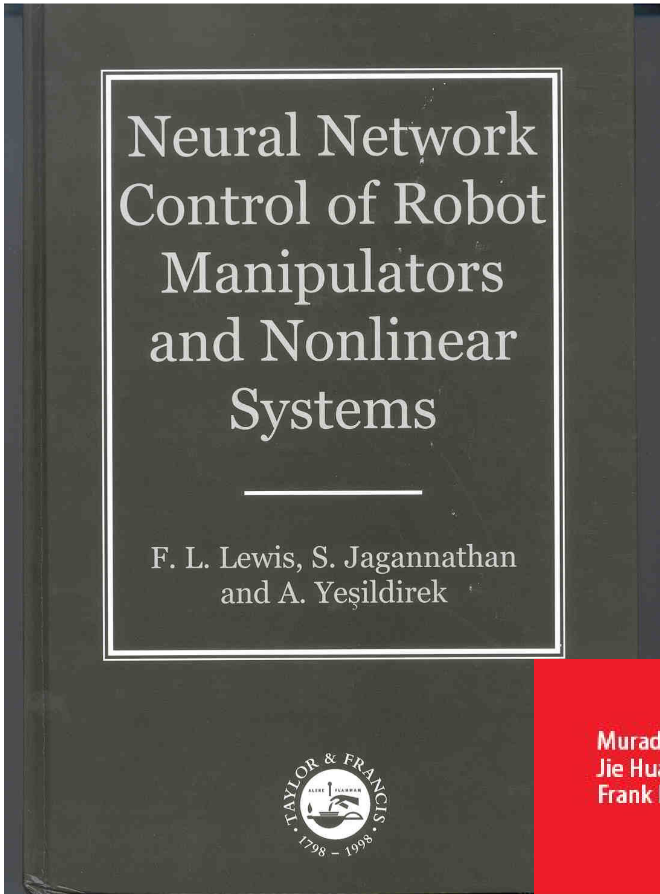
Issues in Feedback Control

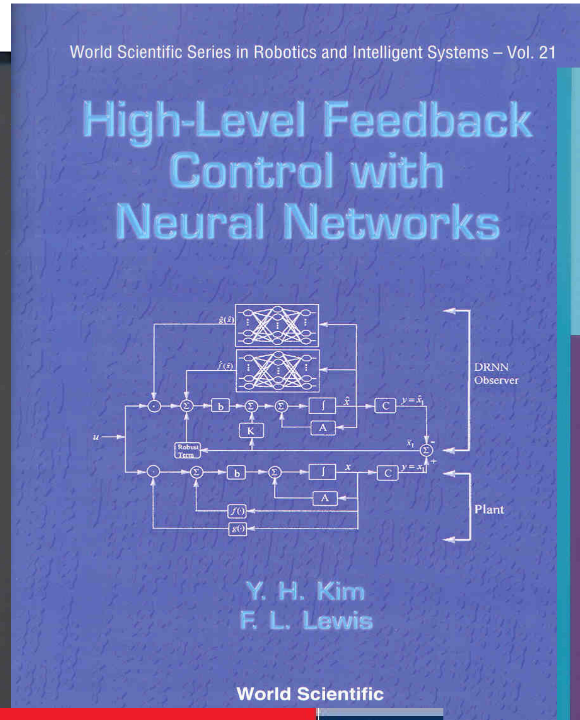


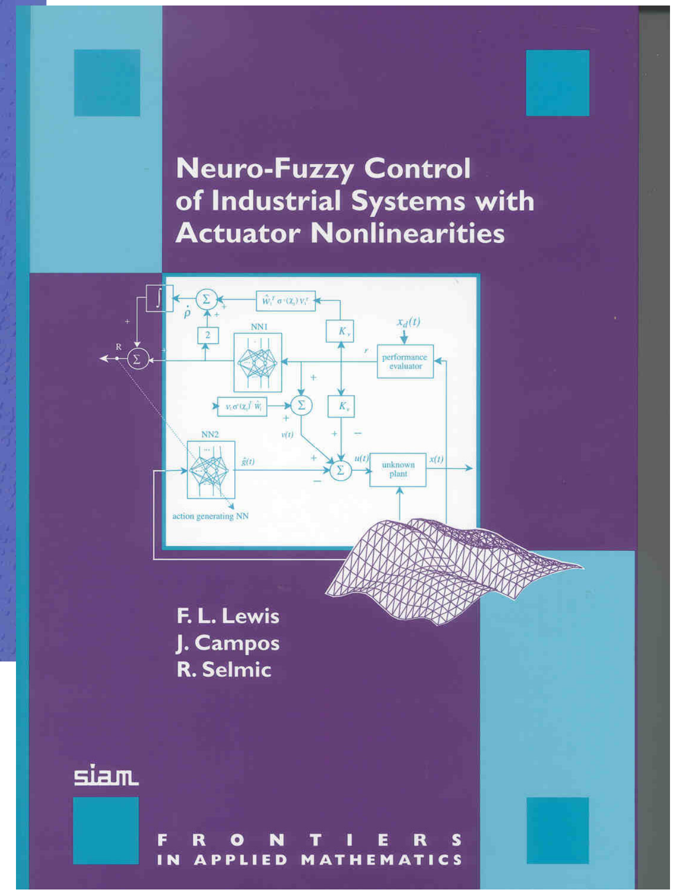
Stability
Tracking
Boundedness
Robustness
- to disturbances
- to unknown dynamics

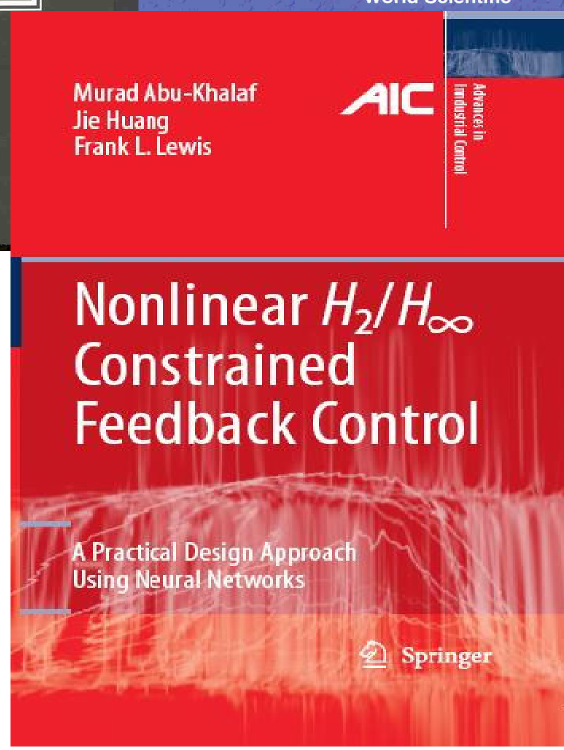
Unknown Process dynamics
Process Nonlinearities
Unknown Disturbances

Feedback Linearization
Adaptive Control
Neural Networks for Control
Neural-adaptive Control









6 US Patents

Sponsored by:
China Qian Ren
China Project 111
US NSF, ARO, ONR,
AFOSR

Example 1. Linear System

Feedback Linearization

$$y = \frac{1}{s^2 + a_1 s + a_2} u \qquad \ddot{y} + a_1\dot{y} + a_2 y = u$$

desired to track a reference input $y_d(t)$

Tracking error $e = y_d - y$

$$\ddot{e} = \ddot{y}_d + a_1\dot{y} + a_2 y - u$$

Sliding variable $r = \dot{e} + \Lambda e$

Error dynamics $\dot{r} = \ddot{e} + \Lambda\dot{e} = \ddot{y}_d + \Lambda\dot{e} + a_1\dot{y} + a_2 y - u$

Define Auxiliary input $u = v + \ddot{y}_d + \Lambda\dot{e}$

Then Error dynamics $\dot{r} = a_1\dot{y} + a_2 y - v$

Of the form $\dot{r} = f(x) - v$

Unknown parameters

Known Regression Vector

Unknown function

$$f(x) = a_1\dot{y} + a_2 y = \begin{bmatrix} a_1 & a_2 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = W^T \phi(x)$$

Example 2. Nonlinear Lagrange System

Feedback Linearization

$$\ddot{y} + d(y,\dot{y}) + k(y) = u$$

unknown nonlinear friction

unknown nonlinear damping term

desired to track a reference input $y_d(t)$

Lagrangian System Appears in:
- Process control
- Mechanical systems
- Robots

Tracking error $e = y_d - y$

$$\ddot{e} = \ddot{y}_d + d(y,\dot{y}) + k(y) - u$$

Sliding variable $r = \dot{e} + \Lambda e$

Error dynamics $\dot{r} = \ddot{e} + \Lambda\dot{e} = \ddot{y}_d + \Lambda\dot{e} + d(y,\dot{y}) + k(y) - u$

Define Auxiliary input $u = v + \ddot{y}_d + \Lambda\dot{e}$

Then Error dynamics $\dot{r} = f(x) - v$

with $f(x) = d(y,\dot{y}) + k(y)$

Assume Linear in the Parameters (LIP)

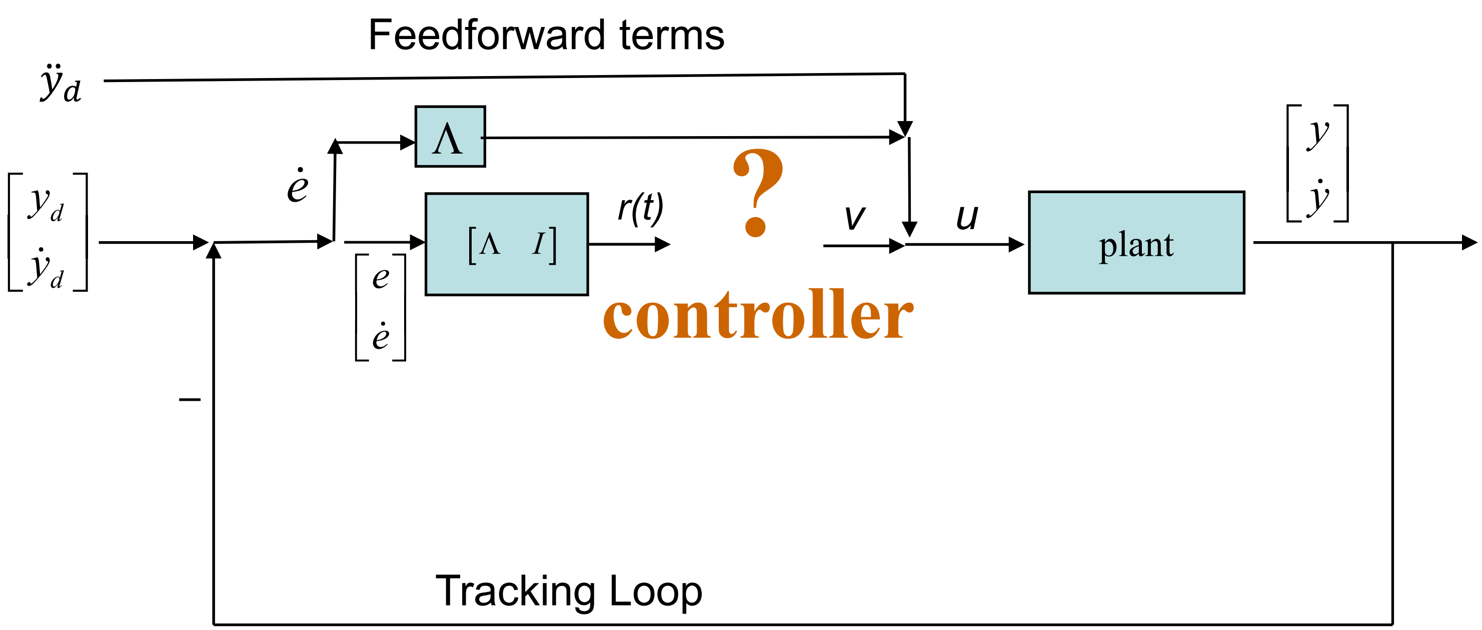
$$f(x) = \begin{bmatrix} D & K \end{bmatrix} \begin{bmatrix} d_1(y,\dot{y}) \\ k_1(y,\dot{y}) \end{bmatrix} = W^T \phi(x)$$

Known possibly nonlinear regression function

Unknown parameters

Feedback Linearization Controller

$$r = \dot{e} + \Lambda e \qquad\qquad u = v + \ddot{y}_d + \Lambda \dot{e}$$



The equations give the FB controller structure

Feedback Linearization Controller

$$r = \dot{e} + \Lambda e$$

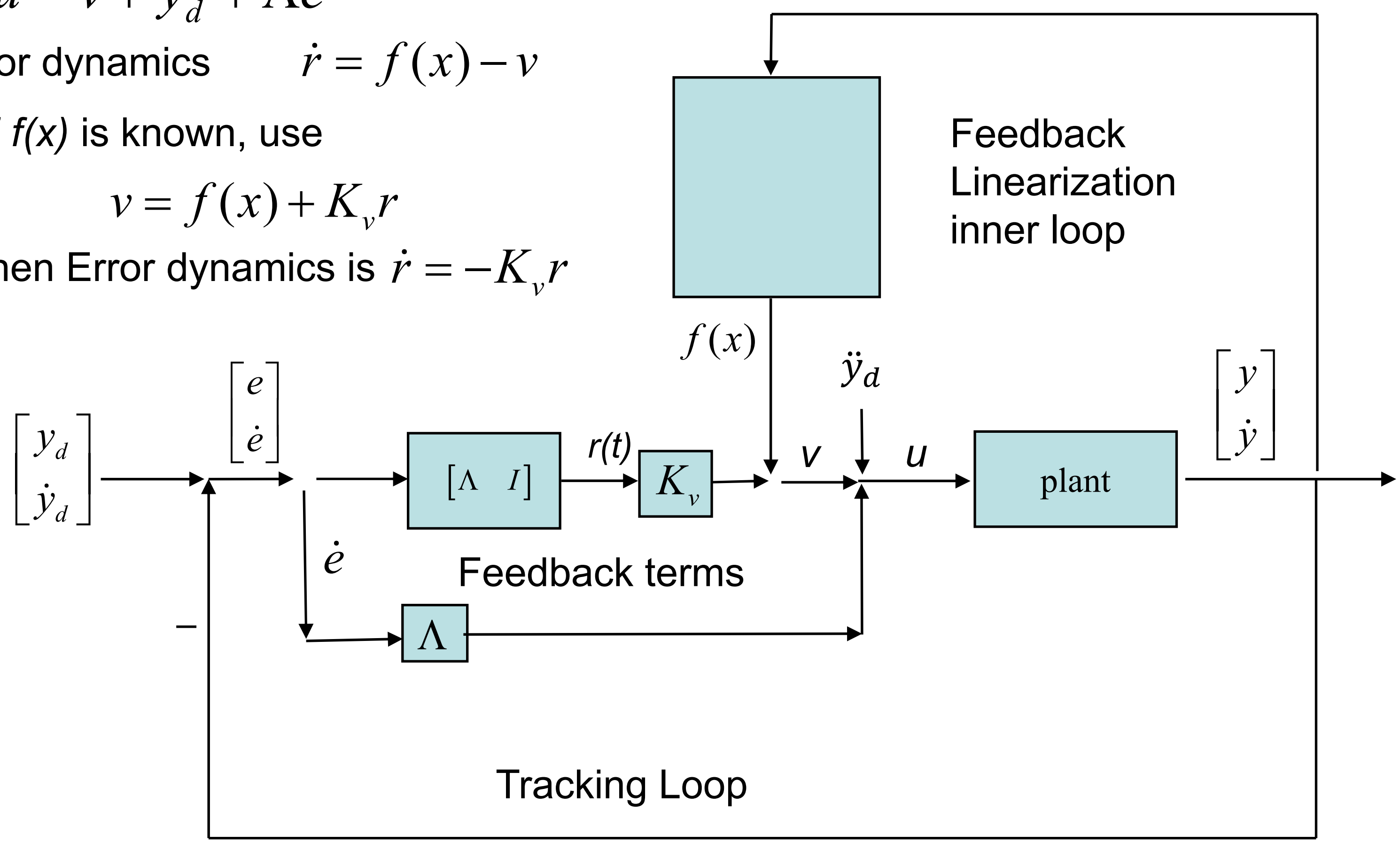
$$u = v + \ddot{y}_d + \Lambda \dot{e}$$

Error dynamics $\quad \dot{r} = f(x) - v$

If *f(x)* is known, use

$$v = f(x) + K_v r$$

Then Error dynamics is $\dot{r} = -K_v r$



The equations give the FB controller structure

Adaptive Control

Error Dynamics

$$\dot{r} = f(x) - v$$

r(t)= control error

Control input

Unknown nonlinearities

Assume: *f(x)* is known to be of the structure

$$f(x) = W^T \phi(x)$$

Known basis set= regression vector – DEPENDS ON THE SYSTEM

Unknown parameter vector

LINEAR-IN-THE-PARAMETERS (LIP)

Error Dynamics

$$\dot{r} = W^T \phi(x) - v$$

Adaptive Control

Controller

Pos. def. control gain

$$v = \hat{f}(x) + K_v r = \hat{W}^T(t)\phi(x) + K_v r$$

ESTIMATE of unknown parameters

closed-loop system becomes

$$\dot{r} = W^T\phi(x) - v = W^T\phi(x) - \hat{W}^T\phi(x) - K_v r$$
$$\dot{r} = \tilde{W}^T\phi(x) - K_v r$$

Est. error drives the control error

Parameter estimation error

$$\tilde{W}(t) = W - \hat{W}(t)$$

Parameter estimate is updated (tuned) using the adaptive tuning law

$$\frac{d\hat{W}}{dt} = \dot{\hat{W}} = F\phi(x)r^T$$

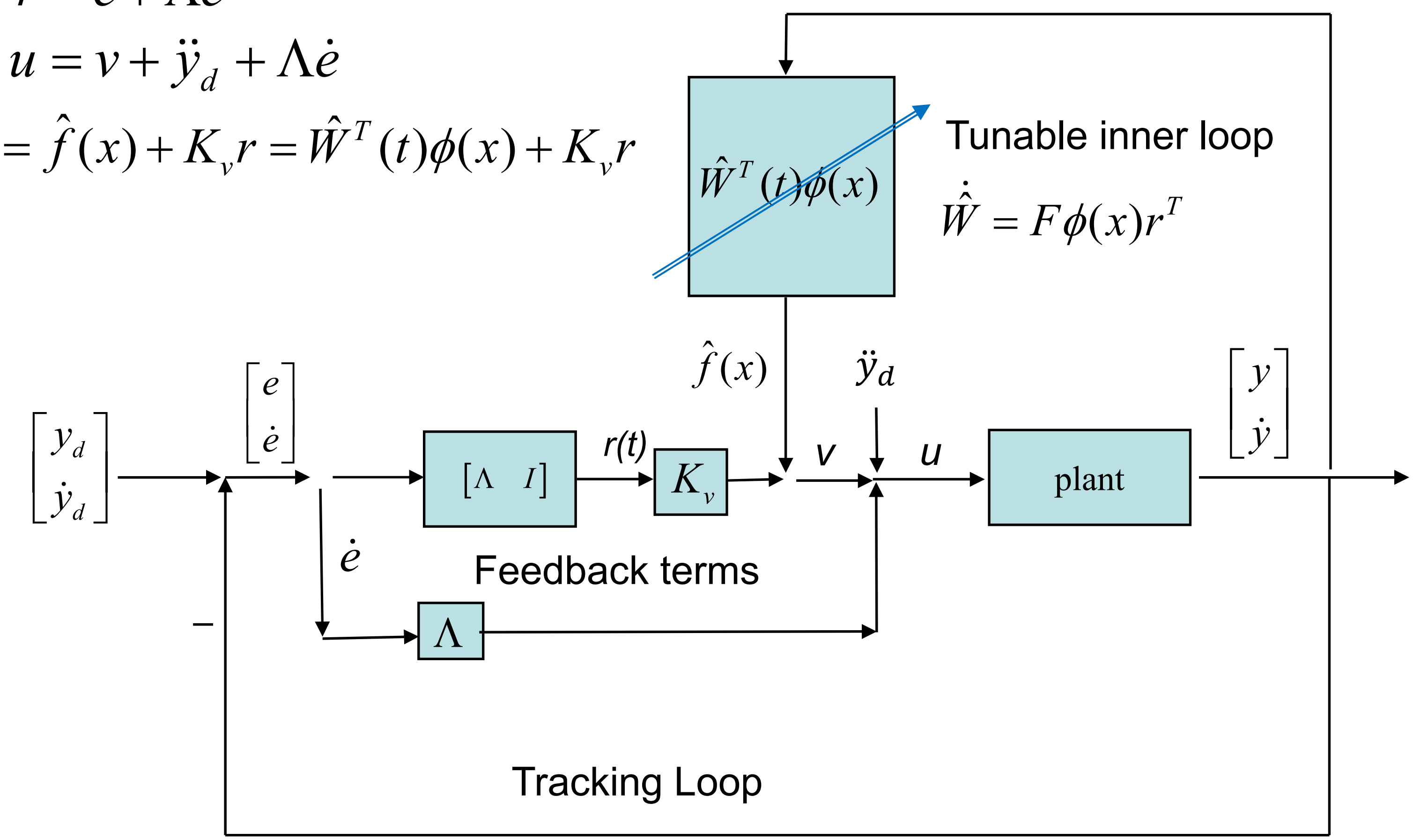
Feedback Linearization Adaptive Controller

A dynamic controller

$$r = \dot{e} + \Lambda e$$

$$u = v + \ddot{y}_d + \Lambda \dot{e}$$

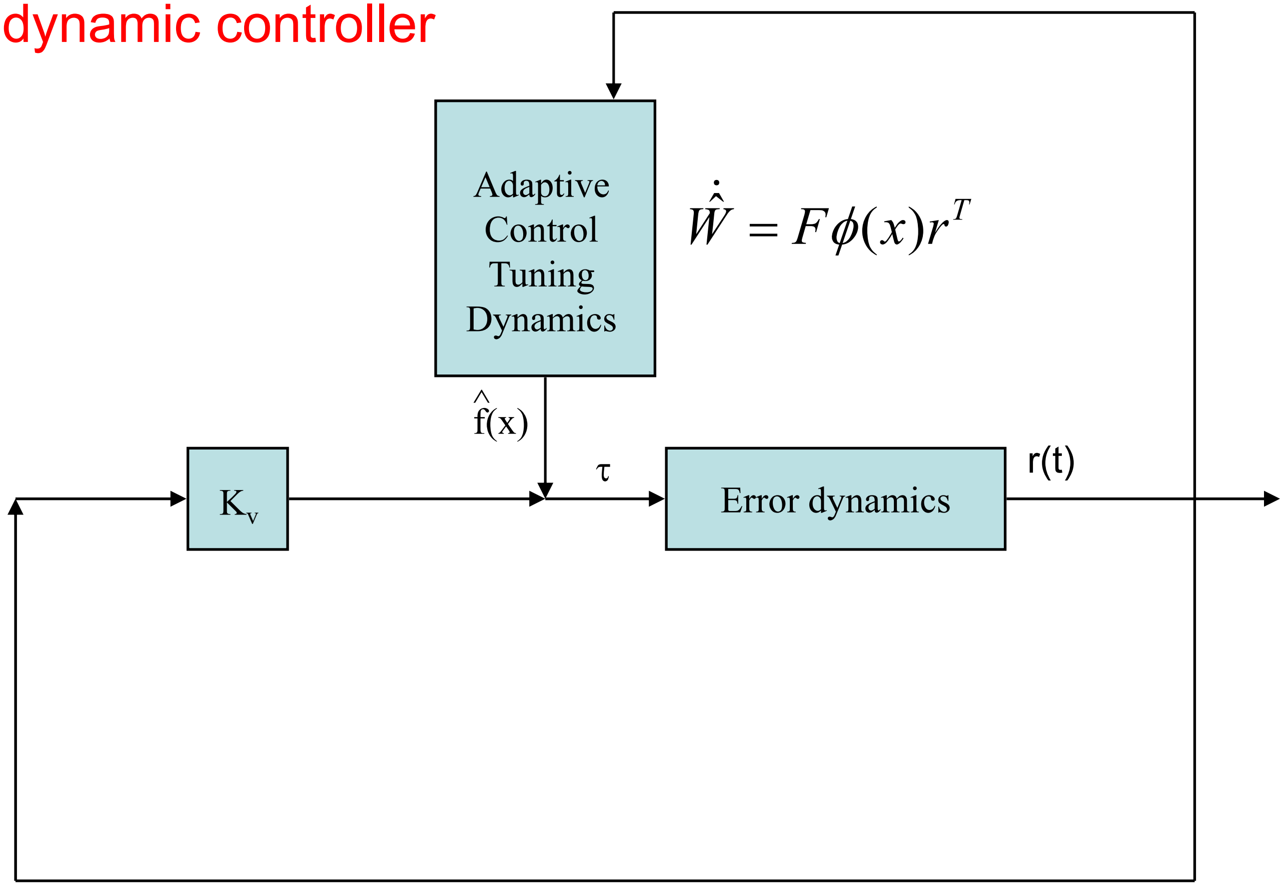
$$v = \hat{f}(x) + K_v r = \hat{W}^T(t)\phi(x) + K_v r$$



The equations give the FB controller structure

Adaptive Controller

A dynamic controller



Must define extra state variables for adaptive control parameters $\hat{W}(t)$
Extra complications to implement adaptive controller

Adaptive Control

Performance of Adaptive Controller: Using the adaptive controller, the closed-loop system is asymptotically stable, i.e. the control error *r(t)* goes to zero.

If an additional Persistence of Excitation (PE) condition holds, the parameter estimates converge to the actual unknown parameters.

Proof:

$$L = \tfrac{1}{2} r^T r + \tfrac{1}{2} tr\{\tilde{W}^T F^{-1} \tilde{W}\} \qquad > 0$$

Frobenius Norm

$$\dot{L} = r^T \dot{r} + tr\{\tilde{W}^T F^{-1} \dot{\tilde{W}}\}$$

$$\dot{r} = \tilde{W}^T \phi(x) - K_v r$$

Error dynamics

$$\dot{L} = r^T \left(\tilde{W}^T \phi(x) - K_v r\right) + tr\{\tilde{W}^T F^{-1} \dot{\tilde{W}}\} = -r^T K_v r + tr\{\tilde{W}^T \left(F^{-1}\dot{\tilde{W}} + \phi(x) r^T\right)\}$$

$$\dot{\hat{W}} = F\phi(x) r^T \quad \text{or} \quad \dot{\tilde{W}} = -F\phi(x) r^T$$

Parameter tuning law

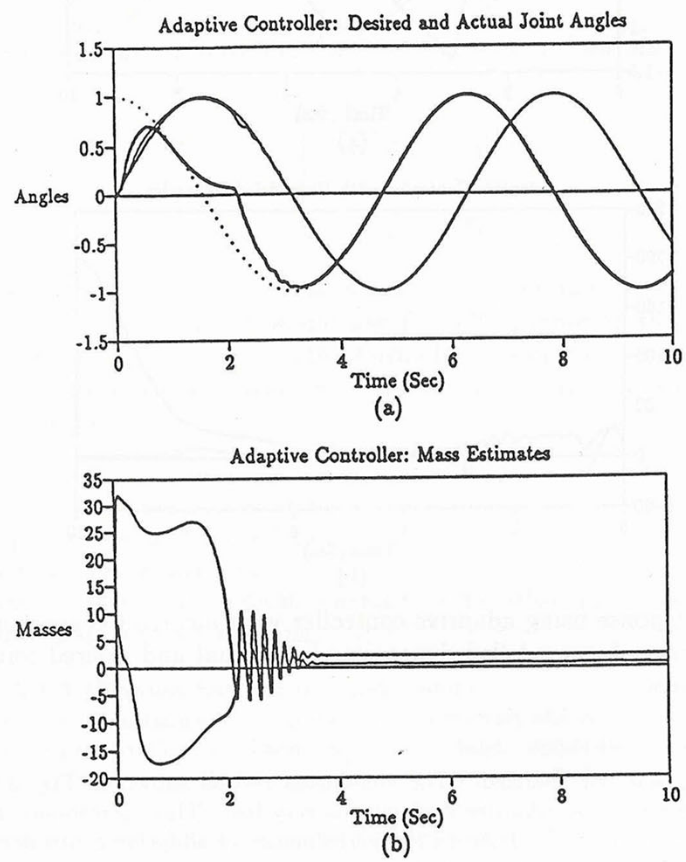
$$\dot{L} = -r^T K_v r \qquad \le 0$$

Therefore Lyapunov shows that $r(t), \tilde{W}(t)$ are bounded

Typical Behavior of Adaptive Controllers

Control errors go to zero and the parameter estimates converge.

This assumes that $f(x) = W^T \phi(x)$ holds exactly, and that there are no disturbances in the system.



Control errors go to zero

Parameter estimates converge

Figure 3.4.3: Response using adaptive controller. (a) Actual and desired joint angles. (b) Mass estimates.

Robust Control

Error dynamics

$$\dot{r} = f(x) - \tau$$

Control input

Unknown nonlinearities

Actual function $f(x) = W^T \phi(x)$

Assume

know a fixed nominal value or estimate $\hat{f}(x) == \hat{W}^T \phi(x)$ for unknown *f(x)*,

estimation error $\tilde{f} = f(x) - \hat{f}(x)$ is bounded like

$$\left\| \tilde{f}(x) \right\| \le F(x)$$

Known bounding function, maybe nonlinear

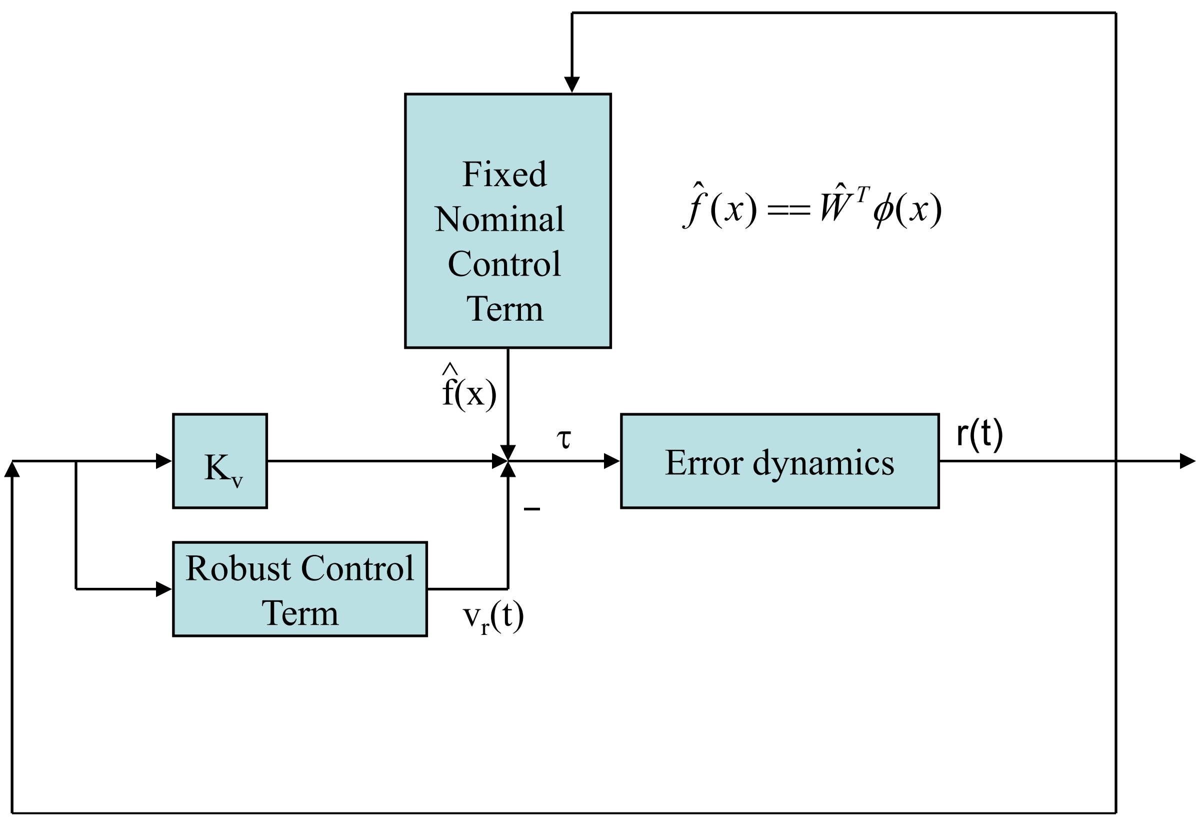
Controller

$$\tau = \hat{f}(x) + K_v r - v_r$$

$$v_r = \begin{cases} -r \dfrac{F(x)}{\|r\|}, & \|r\| \ge \varepsilon \\ -r \dfrac{F(x)}{\varepsilon}, & \|r\| < \varepsilon \end{cases}$$

Robust Controller

A NON Dynamic controller



Robust Controller

Robust Controller A NON Dynamic controller

Example. Linear System $y = \dfrac{1}{s^2 + a_1 s + a_2} u$ $\qquad \ddot{y} + a_1 \dot{y} + a_2 y = u$

Actual system $\ddot{y} + 3\dot{y} + 5y = u$

Assumed estimates $\ddot{y} + 4\dot{y} + 2y = u$

$$\dot{r} = \ddot{e} + \Lambda \dot{e} = \ddot{y}_d + \Lambda \dot{e} + a_1 \dot{y} + a_2 y - u$$

$$= \ddot{y}_d + \Lambda \dot{e} + f(x) - u$$

$$f(x) = a_1 \dot{y} + a_2 y = \begin{bmatrix} a_1 & a_2 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = W^T \phi(x) = \begin{bmatrix} 3 & 5 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

need $\quad u = \ddot{y}_d + \Lambda \dot{e} + f(x) + k_v r$

pick $\quad u = \ddot{y}_d + \Lambda \dot{e} + \hat{f}(x) + k_v r$

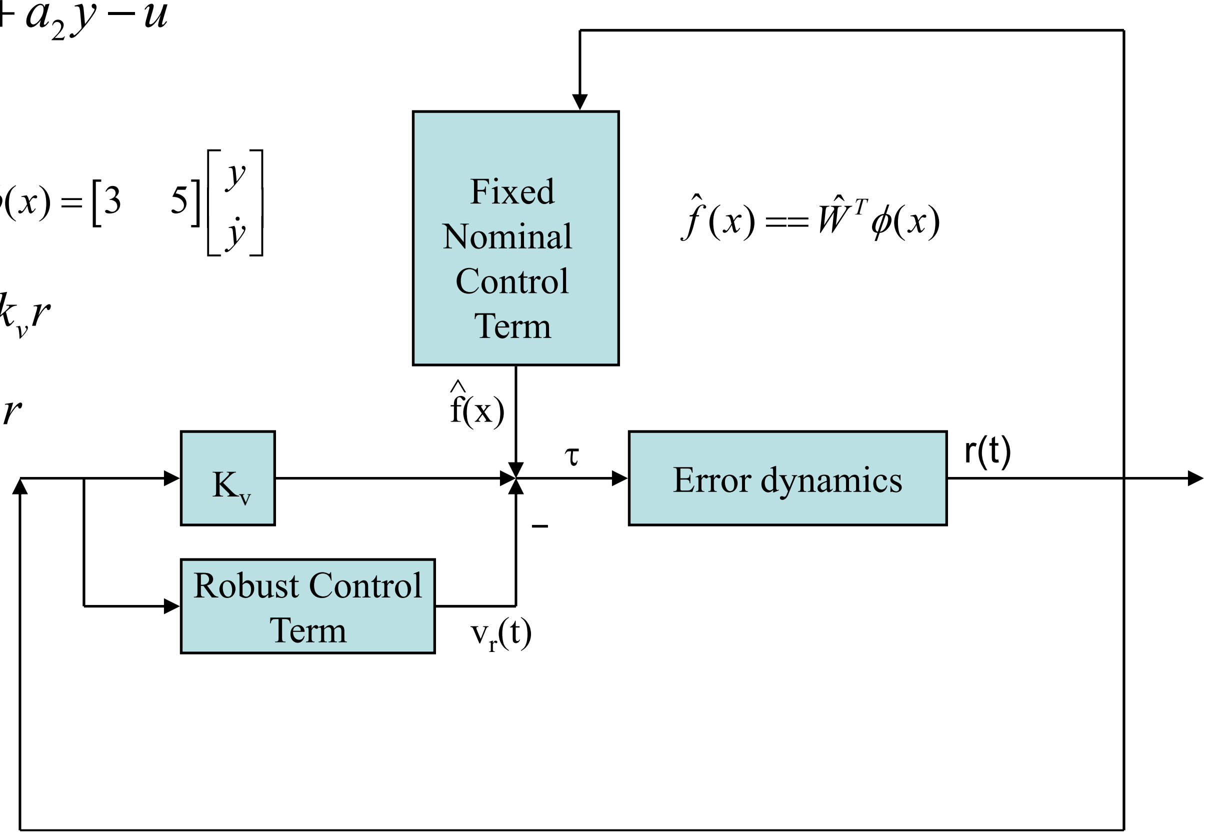
$$\hat{f}(x) == \hat{W}^T \phi(x) = \begin{bmatrix} 4 & 2 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

Then error dynamics is

$$\dot{r} = W^T \phi(x) - \hat{W}^T \phi(x) - K_v r$$

$$= \tilde{W}^T \phi(x) - K_v r$$

This is a system with a bounded disturbance



Robust Controller

Closed-loop Error dynamics

$$\dot{r} = f(x) - \tau = f(x) - \left(\hat{f}(x) + K_v r - v_r\right)$$

$$\dot{r} = \tilde{f}(x) - K_v r + v_r \qquad\qquad \dot{r} = W^T\phi(x) - \hat{W}^T\phi(x) - K_v r = \tilde{W}^T\phi(x) - K_v r$$

Performance of Robust Controller:

With this control, the closed-loop system is bounded stable with $\|r\|$ bounded with a magnitude near ε

Proof:

$$L = \tfrac{1}{2} r^T r$$

$$\dot{L} = r^T \dot{r}$$

$$\dot{L} = r^T\left(\tilde{f}(x) - K_v r + v_r\right) = -r^T K_v r + r^T\left(\tilde{f}(x) + v_r\right)$$

$$\dot{L} \le -\sigma_{\min}(K_v)\|r\|^2 + \|r\| F(x) + r^T v_r$$

$$v_r = \begin{cases} -r\dfrac{F(x)}{\|r\|}, & \|r\| \ge \varepsilon \\ -r\dfrac{F(x)}{\varepsilon}, & \|r\| < \varepsilon \end{cases}$$

Case 1: $\|r\| \ge \varepsilon$

$$\dot{L} \le -\sigma_{\min}(K_v)\|r\|^2 + \|r\| F(x) - \|r\|^2 F(x)/\|r\|$$

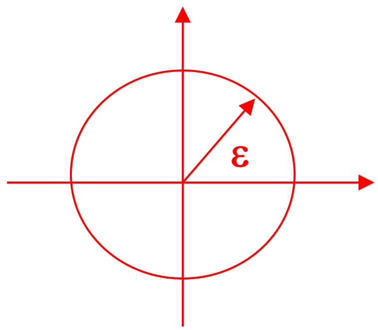
$$= -\sigma_{\min}(K_v)\|r\|^2 \qquad \le 0$$

Case 2: $\|r\| < \varepsilon$

$$\dot{L} \le -\sigma_{\min}(K_v)\|r\|^2 + \|r\| F(x) - \|r\|^2 F(x)/\varepsilon$$

$$= -\sigma_{\min}(K_v)\|r\|^2 + \|r\| F(x)\left(1 - \|r\|/\varepsilon\right)$$

indefinite



Typical Behavior of Robust Controllers

Control error does not go to zero but does indeed stay small.

Robust Control

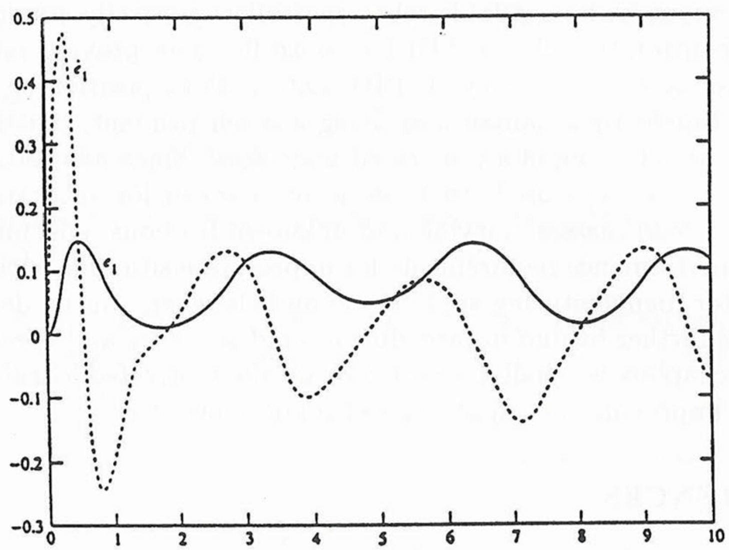


Figure 3.4.6: Typical behavior of robust controller.

Errors are bounded

Adaptive Control

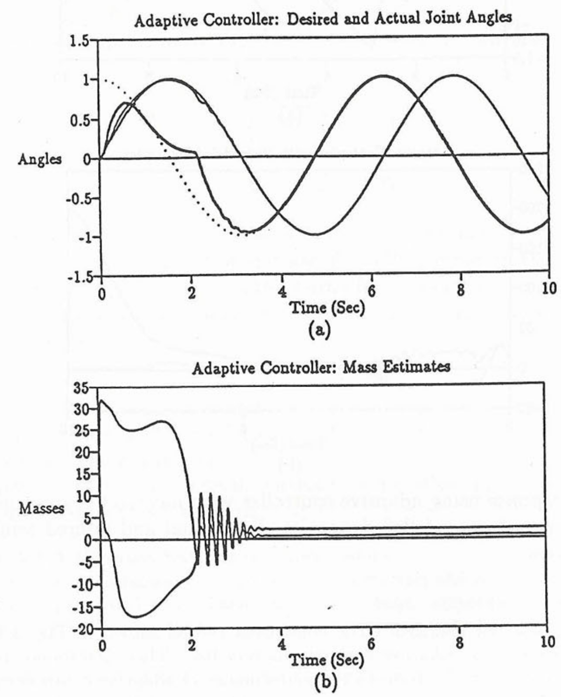
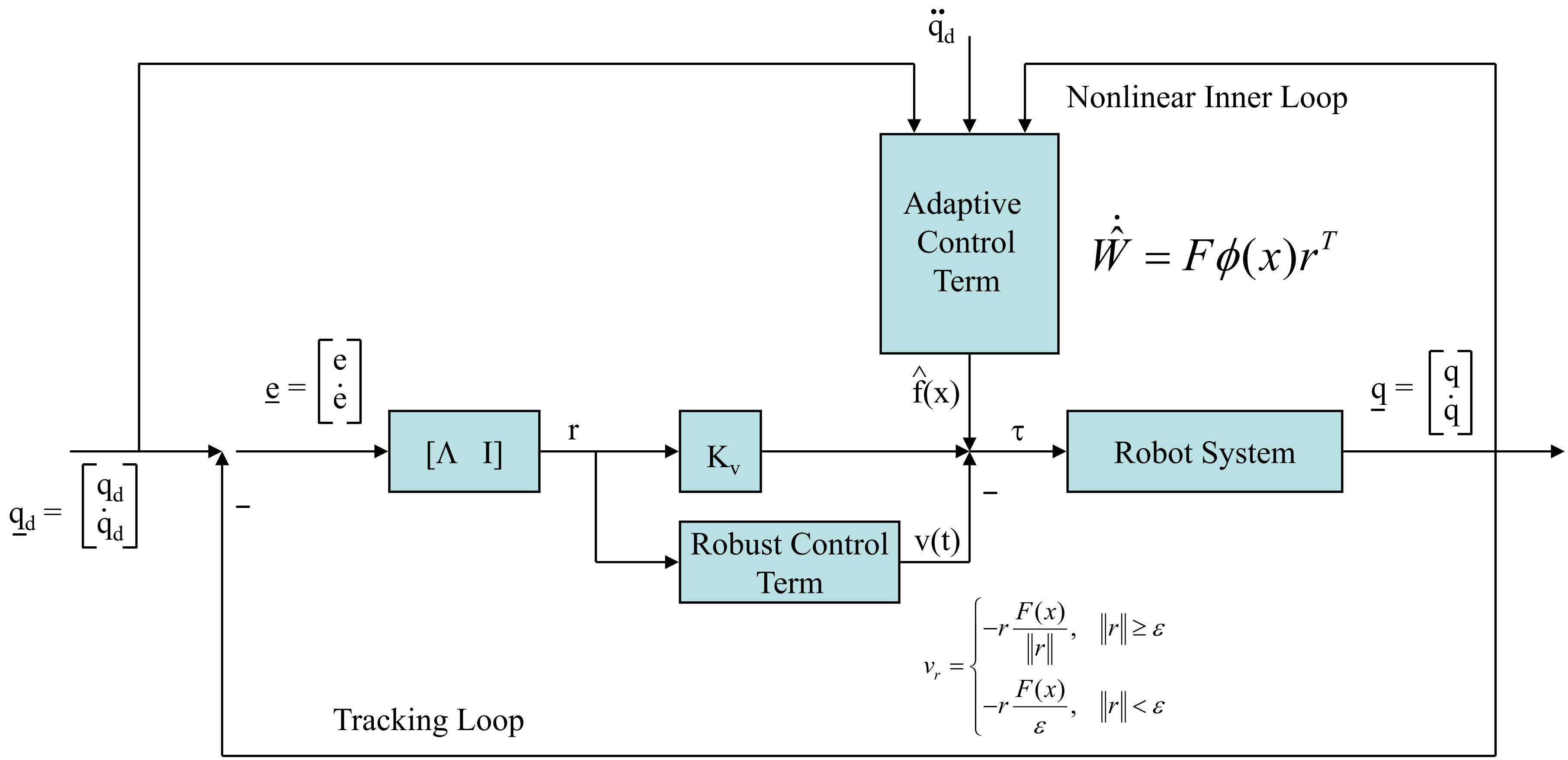


Figure 3.4.3: Response using adaptive controller. (a) Actual and desired joint angles. (b) Mass estimates.

Control errors go to zero

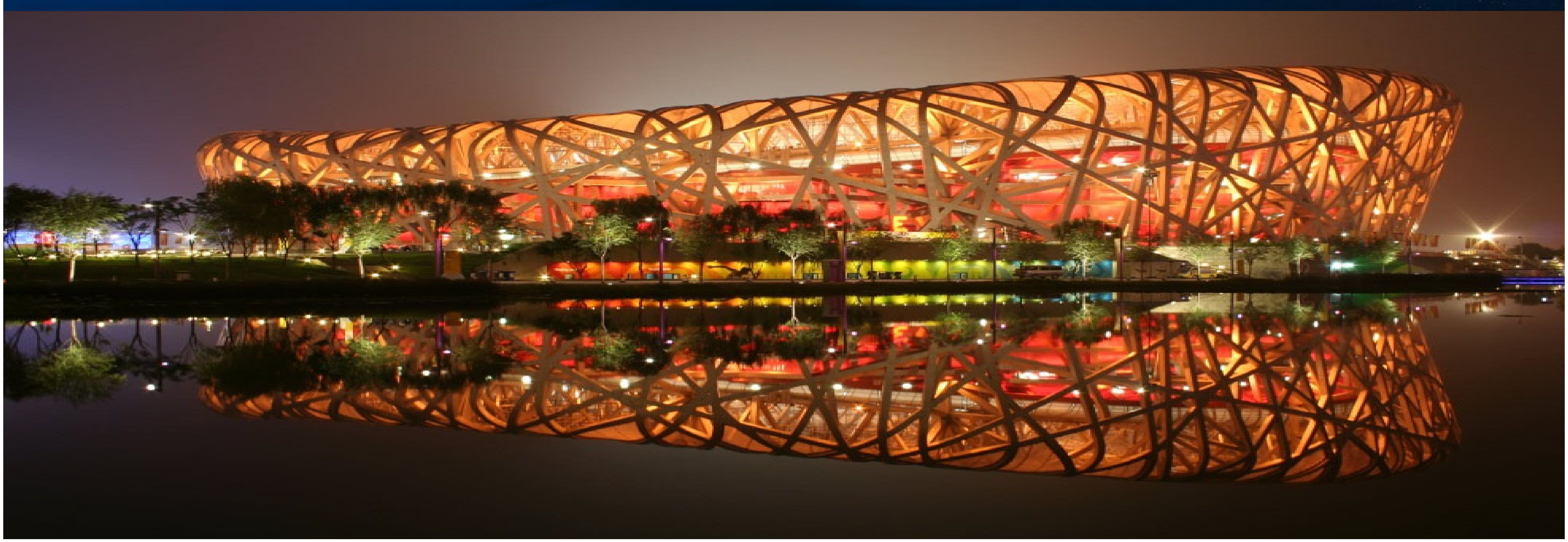
Parameter estimates converge

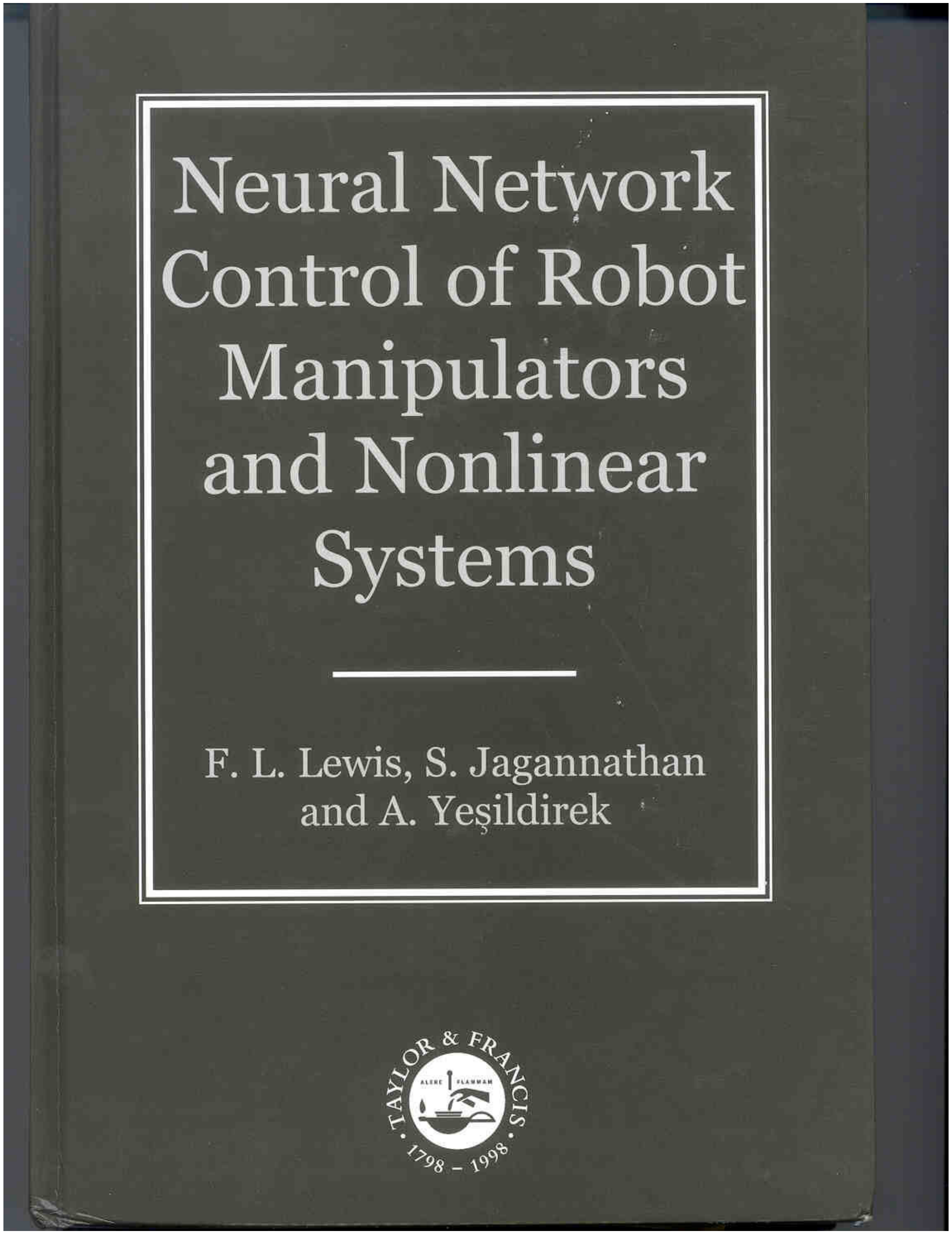
Adaptive plus robust control



Multiloop Nonlinear Controller Structure

Neural Networks for Control





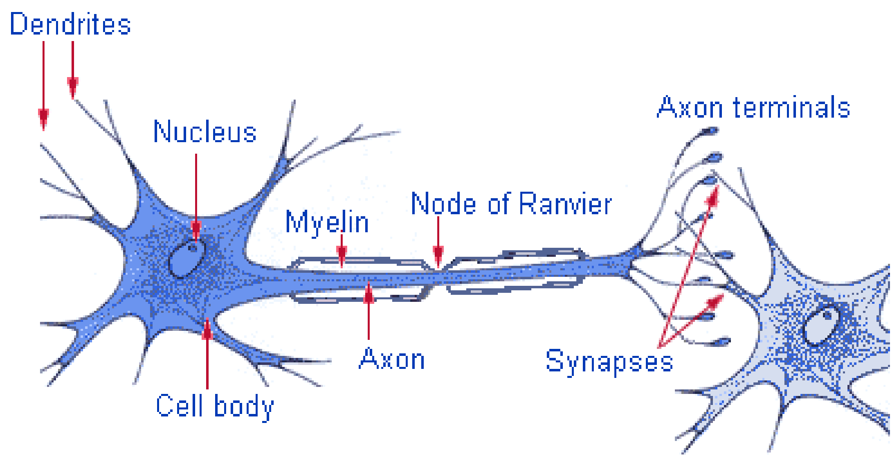
F.L. Lewis, S. Jagannathan, and A. Yesildirek,

Neural Network Control of Robot Manipulators and Nonlinear Systems,

Taylor and Francis, London, 1999.

NN control in Chapter 4

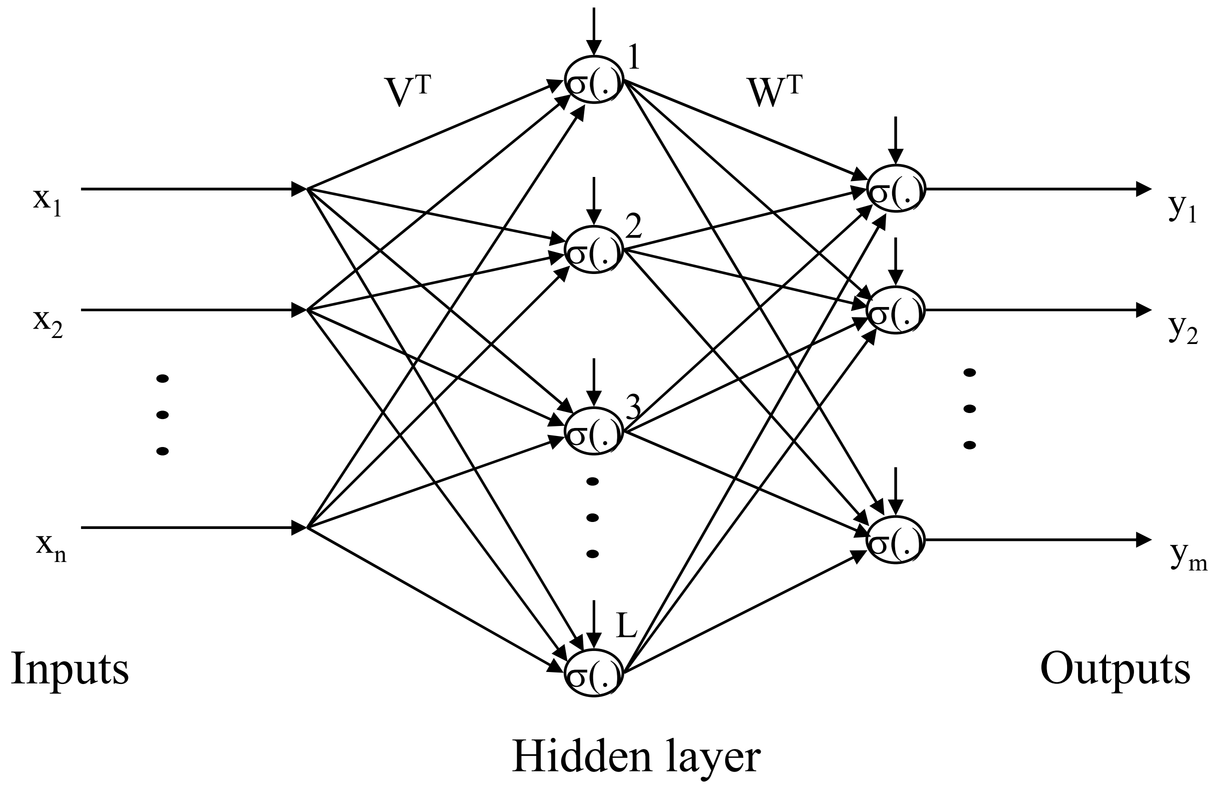
Neural Network Properties



Nervous system cell.
http://www.sirinet.net/~jgjohnso/index.html

- Learning
- Recall
- Function approximation
- Generalization
- Classification
- Association
- Pattern recognition
- Clustering
- Robustness to single node failure
- Repair and reconfiguration

Two-layer feedforward static neural network (NN)



Summation eqs

$$y_i = \sigma\left(\sum_{k=1}^{K} w_{ik} \sigma\left(\sum_{j=1}^{n} v_{kj} x_j + v_{k0} \right) + w_{i0} \right)$$

Matrix eqs

$$\begin{aligned} y &= W^T \sigma(V^T x) \\ &= W^T \phi(x) \end{aligned}$$

Extra freedom to select basis set by tuning first-layer weights V.

Neural Network Universal Approximation Property

Let *f(x)* be any smooth nonlinear function

Then *f(x)* can be approximated by

$$f(x) = \sum_{i=1}^{L} w_i \phi_i(x) \; + \quad \varepsilon(x)$$

For appropriate choice of the basis functions $\phi_i(x), \quad i = 1, L$

Moreover, the approximation error $\varepsilon(x)$ goes uniformly to zero as $L \to \infty$

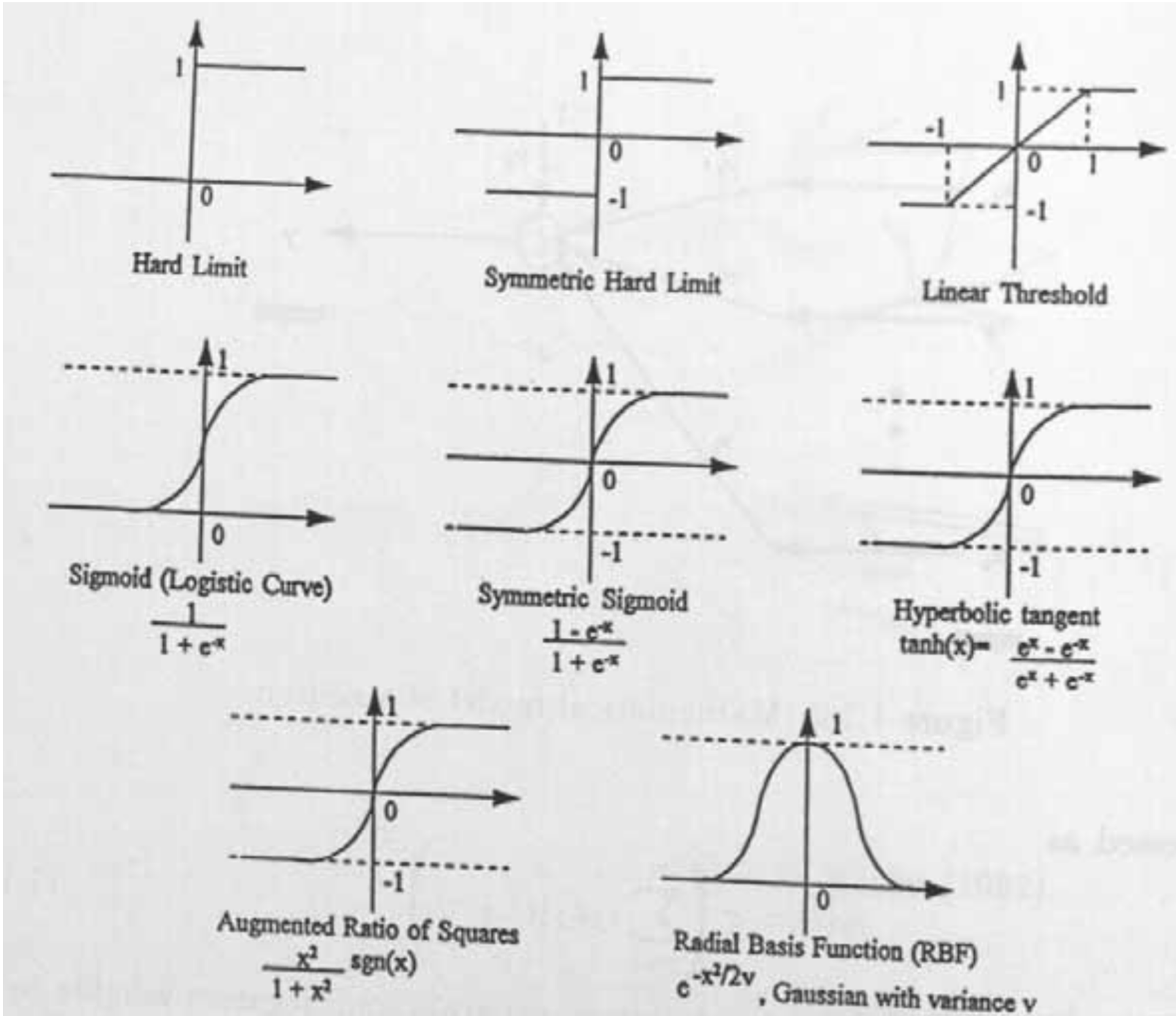
This was shown by Weierstrass for polynomial bases functions (Taylor series)

Hornik and Stinchcomb, Sandberg showed that $\varepsilon(x)$ is bounded on a compact set
For a large class of approximating functions

Need to find the unknown weights w_i

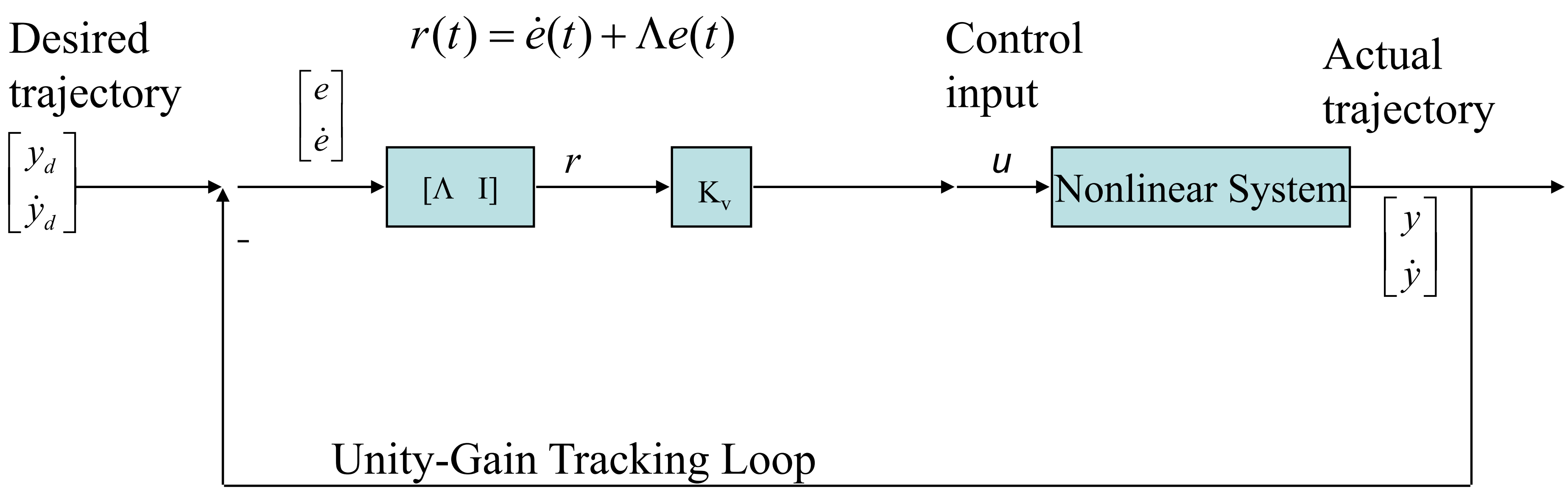
Do this by NN learning – tuning the NN weights

Common activation functions $\sigma(.)$



Industry Standard- PD Controller

Easy to implement with COTS controllers
Fast
Can be implemented with a few lines of code- e.g. MATLAB



But -- Cannot handle-
High-order unmodeled dynamics
Unknown disturbances
High performance specifications for nonlinear systems
Actuator problems such as friction, deadzones, backlash

Feedback Linearization Adaptive Controller

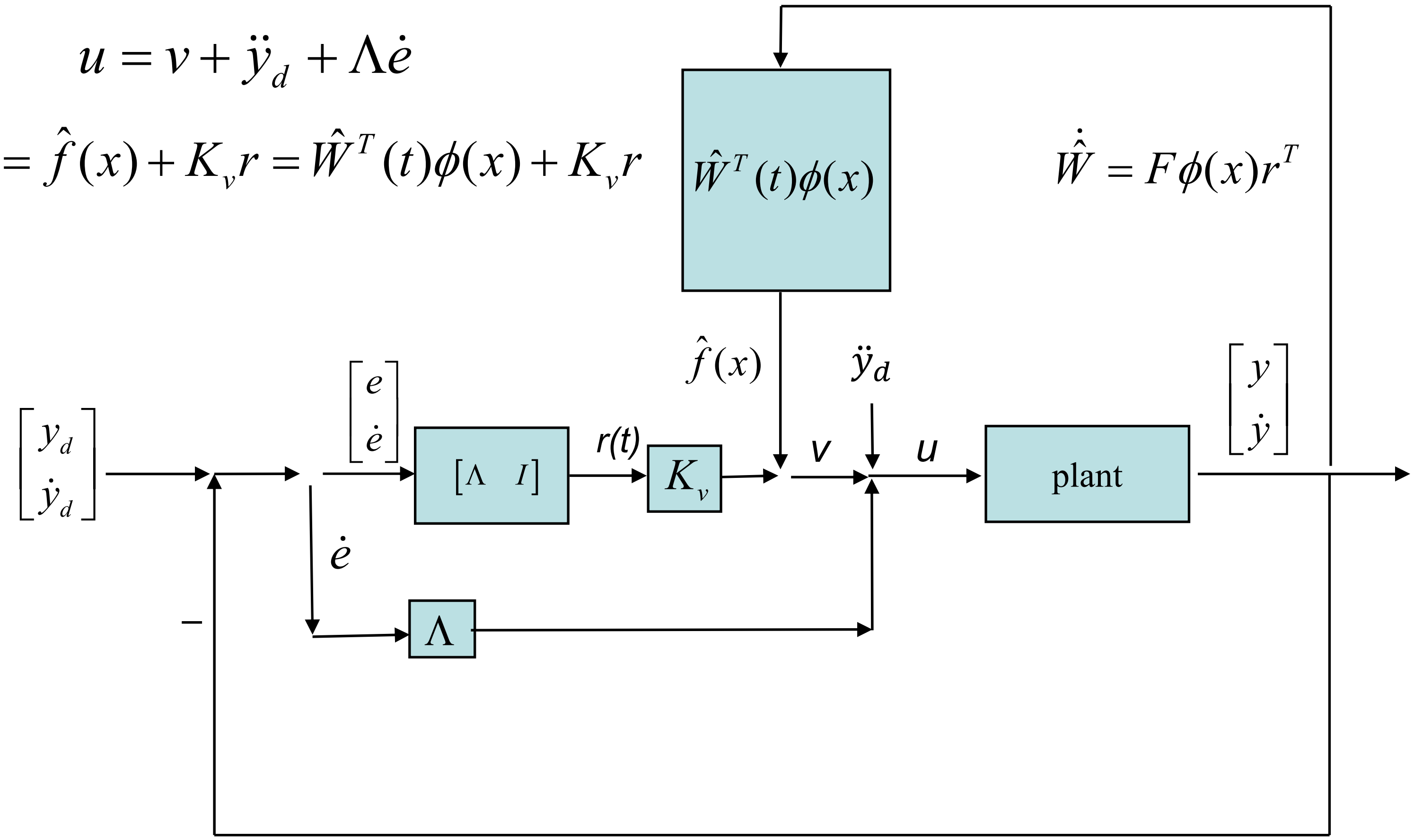
A dynamic controller

$$r = \dot{e} + \Lambda e$$

$$u = v + \ddot{y}_d + \Lambda \dot{e}$$

$$v = \hat{f}(x) + K_v r = \hat{W}^T(t)\phi(x) + K_v r$$

$$\dot{\hat{W}} = F\phi(x)r^T$$



The equations give the FB controller structure

Control System Design Approach

Robot dynamics $$M(q)\ddot{q}+V_m(q,\dot{q})\dot{q}++F(\dot{q})+G(q)+\tau_d=\tau$$

Tracking Error definition $$e(t)=q_d(t)-q(t)$$

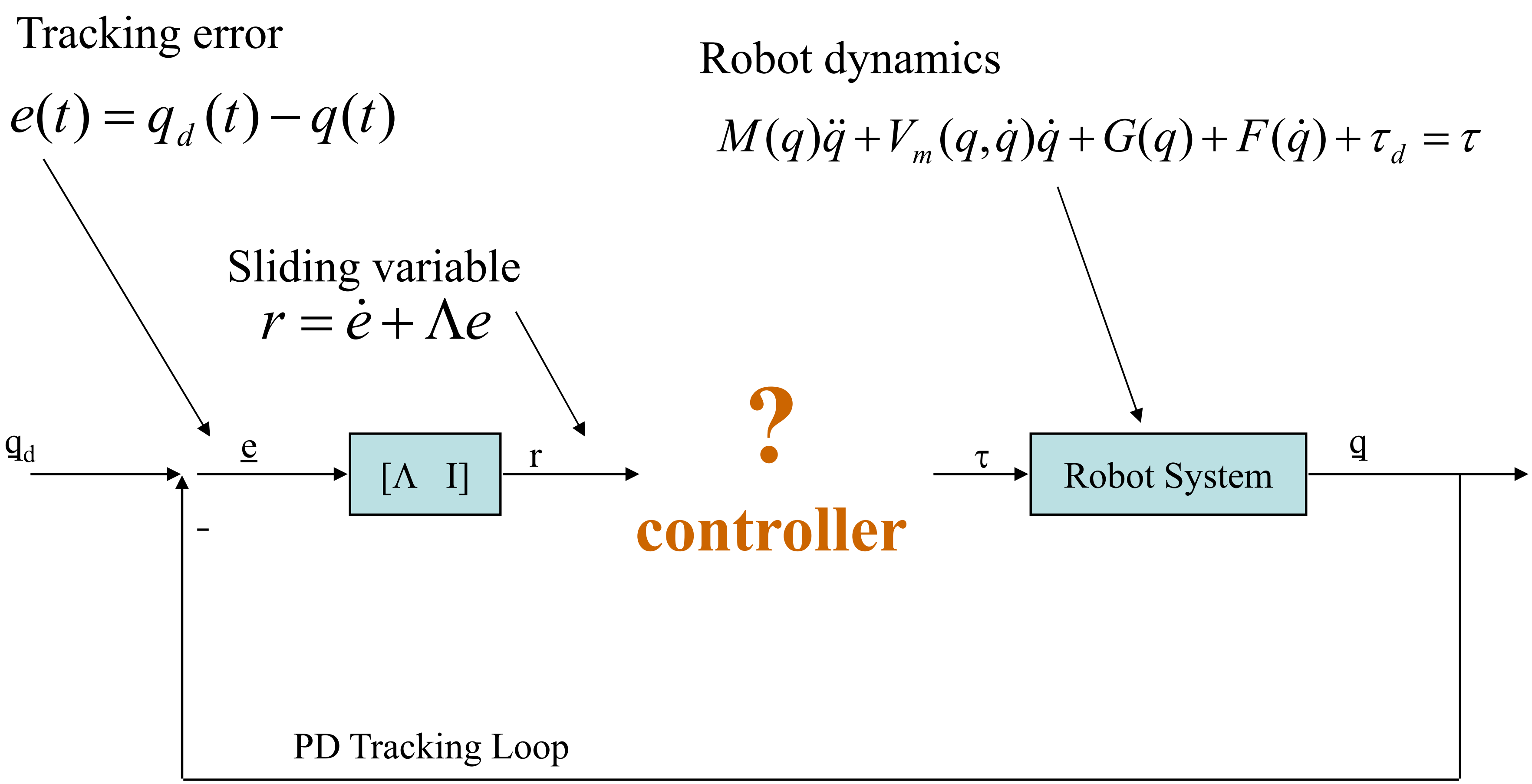
Siding variable $$r=\dot{e}+\Lambda e$$

Error dynamics $$M\dot{r}=M(\ddot{e}+\Lambda\dot{e})=\mathrm{M}(\ddot{q}_d(t)-\ddot{q}(t)+\Lambda\dot{e})$$

$$M\dot{r}=-V_m r+f(x)+\tau_d-\tau$$

Where the unknown function is

$$f(x)=M(q)(\ddot{q}_d+\Lambda\dot{e})+V_m(q,\dot{q})(\dot{q}_d+\Lambda e)+F(\dot{q})+G(q)$$



The equations give the FB controller structure

Control System Design Approach

Robot dynamics $$M(q)\ddot{q}+V_m(q,\dot{q})\dot{q}+G(q)+F(\dot{q})+\tau_d=\tau$$

Tracking Error definition $$e(t)=q_d(t)-q(t) \qquad\qquad r=\dot{e}+\Lambda e$$

Error dynamics $$M\dot{r}=-V_m r+f(x)+\tau_d-\tau$$

UNKNOWN FN.

Universal Approximation Property

Approx. unknown function by NN $$f(x)=W^T\sigma(V^Tx)+\varepsilon$$

Define control input $$\tau=\hat{W}^T\sigma(\hat{V}^Tx)+K_v r-v$$

Closed-loop dynamics

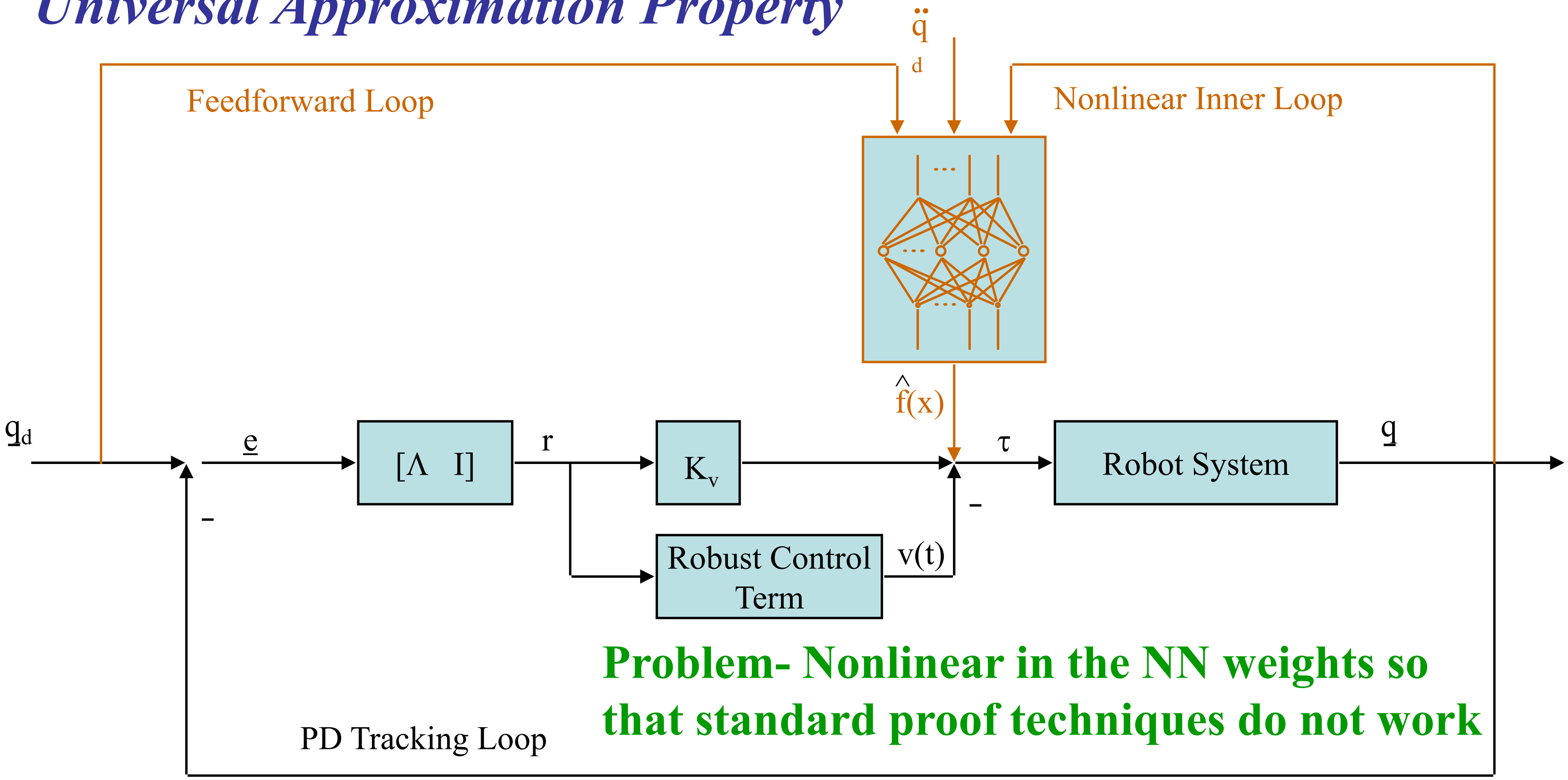
$$M\dot{r}=-V_m r-K_v r+W^T\sigma(V^Tx)+\varepsilon-\hat{W}^T\sigma(\hat{V}^Tx)+\tau_d+v(t)$$

$$M\dot{r}=-V_m r-K_v r+\tilde{f}+\tau_d+v(t)$$

Neural Network Robot Controller

Universal Approximation Property

Feedback linearization



Problem- Nonlinear in the NN weights so that standard proof techniques do not work

Easy to implement with a few more lines of code

Learning feature allows for on-line updates to NN memory as dynamics change

Handles unmodelled dynamics, disturbances, actuator problems such as friction

NN universal basis property means no regression matrix is needed

Nonlinear controller allows faster & more precise motion

Adaptive part

Robust part

Theorem 1 (NN Weight Tuning for Stability)

Let the desired trajectory $q_d(t)$ and its derivatives be bounded. Let the initial tracking error be within a certain allowable set U. Let Z_M be a known upper bound on the Frobenius norm of the unknown ideal weights Z.

Take the control input as

$$\tau = \hat{W}^T \sigma(\hat{V}^T x) + K_v r - v \qquad \text{with} \qquad v(t) = -K_Z(\|Z\|_F + Z_M)r .$$

Let weight tuning be provided by

$$\dot{\hat{W}} = F\hat{\sigma}r^T - F\hat{\sigma}'\hat{V}^T x r^T - \kappa F\|r\|\hat{W} , \qquad \dot{\hat{V}} = Gx(\hat{\sigma}'^T \hat{W} r)^T - \kappa G\|r\|\hat{V}$$

with any constant matrices $F = F^T > 0, G = G^T > 0$, and scalar tuning parameter $\kappa > 0$. Initialize the weight estimates as $\hat{W} = 0, \hat{V} = random$.

Then the filtered tracking error $r(t)$ and NN weight estimates $\hat{W}, \hat{V}$ are uniformly ultimately bounded. Moreover, arbitrarily small tracking error may be achieved by selecting large control gains K_v.

Backprop terms-
Werbos

Extra robustifying terms-
Narendra's e-mod extended to NLIP systems

Stability Proof based on Lyapunov Extension

Define a Lyapunov Energy Function

$$L = \tfrac{1}{2} r^T M r + \tfrac{1}{2} tr(\tilde{W}^T \tilde{W}) + \tfrac{1}{2} tr(\tilde{V}^T \tilde{V})$$

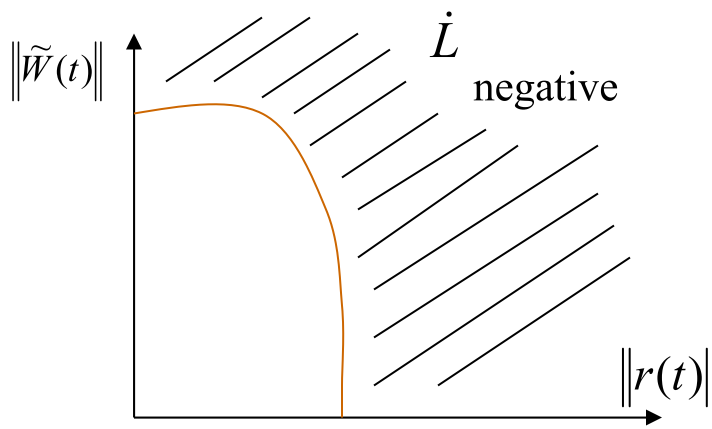
Differentiate

$$\dot{L} = -r^T K_v r + \tfrac{1}{2} r^T (\dot{M} - 2V_m) r$$
$$+ tr\, \tilde{W}^T (\dot{\tilde{W}} + \hat{\sigma} r^T - \hat{\sigma}' \hat{V}^T x r^T)$$
$$+ tr\, \tilde{V}^T (\dot{\tilde{V}} + x r^T \hat{W}^T \hat{\sigma}') + r^T (w + v)$$

Problems—

1. How to characterize the NN weight errors as 'small'?- use Frobenius Norm
2. Nonlinearity in the parameters requires extra care in the proof

Using certain special tuning rules, one can show that the energy derivative is negative outside a compact set.



This proves that all signals are bounded

UUB- uniform ultimate boundednes

Adaptive Control

Parameter Convergence and Persistence of Excitation:

Error dynamics

$$\dot{r} = \tilde{W}^T \phi(x) - K_v r$$

So that

$$z(t) \equiv \tilde{W}^T \phi(x) = \dot{r} + K_v r$$ Is bounded

Dynamics of param. est. error

$$\dot{\tilde{W}} = -F\phi(x) r^T$$

$$z^T(t) \equiv \phi^T(x)\tilde{W} = (\dot{r} + K_v r)^T$$

Lyapunov showed that $r(t) \to 0, \dot{r}(t) \to 0$ so input and output go to zero

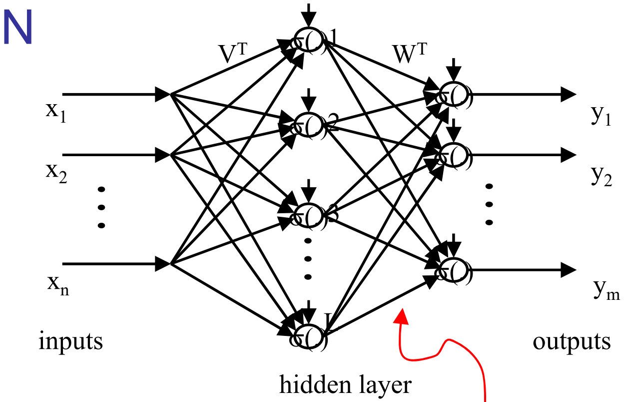
So the state $\tilde{W}(t)$ goes to zero if the system is uniformly completely observable

$$\alpha_1 I \le \int_t^{t+T} \phi(x(\sigma))\phi^T(x(\sigma))\, d\sigma \le \alpha_2 I$$

This is the same as a persistence of excitation condition on the regression vector

Therefore, if $\phi(x)$ is PE, the parameters converge.

Special case- Linear in the parameters – 1 layer NN



Fixed basis set

Tune 2[nd] layer weights

NO REGRESSION VECTOR NEEDED!

Take the control input as

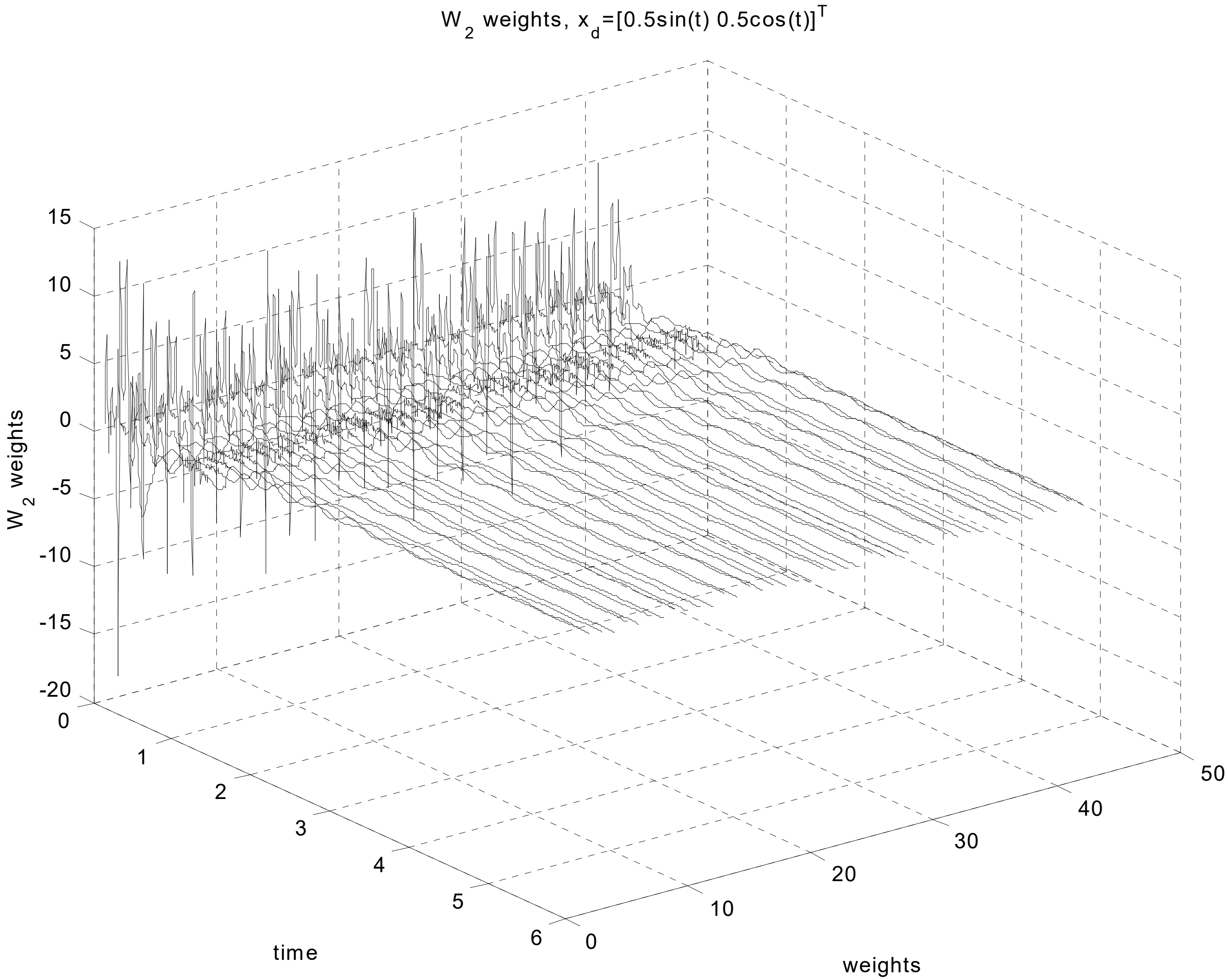
$$\tau = \hat{W}^T \sigma(\hat{V}^T x) + K_v r - v \qquad \text{with} \qquad v(t) = -K_Z(\|Z\|_F + Z_M) r\,.$$

Let weight tuning be provided by

$$\dot{\hat{W}} = F\hat{\sigma} r^T - \cancel{F\hat{\sigma}'\hat{V}^T x r^T} - \kappa F\|r\|\hat{W}\,, \qquad \cancel{\dot{\hat{V}} = Gx(\hat{\sigma}'^T \hat{W} r)^T - \kappa G\|r\|\hat{V}}$$

STILL BETTER THAN ADAPTIVE CONTROL

This is a universal basis set good for a class of systems
Not a regression vector good for only one system



NN weights converge to the best learned values for the given system

Structured Control NN

$$M(\dot{q})\ddot{q}+V_m(q,\dot{q})\dot{q}+G(q)+F(\dot{q})+\tau_d = B(q)\tau$$

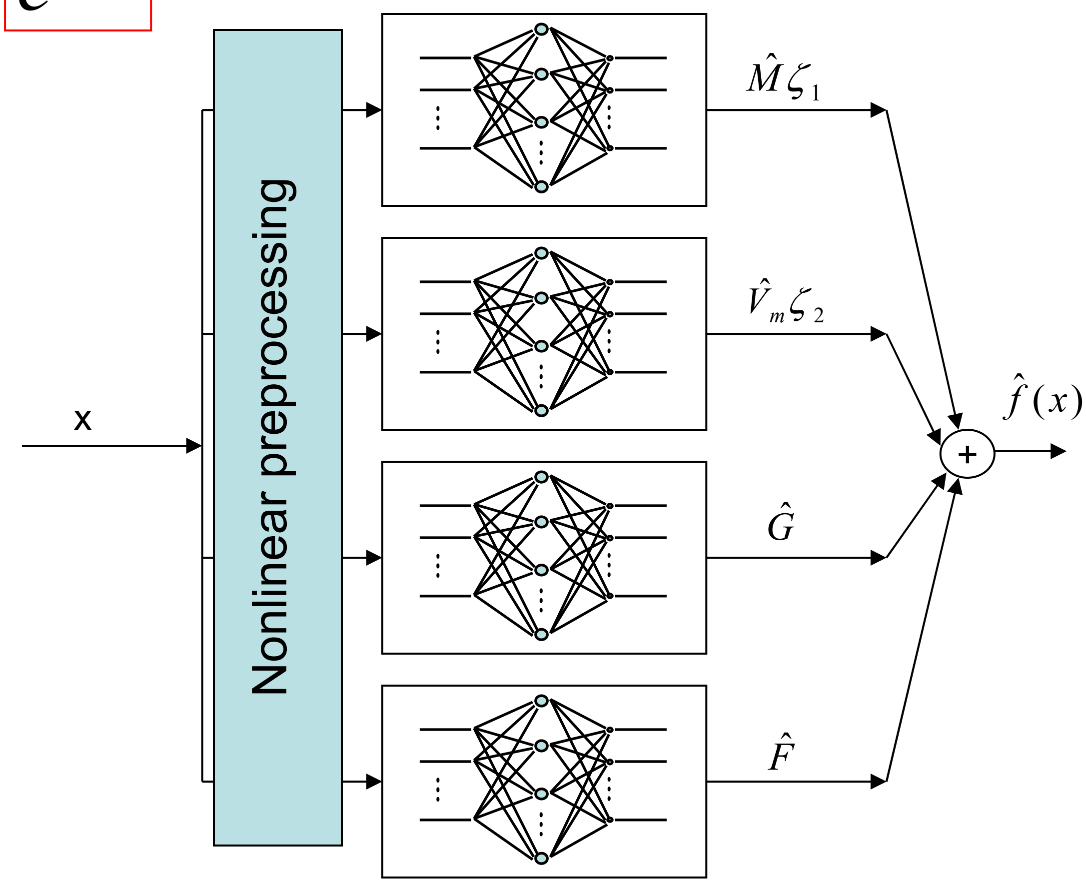
$\boxed{\sin(q_i),\ \cos(q_i)}$

$\boxed{\mathrm{sgn}(\dot{q}_i)}$

$\boxed{e^{-|q_i|}}$

$\boxed{\dot{q}_i\dot{q}_j}$ coriolis

$\boxed{\dot{q}_i^2}$ centripetal

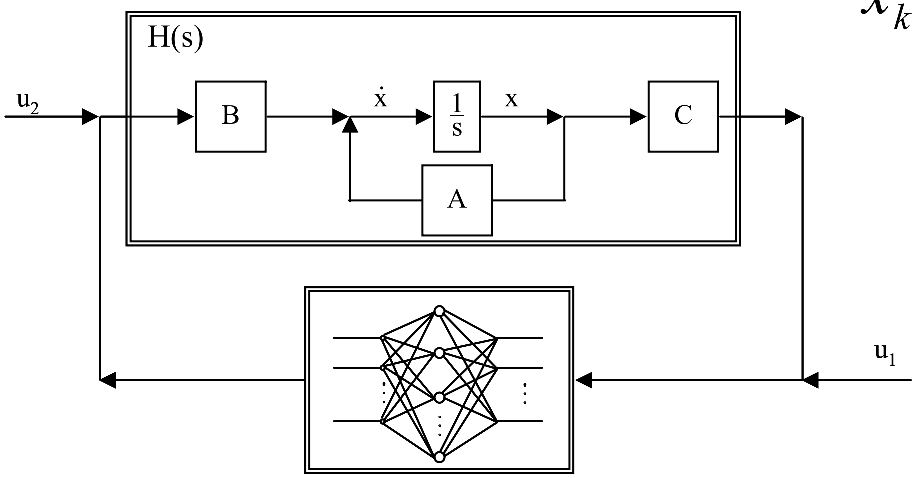


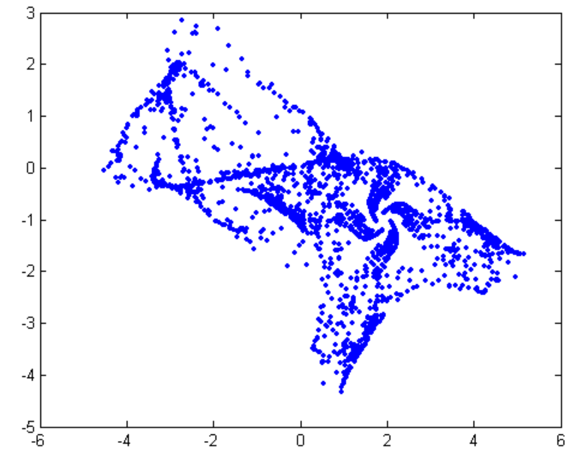
Partitioned neural network

Chaos in Dynamic Neural Networks

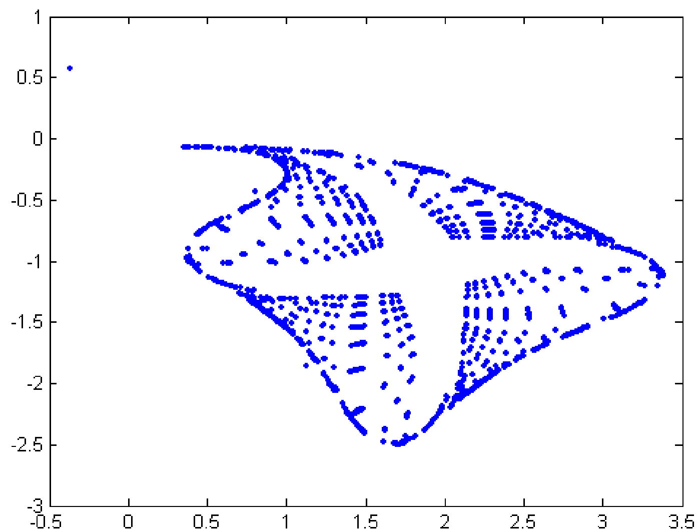
Recurrent or Dynamic NN

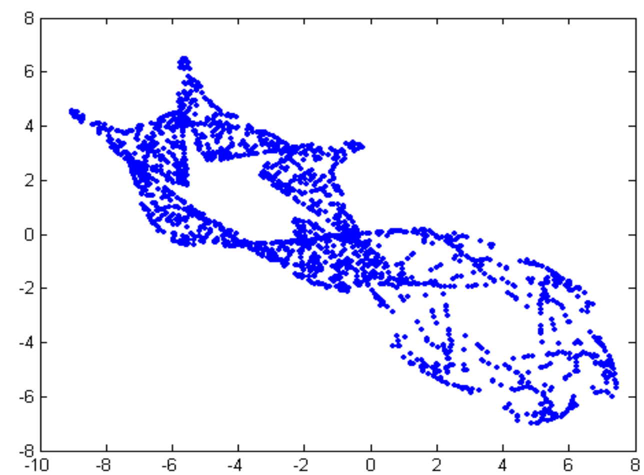
$$x_{k+1} = Ax_k + W^T\sigma(V^T x_k) + u_k$$





c.f. Ron Chen





Kung Tz 500 BC

Confucius

孔子

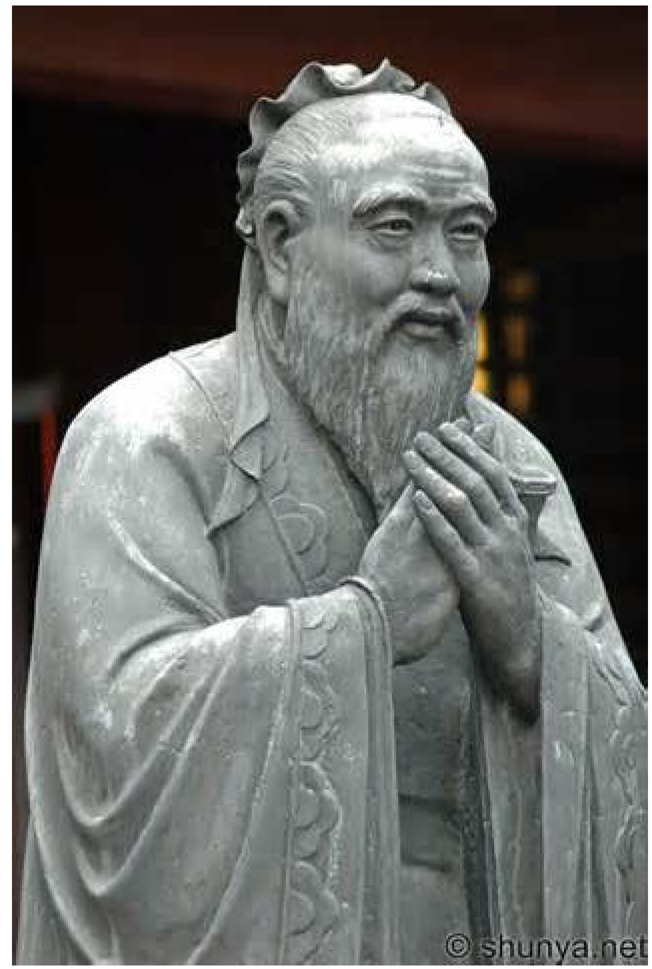
Man's relations to
- Family
- Friends
- Society
- Nation
- Emperor
- Ancestors

Tian xia da tong
Harmony under heaven

Archery
Chariot driving

Music
Rites and Rituals

Poetry
Mathematics

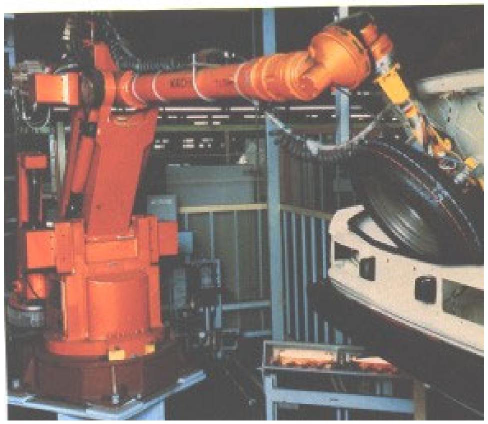


124 BC - Han Imperial University in Chang-an

Handling High-Frequency Dynamics

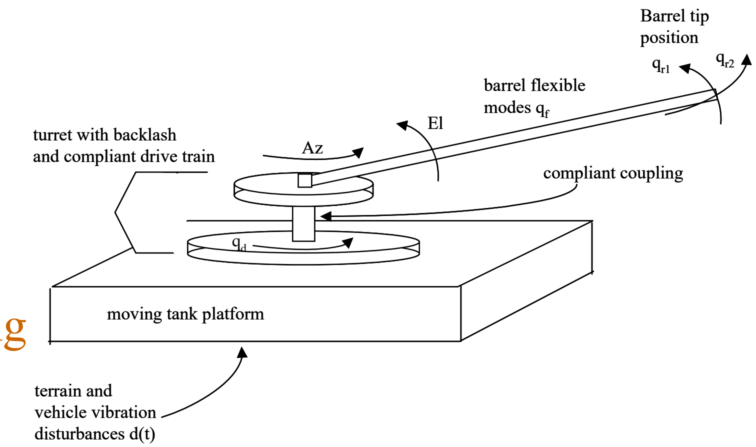
Relevance- Machine Feedback Control

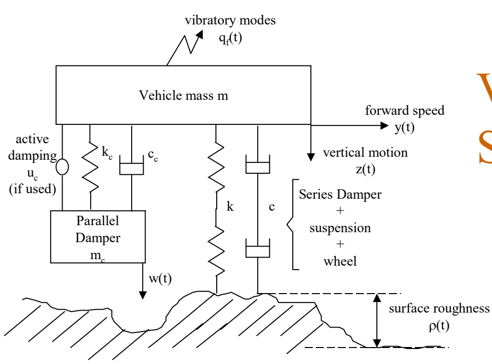
High-Speed Precision Motion Control with unmodeled dynamics, vibration suppression, disturbance rejection, friction compensation, deadzone/backlash control



Industrial Machines

Satellite pointing
Land Systems



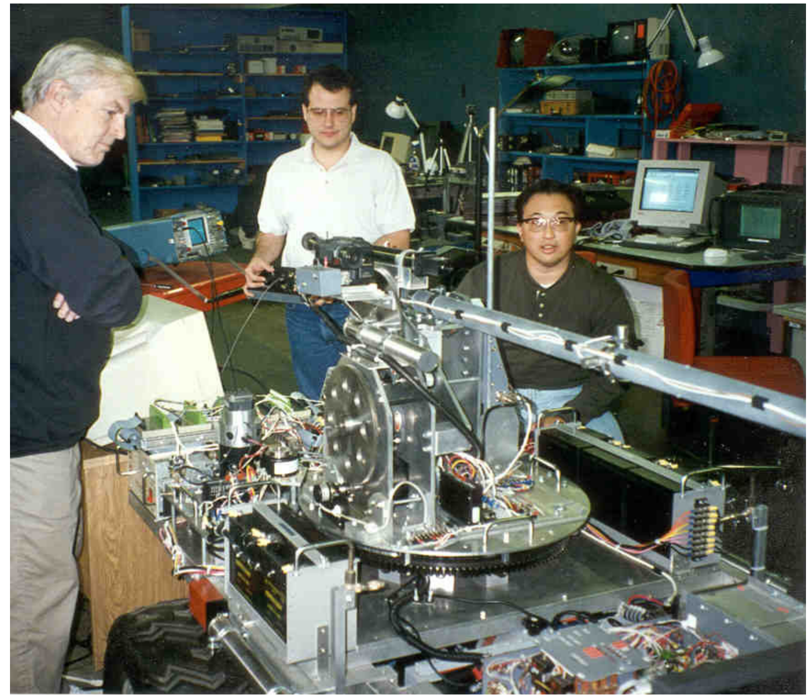


Single-Wheel/Terrain System with Nonlinearities

Vehicle Suspension

Aerospace

Force Control



Flexible pointing systems

Vehicle active suspension

SBIR Contracts

What about practical Systems?

Two types of high frequency modes in Industrial Processes

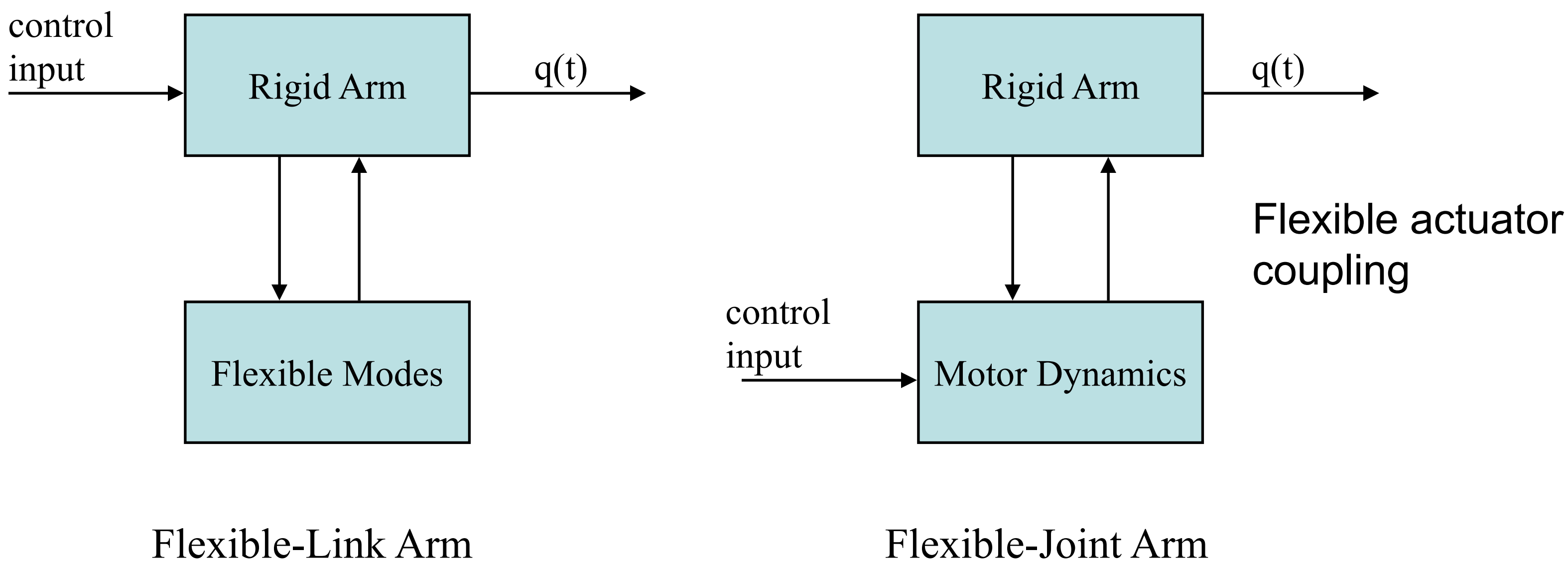


Fig. 5.2.4 Two canonical control problems with high-frequency modes. (a) Flexible-link robot arm. (b) Flexible-joint robot arm.

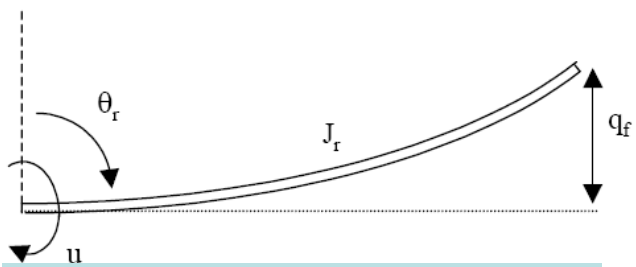
Flexible Systems with Vibratory Modes

Rigid dynamics

$$\begin{bmatrix} M_{rr} & M_{rf} \\ M_{fr} & M_{ff} \end{bmatrix}\begin{bmatrix} \ddot{q}_r \\ \ddot{q}_f \end{bmatrix} + \begin{bmatrix} V_{rr} & V_{rf} \\ V_{fr} & V_{ff} \end{bmatrix}\begin{bmatrix} \dot{q}_r \\ \dot{q}_f \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ o & K_{ff} \end{bmatrix}\begin{bmatrix} q_r \\ q_f \end{bmatrix} + \begin{bmatrix} F_r \\ 0 \end{bmatrix} + \begin{bmatrix} G_r \\ 0 \end{bmatrix} = \begin{bmatrix} B_r \\ B_f \end{bmatrix}\tau$$

Flexible dynamics

Problem- only one control input !



Flexible link pointing system

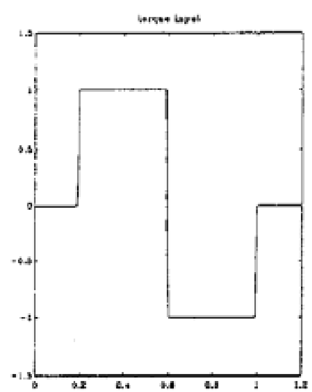


Fig. 2 Acceleration/deceleration torque profile.

acceleration

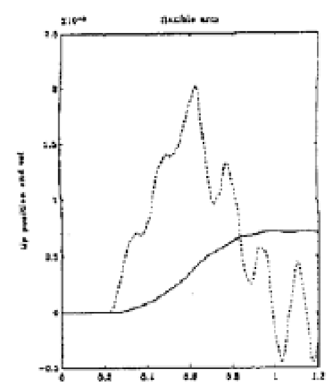


Fig. 3a Open-loop response of flexible arm. Tip position (solid) and vel. (dashed)

velocity
position

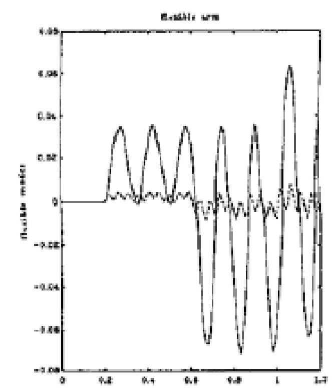


Fig. 3b Open-loop response of flexible arm. Flexible modes.

Flex. modes

Singular Perturbation Theory

$$\begin{bmatrix} M_{rr} & M_{rf} \\ M_{fr} & M_{ff} \end{bmatrix}\begin{bmatrix} \ddot{q}_r \\ \ddot{q}_f \end{bmatrix}+\begin{bmatrix} V_{rr} & V_{rf} \\ V_{fr} & V_{ff} \end{bmatrix}\begin{bmatrix} \dot{q}_r \\ \dot{q}_f \end{bmatrix}+\begin{bmatrix} 0 & 0 \\ o & K_{ff} \end{bmatrix}\begin{bmatrix} q_r \\ q_f \end{bmatrix}+\begin{bmatrix} F_r \\ 0 \end{bmatrix}+\begin{bmatrix} G_r \\ 0 \end{bmatrix}=\begin{bmatrix} B_r \\ B_f \end{bmatrix}\tau$$

Slow subsystem

NN design

$$\overline{M}_{rr}\ddot{\overline{q}}_r + \overline{V}_{rr}\dot{\overline{q}}_r + \overline{F}_r + \overline{G}_r = \overline{B}_r\overline{\tau}$$

TWO Control Inputs!!

Fast Subsystem

$$\frac{d}{dT}\begin{bmatrix} \tilde{q}_f \\ \dot{\tilde{q}}_f \end{bmatrix}=\begin{bmatrix} 0 & I \\ -\overline{H}_{ff}\tilde{K}_{ff} & 0 \end{bmatrix}\begin{bmatrix} \tilde{q}_f \\ \dot{\tilde{q}}_f \end{bmatrix}+\begin{bmatrix} 0 \\ \overline{B}_f \end{bmatrix}\tau_f$$

Time scaling $T = \frac{t}{\varepsilon}$

Linear Design

Tikhonov’s Theorem

$$q_r = \overline{q}_r + O(\varepsilon)$$

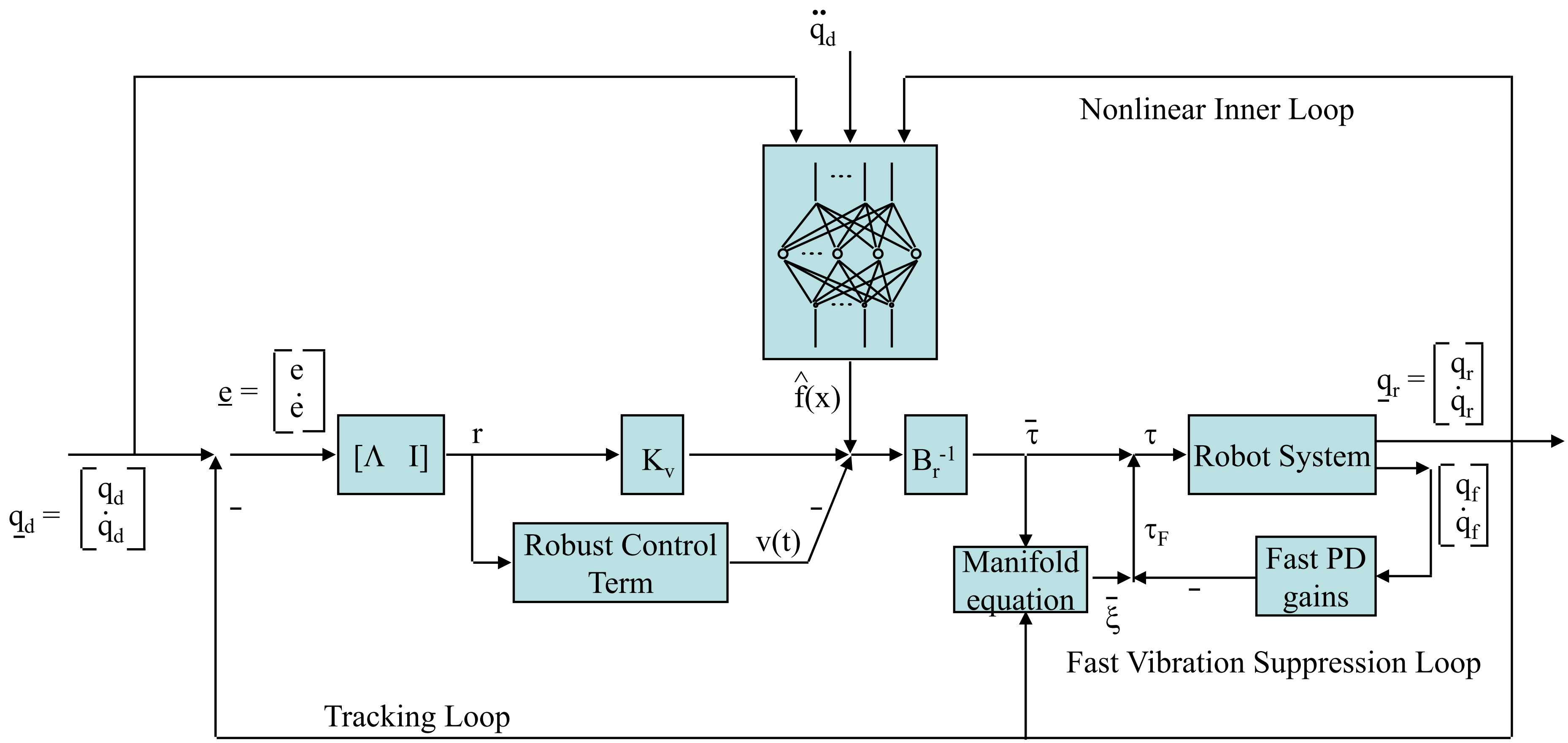
$$q_f = \varepsilon^2(\tilde{q}_f + \overline{q}_f)$$

Comes from Manifold Equation

$$\tau = \overline{\tau} + \tau_f$$

Singular Perturbations

Add an extra feedback loop
Use passivity to show stability



Neural network controller for Flexible-Link robot arm

Coupled Systems

Flexible actuators

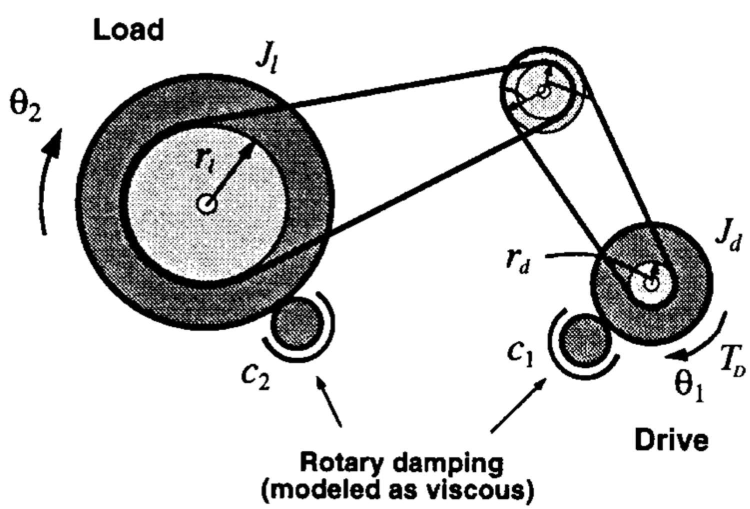
Problem- only one control input !

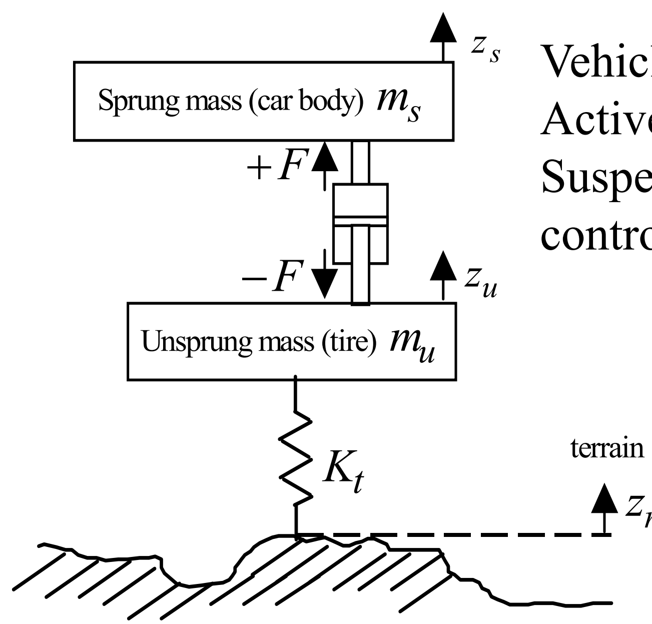
Robot mechanical dynamics

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d = K_T i$$

$$L\dot{i} + R(i,\dot{q}) + \tau_e = u_e$$

Motor electrical dynamics





Vehicle
Active
Suspension
control

Backstepping Design

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d = K_T i$$

$$L\dot{i} + R(i,\dot{q}) + \tau_e = u_e$$

1. Outer Loop Design

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d = K_T i_d + K_T(i - i_d)$$

Design desired current i_d using NN number 1

2. Inner Loop Design

Design inner NN control loop to make error $\eta = i_d - i$ small

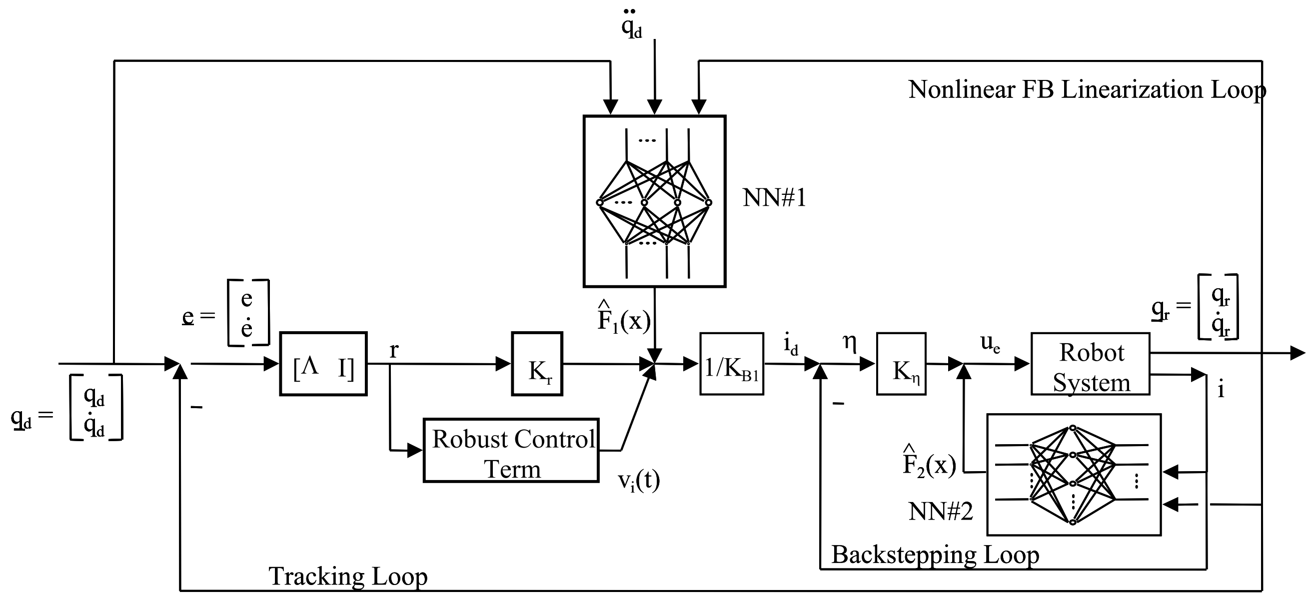
$$L\dot{\eta} + F(X) - \tau_e = -u_e$$

Unknown Nonlinear term

Advantage of NN Backstepping-
DO NOT need to compute the form of F(X) and find a linear parameterization

Backstepping

Add an extra feedback loop
Two NN needed
Use passivity to show stability



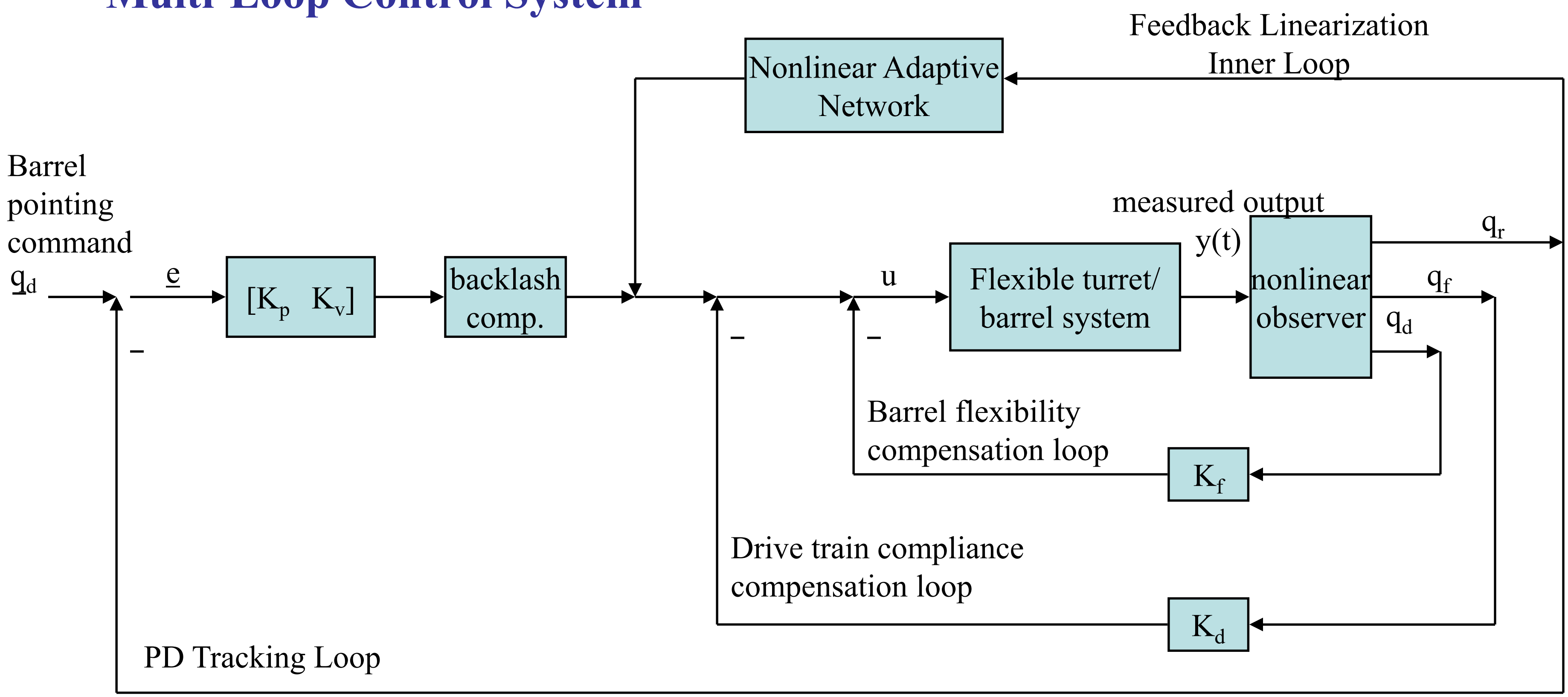
Neural network backstepping controller for Flexible-Joint robot arm

Advantages over traditional Backstepping- no regression functions needed

Modern Nonlinear Control Theory

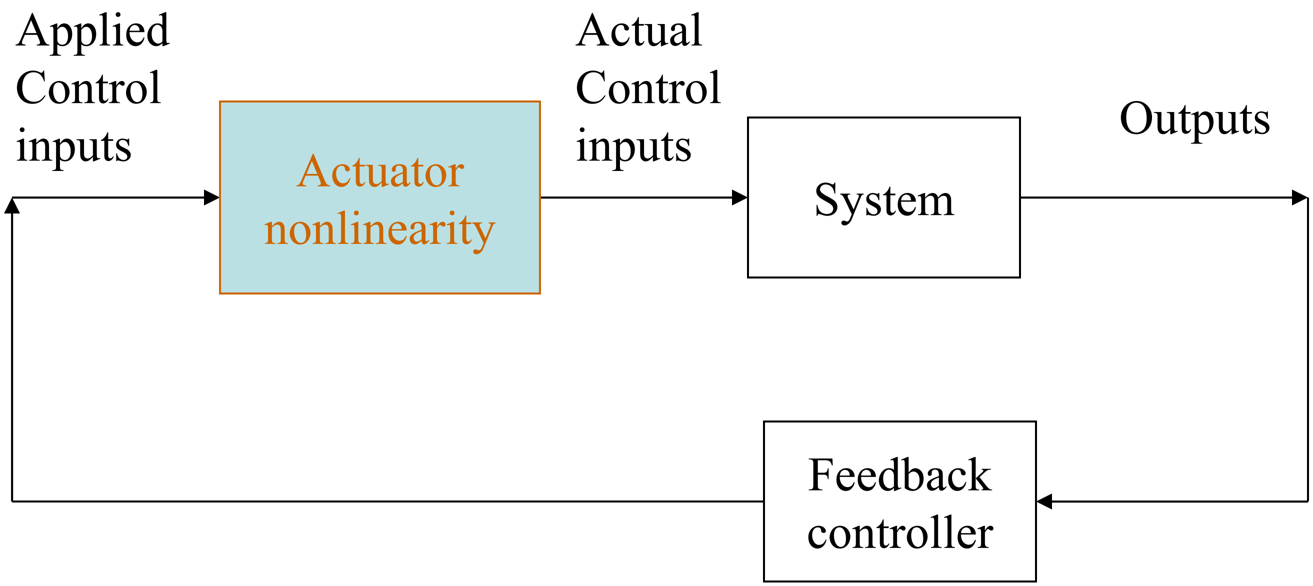
Multi-Loop Control System

Structure of controller with adaptive backlash compensation and an additional loop for rejection of drive train compliance

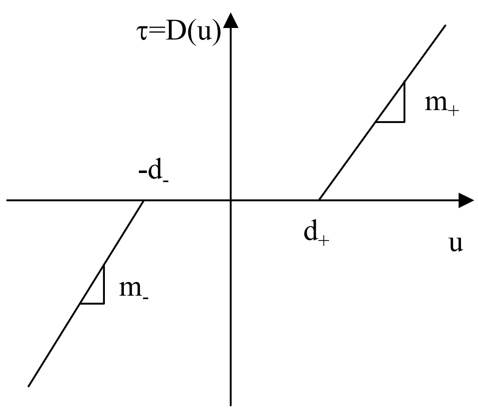


Actuator Dynamics

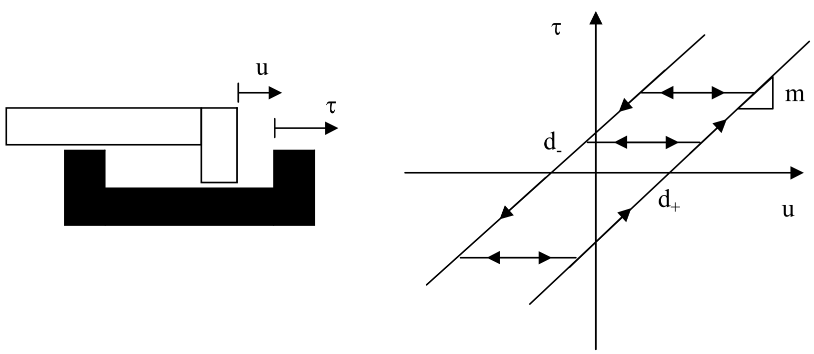
Actuator Nonlinearities



Friction



Deadzone



Backlash

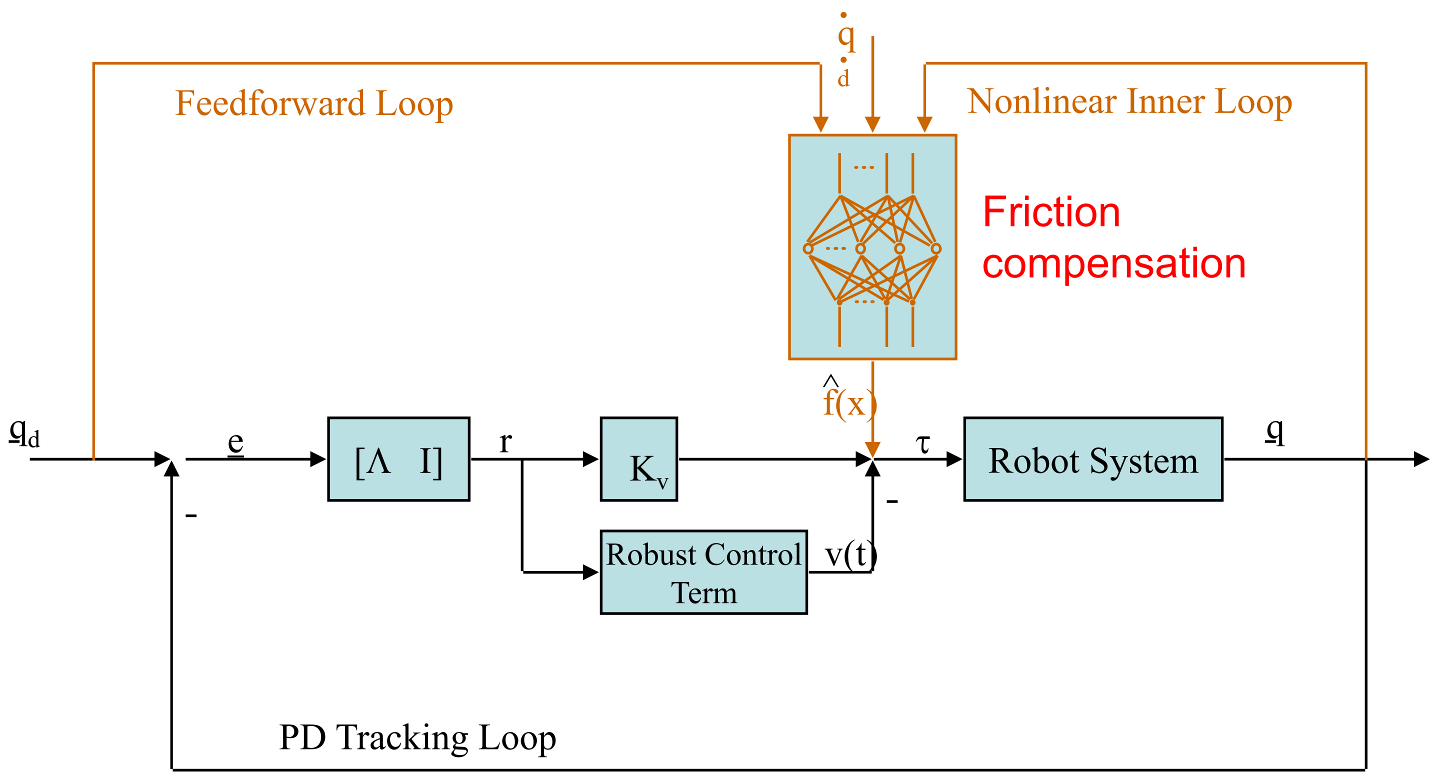
Friction Compensation

Lagrangian System Dynamics – e.g. Robot

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = \tau$$

Friction

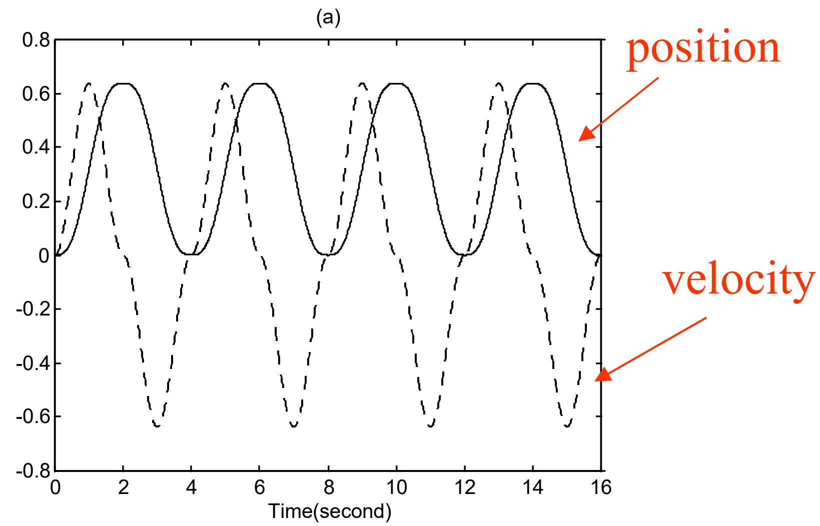
Use the standard NN from Lecture 1

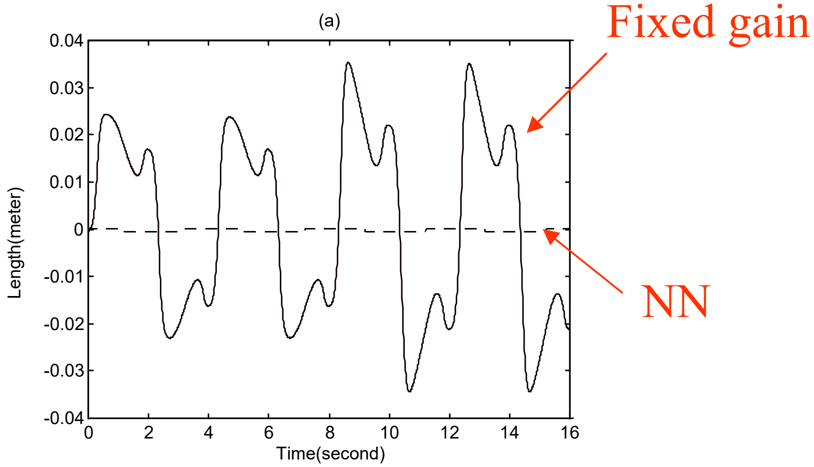


NN Friction Compensator

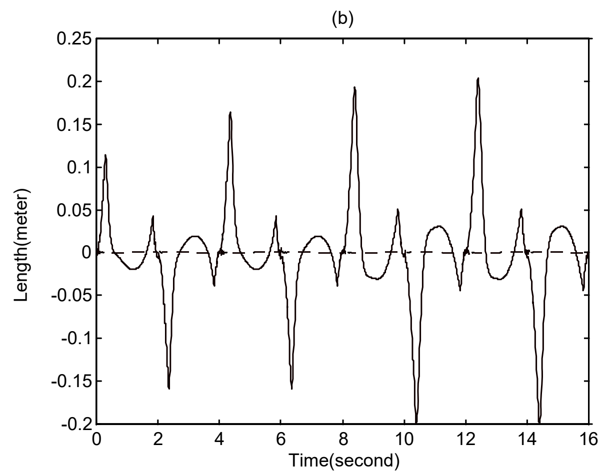
Trajectory Tracking Controller

Desired trajectory





Position

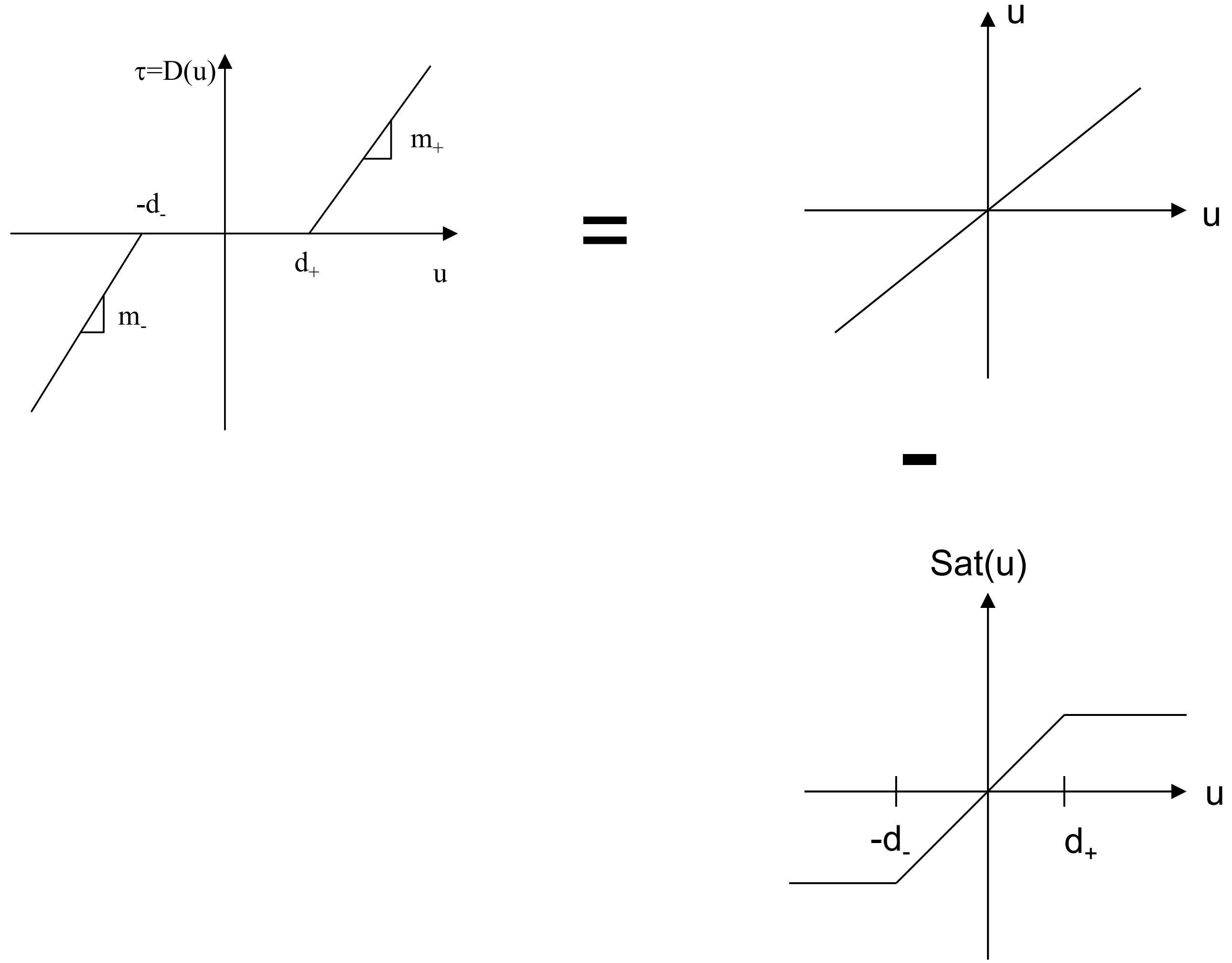


Velocity

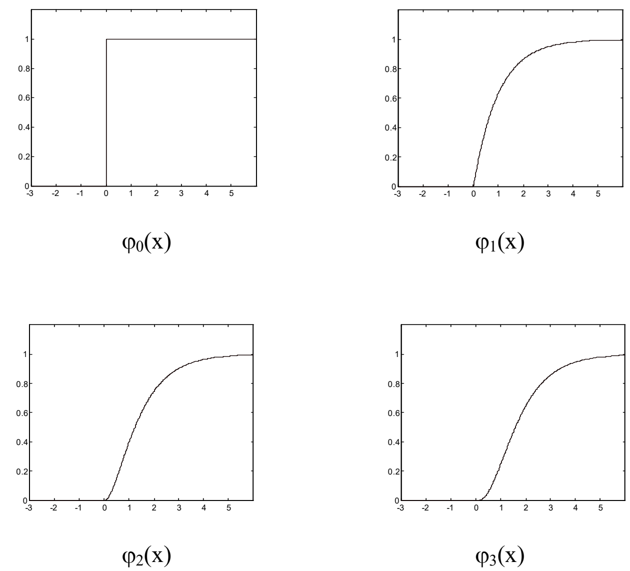
Tracking errors- solid = fixed gain controller, dashed= NN controller

Deadzone

Key fact

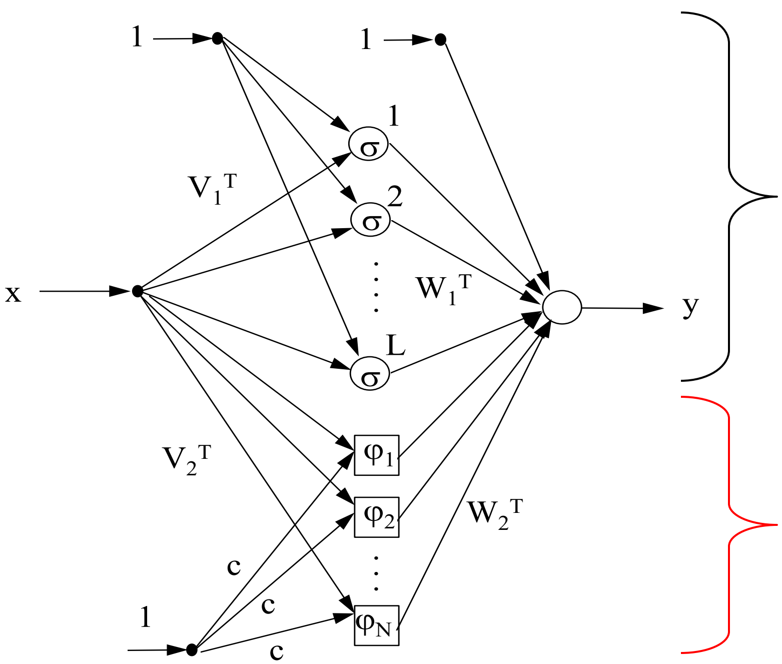


NN Approximation of Functions with Jumps



Jump Function Basis Set

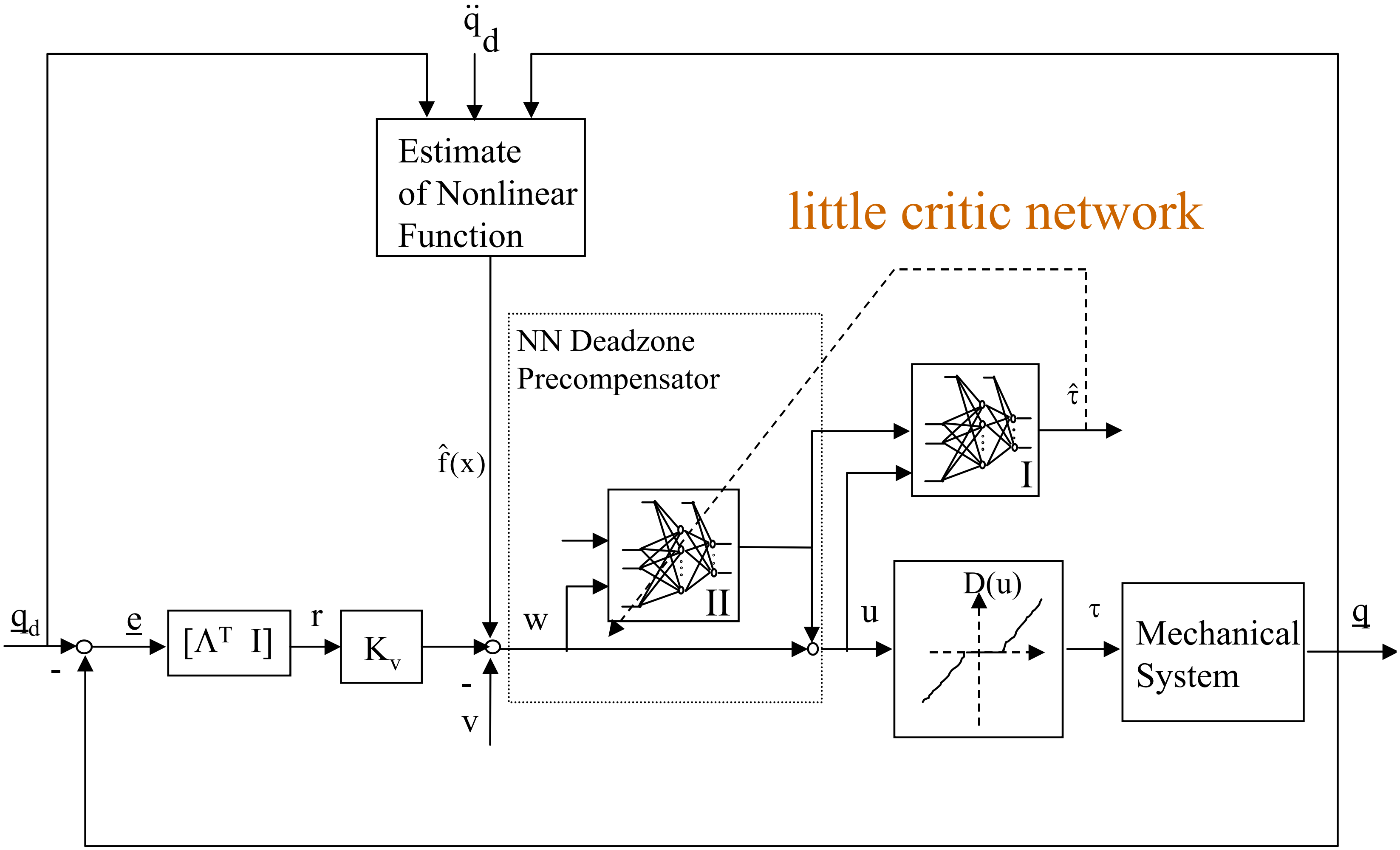
Augmented NN



Standard NN

Extra Nodes with
Jump Activation Functions

NN in Feedforward Loop- Deadzone Compensation

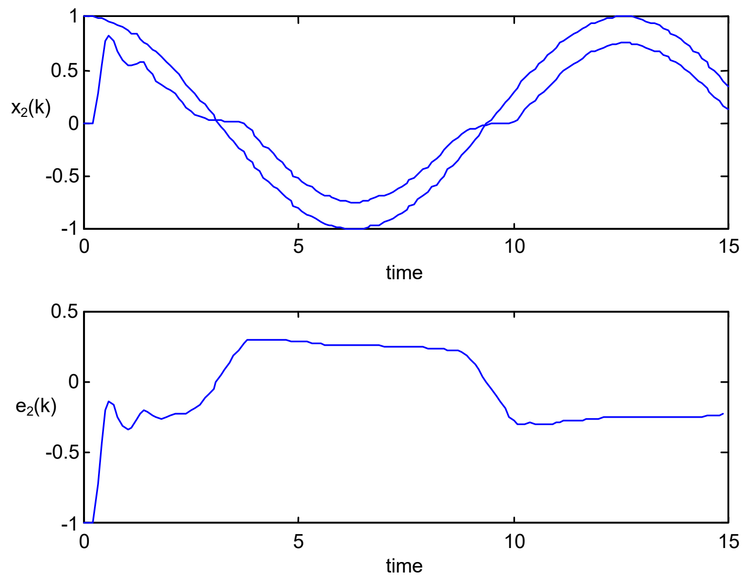


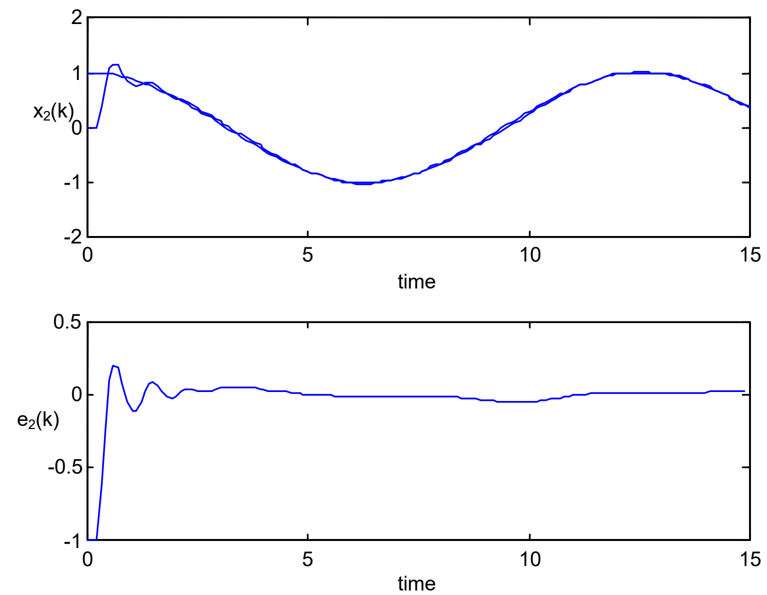
$$\hat{W}_i = T\sigma_i(U_i^T w) r^T \hat{W}^T \sigma'(U^T u) U^T - k_1 T\|r\|\hat{W}_i - k_2 T\|r\|\|\hat{W}_i\|\hat{W}_i$$

$$\hat{W} = -S\sigma'(U^T u) U^T \hat{W}_i \sigma_i(U_i^T w) r^T - k_1 S\|r\|\hat{W}$$

Acts like a 2-layer NN
With enhanced
backprop tuning !

Performance Results





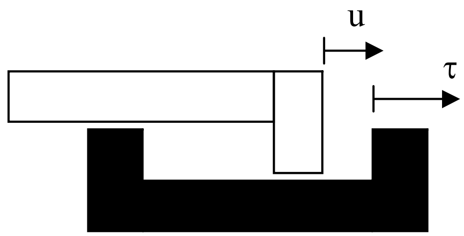
PD control-
deadzone chops out the middle

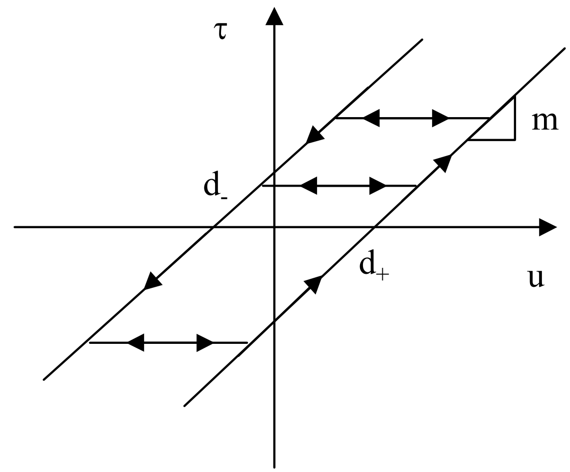
NN control fixes the problem

Backlash

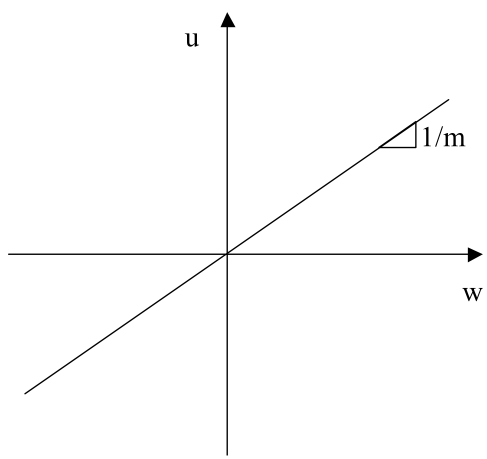
$$\dot{\tau} = B(\tau, u, \dot{u})$$

A dynamic nonlinearity

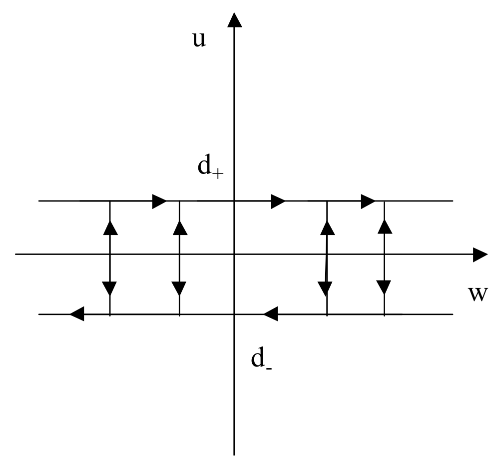




$$\dot{\tau} = B(\tau, u, \dot{u}) \; =$$

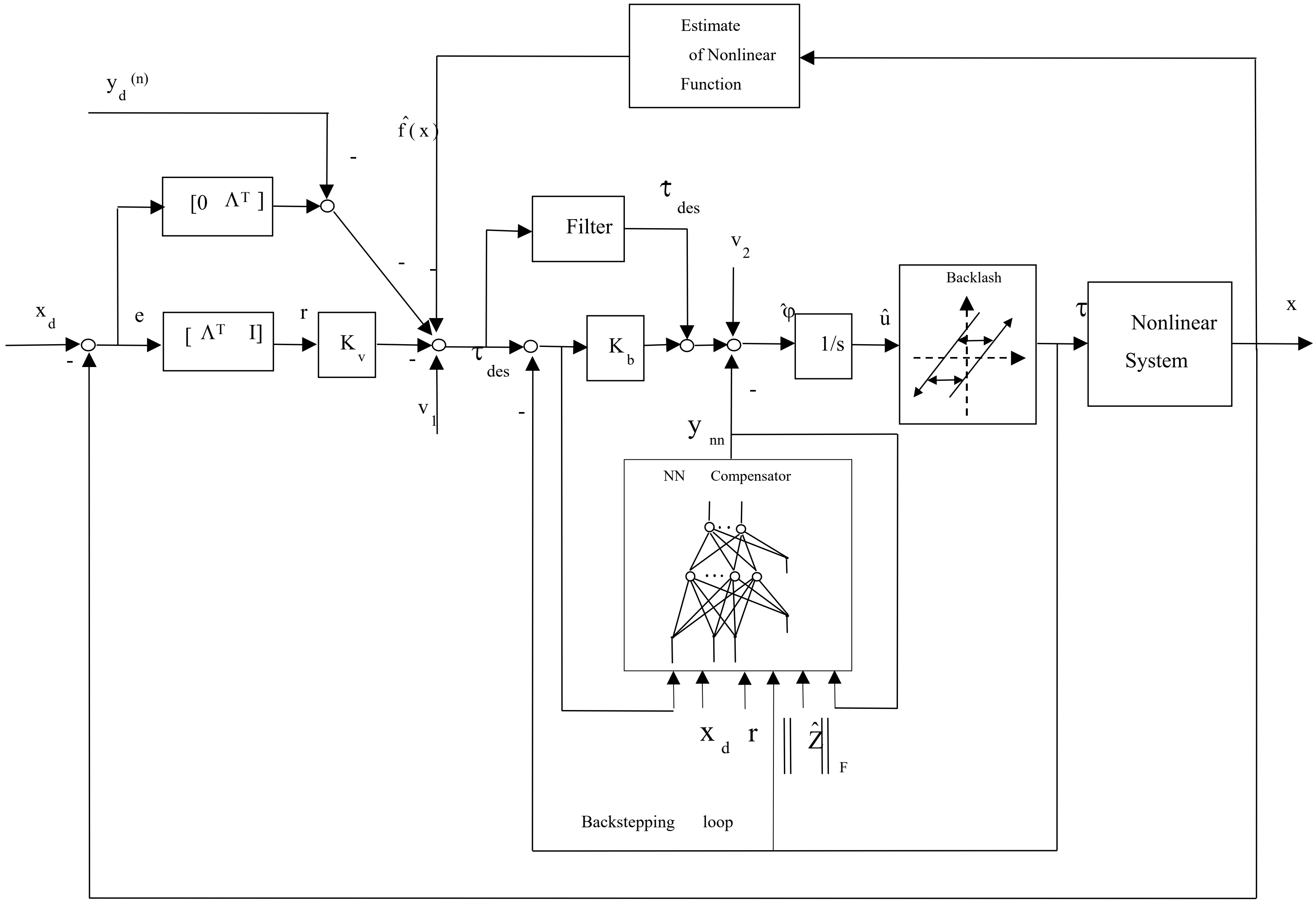


+



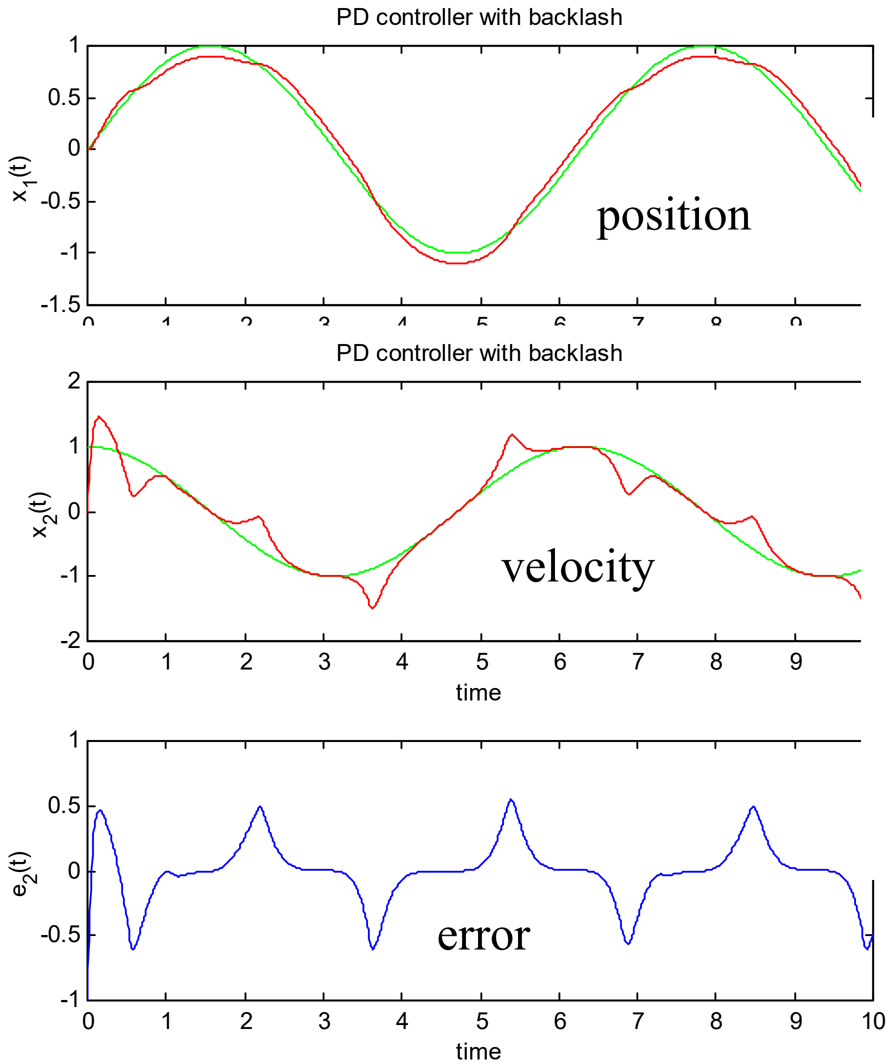
Nominal feedforward path + Unknown dynamic nonlinearity

Dynamic Inversion NN compensator for system with Backlash

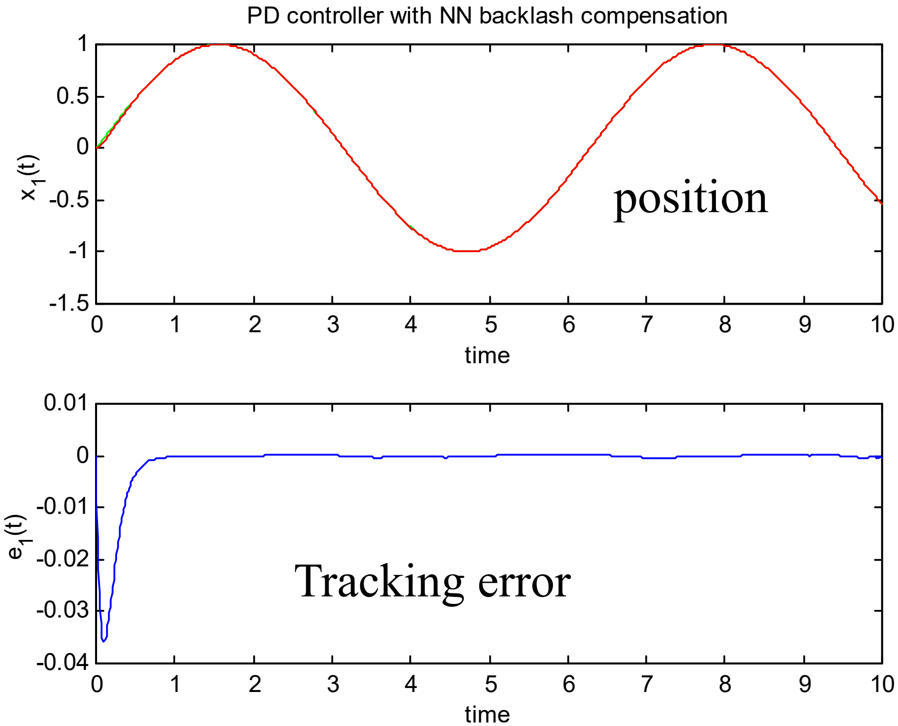


U.S. patent- Selmic, Lewis, Calise, McFarland

Performance Results



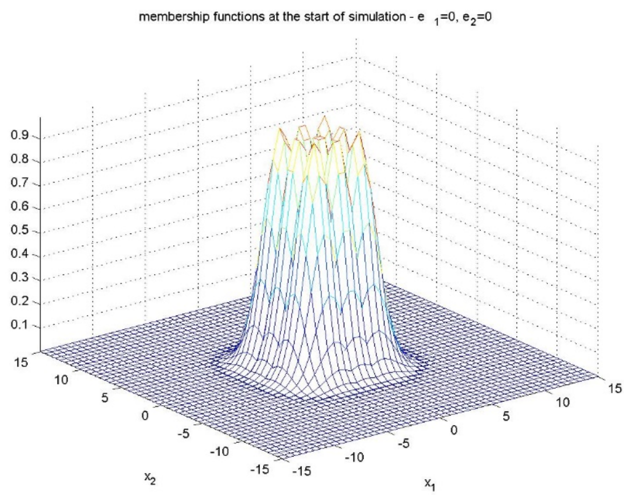
PD control-
backlash chops off tops & bottoms

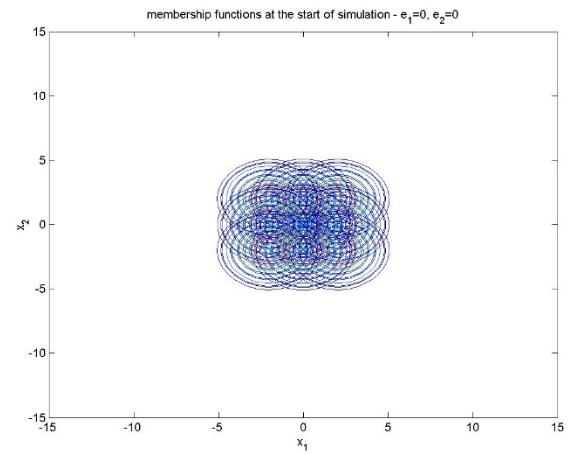


NN control fixes the problem

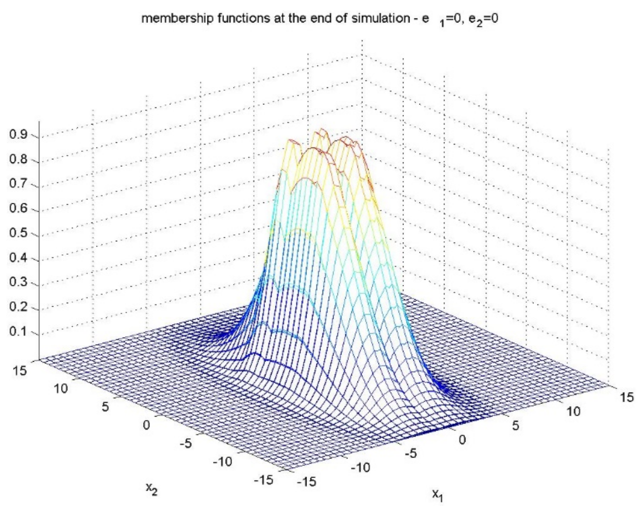
Dynamic Focusing of Awareness

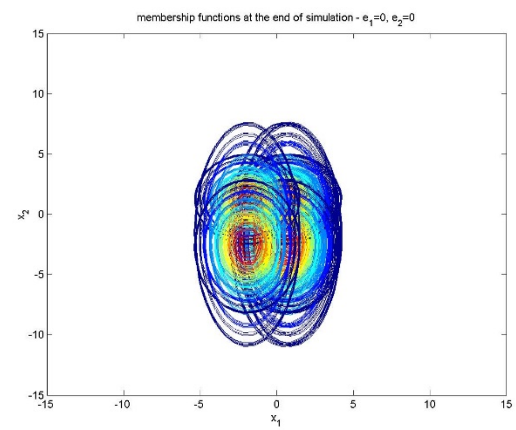
Initial MFs





Final MFs





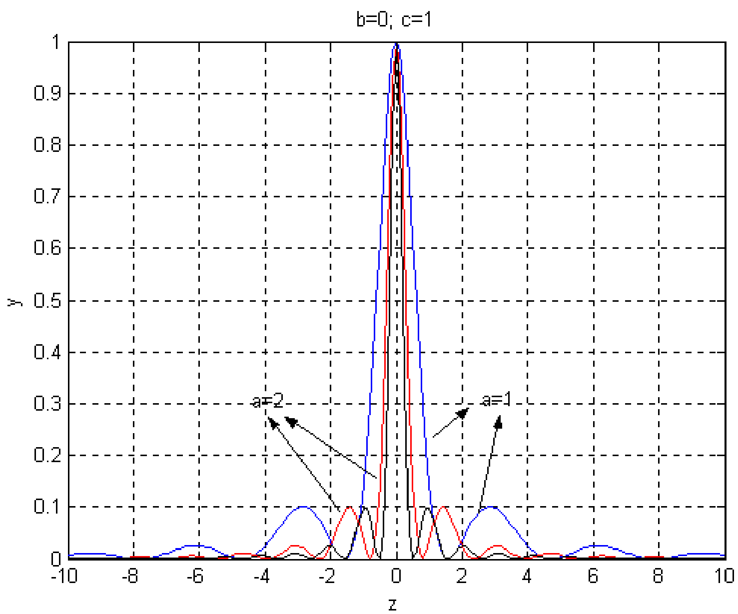
Depend on desired reference trajectory

Elastic Fuzzy Logic- c.f. P. Werbos

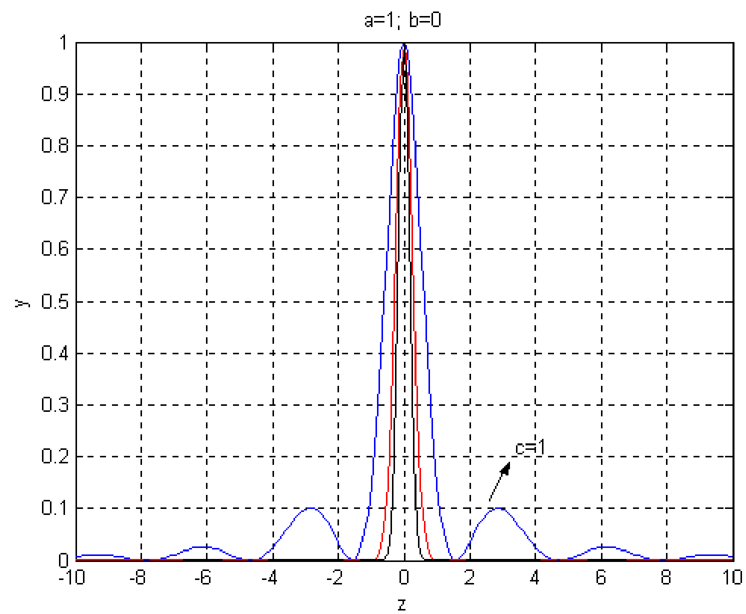
$$\phi(z,a,b,c) = \phi_B(z,a,b)^{c^2}$$

Weights importance of factors in the rules

$$\phi(z,a,b,c) = \left[\frac{\cos^2(a(z-b))}{1+a^2(z-b)^2} \right]^{c^2}$$



Effect of change of membership function spread "a"



Effect of change of membership function elasticities "c"

Elastic Fuzzy Logic Control

Control

$$u(t) = -K_v r - \hat{g}(x, x_d)$$

Tune Control Rep. Values

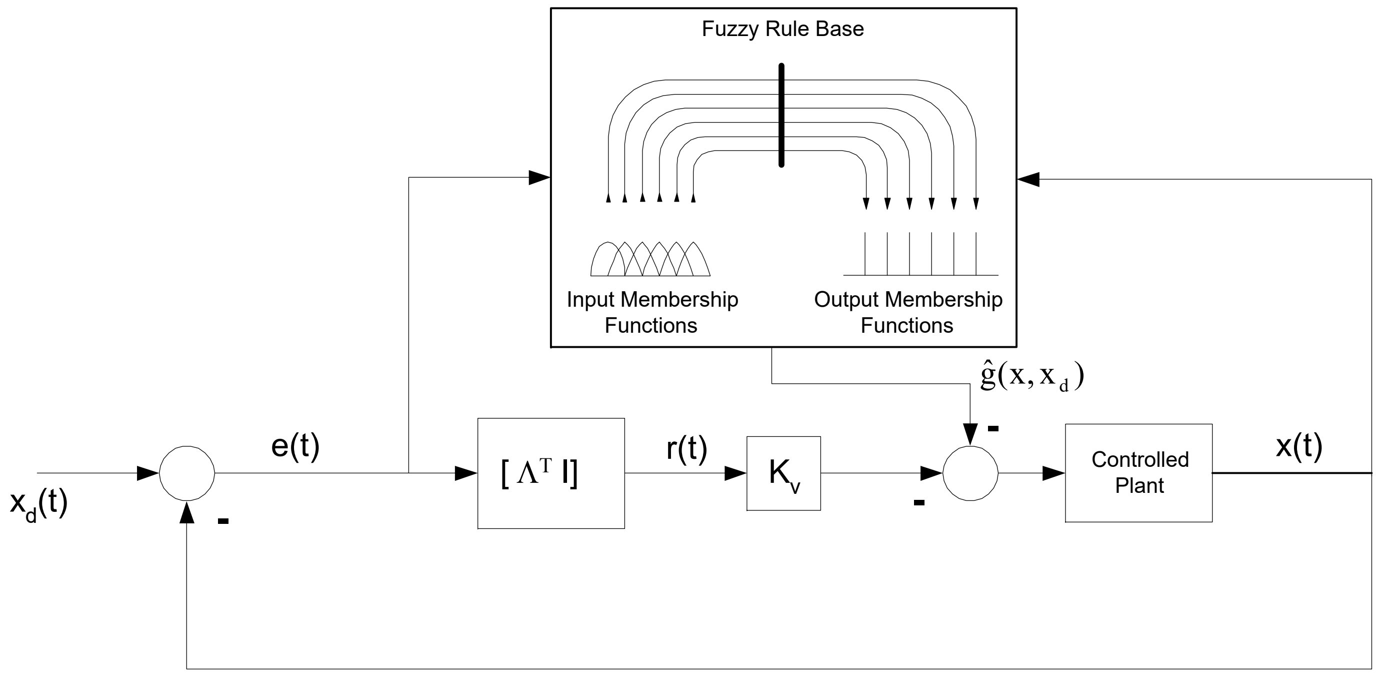
$$\dot{\hat{W}} = K_W(\hat{\Phi} - A\hat{a} - B\hat{b} - C\hat{c})r^T - k_W K_W \hat{W}\|r\|$$

Tune Membership Functions

$$\dot{\hat{a}} = K_a A^T \hat{W} r - k_a K_a \hat{a}\|r\|$$

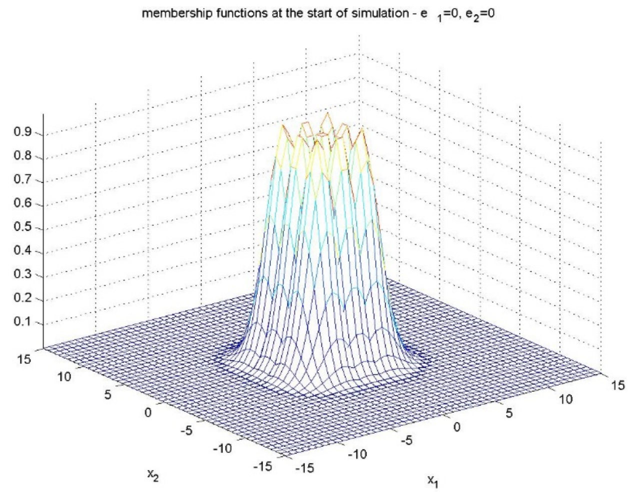
$$\dot{\hat{b}} = K_b B^T \hat{W} r - k_b K_b \hat{b}\|r\|$$

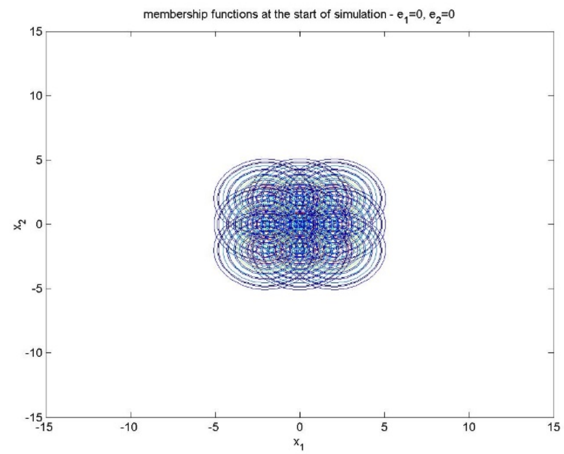
$$\dot{\hat{c}} = K_c C^T \hat{W} r - k_c K_c \hat{c}\|r\|$$



Better Performance

Initial MFs





Final MFs