Approximate Model Inversion Based Model Reference Adaptive Control (AMI-MRAC)

Recent Advances in Model Reference Adaptive Control: Theory and Applications

ORGANIZERS

ANTHONY CALISE, *Georgia Institute of Technology*
ERIC JOHNSON, *Georgia Institute of Technology*
TANSEL YUCELEN, *Georgia Institute of Technology*
GIRISH CHOWDHARY, *Massachusetts Institute of Technology*

AIAA Guidance, Navigation, and Control Conference

11-12 Aug 2012, Minneapolis, MN

Approximate Model Inversion (AMI) Based MRAC

Plant

$$\ddot{x} = f(x, \dot{x}, u),$$

- $x \in \Re^n$: system states
- $u \in \Re^m$: control inputs (multiple input), controllable
- f satisfies conditions for a unique solution, unknown

Problem Statement: Model Reference Adaptive Control

Design a control law $u(t)$ such that the plant tracks the reference model

$$\ddot{x}_{rm} = f_{rm}(x_{rm}, \dot{x}_{rm}, r)$$

- $x_{rm} \in \Re^n$: system states
- $r \in \Re^l$: Exogenous inputs
- f_{rm} bounded input bounded output stable and continuously differentiable

Approach

- Design a pseudo control $\nu \in \Re^n$ such that $x \to x_{rm}$
- If plant model were known and invertible, then we can simply set $u = f^{-1}(x, \dot{x}, \nu)$, in this case $\nu = \ddot{x}_{desired}$
- However, this is usually not the case, so choose an inversion model $\hat{f}$ and let $u = \hat{f}^{-1}(x, \nu)$

Assumption

The approximate inversion model $\hat{f}: \Re^{n+l} \to \Re^n$ is continuous,
The operator $\hat{f}^{-1}: \Re^{2n} \to \Re^l$ exists and assigns a unique $u \in \Re^l$ to every unique $(x, \dot{x}, \nu) \in \Re^{2n}$

- So, we have $\nu = \hat{f}(x, \dot{x}, u)$
- Realize that this usually means dim($\ddot{x}$)=dim(u). This is a matter of finding the right representation for most feedback linearizable systems
- E.g. in rotorcraft control, dim inputs 4, and 4 key directly affected states (p, q, r, a_z)

AMI-MRAC

- Alternative system descriptions that can be handled:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = f(x_1, x_2, u)$$

- As an example, consider systems affine in control

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = Gx_2 + B(\Theta(x_1, x_2) - u)$$

- In this case, if the pair (G, B) is controllable, we can use $\nu = Bu(t)$ is a valid choice for inversion model
- The model error here is: $\Delta(x_1, x_2) = Gx_2 + B\Theta(x_1, x_2)$

Control Action

- Control task: Design a pseudo control ν that achieves $e = \begin{bmatrix} x_{rm} - x \\ \dot{x}_{rm} - \dot{x} \end{bmatrix} \to 0$
- The combined control action has the following three parts
- Linear feedback

$$\nu_{pd} = [K_p \quad K_d] \begin{bmatrix} x_{rm} - x \\ \dot{x}_{rm} - \dot{x} \end{bmatrix}$$

- Feedforward

$$\nu_{rm} = \ddot{x}_{rm}$$

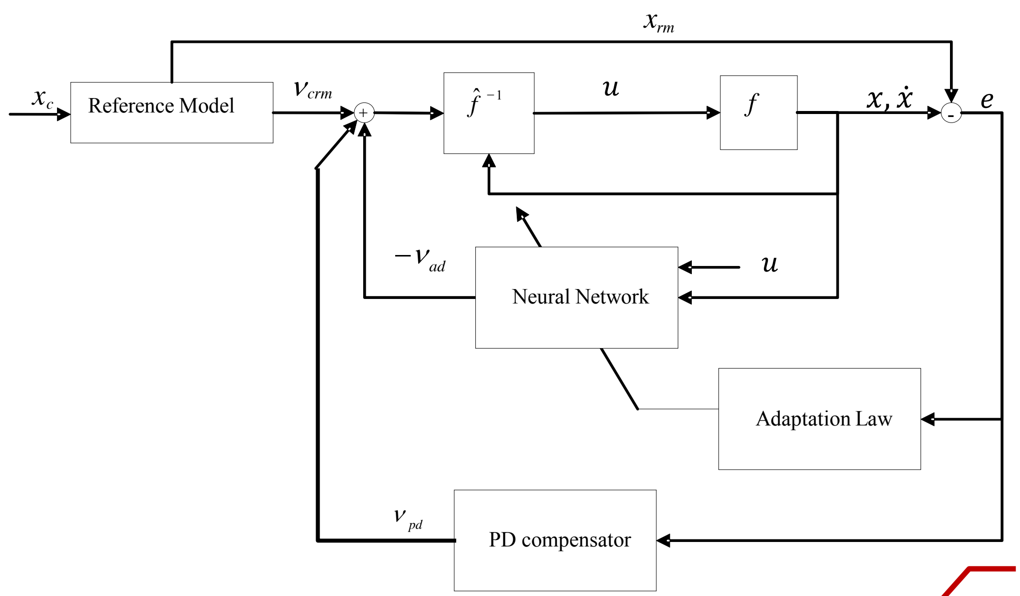
- Adaptive

$$\nu_{ad}$$

- Combined control action:

$$\nu = \nu_{pd} + \nu_{rm} - \nu_{ad}$$

Tracking Error Dynamics



Modeling error $\Delta \in \Re^n$

$$\ddot{x} = \hat{f}(x,u) + \left[f(x,u) - \hat{f}(x,u)\right]$$

- So, $\ddot{x} = v + \Delta(x,u)$

Tracking error dynamics

- Let's differentiate e,

$$\dot{e} = \begin{bmatrix} \dot{x}_{rm} - \dot{x} \\ \ddot{x}_{rm} - \ddot{x} \end{bmatrix}$$

- Now, $\ddot{x} = \hat{f}(x,u) + [f(x,u) - \hat{f}(x,u)] = \nu + \Delta(x,u)$

$$\dot{e} = \begin{bmatrix} \dot{x}_{rm} - \dot{x} \\ \nu_{rm} - (\nu + \Delta) \end{bmatrix} = \begin{bmatrix} \dot{x}_{rm} - \dot{x} \\ \cancel{\nu_{rm}} - (\nu_{pd} + \cancel{\nu_{rm}} - \nu_{ad} + \Delta) \end{bmatrix}$$

Let $A = \begin{bmatrix} 0 & I \\ K_{pd} & K_d \end{bmatrix}$, and $B = \begin{bmatrix} 0 \\ I \end{bmatrix}$

Tracking error dynamics

$$\dot{e} = Ae + B(\nu_{ad} - \Delta)$$

Choice of adaptive element

Structured Uncertainty

Given that there exists a constant matrix W^* and known basis functions $\Phi(x,u)$ such that

$$\Delta(x,u) = {W^*}^T \Phi(x,u)$$

Then, choose adaptive element:

$$\nu_{ad} = W^T \Phi(x,u)$$

Unstructured Uncertainty

Given $\Delta(x,u)$ is continuous and defined over a compact set, then :

$$\Delta(x,u) = g(x,u,W) + \tilde{\epsilon}$$

where g is some universal function approximator parameterized by W

Adaptive law

- So, $\nu = \nu_{pd} + \nu_{rm} - W^T(t)\Phi(x,u)$
- In traditional direct adaptive control, the aim is to create an adaptive law $\dot{W}$ such that $e \rightarrow 0$
- The key idea is going to be to update the weights such that some reward is maximized

Instantaneous Error Minimization

Updates weights in the direction of maximum reduction of quadratic cost of the instantaneous tracking error

$$\dot{W}(t) \cong -\Gamma \frac{\partial\left(e(t)^T e(t)\right)}{\partial W}$$

MRAC with Instantaneous data

Instantaneous Error Minimization

Updates weights in the direction of maximum reduction of quadratic cost of the instantaneous tracking error

$$\dot{W}(t) \cong -\Gamma \frac{\partial\left(e(t)^T e(t)\right)}{\partial W}$$

- Letting P be the solution of the closed loop Lyapunov equation and $\Gamma > 0$ the learning rate, this results in the following rank -1 adaptive law:

$$\dot{W} = -\Gamma\Phi(x)e^T PB$$

- This adaptive law requires PE to ensure exponential stability of $\left(e, \tilde{W} \equiv 0\right)$

Derivation of adaptive law

- Recall tracking error dynamics:

$$\dot{e} = Ae + B(\nu - \Delta)$$

- $\nu - \Delta = W^T(t)\Phi(x,u) - {W^*}^T\Phi(x,u) = \tilde{W}^T\Phi(x,u)$
- Since A is Hurwitz by design, we have that for every positive definite Q, a positive definite matrix P exists such that:

$$AP + PA^T + Q = 0$$

- Let's start with a positive definite Lyapunov Candidate: $V\left(e, \tilde{W}\right) = \frac{1}{2}e^T Pe + \frac{1}{2}\tilde{W}^T\Gamma^{-1}\tilde{W}$

Derivation of adaptive law

- Lie derivative of the Lyapunov candidate

$$\dot V(e,\widetilde W) = -e^TQe + e^TPB(\nu-\Delta) + \dot W^T\Gamma^{-1}\widetilde{\mathrm{W}}$$

$$\dot V(e,\widetilde W) = -e^TQe + e^TPB\widetilde W^T\Phi(x,\mathrm{u}) + \widetilde W^T\Gamma^{-1}\dot W$$

- Notice that e^TPB is a scalar, so

$$\dot V(e,\widetilde W) = -e^TQe + \widetilde W^T\left(\Gamma^{-1}\dot W + e^TPB\Phi(x,\mathrm{u})\right)$$

- Set the term in the bracket to zero, so

$$\dot W = \Gamma e^TPB\Phi(x,u)$$

$$\dot V(e,\widetilde W) \le -\frac{1}{2}\lambda_{min}(Q)||e||^2$$

- From here use Barbalat's lemma to show that $e\to 0$ asymptotically, and weights bounded (when no noise)

Entire closed loop system

- The entire closed loop system has two coupled differential equations:

$$\dot{e} = Ae + B\widetilde{W}^T\Phi(x,u)$$

$$\dot{\widetilde{W}} = -\Gamma e^T PB\Phi(x,u)$$

- Here we assumed W^* to be a constant
- Notice no feedback from $\widetilde{W}$ in the second equation
- This could lead to issues with boundedness of W, especially if there is noise in the system
- When does $\widetilde{W} \rightarrow 0$? (Persistency of excitation)

Some relevant existing work

- Baseline adaptive law only guarantee $e(t) \to 0$ asymptotically, no performance guarantees
- Classic modifications for weight boundedness:
  - σ-mod (Ioannou 84), e-mod (Narendra 86), projection based adaptation
- Intelligent excitation (Cao 07):
  - Excitation enforced as a function of tracking error
  - Increased control cost due to added excitation
- L_1 adaptive control (Cao, Hovakimyan 08)
- Q-modification (Volyanksyy 09)
- Direct Recursive Least Squares Adaptation (Ngyuen 2006)

Example: Inverted pendulum

- A nonlinear system which resembles an inverted pendulum

$$\ddot{x} = \delta + \sin(x) - |\dot{x}|\dot{x} + 0.5e^{x\dot{x}}$$

- Approximate inversion model: $\nu = \delta$
- Model error $\Delta = \begin{bmatrix} 1 & -1 & 0.5 \end{bmatrix} \begin{bmatrix} \sin(x) \\ |\dot{x}|\dot{x} \\ e^{x\dot{x}} \end{bmatrix}$
- Second order reference model
- Simulation file: invpendu_dmrac

Wing Rock dynamics simulation

- Highly swept-back aircraft are susceptible to Wing Rock: lightly damped oscillations
- ϕ: roll angle, p roll rate, δ_a aileron

A model for Wing Rock dynamics (Monahemi 96)

$$\dot{\phi} = p$$
$$\dot{p} = \delta_a + \Delta(\phi, p)$$

- Inversion model: $\nu = \delta$
- Task: track roll commands in presence of wing rock dynamics:

$$\Delta(\phi, p) = 0.8 + 0.23\phi + 0.69p - 0.62|\phi| + 0.01|p|p + 0.02p^3$$

- Second order reference model used
- Simulation file: wingrock_dmrac_simple

Issues

- Boundedness of $\tilde{W}$ in presence of noise etc
- Transient response
- Actuator saturation
- Imperfect selection of basis

Neuroadaptive Control

Recent Advances in Model Reference Adaptive Control: Theory and Applications

ORGANIZERS

ANTHONY CALISE, *Georgia Institute of Technology*
ERIC JOHNSON, *Georgia Institute of Technology*
TANSEL YUCELEN, *Georgia Institute of Technology*
GIRISH CHOWDHARY, *Massachusetts Institute of Technology*

Introduction

- Neural Networks are universal function approximators
- That means, given any continuous function on a compact domain, we have a tool to form a good approximation without having to know more
- Single Hidden Layer, and Radial Basis Function NN have been widely used as adaptive elements

Model Reference Adaptive Control

Plant

$$\dot{x}(t) = \mathrm{A}x(t) + \mathrm{B}\left(u(t) + \Delta\big(x(t)\big)\right),$$

- $x \in \Re^n$: system states, $u \in \Re$: control inputs, $\Delta\big(x(t)\big)$: uncertainty
- $\mathrm{A} \in \Re^{n\times n}$, $\mathrm{B} = [0, \ldots, 1]^T$, easily generalizes to multi-input cases

Problem Statement: Model Reference Adaptive Control

Design a control law $u(t)$ such that the plant tracks the reference model

$$\dot{x}_{rm}(t) = \mathrm{A_{rm}}x_{rm}(t) + \mathrm{B_{rm}}\mathrm{r(t)},$$

- $x_{rm}(t) \in \Re^n$: system states, $r(t) \in \Re$: Exogenous inputs
- Assumed bounded input bounded output stable
- Chosen to characterize the desired response

Control Action

- The tracking control law has the following three parts:

- Linear feedback

$$u_{pd} = K(x_{rm} - x)$$

- Linear feedforward:

$$u_{rm} = K_r[x_{rm}, r]$$

- Adaptive part

$$u_{ad}$$

Tracking control law:

$$u = u_{pd} + u_{rm} - u_{ad}$$

- Tracking error dynamics, $(e = x_{rm} - x)$

$$\dot{e} = (A - BK)e + B(u_{ad} - \Delta)$$

Approximate Model Inversion (AMI) Based MRAC

Plant

$$\ddot{x} = f(x, \dot{x}, u),$$

- $x \in \Re^n$: system states
- $u \in \Re^m$: control inputs (multiple input), controllable
- f satisfies conditions for a unique solution, unknown

Problem Statement: Model Reference Adaptive Control

Design a control law $u(t)$ such that the plant tracks the reference model

$$\ddot{x}_{rm} = f_{rm}(x_{rm}, \dot{x}_{rm}, r)$$

- $x_{rm} \in \Re^n$: system states
- $r \in \Re^l$: Exogenous inputs
- f_{rm} bounded input bounded output stable and continuously differentiable

Approach

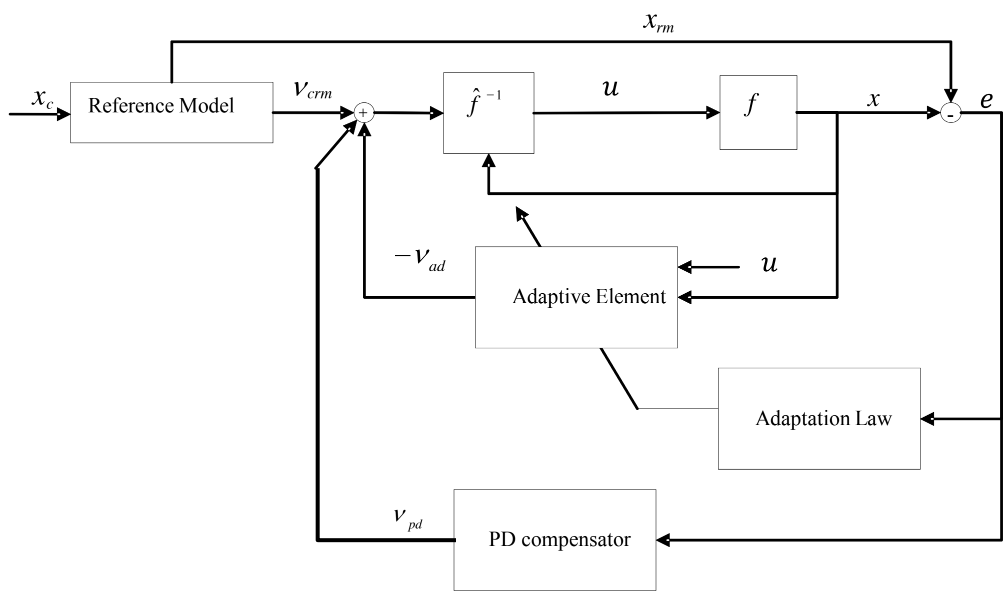
- Design a pseudo control $\nu \in \Re^n$ such that $x \to x_{rm}$
- If plant model were known and invertible, then we can simply set $u = f^{-1}(x, \nu)$
- However, this is usually not the case, so choose an inversion model $\hat{f}$ and let $u = \hat{f}^{-1}(x, \nu)$

Assumption

The approximate inversion model is continuous, and invertible w.r.t. u. I.e. the operator $\hat{f}^{-1}: \Re^{2n} \to \Re^m$ exists and assigns for every unique $(x, \nu) \in \Re^{2n}$ a unique $u \in \Re^m$

- Combined control action: $\nu = -Ke + \dot{x}_{rm} - \nu_{ad}$

Tracking Error Dynamics



$$\nu = \nu_{pd} + \nu_{rm} - \nu_{ad}$$

- Tracking error dynamics:

$$\dot{e} = Ae + B(\nu_{ad} - \Delta)$$

Choice of adaptive element

Structured Uncertainty

Given that there exists a constant matrix W^* and known basis functions $\Phi(x,u)$ such that

$$\Delta(x,u) = {W^*}^T \Phi(x,u)$$

Then, choose adaptive element:

$$\nu_{ad} = W^T \Phi(x,u)$$

Unstructured Uncertainty

Given $\Delta(x,u)$ is continuous and defined over a compact set, then :

$$\Delta(x,u) = g(x,u,W) + \tilde{\epsilon}$$

where g is some universal function approximator parameterized by W

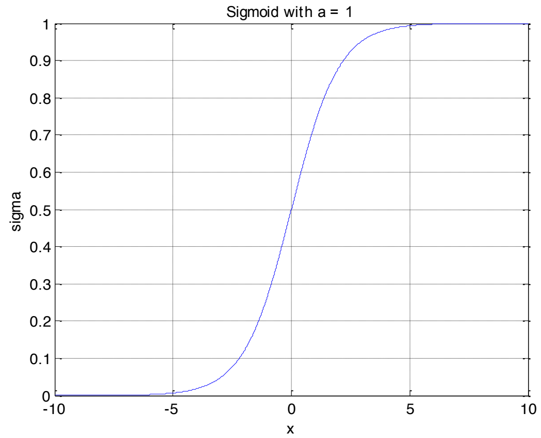
Single Hidden Layer Neural Network

- Single Hidden Layer (SHL) feedforward NN are universal approximators

$$\nu_{ad} = W^T(t)\sigma(V^T\bar{x})$$

- $\bar{x} = [b_a \;\; x]$, where b_a is the input bias (constant)

- $\sigma = \begin{bmatrix} b_w \\ \frac{1}{1+e^{a_1 z_1}} \\ \vdots \\ \frac{1}{1+e^{a_{n_2} z_{n_2}}} \end{bmatrix} \in \Re^{n_2+1}$, sigmoidal activation function



$$V = \begin{bmatrix} \theta_{1,1} & \cdots & \theta_{1,n2} \\ v_{1,1} & \cdots & \theta_{1,n2} \\ \vdots & \ddots & \vdots \\ v_{n1,1} & \cdots & \theta_{n1,n2} \end{bmatrix} \in \Re^{(n_1+1)\times n_2} \qquad W = \begin{bmatrix} \theta_{1,1} & \cdots & \theta_{1,n2} \\ w_{1,1} & \cdots & w_{1,n2} \\ \vdots & \ddots & \vdots \\ w_{n2,1} & \cdots & w_{n2,n3} \end{bmatrix} \in \Re^{(n_{2_1}+1)\times n_3}$$

Universal approximation property

- Given sufficient number of hidden layer neurons (n_2), and a corresponding set of ideal weights (W^*, V^*) there exists $\bar{\epsilon}$ such that

$$\bar{\epsilon} = \sup_{\bar{x} \in \mathrm{D}} ||{W^*}^T \sigma\left({V^*}^T \bar{x}\right) - \Delta(z)||$$

- Ideally we would like to update W, V such that they approach a compact neighborhood of W^*, V^*
- Direct adaptive control is happy with just updating W, V such that tracking error e stays bounded

NN training law

- The "Classical" Error backpropagation based learning Law:

$$\dot{W} = -(\sigma - \sigma' V^T \bar{x}) r^T \Gamma_w$$

$$\dot{V} = -\Gamma_V \bar{x} r W^T \sigma'$$

where $$r^T = e^T PB$$

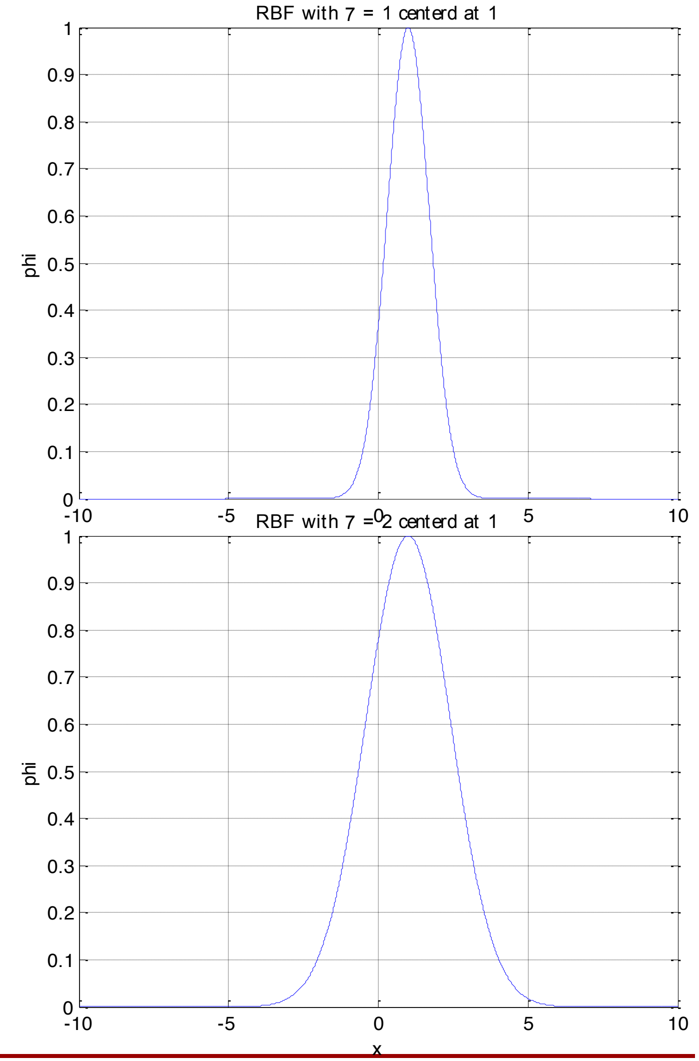
$$\sigma' = diag\left(\sigma(V^T\bar{x})\left(I - diag\left(\sigma(V^T\bar{x})\right)\right)\right)$$

- This adaptive law guarantees uniform ultimate boundedness of $(e, \widetilde{W})$ (Lewis et al.)

Radial Basis Function NN

- Radial Basis Functions (RBFs) are Gaussian Kernels
- Select n_2 centers x_{c_j}
- Select a width parameter
- Add a bias term, then $\Phi(\bar{x}) \in \Re^{n_2+1}$

$$\Phi(\bar{x}) = \begin{bmatrix} b_w \\ e^{\frac{-||x_{c_1}-\bar{x}||}{\mu}} \\ \vdots \\ e^{\frac{-||x_{c_{n_2}}-\bar{x}||}{\mu}} \end{bmatrix}$$



Universal Approximation with RBFs

- Given fixed number of RBFs with a fixed width μ there exists an ideal set of weights W^* such that

$$\bar{\epsilon} = \sup_{\bar{x} \in \mathrm{D}} ||{W^*}^T \Phi(\bar{x}) - \Delta(z)||$$

- $\bar{\epsilon}$ can be made arbitrarily small given sufficient number of RBFs
- RBF NN $\nu_{ad} = W^T(t)\Phi(\bar{x})$
- Ideally, we would like $W(t) \rightarrow \mathrm{W}^*$
- Traditionally, adaptive control is happy with keeping e bounded

RBF adaptive law

- The following adaptive law guarantees that the tracking error stays bounded in a compact neighborhood

$$\dot{W} = -\Gamma\Phi(\bar{x})e^T PB$$

- Can use $\sigma -$ mod, or $e -$ mod to guarantee boundedness in presence of noise:

$$\dot{W} = -\Gamma\Phi(\bar{x})e^T PB - \kappa W$$

Which NN to choose?

- Benefits of SHL NN:
  - Don't need to select centers
  - More general parameterization
- Issues:
  - Nonlinearly parameterized, which makes analysis hard
  - Rank-1 updates do not leverage the power of this NN
- Benefits of RBF NN
  - Linearly parameterized, relatively straight forward to analyze
  - Lends to the theory of Gaussian Kernels and Reproducing Kernel Hilbert Spaces
- Issues:
  - How to select the centers (see Concurrent Learning Talk)
  - How many RBFs to use?
  - Can run into issues with bias estimation if the centers are far away

Example: Inverted pendulum (SHL NN)

- A nonlinear system which resembles an inverted pendulum

$$\ddot{x} = \delta + \sin(x) - |\dot{x}|\dot{x} + 0.5e^{x\dot{x}}$$

- Approximate inversion model: $\nu = \delta$
- Model error $\Delta = \begin{bmatrix} 1 & -1 & 0.5 \end{bmatrix} \begin{bmatrix} \sin(x) \\ |\dot{x}|\dot{x} \\ e^{x\dot{x}} \end{bmatrix}$
- Second order reference model

Example Inverted Pendulum

Software:NN_InvertedPendulum_simple

Try the following

- Control without NN: set
  - gammaW=0; (line 61)
  - gammaV=0; (line 62)

Wing Rock dynamics(RBF NN)

- Highly swept-back aircraft are susceptible to Wing Rock: lightly damped oscillations
- ϕ: roll angle, p roll rate, δ_a aileron

A model for Wing Rock dynamics (Monahemi 96)

$$\dot{\phi} = p$$
$$p = \delta_a + \Delta(\phi, p)$$

- Inversion model: $\nu = \delta$
- Task: track roll commands in presence of wing rock dynamics:

$$\Delta(\phi, p) = 0.8 + 0.23\phi + 0.69p - 0.62|\phi| + 0.01|p|p + 0.02p^3$$

- Second order reference model used

Example wingrock with RBF

Software: wingrock_rbf_simple

Things to try

- Turn off adaptive control by setting gammaW=0
- Add sigma mod: by setting kappa=0.1
- Add e-mod by setting zeta=0.1
- What effect does the bias term ($W^*(1)$) have (this simulates trim)

Issues

- Boundedness of $\tilde{W}$ in presence of noise etc
- Transient response
- Actuator saturation
- In case of RBFs, how do we distribute the centers?
  - See paper *"A Reprodcing Kernel Hilbert Space Approach to Adaptive Control", Kingravi, Chowdhary, Vela, Johnson, CDC 2011*
- In general, traditional adaptive laws don't guarantee that the weights approach their ideal values (as dictated by the universal approximation property)
- Persistency of excitation is needed to guarantee this