Neural Networks
Lecture 10: Fault Detection and Isolation (FDI) Using Neural Networks

H.A. Talebi
Farzaneh Abdollahi

Department of Electrical Engineering

Amirkabir University of Technology

Winter 2011

Additive FDI

- Consider the nonlinear dynamics:

$$\dot{x} = f(x,u) + N_1(x,u) + T_F(x,u,t) \tag{1}$$

  - f: known nonlinear fcn.
  - $N_1(x,u)$: Unmodeled dynamics and uncertainties of nominal model (fault free), estimated by NN1
  - $T_F(x,u,t)$: vector valued fcn. of unknown actuator fault
- Objective: Diagnose and Isolate Actuator Fault: (T_F)
- By defining a Hurwitz matrix A, (1) can be rewritten

$$\dot{x} = Ax + g(x,u) + N_1(x,u) + T_F(x,u,t)$$

  - $g(x,u) = f(x,u) - Ax$
- $M(s) = (sI - A)^{-1}$

Additive FDI

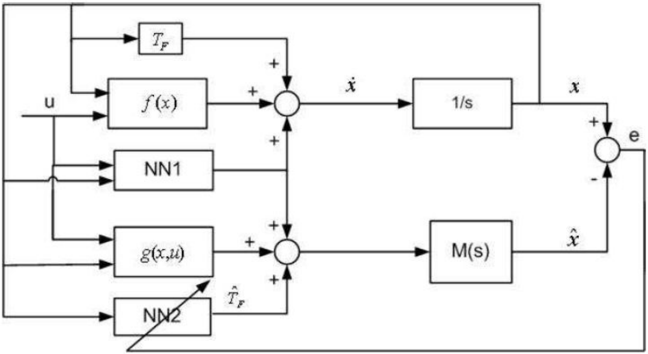


Fig. 4.1 Structure of the neural network estimator for actuator fault detection.

NN2

- T_f is estimated by NN_2: $\hat{T}_f(x, \hat{W}, \hat{V}, t) = \hat{W}\sigma(\hat{V}\bar{x})$
  - A three layer MLP
  - Series Parallel identifier
- $\therefore$ The estimated model

$$\dot{\hat{x}} \quad = \quad A\hat{x} + g(x, u) + N_1(x, u) + \hat{W}\sigma(\hat{V}\bar{x})$$

- It has been shown that by updating the weights of NN_2:
  $\dot{\hat{W}} = -\eta_1(\frac{\partial J}{\partial \hat{W}}) - \rho_1\|\tilde{x}\|\hat{W}$
  $\dot{\hat{V}} = -\eta_2(\frac{\partial J}{\partial \hat{V}}) - \rho_2\|\tilde{x}\|\hat{V}$
  - $\tilde{x} = x - \hat{x}$
  - $J = \frac{1}{2}(\tilde{x}^T\tilde{x})$: objective fcn to be minimized
- then $\tilde{x}, \tilde{W} = W - \hat{W}, \tilde{V} = V - \hat{V}$ are ultimately bounded
- The main advantage of the given approach: fault detection and Isolation is provided in a unified fashion

Example:FDI on Robotic Manipulation

- Dynamic of a robot manipulator $M(q)\ddot{q} + C(q,\dot{q}) + G(q) = \tau + \tau_F$
  - q: manipulator joints position
  - $\dot{q}$: joint velocity
  - τ: applied torque
  - τ_F: unknown fault
- Based on the defined model:

$$T_f = \begin{bmatrix} 0 \\ M^{-1}\tau_F \end{bmatrix}; f(x,u) = \begin{bmatrix} \dot{q} \\ M^{-1}(q)(\tau - C(q,\dot{q}) + G(q) + \tau_F) \end{bmatrix}$$

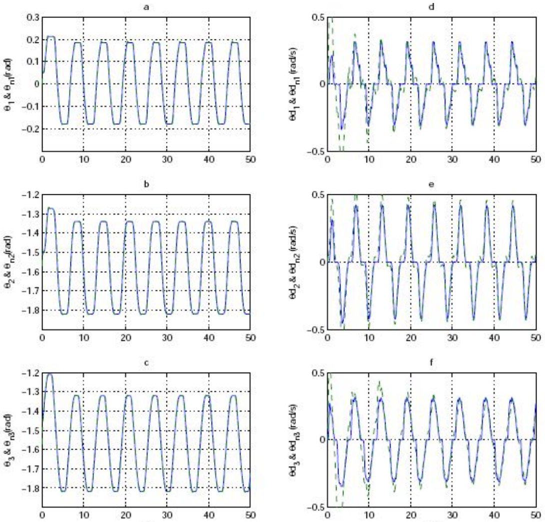
Experimental Results on Puma 560

Example Puma 560

- It has 6 joints
- The unmodelled dynamics (for Ex. friction in joint which are not modeled precisely) are estimated by N_1
- $\therefore$ The modified fault free description of Puma:
  $M(q)\ddot{q} + C(q,\dot{q}) + G(q) + N_1(q,\dot{q},\tau) = \tau$
- Define $\theta = q$
- At this stage, a PD controller is considered to control the robots behavior: $\tau_C = K_p(\theta_{ref} - \theta) + K_v(\dot{\theta}_{ref} - \dot{\theta}) + G$
  - Desired position: $\theta_{ref} = \theta_H + 0.3sin(t)$
  - $\theta_H = [0 \;\; -\pi/2 \;\; -\pi/2 \; 0 \; 0 \; 0]$

Fault Free Experimental Results

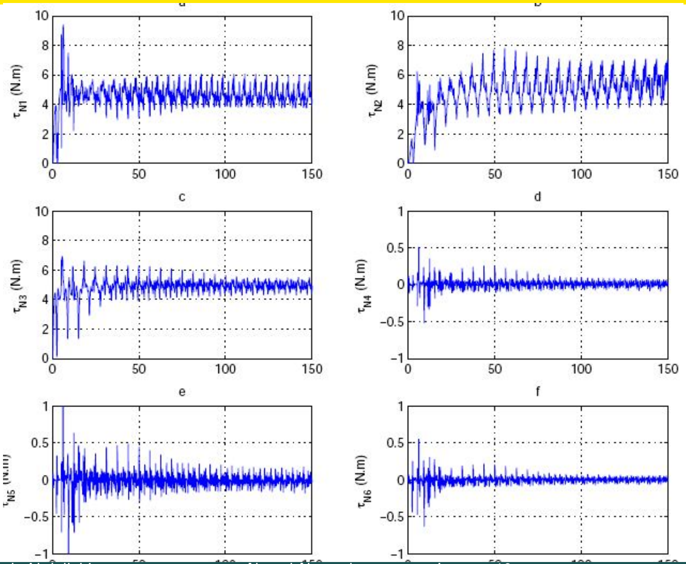
- (a)-(c) joint positions;
- (d)-(f) Joint velocities.
- Solid lines: actual measurements;
- Dashed lines: their estimates.



Faulty Experimental Results

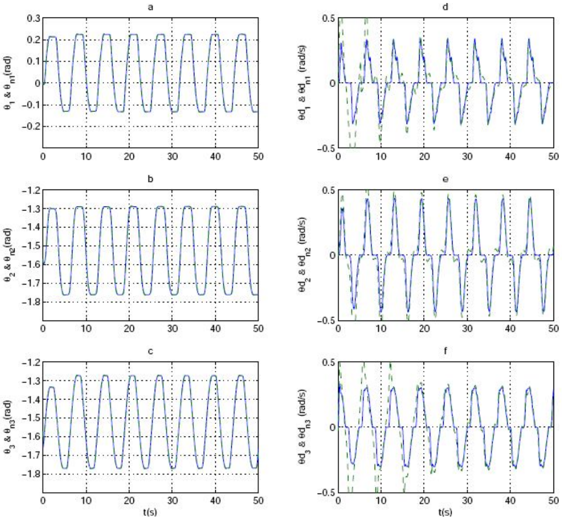
- Consider a constant actuator fault 5Nm in the first three joints: $\tau_F = [5\ 5\ 5\ 0\ 0\ 0]$
- Estimate τ_F by NN2: a three layer NN
  - input layers: 12 neurons $(\theta, \dot{\theta})$
  - hidden layer 10 neurons
  - output layer 6 neurons $\hat{\tau}_F$

The outputs of the neural network estimating the faults



Experimental Results with 5 Nm fault

- (a)-(c) Joint positions
- (d)-(f)Joint velocities.
- Solid lines: actual measurements
- Dashed lines:their estimates



An Actuator Gain FDI [1]

- Consider the nonlinear fault free dynamics:

$$\begin{aligned} \dot{x} &= f(x,u) + \eta_x(x,u,t) \\ y(t) &= Cx(t) + \eta_y(x,u,t) \end{aligned} \qquad (2)$$

  - $u \in \mathcal{R}^m$: input; $x \in \mathcal{R}^n$: state;$y \in \mathcal{R}^m$: output
  - f: known nonlinear fcn.
  - $\eta_x(x,u,t) \in \mathcal{R}^n \times \mathcal{R}^m \times \mathcal{R} \rightarrow \mathcal{R}^n$: Plant unmodeled dynamics and noise
  - $\eta_y(x,u,t) \in \mathcal{R}^n \times \mathcal{R}^m \times \mathcal{R} \rightarrow \mathcal{R}^m$: Sensor unmodeled dynamics and disturbances
  - C: an $m \times n$ conts. matrix
- Assume a multiplicative actuator fault occur:
$u(t) \rightarrow u_f(t) = T_A(x,u,t)u(t)$
  - $T_A(x,u,t) = diag\{T_{Ai}(x,u,t)\}, i = 1, ..., m$ unknown matrix

An Actuator Gain FDI

- $\therefore$ The faulty dynamics:

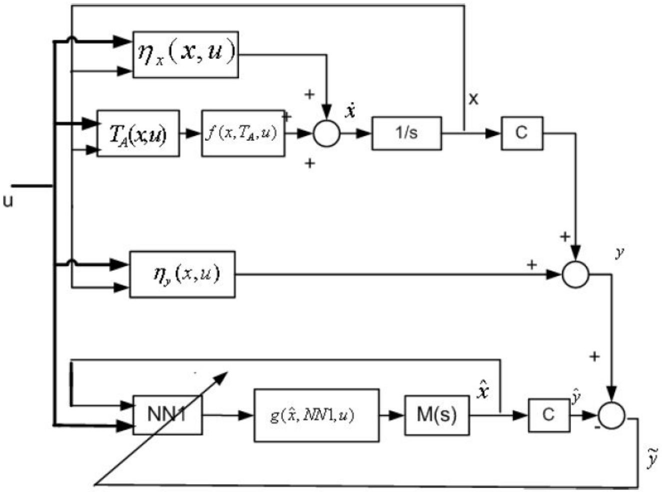
$$\begin{aligned} \dot{x} &= f(x, u_f) + \eta_x(x, u, t) \\ y(t) &= Cx(t) + \eta_y(x, u, t) \end{aligned} \quad (3)$$

- Objective: Diagnose and Isolate Actuator Fault: (T_F)
- By defining a Hurwitz matrix A, (3) can be rewritten

$$\begin{aligned} \dot{x} &= Ax + g(x, u_f) + \eta_x(x, u, t) \\ y(t) &= Cx(t) + \eta_y(x, u, t) \end{aligned}$$

  - $g(x, u) = f(x, u) - Ax$
- $M(s) = (sI - A)^{-1}$

An Actuator Gain FDI



An Actuator Gain FDI

- The following assumptions should be made to facilitate the FDI design
  - The nonlinear sys (3) is observable
  - The nominal closed loop sys is asym. stable ($\dot{x} = f(x, k(Cx))$)
  - η_x, η_y are uniformly bounded $\|\eta_x\| \leq \bar{\eta}_x, \|\eta_y\| \leq \bar{\eta}_y$
  - g is Lipshtiz in x and u: $\|g(x,u) - g(\hat{x},\hat{u})\| \leq l_g \left\| \begin{bmatrix} x - \hat{x} \\ u - \hat{u} \end{bmatrix} \right\|$
- T_f is estimated by NN_1: $\hat{T}_f(x, \hat{W}, \hat{V}, t) = \hat{W}\sigma(\hat{V}\bar{x})$
  - A three layer MLP
- $\therefore$ The estimated model

$$
\begin{aligned}
\dot{x} &= A\hat{x} + g(\hat{x}, \hat{u}_f) \\
\hat{y}(t) &= C\hat{x} \\
\hat{u}_f &= \hat{W}\sigma(\hat{V}\bar{x})u
\end{aligned}
$$

An Actuator Gain FDI

- It has been shown that by updating the weights of NN_1:

$$\dot{\hat{W}} = -\eta_1\left(\frac{\partial J}{\partial \hat{W}}\right) - \rho_1\|\tilde{y}\|\hat{W}$$
$$\dot{\hat{V}} = -\eta_2\left(\frac{\partial J}{\partial \hat{V}}\right) - \rho_2\|\tilde{y}\|\hat{V}$$

  - $\tilde{x} = x - \hat{x}$
  - $\tilde{y} = y - \hat{y}$
  - $\tilde{W} = W - \hat{W}$
  - $\tilde{V} = V - \hat{V}$
  - $J = \frac{1}{2}(\tilde{y}^T\tilde{y})$: objective fcn to be minimized
- then $\tilde{x}, \tilde{y}, \tilde{W}, \tilde{V}$ are ultimately bounded
- This FDI can diagnose and isloate simultaneous fault

Example of Flexible Joint Manipulator [?]

- The dynamical model:

$$\begin{aligned} D_l(q_1)\ddot{q}_1 &+ C_1(q_1,\dot{q}_1)+g(q_1)+B_1\dot{q}_1=\tau_s \\ J\ddot{q}_2 &+ \tau_s+B_2\dot{q}_2=\tau \end{aligned}$$

  - $q_1 \in \mathcal{R}^n$: link position
  - $q_2 \in \mathcal{R}^n$: motor shaft position
  - $g(q_1) \in \mathcal{R}^n$: gravity force
  - $C_1(q_1,\dot{q}_1) \in \mathcal{R}^n$: centrifugal and Coriolis force
  - $B_1 \in \mathcal{R}^{n\times n}, B_2 \in \mathcal{R}^{n\times n}$: viscus damping of input and output shaft
  - $D_1(q_1) \in \mathcal{R}^{n\times n}, J \in \mathcal{R}^{n\times n}$ robot and actuator inertia matrix
  - τ input torque
  - $\tau_s = K(q_2-q_1)+\beta(q_1,q_2,\dot{q}_1,\dot{q}_2)$: reaction torque from rotational spring
  - K: stiffness matrix of rotational spring represent the flexibility between the input and output shaft
  - $\beta(q_1,q_2,\dot{q}_1,\dot{q}_2)$: a combination of nonlinear spring and friction at output shafts of the manipulator
  - The output vector $y=[q_{21}\ \ q_{22}\ \ \dot{q}_{21}\ \ \dot{q}_{22}]^T$

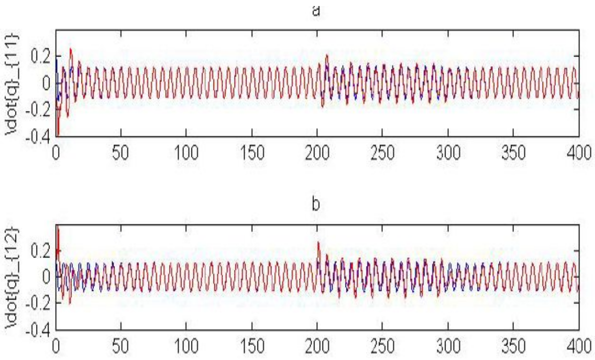
Numerical Data of Two Link Flexible Joint Manipulator

- $J = diag\{1.16, 1.16\}$, $m = diag\{1, 1\}$
- $l_1 = l_2 = 1m$, $K = diag\{100, 100\}, \eta = 0.3, \rho_1 = \rho_2 = 0.001$
- $A = -2I_8$
- A PD controller is used to stabilize the open-loop system
- β is uncertainty in dynamics and unknown
- Sensory measurement is corrupted with a 10% Gaussian noise
- The NN FDI:
  - A Three layer NN
  - 10 neurons in input layer; 10 neurons in hidden layer; 2 neurons in output layer (faults in actuator τ_1 and τ_2
  - The actuator fault in both joints:

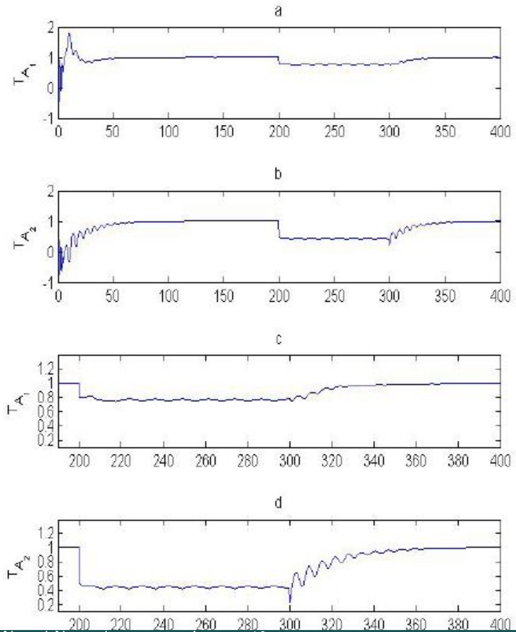
$$T_A = \begin{cases} diag\{0.7\ 0.4\} & t \in [200\ 300] secs \\ diag\{1\ 1\} & otherwise \end{cases}$$

Example Flexible Joint Manipulator

- (a) angular velocities of the first link
- (b) angular velocity of the second link
- Red lines: real signals; Blue lines: the estimated signals by NN



- (a) Estimated fault in the first actuator
- (b) Estimated fault in the second actuator
- (c) Estimated fault in the first actuator; a closer look
- (d) Estimated fault in the second actuator; a closer look



A Sensor and Actuator FDI [1]

- Consider the nonlinear dynamics:

$$\begin{aligned} \dot{x} &= f(x,u) + \eta_x(x,u,t) + T_A(x,u) \\ y(t) &= Cx(t) + \eta_y(x,u,t) + T_s(x,u) \end{aligned} \tag{4}$$

  - $u \in \mathcal{R}^m$: input; $x \in \mathcal{R}^n$: state;$y \in \mathcal{R}^m$: output
  - f: known nonlinear fcn.
  - $\eta_x(x,u,t) \in \mathcal{R}^n \times \mathcal{R}^m \times \mathcal{R} \to \mathcal{R}^n$: Plant unmodeled dynamics and noise
  - $\eta_y(x,u,t) \in \mathcal{R}^n \times \mathcal{R}^m \times \mathcal{R} \to \mathcal{R}^m$: Sensor unmodeled dynamics and disturbances
  - C: an $m \times n$ conts. matrix
  - $T_s \in \mathcal{R}^n \times \mathcal{R}^m \to \mathcal{R}^m$: unknown sensor fault
  - $T_A \in \mathcal{R}^n \times \mathcal{R}^m \to \mathcal{R}^n$: unknown actuator fault

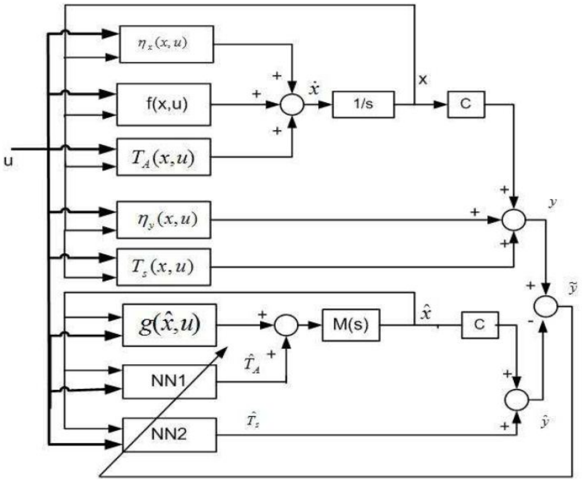
A Sensor and Actuator FDI

- Objective: Diagnose and Isolate Actuator and Sensor Faults: (T_A, T_S)
- By defining a Hurwitz matrix A, (4) can be rewritten

$$\begin{aligned} \dot{x} &= Ax + g(x,u) + \eta_x(x,u,t) + T_A(x,u) \\ y(t) &= Cx(t) + \eta_y(x,u,t) + T_s(x,u) \end{aligned}$$

  - $g(x,u) = f(x,u) - Ax$
- $M(s) = (sI - A)^{-1}$

A Sensor and Actuator FDI



A Sensor and Actuator FDI

- The following assumptions should be made
  - Sensor and actuator fault will not occur simultaneously
  - The nonlinear sys (4) is observable
  - The nominal closed loop sys is asym. stable ($\dot{x} = f(x, k(Cx))$)
  - η_x, η_y are uniformly bounded $\|\eta_x\| \leq \bar{\eta}_x, \|\eta_y\| \leq \bar{\eta}_y$
  - g is Lipshtiz in x: $\|g(x,u) - g(\hat{x},u)\| \leq l_g \|x - \hat{x}\|$
- T_A is estimated by NN_1 and T_S is estimated by NN_2 :
  $\hat{T}_A(\hat{x}, \hat{W}_1, \hat{V}_1, t) = \hat{W}_1 \sigma(\hat{V}_1 \bar{\hat{x}})$
  $\hat{T}_S(\hat{x}, \hat{W}_2, \hat{V}_2, t) = \hat{W}_2 \sigma(\hat{V}_2 \bar{\hat{x}})$
  - Each is a three layer MLP
- $\therefore$ The estimated model

$$
\begin{aligned}
\dot{\hat{x}} &= A\hat{x} + g(\hat{x}, u) + \hat{W}_1 \sigma(\hat{V}_1 \bar{\hat{x}}) \\
\hat{y}(t) &= C\hat{x} + \hat{W}_2 \sigma(\hat{V}_2 \bar{\hat{x}})
\end{aligned}
$$

A Sensor and Actuator FDI

- It has been shown that by updating the weights of NN_1 and NN_2:
  $\dot{\hat{W}}_1 = -\eta_1(\frac{\partial J}{\partial \hat{W}_1}) - \rho_1\|\tilde{y}\|\hat{W}_1$
  $\dot{\hat{V}}_1 = -\eta_2(\frac{\partial J}{\partial \hat{V}_1}) - \rho_2\|\tilde{y}\|\hat{V}_1$
  $\dot{\hat{W}}_2 = -\eta_3(\frac{\partial J}{\partial \hat{W}_2}) - \rho_3\|\tilde{y}\|\hat{W}_2$
  $\dot{\hat{V}}_3 = -\eta_4(\frac{\partial J}{\partial \hat{V}_3}) - \rho_4\|\tilde{y}\|\hat{V}_4$
  - $\tilde{x} = x - \hat{x}$
  - $\tilde{y} = y - \hat{y}$
  - $\tilde{W}_i = W_i - \hat{W}_i, i = 1, 2$
  - $\tilde{V}_i = V_i - \hat{V}_i, i = 1, 2$
  - $J = \frac{1}{2}(\tilde{y}^T\tilde{y})$: objective fcn to be minimized
- then $\tilde{x}, \tilde{y}, \tilde{W}_i, \tilde{V}_i, i = 1, 2$ are ultimately bounded

Example of Satellite Attitude Control

- Dynamical equation of satellite attitude model with magnetorquer actuator $H\dot{\omega} = -S(\omega)H\omega + BM$
  - $\omega = [\omega_1 \ \omega_2 \ \omega_3]^T$: angular velocity of satellite
  - $B(t) = \begin{bmatrix} 0 & B_z(t) & -B_y(t) \\ -B_z(t) & 0 & B_x(t) \\ B_y(t) & -B_x(t) & 0 \end{bmatrix}$: magnetic field matrix
  - H: Symmetric P.D. inertia matrix
  - $S(\omega) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}$: cross product matrix
  - M: magnetic dipole of the coil

Example of Satellite Attitude Control

- To define it in standard state space representation:
  $u = M; f(x,u) = H^{-1}(B(t)u - S(\omega)H\omega); x = \omega$
- the unmodeled dynamics: $\eta_x = H^{-1}(B(t) - B)u; B = \frac{1}{T}\int_0^T B(t)dt$
- Magnetorquer fault (actuator fault) is considered as an additive term τ_a:

$$\dot{x} = f(x,u) + \eta_x + T_A; T_A = H^{-1}\tau_a$$

- Magnetometer fault (sensor fault): $B_m = B + EB + \Delta B + \nu$
  - B_m: measured magnetic field
  - B: true magnetic field
  - ΔB: magnetic bias
  - ν measurement noise
  - $E = \begin{bmatrix} 0 & E_z & -E_y \\ -E_z & 0 & E_x \\ E_y & -E_x & 0 \end{bmatrix}$: magnetormeter misalignment matrix
  - any fault or misalignment can be considered as uncertainty vector η_y or additive fault T_S

Numerical Data of ϕrsted Satellite

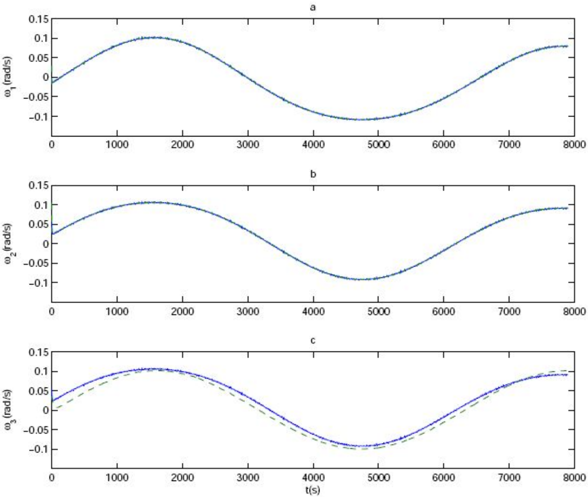
- $H = \begin{bmatrix} 181.78 & 0 & 0 \\ 0 & 181.25 & 0 \\ 0 & 0 & 1.28 \end{bmatrix} kg.m^2$
- All sensory measurements are assumed to be corrupted with 10% Gaussian noise
- The actuator fault is estimated by NN_1 and sensor fault by NN_2. They have:
    - 3 layers
    - 6 neurons in input layer; 5 neurons in hidden layer with sigmoidal activation fcn; 3 neurons in output layer with linear activation fcn.
    - $\eta_1 = \eta_2 = 20; \eta_3 = \eta_4 = 0.1; \rho_i = 1e-6, i = 1, 2, 3, 4$
    - $A = -2I_3$
    - $\hat{T}_A = [\hat{T}_{A1} \;\; \hat{T}_{A2} \;\; \hat{T}_{A3}]^T$: output of NN_1
    - $\hat{T}_S = [\hat{T}_{S1} \;\; \hat{T}_{S2}]^T$: output of NN_2

Example of Satellite Attitude Control

- Note that: the output of the networks do not approach to zero due to presence of η_y and η_x
- Therefore a threshold should be considered for fault detection
- The threshold can be chosen by by adaptive , fuzzy or ...approached
- In this simulation it has been chosen by trial and error
- A fault is detected if:
  $|\hat{T}_{A_i}| > th_{ai}, i = 1, ..., 3$ (actuator fault)
  $|\hat{T}_{S_i}| > th_{si}, i = 1, ..., 2$ (sensor fault)
  $\forall t \in [t_f, t_f + t_r]$
  - $th_{ai} = [-0.02\ 0.02]$ threshold of ith actuator
  - $th_{si} = [-0.02\ 0.02]$ threshold of ith sensor
  - $t_r = 30sec.$

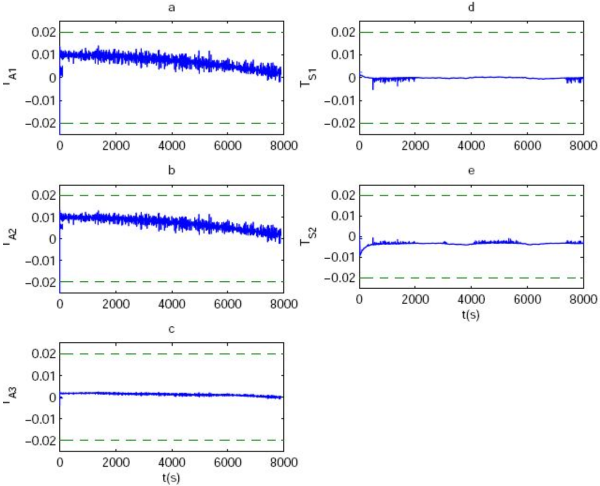
Fault Free angular velocities, trajectory: $0.1 sin(0.001t)$

- Solid lines: actual velocities
- Dashed lines: Estimated velocities



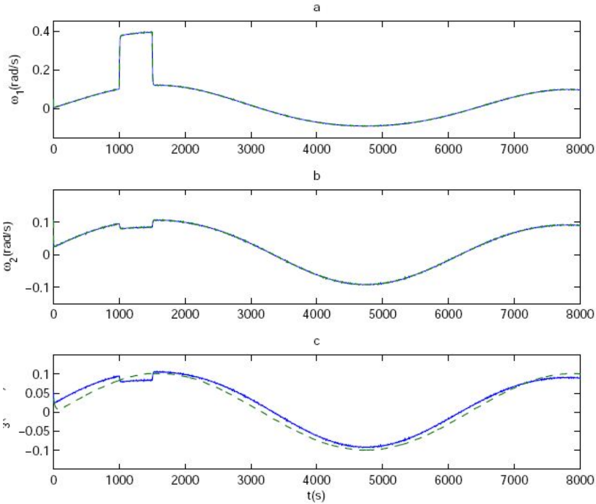
NN_1 and NN_2 responses to fault free operation

- Solid lines: estimated faults; Dashed lines: thresholds
- (a)-(c): Estimated actuator faults
- (d)-(e): Estimate sensor faults



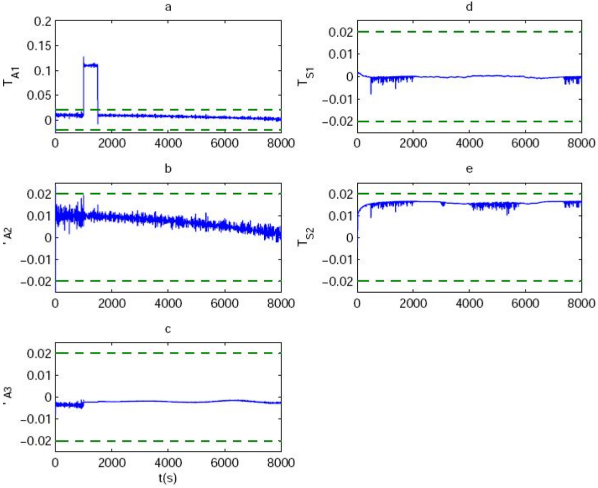
Angular velocities with bias fault: $T_A = [0.1\ 0\ 0]^T$

- Solid lines: actual velocities
- Dashed lines: Estimated velocities



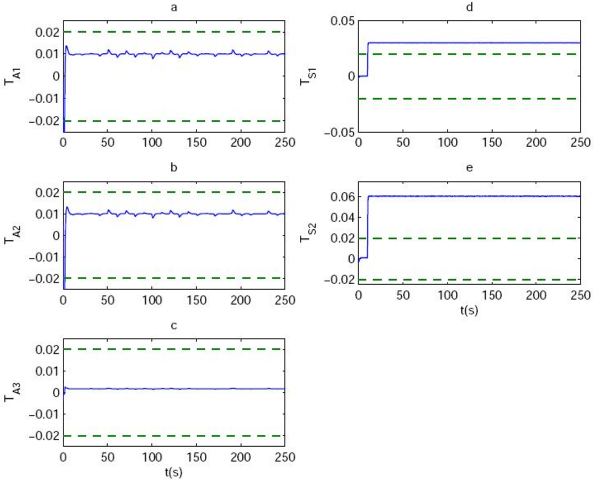
NN_1 and NN_2 responses to bias fault: $T_A = [0.1\ 0\ 0]^T$

- Solid lines: estimated faults; Dashed lines: thresholds
- (a)-(c): Estimated actuator faults
- (d)-(e): Estimate sensor faults



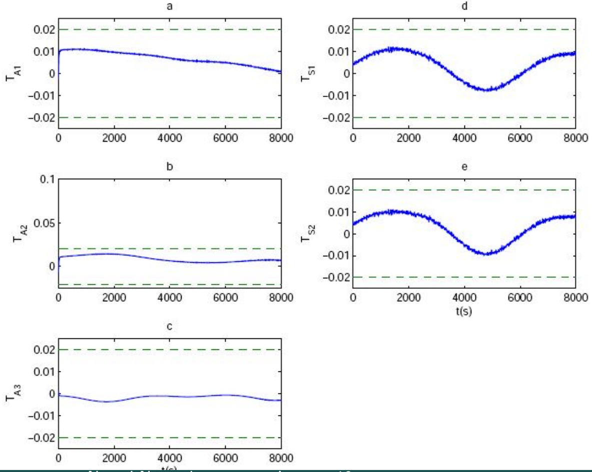
NN_1 and NN_2 responses; traj: $0.1 rad/s$, fault: $T_S = [0.03\ 0.05]^T$

- Solid lines: estimated faults; Dashed lines: thresholds
- (a)-(c): Estimated actuator faults
- (d)-(e): Estimate sensor faults



NN_1 and NN_2 responses; traj: $0.1 sin(0.001t)$, fault free, with sensor and state uncertainties

- Solid lines: estimated faults; Dashed lines: thresholds
- (a)-(c): Estimated actuator faults
- (d)-(e): Estimate sensor faults



References

H.A.Talebi, F. Abdollahi, R.V.Patel, and K.Khorasani, *Neural Network-Based State Estimation of Nonlinear Systems: Application to Fault Detection and Isolation*. Springer, 2009.