MODEL-BASED AND DATA-DRIVEN TOOLS AND THEIR APPLICATION TO FAULT DIAGNOSIS IN ENGINEERING SYSTEMS AND INFORMATION RETRIEVAL

Setu Madhavi Namburu

A Thesis
Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science
At the
University of Connecticut
March, 2006

Publications

- Namburu, S. M., M. Azam, J. Luo, K. Choi and K. Pattipati, "Data-driven Modeling, Fault Diagnosis and Sensor Selection in HVAC Chillers", Submitted to *IEEE Transactions on Automation Science and Engineering*, December 2005.
- Namburu, S. M., J. Luo, M. Azam, K. Choi, and K. R. Pattipati, "Fault Detection, Diagnosis and Data-driven Modeling in HVAC Chillers," in *Proc. SPIE Conf.*, Orlando, FL, March 2005.
- Namburu, S. M., H. Tu, J. Luo, and K. R. Pattipati, " Supervised Learning Algorithms for Text Categorization", *IEEE Aerospace Conference*, Bigsky, MT, March 2005.
- Choi, K., S. Namburu, M. Azam, J. Luo, K. R. Pattipati, and Patterson-Hine, A., "Fault Diagnosis in HVAC Chillers: Adaptability of a Data-driven Fault Detection and Isolation Approach," *IEEE Instrumentation & Measurement Magazine*, #1094-6969, August 2005. (updated version of the above paper)
- Luo, J., S. M. Namburu, K. R. Pattipati, L. Qiao, and S. Chigusa, "Integrated Model-based and Data-driven Diagnosis of Automotive Anti-Lock Braking Systems", Submitted to *IEEE Transactions on SMC – Part A*, April 2005.
- Luo J., M. Namburu, K. Pattipati, L. Qiao, and S. Chigusa, "Integrated Model-base and Data-driven Diagnostic Strategies Applied to Anti-Lock Braking System", *IEEE Aerospace Conference*, Big Sky, MT, March 2005.
- Choi, K., S. Namburu, M. Azam, J. Luo, K. R. Pattipati, and Patterson-Hine, A., "Fault Diagnosis in HVAC Chillers using Data-driven Techniques," *IEEE AutoTestCon*, San Antonio, TX, September 2004.

  ⇒ Best Technical Paper Award at 2004 AUTOTESTCON

Presentation Outline

- Introduction
  - Motivation
  - Fault Detection and Diagnosis (FDD)
  - Information Retrieval
- Model-based and Data-driven Tools
  - Theoretical Background
  - Software Tools
- Application Examples
  - For FDD in Engineering Systems
    - ➔ Simple Gravity Flow Tank System
    - ➔ Automotive Suspension System
    - ➔ Automotive Engine
    - ➔ HVAC Chiller
  - For Information Retrieval
    - ➔ Text Categorization
- Conclusion and Future Research

Introduction

- Motivation
  - The increasing complexity of modern engineering systems
  - The growing need for more intelligent automated system health monitoring procedures
  - Goal: To develop generic interactive user interfaces for model-based and data-driven techniques and show their application to many real-world examples
- Fault Detection and Diagnosis
  - Fault Detection: The indication that something is going wrong in the monitored system
  - Fault Diagnosis: The determination of exact location of the fault (the component which is faulty) and it's magnitude of severity
- Information Retrieval
  - Handling of massive volumes of online documents, intelligence reports, web pages, E-mails, news, etc. in modern information society

Model-based Tools

- Requires a mathematical/physics based model of the system, significant amount of expertise and system specific knowledge for application
- Accurate diagnosis is possible as it incorporates physical understanding of the system to monitoring
- Difficult to apply for FDD in complex engineering systems
- Techniques considered
  - Discrete linear observer
  - Parameter estimation via Equation Error Method
  - Parameter estimation via Extended Kalman Filter

Discrete Linear Observer

Consider a discrete-time, time-invariant, linear dynamic model

$$x(t+1) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t)$$

An observer to reconstruct the system variables based on the measured inputs and outputs $u(t)$ and $y(t)$ is given by

$$\hat{x}(t+1) = A\hat{x}(t) + Bu(t) + He(t)$$

$$e(t) = y(t) - C\hat{x}(t)$$

The state estimation error $e_x(t)$ is given by

$$e_x(t) = x(t) - \hat{x}(t)$$

$$e_x(t+1) = (A - HC)e_x(t)$$

The state error is vanished to zero by a proper design of the observer feedback H

Parameter Estimation via Equation-Error Method

- A Five-step parameter estimation method
  - Obtain the theoretical model of the system relating the measurable input and output variables
    $$\underline{y}(t) = f\{\underline{u}(t), \underline{\theta}_0\}$$
  - Determine the relationship between the model parameters $\underline{\theta}$ and the physical system coefficients $\underline{p}$
    $$\underline{\theta} = g(\underline{p})$$
  - Identify the model parameter vector $\underline{\theta}$ from the measured variables
    $$\underline{Y}^N = \{\underline{y}(k) : 0 \le k \le N\} \text{ and } \underline{U}^N = \{\underline{u}(k) : 0 \le k \le N\}$$
  - Calculate the system coefficients (parameters): $\underline{p} = g^{-1}(\underline{\theta})$ and deviations from nominal coefficients
    $$\underline{p}_0 = g^{-1}(\underline{\theta}_0) \quad \text{Viz.,} \quad \Delta\underline{p} = \underline{p} - \underline{p}_0$$
  - Diagnose faults by using the relationship between system faults and deviations in the coefficients $\Delta\underline{p}$

Parameter Estimation via Equation-Error Method (cont'd)

- (Example) Consider SISO discrete-time model

  $$a_n y^{(n)}(t) + \ldots + a_1 \dot{y}(t) + y(t) = b_0 u(t) + \ldots + b_m u^{(m)}(t)$$

  Written as

  $$y(k) = \underline{\psi}^T(k)\underline{\theta} + e(k) \quad \text{for} \quad 0 \le k \le N$$

  Where $\underline{\theta}^T = [a_1, a_2, \ldots, a_n \mid b_0, b_1, \ldots, b_m]$ is the parameter vector,
  $\underline{\psi}^T(k) = [-\dot{y}(k), -\ddot{y}(k), \ldots, -y^{(n)}(k), u(k), \dot{u}(k), \ldots, u^{(m)}(k)]$ is the data vectorand *e(t)* represents equation-error

- The estimate of the parameter vector is computed via least-squares or recursive least-squares algorithms

Parameter Estimation via Equation-Error Method (cont'd)

- The equation-error is given by

$$e(k) = y(k) - \underline{\psi}^T(k)\underline{\theta}$$

- The minimization of the sum of least squares gives

$$V = \sum_{k=1}^{N} e^2(k) = \underline{e}^T \underline{e}$$

  – Is called least-squares criterion for linear regression. It is a quadratic function of θ

- Leads to least-squares (LS) estimate

$$\underline{\hat{\theta}}(N) = \arg\min V = \left[\underline{\psi}^T \underline{\psi}\right]^T \underline{\psi}^T \underline{y}$$

- The estimate of the parameter vector can be computed via Recursive Least Squares

$$\underline{\hat{\theta}}(k+1) = \underline{\hat{\theta}}(k) + \underline{\alpha}(k)[y(k+1) - \underline{\psi}^T(k+1)\underline{\hat{\theta}}(k)]$$

$$\underline{\alpha}(k) = \frac{1}{\underline{\psi}^T(k+1)P(k)\underline{\psi}(k+1)+1} P(k)\underline{\psi}(k+1)$$

$$P(k+1) = [I - \underline{\alpha}(k)\underline{\psi}^T(k+1)]P(k)$$

where P is the covariance matrix and α is forgetting factor

Simultaneous State and Parameter Estimation via EKF

Augmented Parameters as States:

Initialize with x_0, P_0

State Estimation:

$$\dot{\hat{x}} = f(\hat{x}, u, \hat{\theta}(k/k) \Rightarrow \hat{x}(k+1/k) = \hat{x}(t_{k+1}/t_k)$$

Covariance propagation:

$$\dot{P}_{xx} = \frac{\partial f}{\partial x^T} P_{xx} + P_{xx} \frac{\partial f^T}{\partial x} + Q \Rightarrow P_{xx}(k+1/k) = P(t_{k+1}/t_k)$$

Predicted Output and Error:

$$\hat{y}(k+1) = H\hat{x}(k+1/k)$$

$$e(k+1) = y(k+1) - \hat{y}(k+1)$$

Simultaneous State and Parameter Estimation via EKF (cont'd)

Prediction Error Covariance:

$$A(k+1) = R + H.P_{xx}(k+1/k)H^T$$

Filter Gain :

$$K(k+1) = P_{xx}(k+1/k).H^T.A^{-1}(k+1)$$

Gradient of Prediction:

$$\frac{d\hat{y}(k+1)}{d\theta} = H.\frac{d\hat{x}(k+1/k)}{d\theta}$$

Covariance Update:

$$P_{xx}(k+1/k+1) = [I_n - K(k+1).H].P_{xx}(k+1/k)$$

State Update:

$$\hat{x}(k+1/k+1) = \hat{x}(k+1/k) + K(k+1).e(k+1)$$

Data-driven Tools

- Suitable for the cases where no mathematical model of the system is available
- Requires large amount of monitored data
- Techniques considered
  - For Fault Detection
    - ➔ Generalized Likelihood Ratio Test (GLRT)
  - For Fault Diagnosis
    - ➔ Support Vector Machines (SVM)
    - ➔ Principal Component Analysis (PCA)
    - ➔ Partial Least Squares (PLS)

Generalized Likelihood Ratio Test (GLRT)

- The Generalized Likelihood Ratio (GLR) test is used to detect changes in residuals. In our case, the mean ω_0 and variance σ^2 before the changes are known, and the mean ω_1 after the change is unknown
- The log-likelihood ratio for observations from time j up to time k is

$$R_j^k(\omega_1) = \sum_{i=j}^{k} \log \frac{P_{\omega_1}(r_i)}{P_{\omega_0}(r_i)}$$

$P_\omega(r_i)$: the prob. density function of the residual, r, at time index i about the mean value ω

- To estimate change time and mean ω_1, MLE formulation is

$$g_k = \max_{1 \le j \le k} \sup_{\omega_1} R_j^k(\omega_1)$$

- Since the residual was found to be almost Gaussian under normal conditions for chiller application discussed later, the log-likelihood ratio above simplifies to

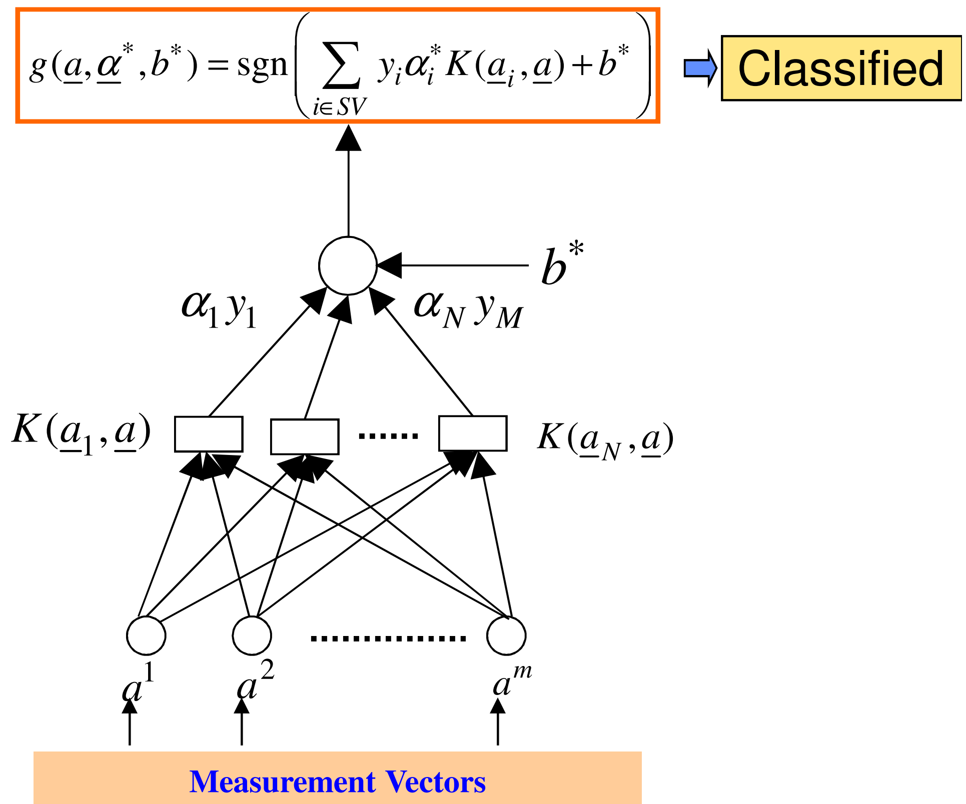
$$R_j^k = \frac{\omega_1 - \omega_0}{\sigma^2} \sum_{i=j}^{k} \left(r_i - \frac{\omega_1 + \omega_0}{2} \right)$$

- Denoting $\eta = \omega_1 - \omega_0$, the MLE of η is $\hat{\eta}_j = \frac{1}{k-j+1} \sum_{i=j}^{k} (r_i - \omega_0)$,
- and the decision function becomes

$$g_k = \frac{1}{2\sigma^2} \max_{1 \le j \le k} \frac{1}{k-j+1} \left[\sum_{i=j}^{k} (r_i - \omega_0) \right]^2 .$$

Support Vector Machines

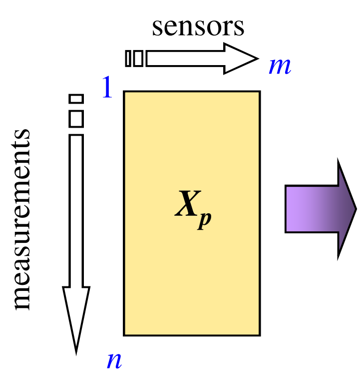
- Based on the *Structural Risk Minimization* principle from computational learning theory
- Find a *hypothesis H* which guarantees the lowest *true error*
- Represent patterns in a higher dimension
- Universal learners: linear threshold function, polynomial classifiers, *radial basis function* (RBF) networks, and three-layer sigmoid neural nets
- Independent of the dimensionality of the feature space



Multi-class classification based on voting of pair-wise classifiers

Principal Component Analysis

- Reduces the dimensionality of a data set
- Produces a representation in a way that keeps the correlation structures among the process variables, and is optimal in terms of capturing the variation in data



SVD

$$\frac{1}{\sqrt{n-1}} X_p = U \Sigma V^T \rightarrow \mathbf{P}$$

$\Sigma \in \Re^{n\times n}$: diagonal matrix (nonnegarive real **singular values**)

$U \in \Re^{n\times n}$ and $V \in \Re^{m\times m}$: unitary matrices

$\underbrace{\mathbf{P}}_{L}$

***L*: # of principal components**

Hotelling's statistic *(T^2)* for the lower-dimensional space:

$$T^2 = X_{test}^T P \Sigma_L^{-2} P^T X_{test}$$

Σ_L : the first L rows and columns of Σ

T_α^2 : T^2 threshold using the probability distribution

$$T_\alpha^2 = \frac{L(K-1)(K+1)}{K(K-L)} F_{L,K-L,\alpha}$$

F distribution at significant level α with L and $(n-L)$ degrees of freedom

$T^2 \le T_\alpha^2$?

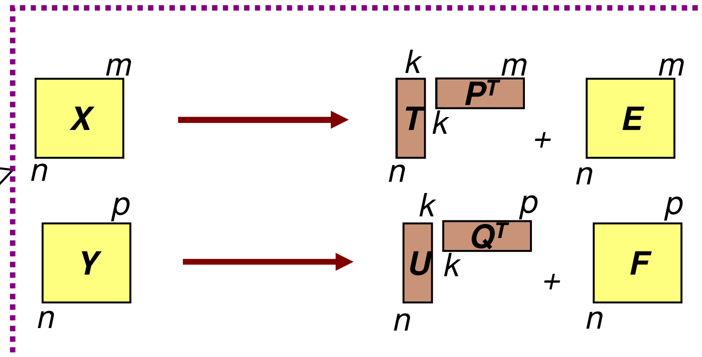
YES

Classified

Partial Least Squares

- A supervised dimensionality reduction technique
  - maximizes the covariance between the independent training matrix *X* and the
  - dependent matrix *Y* for each component of the reduced space
- Builds a regression model between *X* and *Y*

X: Training Data
Y: Known set of fault classes
n: Number of patterns (steady-state measurements)
m: Number of features (sensors)
p: Number of fault classes



T, U: Score Matrices (Latent Vectors)
P,Q: Loading Matrices
E,F: Residuals
k: Model reduction order

The PLS Regression Model:

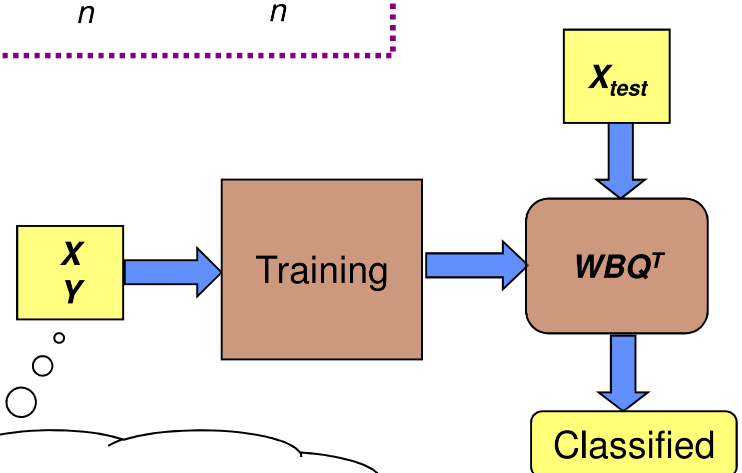
$T = XW$
Where $W \propto P$

Regression coefficient matrix

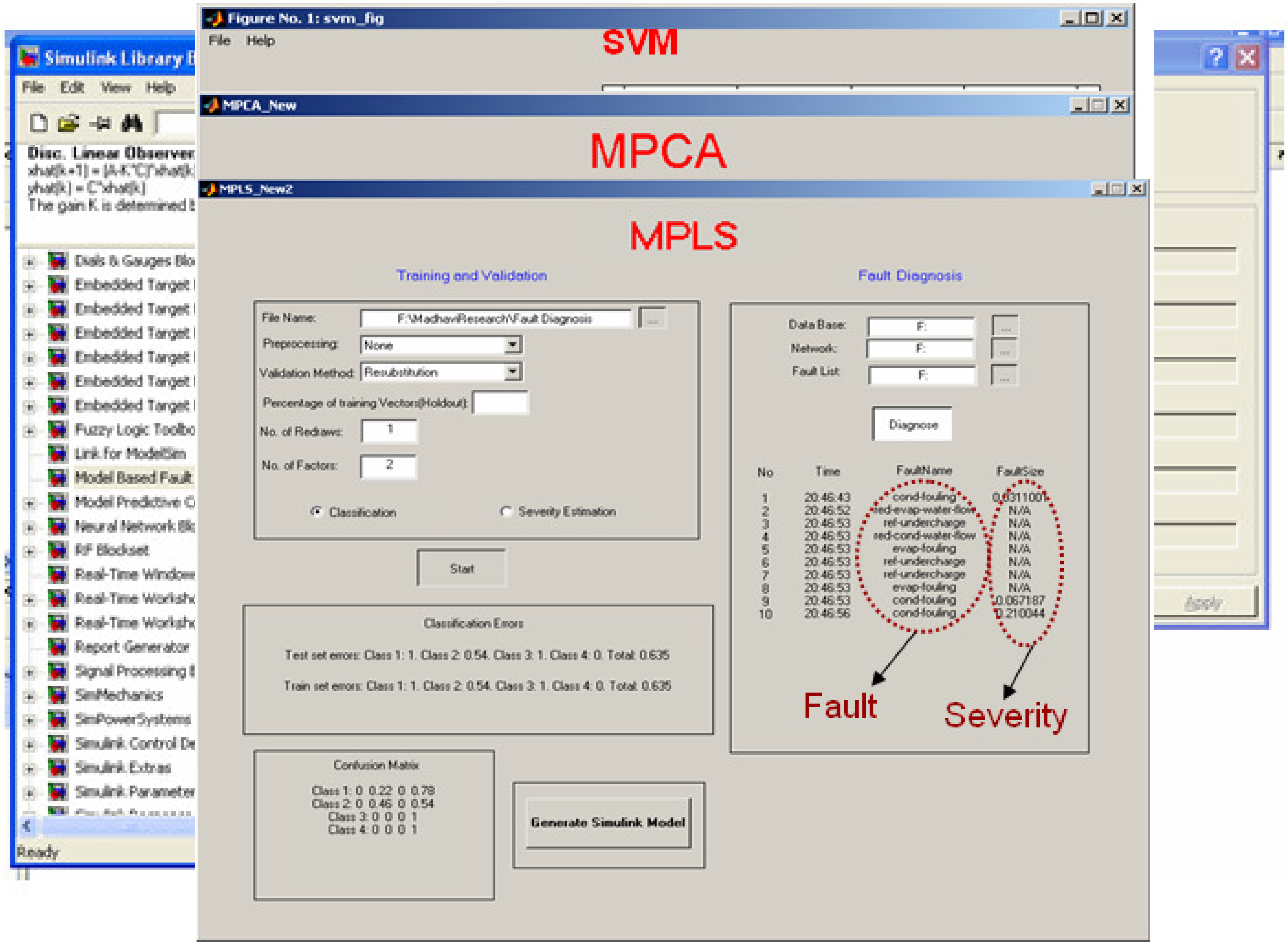
$$Y = TBQ^T + F$$

Optimal *W,B,Q*

For estimation, Y contains the fault severity values



Generic Software Tools



Model-based Fault Diagnosis Toolset

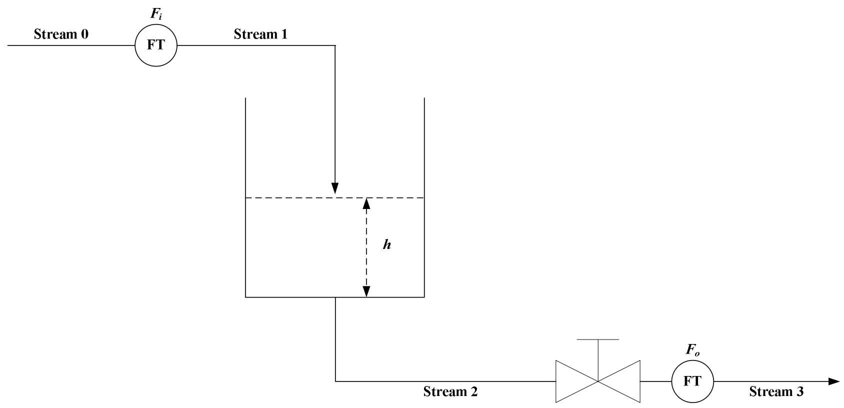
User-friendly Interfaces for SVM, PCA and PLS

Application Examples

- For FDD in Engineering Systems
  - ➔ Simple Gravity Flow Tank System (Model-based)
  - ➔ Automotive Suspension System (Model-based)
  - ➔ Automotive Engine (Data-driven)
  - ➔ HVAC Chiller (Data-driven)
- For Information Retrieval
  - ➔ Text Categorization (Data-driven)

Simple Gravity Flow Tank System

☆ System Description



Material-balance Equation

$$A_c \frac{dh}{dt} = F_i - ch$$

where A_c : The cross-sectional area of the tank

h : liquid level

c : constant which depends on the valve

F_i : measured inlet flow rate

- State-space Equations

$$\frac{d\,\mathrm{x}}{dt} = A\,\mathrm{x}(t) + B\,\mathrm{u}(t)$$

$$\mathrm{y}(t) = C\,\mathrm{x}(t)$$

where u = *Fi*; y = *Fo*; x = *h*; *A* = -*c/Ac*;
B = *1/Ac*, and *C* = *c*

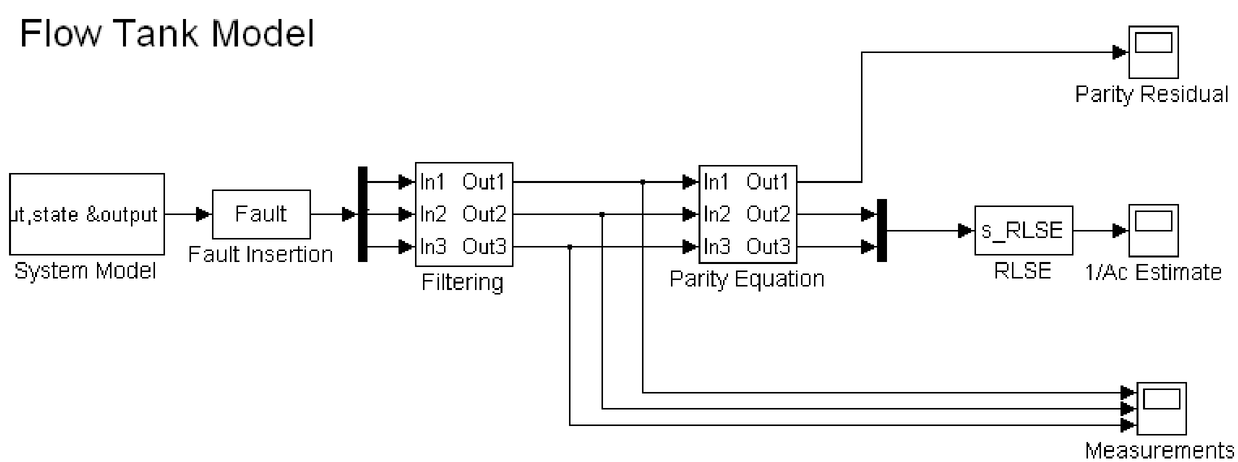
- Measurement Variables & Faults
  - 3 Monitored Variables
    - Inlet flow rate (*Fi*)
    - Outlet flow rate (*Fo*)
    - Liquid level (*h*)
  - One Parametric Fault (*B*)

Simple Gravity Flow Tank System (cont'd)

- FDD Approach
  - Parity equation from state-space model

$$r(t) = \dot{x} + \frac{c}{A_c}x(t) - \frac{1}{A_c}u(t) = \dot{x} - Ax(t) - Bu(t)$$

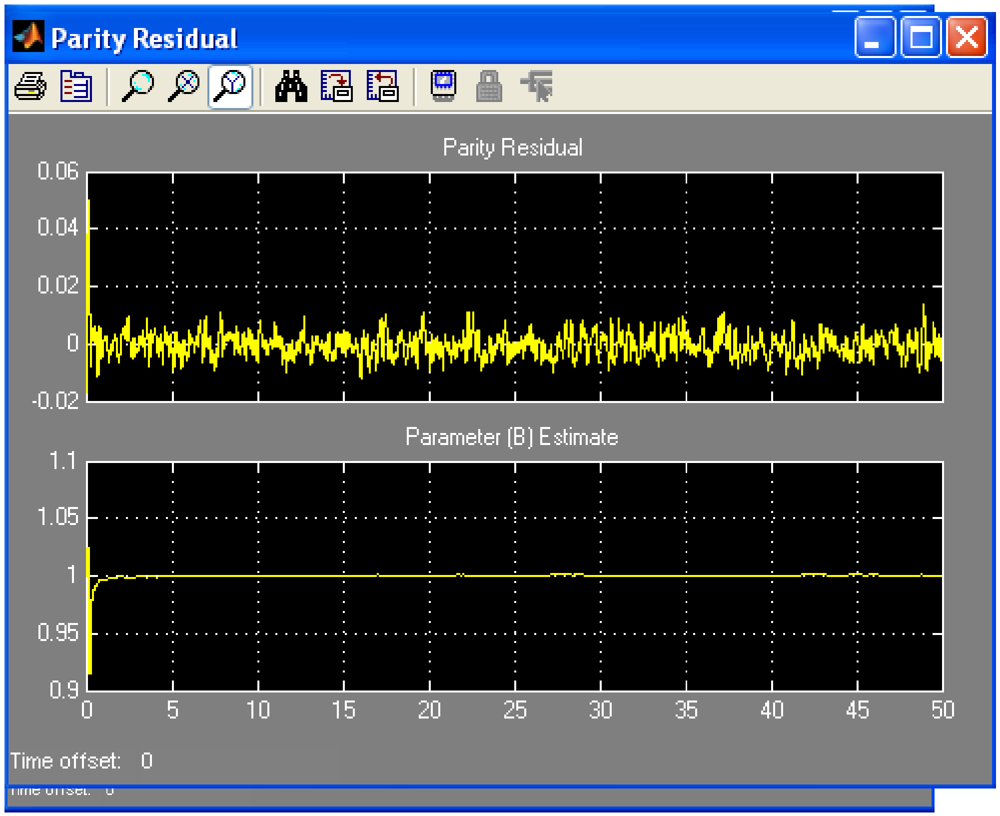
  - Parameter estimation via RLSE



Simulink Model of the Flow Tank System with RLSE Module

Simple Gravity Flow Tank System (cont'd)

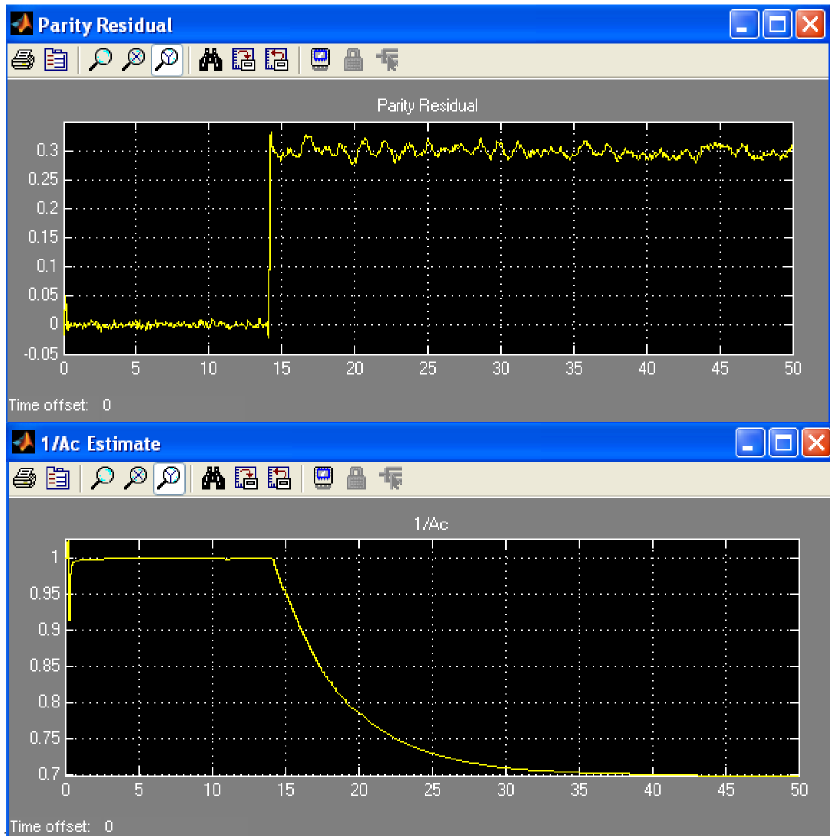
- Experimental Results (Nominal Condition)



Parity Residual and Parameter *B* Estimate at Nominal Conditions
The Measured Input (inlet flow rate), The Measured State (liquid height) and
The Measured Output (outlet flow rate) of the Gravity Tank during Normal Operating Conditions

Simple Gravity Flow Tank System (cont'd)

- Experimental Results (Faulty Condition)



The Measured Input ... ght) and The Measured Output (outlet flow rate) of the Gravity Tank during when a Leak (50% drop) in Stream 1 occurs at t = 14sec

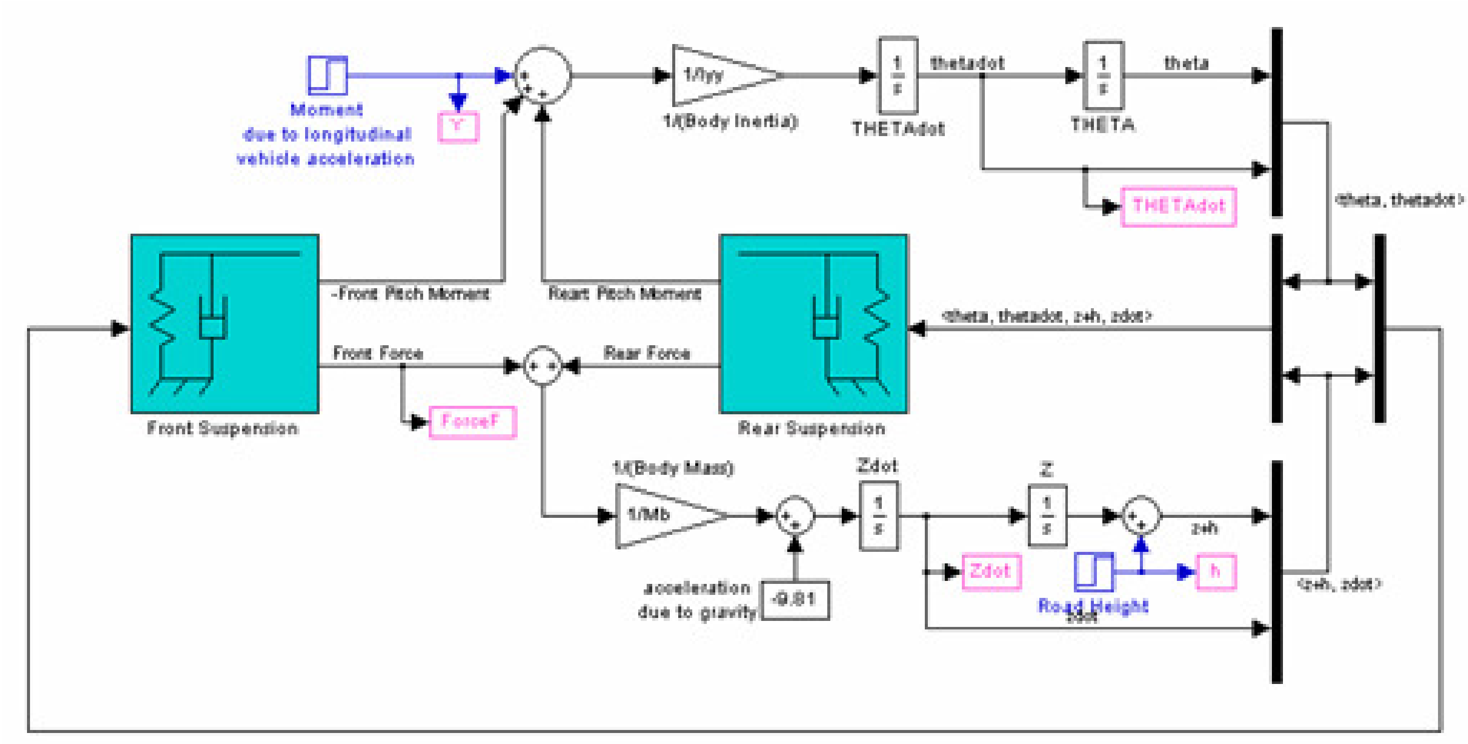
Parity Residual and Parameter *B* Estimate when a Leak in Stream 1 Occurs at $t = 14sec$

Application Examples

- For FDD in Engineering Systems
  - ➔ Simple Gravity Flow Tank System (Model-based)
  - ➔ Automotive Suspension System (Model-based)
  - ➔ Automotive Engine (Data-driven)
  - ➔ HVAC Chiller (Data-driven)
- For Information Retrieval
  - ➔ Text Categorization (Data-driven)

Automotive Suspension System

- System Description



The Simulink Two Degree of Freedom Suspension Model

Automotive Suspension System (cont'd)

- State-space Equations

$$\dot{X}=\begin{bmatrix}\dot{\theta}\\ \ddot{\theta}\\ \dot{z}\\ \ddot{z}\end{bmatrix}=\begin{bmatrix}0 & 1 & 0 & 0\\ -\frac{2}{I_{yy}}\left(L_f^2K_f+L_r^2K_r\right) & -\frac{2}{I_{yy}}\left(L_f^2C_f+L_r^2C_r\right) & \frac{2}{I_{yy}}\left(L_fK_f-L_rK_r\right) & \frac{2}{I_{yy}}\left(L_fC_f-L_rC_r\right)\\ 0 & 0 & 1 & 0\\ \frac{2}{M_b}\left(K_fL_f-K_rL_r\right) & \frac{2}{M_b}\left(C_fL_f-C_rL_r\right) & \frac{2}{M_b}\left(-K_f-K_r\right) & \frac{2}{M_b}\left(-C_f-C_r\right)\end{bmatrix}\begin{bmatrix}\theta\\ \dot{\theta}\\ z\\ \dot{z}\end{bmatrix}+\begin{bmatrix}0\\ 0\\ 0\\ -g\end{bmatrix}+\begin{bmatrix}0\\ 1/I_{yy}\\ 0\\ 0\end{bmatrix}M_y$$

$$Y=\begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\theta\\ \dot{\theta}\\ z\\ \dot{z}\end{bmatrix}+\begin{bmatrix}v1\\ v2\end{bmatrix}$$

Where $\theta,\dot{\theta}$ = pitch (rotational) angle and rate of change

$z,\dot{z}$ = bounce (vertical) distance and velocity

Kf, Cf = front suspension spring rate and damping rate at each wheel

Lf = horizontal distance from body center of gravity to front suspension

Kr, Cr = rear suspension spring rate and damping rate at each wheel

Lr = horizontal distance from body center of gravity to rear suspension

Mb = body mass

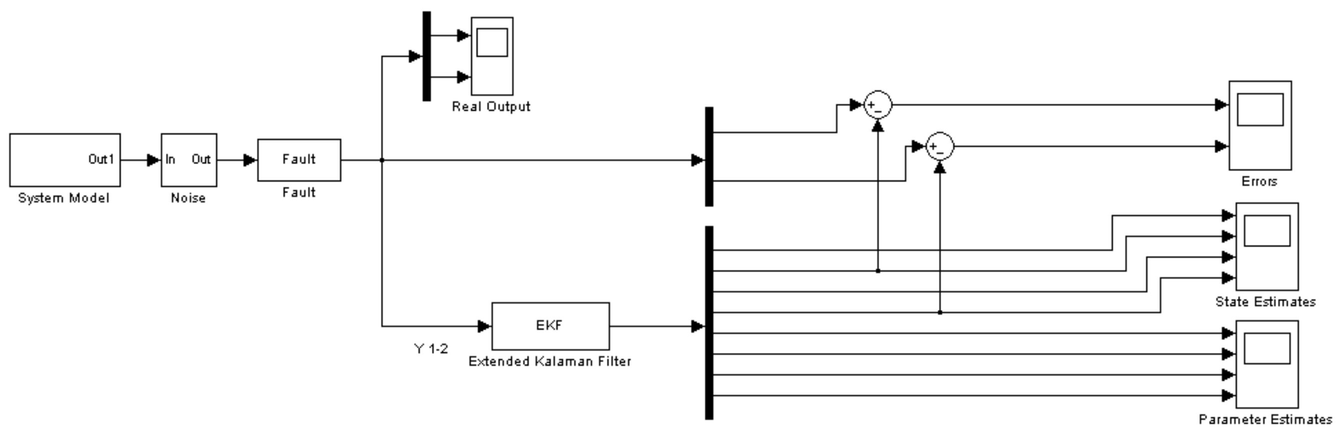
g = gravitational acceleration

Iyy = body moment of inertia about center of gravity

My = moment introduced by vehicle acceleration

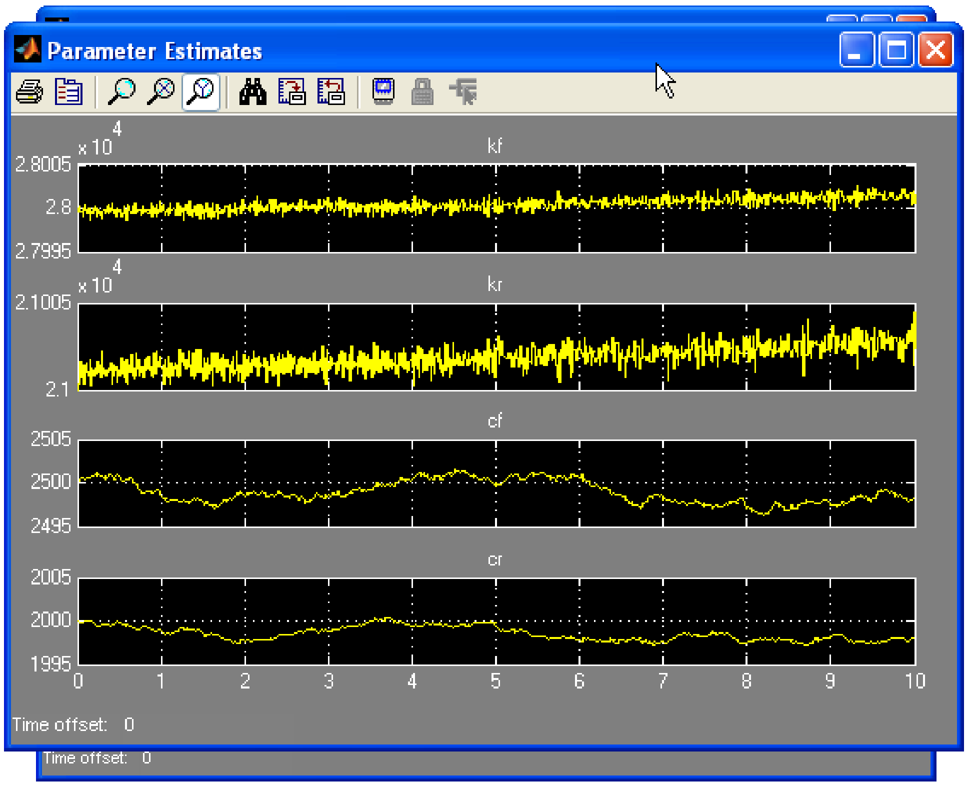
Automotive Suspension System (cont'd)

- Measure Variables and Fault Universe
  - Two Monitored Variables
    - ➔ The rate of change of pitch angle
    - ➔ Bounce velocity
  - Four Parametric Faults
    - ➔ Front suspension rate (*Kf*)
    - ➔ Front damping Rate (*Cf*)
    - ➔ Rear suspension rate (*Kr*)
    - ➔ Rear damping rate (*Cr*)
- FDD Approach
  - Parameter Estimation via EKF



Automotive Suspension System (cont'd)

- Experimental Results (Nominal Condition)

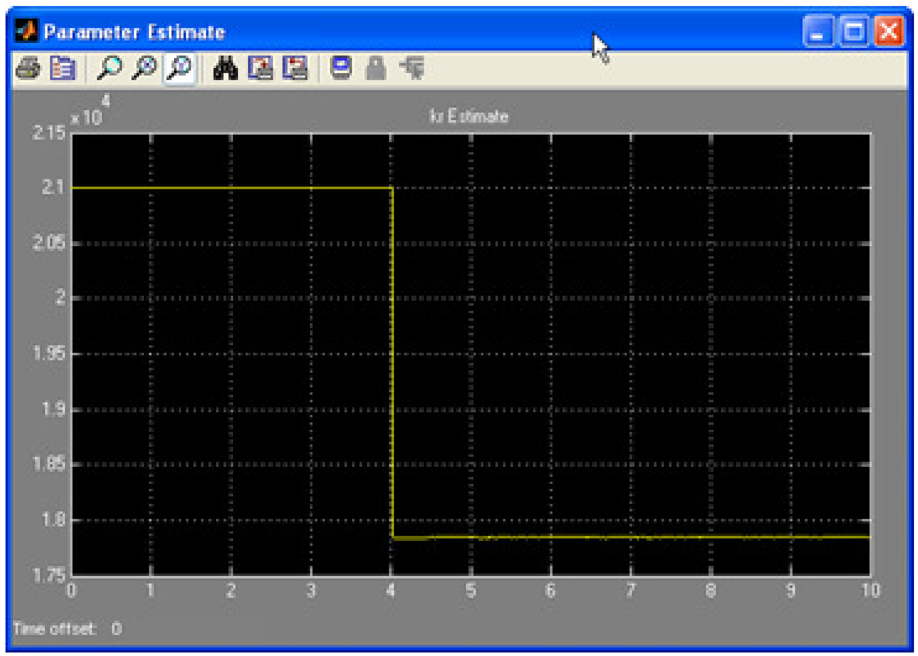


Nominal Parameter Estimates via EKF

State Estimates of the Suspension System via EKF at Nominal Conditions

Automotive Suspension System (cont'd)

- Experimental Results (Faulty Condition)



Kf Estimate via EKF when 10% Fault (*Kf* = 25200) is Inserted at *t* = 2sec

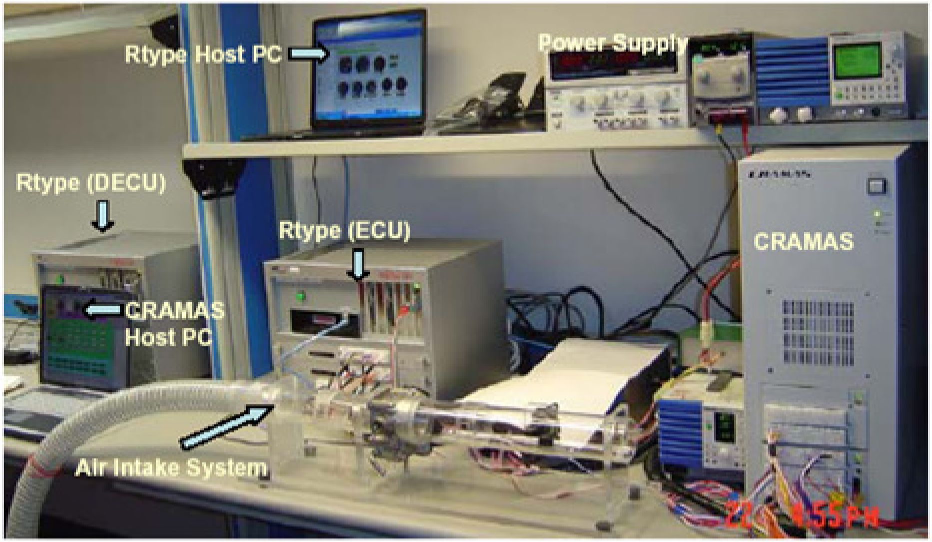
Kr Estimate via EKF when 15% Fault (*Kr* = 17850) is Inserted at *t* = 4sec

Application Examples

- For FDD in Engineering Systems
  - → Simple Gravity Flow Tank System (Model-based)
  - → Automotive Suspension System (Model-based)
  - → Automotive Engine (Data-driven)
  - → HVAC Chiller (Data-driven)
- For Information Retrieval
  - → Text Categorization (Data-driven)

Automotive Engine

- System Description



CRAMAS Set-up for Engine Simulation

Automotive Engine (cont'd)

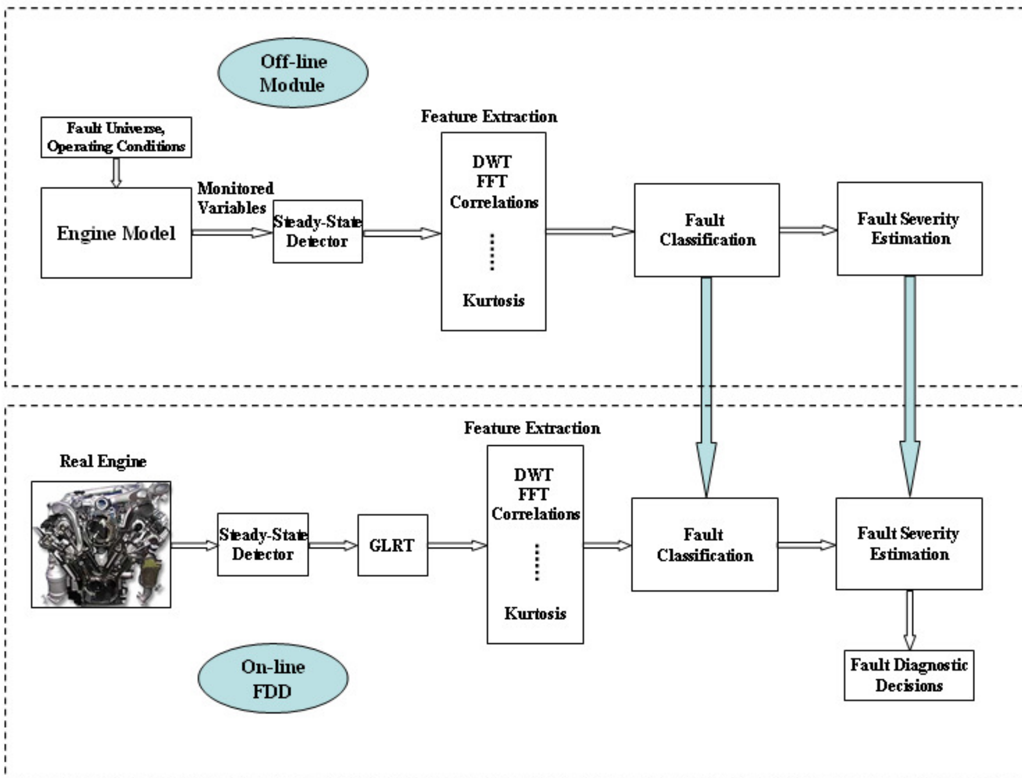
- Engine Fault Universe and Monitored Variables

Fault List		Monitored Variables	
Name	**Notation**	**Name (units)**	**Notation**
Air Flow Sensor Fault	AFS	Air Pressure (KPa)	P_m
Leakage in Air Intake System	AIS_leak	Amount of Fuel Injected (mg)	FC
Blockage of Air Filter	AF_blockage	Air to Fuel Ratio	A/F
Throttle Angle Sensor Fault	TAS	Engine Speed (rpm)	NE
Less Fuel Injection	LFI	Vehicle Speed (rpm)	SPD
Added Engine Friction	AEF	Throttle Angle (deg)	TA
Air/Fuel Sensor Fault	AFuel_S	Air Flow Meter (Volts)	VG2
Engine Speed Sensor Fault	ESS		

- Operating Conditions: PA: 15,18 and 20 degrees
- Fault Severity Levels: 4%, 5%, 6%, 8%, 10%, and 12%
- Challenge: Mode invariant FDD Solution

Automotive Engine (cont'd)

- FDD Approach
  - Fault Detection via GLRT
  - Fault Isolation via Trend Testing on Wavelets and PLS
  - Fault Estimation via PLS

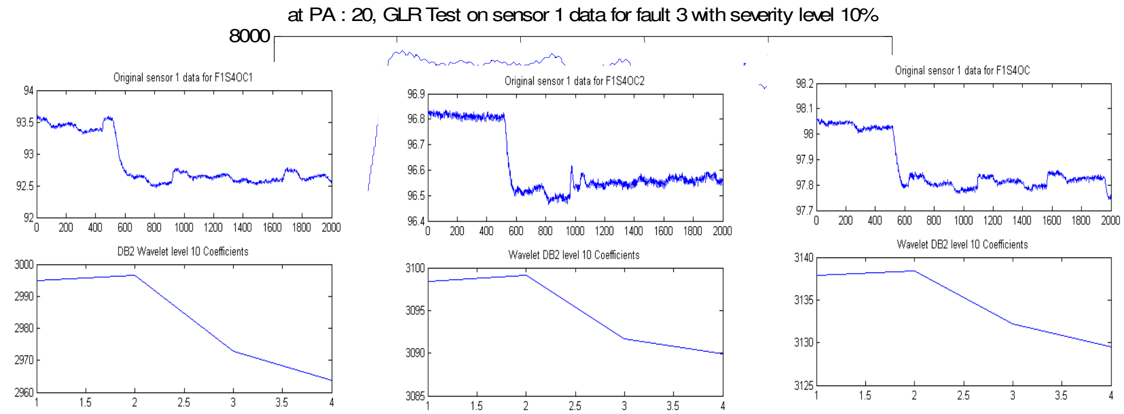


Generic Fault Detection and Diagnosis Scheme for Automotive Engine

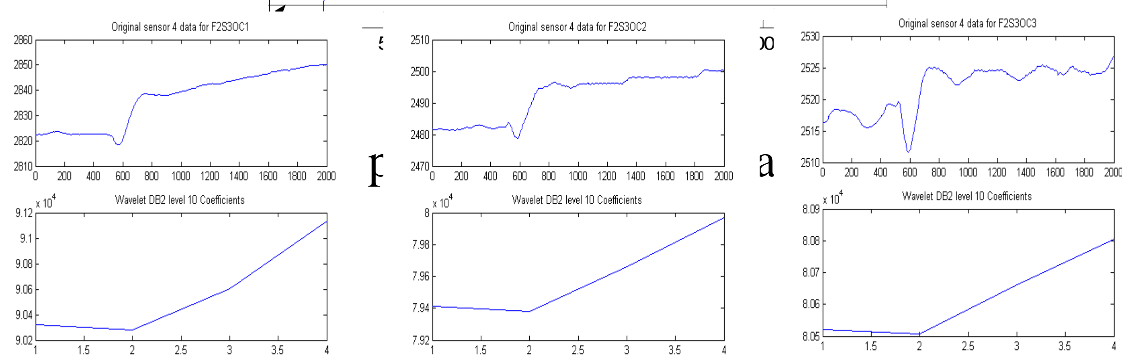
Automotive Engine (cont'd)

- Experimental Results

Mode Invariant Feature Extraction via Wavelets



Original Pressure Signal and DB2L10 Approximate Coefficients Response to 8% AFS Fault at $k = 500$.



Original Engine Speed and DB2L10 Approximate Coefficients Response to 6% AIS_leak fault at $k = 500$.

Automotive Engine (cont'd)

- Experimental Results (cont'd)

	S1T1	S1T2	S2T1	S2T2	S3T1	S3T2	S4T1	S4T2	S5T1	S5T2	S6T1	S6T2	S7T1	S7T2
AFS (F1)	1	0	0	1	0	0	0	0	0	0	0	0	0	1
AIS_leak (F2)	0	1	0	0	0	0	0	1	0	1	0	0	1	0
AF_blockage (F3)	1	0	1	0	0	0	0	0	0	0	0	0	1	0
TAS (F4)	1	0	1	0	0	0	0	0	0	0	1	0	1	0
LFI (F5)	0	0	0	1	0	0	0	0	1	0	0	0	0	0
AEF (F6)	0	0	0	0	0	0	1	0	1	0	0	0	0	0
AFuel_S (F7)	0	0	1	0	0	1	1	0	1	0	0	0	0	0
ESS (F8)	1	0	0	0	0	0	1	0	1	0	0	0	0	0

The accuracies obtained from the two approaches are as follows.
Total classification accuracy via trend testing on wavelet coefficients: **79.86%**
Total classification accuracy via PLS on wavelet coefficients (mean over 20 runs): **98.75%.**

Automotive Engine (cont'd)

- Experimental Results (cont'd)

Confusion matrix of eight engine faults via trend testing and PLS

	Via Trend Testing								Via PLS							
	F1	F2	F3	F4	F5	F6	F7	F8	F1	F2	F3	F4	F5	F6	F7	F8
F1	66.67	0	0	0	0	0	0	33.33	98.57	0	0	0	0	1.43	0	0
F2	0	100	0	0	0	0	0	0	0	99.29	0	0	0	0.71	0	0
F3	0	0	77.78	22.22	0	0	0	0	0	0	96.43	0	0	0	0.71	2.86
F4	0	0	0	100	0	0	0	0	0	0	0	100	0	0	0	0
F5	0	0	0	5.56	94.44	0	0	0	0	0	0	0	100	0	0	0
F6	0	0	0	11.11	38.89	38.89	5.56	5.56	0	2.86	1.43	0	0	95.71	0	0
F7	0	0	0	16.67	0	0	83.33	0	0	0	0	0	0	0	100	0
F8	16.67	0	0	5.56	0	0	0	77.78	0	0	0	0	0	0	0	100

Fault Severity Estimation Average % Errors over 20 runs via PLS

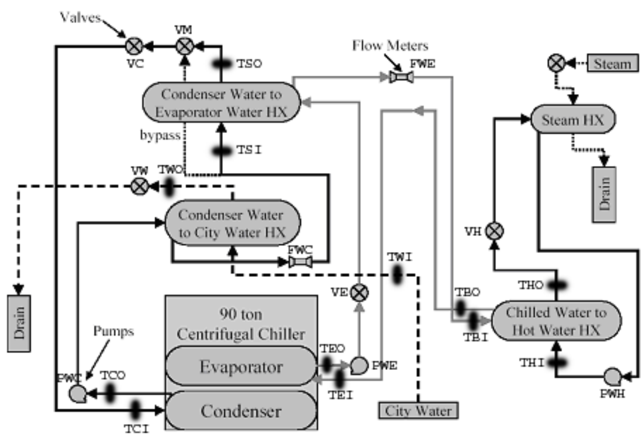
Fault Name	Wavelet Details	Average Errors (%)
AFS (F1)	DB1L9	3.36
AIS_leak (F2)	DB1L9	12.25
AF_blockage (F3)	DB1L9	10.21
TAS (F4)	DB1L9	10.87
LFI (F5)	DB1L9	13.02
AEF (F6)	DB1L10	32.44
AFuel_S (F7)	DB1L9	10.71
ESS (F8)	DB2L10	6.66

Application Examples

- For FDD in Engineering Systems
  - ➔ Simple Gravity Flow Tank System (Model-based)
  - ➔ Automotive Suspension System (Model-based)
  - ➔ Automotive Engine (Data-driven)
  - ➔ HVAC Chiller (Data-driven)
- For Information Retrieval
  - ➔ Text Categorization (Data-driven)

HVAC Chiller

☆ Schematic of Chiller Test Stand



☆ Faults and the Severities

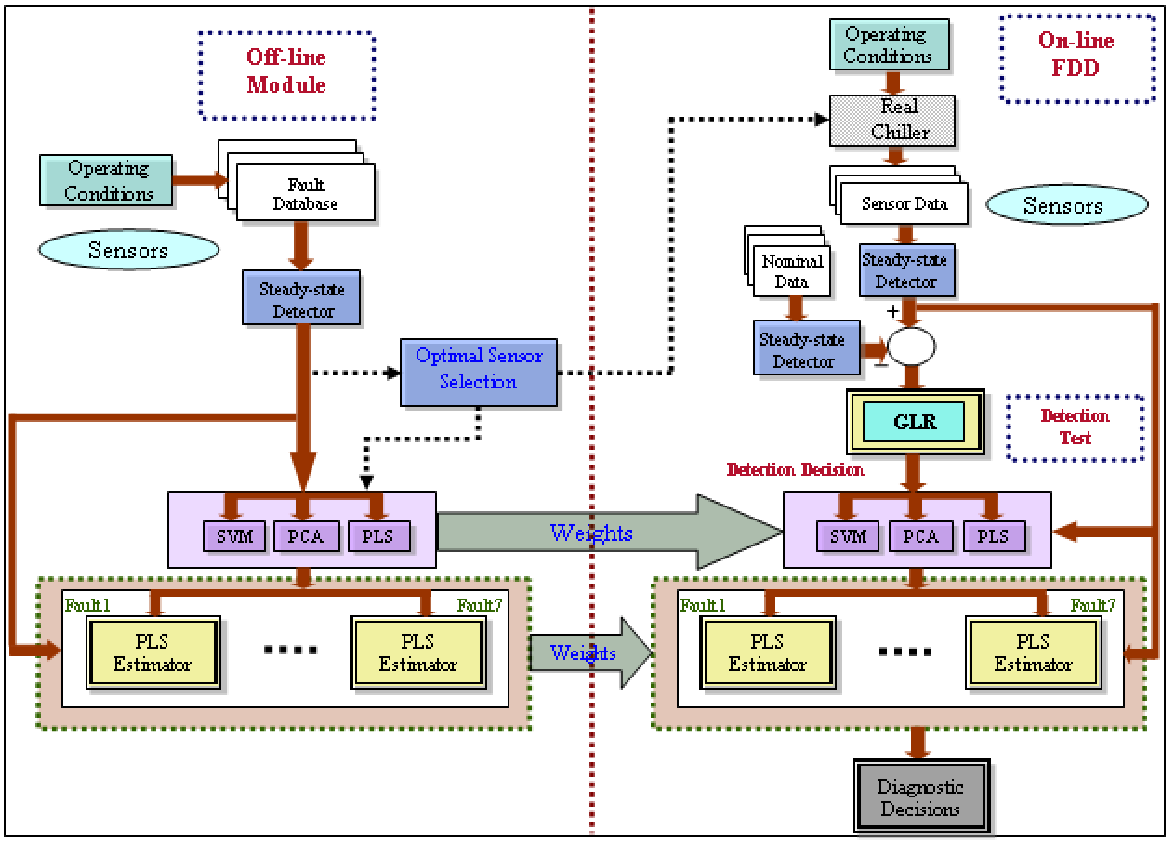
Fault Notation	Description	Severity Levels
RCWF	Reduced condenser water flow	10%, 20%, 30%, 40% reduction
REWF	Reduced evaporator water flow	10%, 20%, 30%, 40% reduction
RL	Refrigerant leak/undercharge	10%, 20%, 30%, 40% reduction
RO	Refrigerant overcharge	10%, 20%, 30%, 40% increase
EO	Excess oil	14%, 32%, 50%, 68% increase
CF	Condenser fouling	12%, 20%, 30%, 45% increase
NCR	Non-condensable in the refrigerant	1%, 2%, 3%, 5% nitrogen addition
DEV	Defective expansion valve	

- 27 Pre-defined Operating Conditions
  - The Input Variables
    - ☆ Evaporator Water Leaving Temperature (TEO)
    - ☆ Condenser Water Entering Temperature (TCI)
    - ☆ The Chiller Cooling Load (Capacity %)

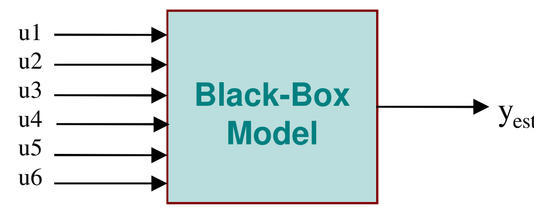
- Measurement Variables
  - 64 Monitored Variables
    - ☆ 48 Measured Variables
    - ☆ 16 Calculated Variables

HVAC Chiller (cont'd)

☆ FDD Process



☆ Chiller Nominal Model Developed via PLS



Inputs:

u1: TEO; u2: TCI;

u3: The chiller cooling load;

u4: $(u3)^2$;

u5: u1*u3; u6: u2*u3

Output:

Sensor Measurement

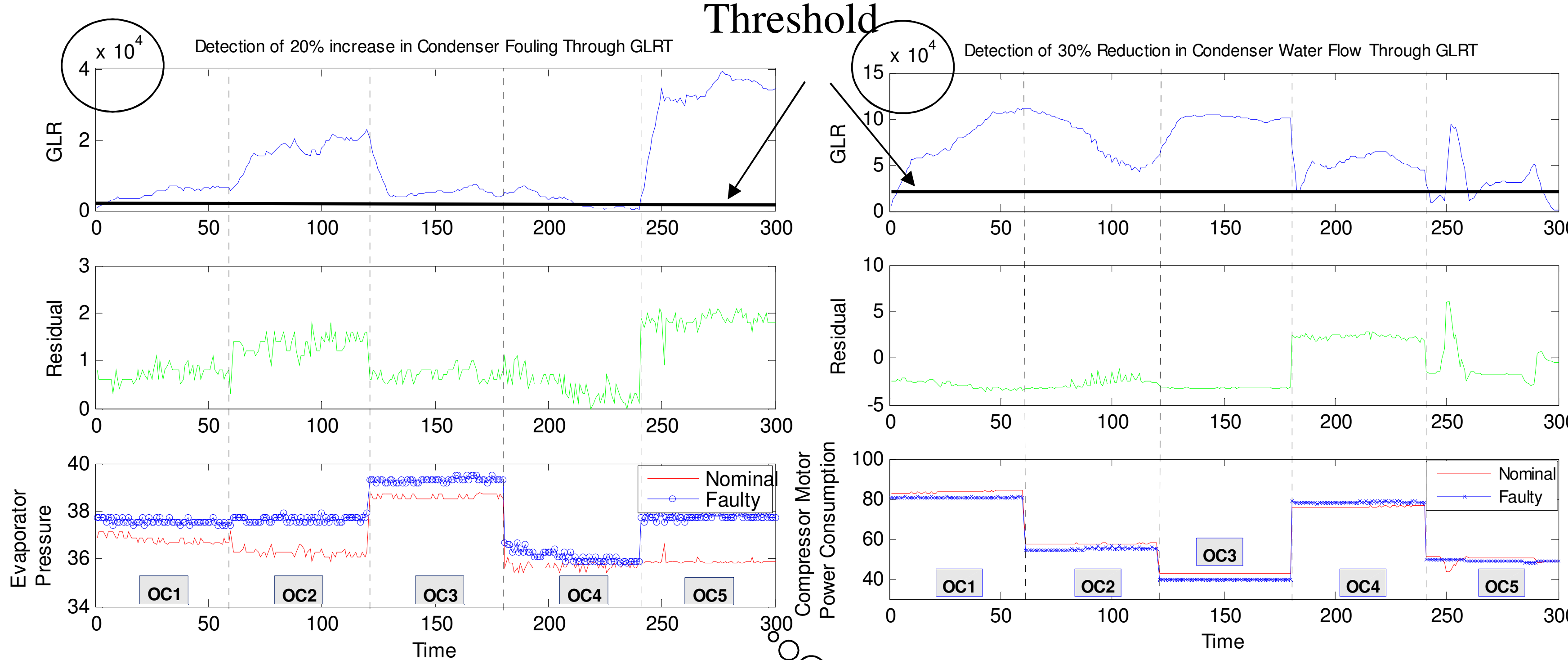
Optimal sensor selection using Genetic Algorithms (GA)

One model for each measurement

HVAC Chiller (cont'd)

☆ Detection of RCWF and CF Faults via GLR

Threshold



100% Fault detection in each operating condition (OC)

HVAC Chiller (cont'd)

✭Fault classification confusion matrix for the data from 48 measured variables

✫ Fault Severity Estimation Results

Faults / Operating Condition	Actual Severities	RCWF (Average % Errors)	REWF (Average % Errors)	RL (Average % Errors)	RO (Average % Errors)	Actual Severities	EO (Average % Errors)	Actual Severities	CF (Average % Errors)	Actual Severities	NCR (Average % Errors)
OC1	0.1	0.9849	1.3325	3.3275	0.08518	0.14	4.3124	0.12	9.7544	0.01	2.4507
	0.2	0.56158	0.67521	4.2133	0.21342	0.32	1.4526	0.2	4.5096	0.02	3.313
	0.3	0.31064	0.39799	2.6577	0.10756	0.5	1.2487	0.3	2.6152	0.03	1.318
	0.4	0.50731	1.5725	1.8546	0.13432	0.68	0.60322	0.45	1.727	0.05	0.40398
OC9	0.1	2.1827	1.2365	10.833	7.4576	0.14	5.3204	0.12	6.725	0.01	5.02
	0.2	0.58717	0.95008	6.3962	3.9268	0.32	2.5752	0.2	4.4604	0.02	3.5608
	0.3	0.36775	0.62639	3.932	2.3581	0.5	0.85241	0.3	2.7412	0.03	1.3368
	0.4	0.53071	0.26202	3.8577	2.2822	0.68	1.4855	0.45	8.3442	0.05	0.65527
OC25	0.1	2.5693	1.8496	9.9797	6.793	0.14	2.9115	0.12	10.591	0.01	17.577
	0.2	1.7763	0.33356	10.918	4.3695	0.32	1.6092	0.2	7.416	0.02	4.2373
	0.3	0.53137	0.39104	8.2514	3.7702	0.5	2.1848	0.3	2.2021	0.03	4.6202
	0.4	0.42957	0.36302	11.243	6.0927	0.68	0.72345	0.45	2.525	0.05	1.6825

Classified as									
CF	PCA	0.15	0.03	0.06	0.09	0.03	**99.35**	0.28	0
	PLS	0.03	0	0.03	0.09	0.03	**99.78**	0.03	0
NCR	SVM	0.03	0	0	0.03	0	0.09	**99.84**	0
	PCA	0	0	0	0.06	0	0.06	**99.88**	0
	PLS	0.03	0	0.06	0.31	0	0	**99.60**	0
DEV	SVM	1.11	0	0	0	0	0	0	**98.89**
	PCA	1.11	0	0	0	0	0	0	**98.89**
	PLS	0.12	0	0.12	0.12	0.12	0	0	**99.51**

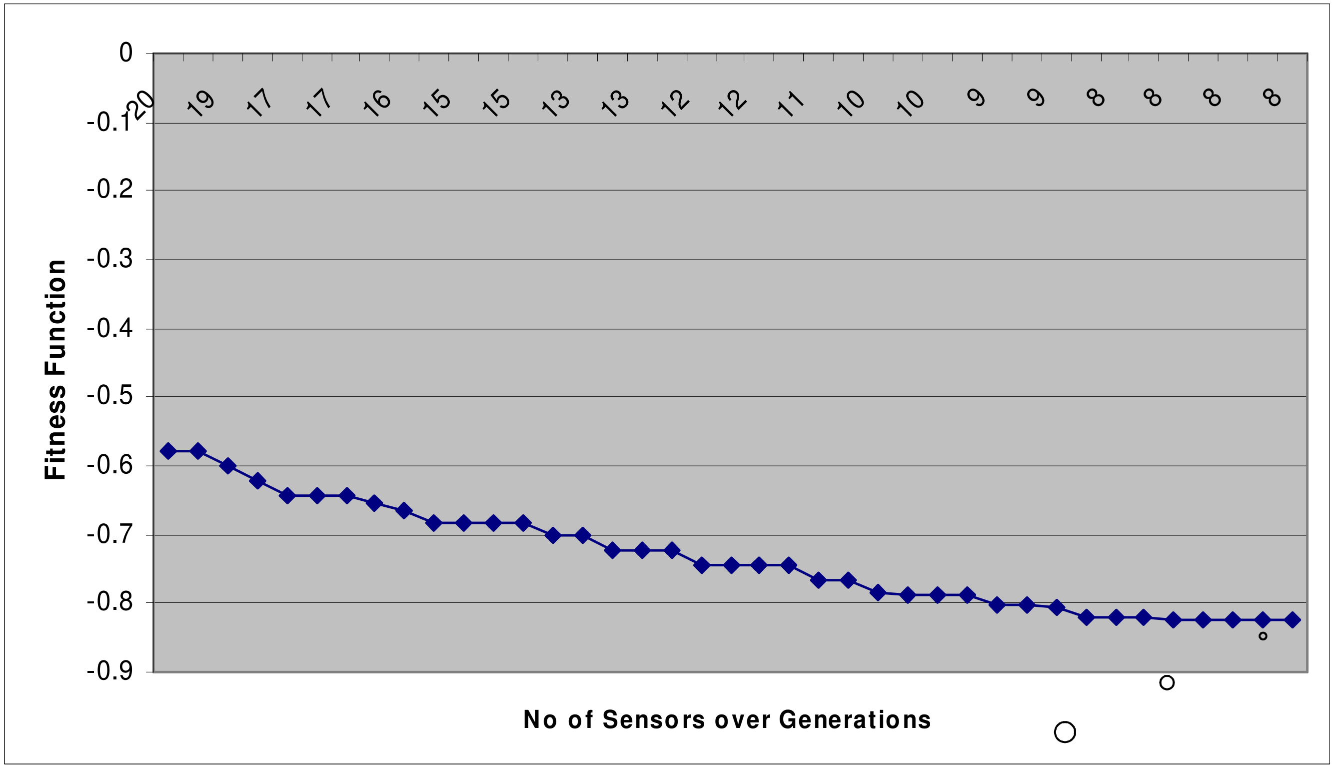
Very good estimates at higher severity levels

HVAC Chiller (cont'd)

✯ SVM is proved to be the best technique for classification via McNemar's Test

$$PLS \underset{100\%}{\leq} PCA \underset{100\%}{\leq} SVM$$

✯ **Evolution of best number of sensors via GA**



Accuracy achieved: 99.19%

Measurement Prediction Results for the Optimal Sensor Suite

☆ **Average % Errors**

Sensors / Operating Condition	TOO	TSO	TBO	T_suc	Unit Status	TO_sump	TWI	THO
1	0.0796	1.2883	0.1764	0.3413	6.0618	2.0633	6.1457	0.9227
2	0.0478	0.6017	0.0902	0.1425	3.2425	2.0833	1.5473	1.7827
3	0.2157	0.8813	0.1718	0.1790	7.2803	3.8624	5.7546	0.8869
4	0.1389	0.6491	0.0598	0.2781	2.5314	1.9654	1.7161	0.3448
5	0.0523	0.6891	0.0796	0.2310	1.9978	1.8703	2.3057	2.1370
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
20	0.1306	0.8574	0.1581	0.5163	0.3443	0.5921	0.7594	0.8949
21	0.0946	2.4728	0.3273	0.5538	0.6740	1.9032	0.8474	2.3227
22	0.2155	0.9980	0.2668	0.4337	2.0762	1.9160	4.5705	0.6502
23	0.1070	0.6952	0.1667	0.6747	3.2159	0.3555	2.6566	0.3229
24	0.0914	1.0979	0.2690	0.3275	3.3450	3.5725	4.1424	0.8248
25	0.1625	1.0720	0.1543	0.3772	2.7118	2.3148	1.9271	1.0256
26	0.2033	0.7903	0.2256	0.4740	3.0374	1.0928	7.1105	0.3848
27	0.2201	1.6295	0.2669	0.6373	3.4958	3.0673	7.4192	2.3457

Leave-one-OC-out cross-validation is performed

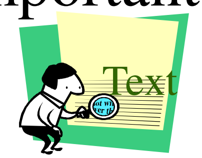
Application Examples

- For FDD in Engineering Systems
  - ➔ Simple Gravity Flow Tank System (Model-based)
  - ➔ Automotive Suspension System (Model-based)
  - ➔ Automotive Engine (Data-driven)
  - ➔ HVAC Chiller (Data-driven)
- For Information Retrieval
  - ➔ Text Categorization (Data-driven)

Text Categorization

➢ Motivation

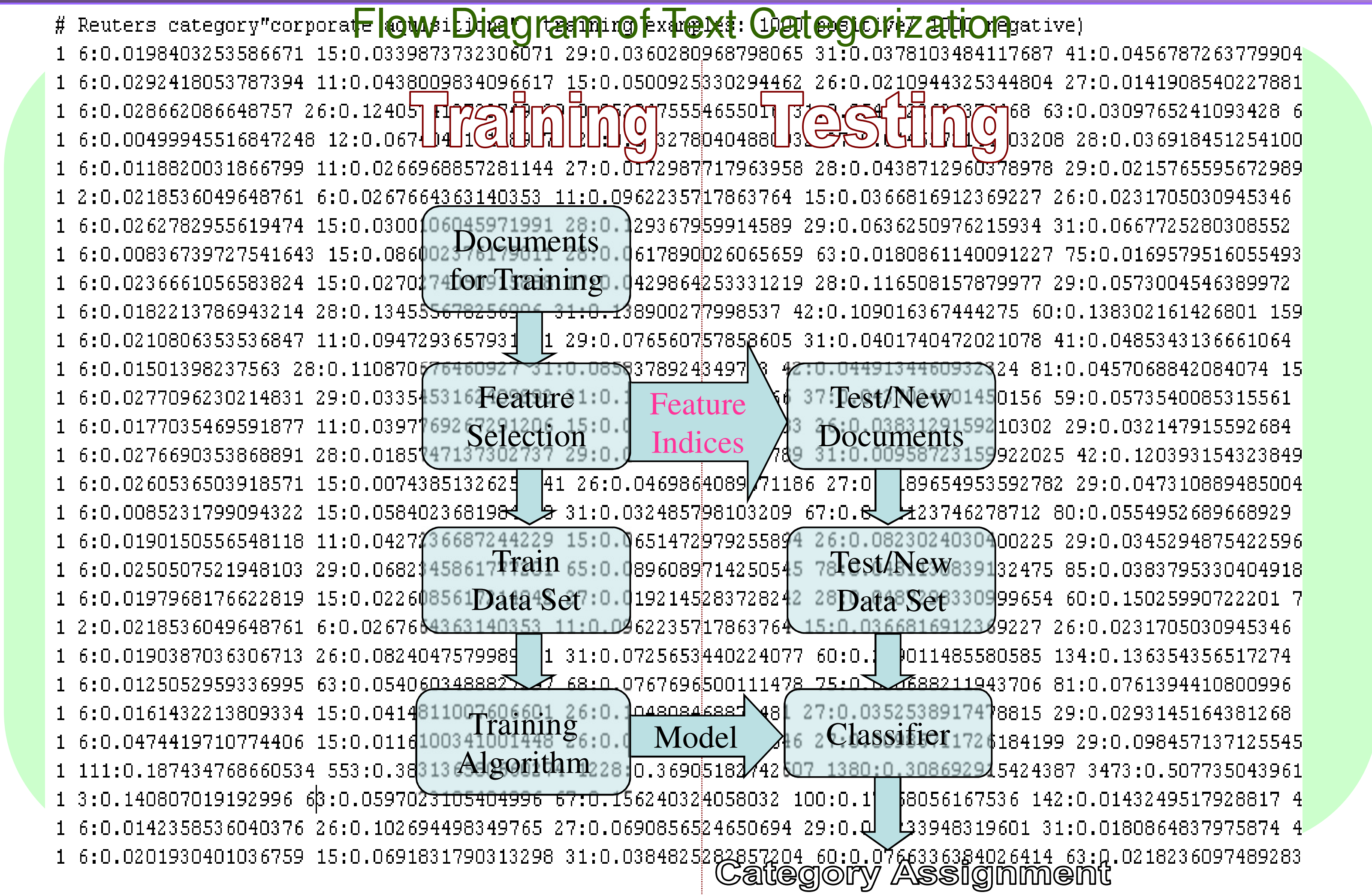
- Modern information sources contain
  - ✫ Massive volumes of online documents, news, intelligence reports, web pages, e-mails, etc.
- Handling of these sources
  - ✫ Largely Manual → Need Automated Engines
- Contents of Information
  - ✫ Images, Voice, Multimedia data, Text, etc.
- Most important content in information retrieval
  - ✫ Text



➢ Goal

- Classification of text documents into a fixed number of predefined categories using two supervised learning algorithms, SVM and PLS

Text Categorization (cont'd)



Documents represented as feature vectors

Initial article in SGML format

Text Categorization (cont'd)

- Feature Selection

Set of documents *(D)*

Class 1 .... Class *p*

Features

Feature Tokenization
Feature Indexing
Feature Reduction

- Single tokens ★
- Phrases
- Stemmed tokens

- High dimensional input space (many features);
- Few irrelevant features;
- Document vectors are sparse;
- Most text categorization problems are linearly separable.

- Feature reduction schemes
  - Stemming
  - ★ Stop word removal
  - Pruning of vocabulary by
    - ★ Mutual information
    - ☆ Document count
    - ☆ Words occurrence count

$$MI(F,Y)=\sum_{F\in\{f,\bar{f}\}}\sum_{Y\in\{C\}}P(F,Y)\log\frac{P(F,Y)}{P(F)P(Y)}$$

- Feature weighting (tokenizing) schemes:

IDF(inv. doc. freq.):	$\log(n/df_i)$
★ TF(term freq.):	tf_{ij}
★ TFIDF:	$tf_{ij}\log(n/df_i)$
LOGTFIDF:	$\log(tf_{ij}+0.5)\log(n/df_i)$
LOGTF:	$\log(tf_{ij})$
BINARY:	1 or 0

Notation:
tf_{ij}: the occurrence frequency of feature *i* in document *j*
n : the total number of documents in the training set
df_i : the number of documents containing the feature *i*

Text Categorization (cont'd) (Datasets)

- Reuter's – 21578 Dataset:
  - Documents about corporate mergers and data acquisitions (ACQ)
    - Classes: ACQ, not ACQ (Binary classification)
    - Number of Documents: 2000(Training), 598(Testing)
    - Number of Features: 9947
    - Performance Measure: Precision-recall break even point
- WebKB Dataset:
  - WWW pages from Carnegie Mellon University from 7 classes (course, department, faculty, project, staff, student and other).
    - Each class data from four different universities and miscellaneous
    - Considered classes: course, faculty, project and student (to be consistent with previous research done)
    - Number of Documents: 4199 (Leave-one-university-out cross-validation)
    - Number of Features: 300
    - Performance Measure: Classification rate

Text Categorization (cont'd) (Datasets)

➢ 20 News Groups Dataset:

- 20,000 Usenet articles from 20 topics

comp.graphics **comp.os.ms-windows.misc** **comp.sys.ibm.pc.hardware** **comp.sys.mac.hardware** **comp.windows.xp**	**rec.autos** **rec.motorcycles** **rec.sport.baseball** **rec.sport.hockey**	**sci.crypt** **sci.electronics** **sci.med** **sci.space**
misc.forsale	**talk.politics.misc** **talk.politics.guns** **talk.politics.mideast** **talk.religion.misc**	**alt.atheism** **soc.religion.christian**

☆ Considered Groups: comp, rec, sci and talk (each separately)

☆ Number of Documents: ≅1000 from each class (two -fold cross-validation)

☆ No. of Features: 5000 from each class

☆ Performance Measure: Classification rate

Text Categorization (cont'd) (Classification Results)

- Precision-Recall Break-even point of Reuter's-21578 data about ACQ

Classifier	Parameters	ACQ
SVM	γ =1.2 G = 5	**96.1**
PLS	k = 11	**96.32**

- Four Classes of WebKB data Results along with Results from previous Research

.Classifier	Accuracy (%)	
SVM	γ = 0.00001 G =1000	γ = 0.000001 G = 1000
	90.21	89.92
PLS	TF k=24	TFIDF k=24
	87.00	**89.6**
NB	87.00	
EM	82.00	
LSI-bg	75.56	

- Four Subgroups of 20 News Groups data

Group Name	SVM (%)		PLS (%)	
	TF	TFIDF	TF	TFIDF
Computers	**74.85** (γ = 0.00002, G = 100)	**77.52** (γ = 0.00001, G = 100)	**71.63** (k = 38)	**75.69** (k = 30)
Recreation	**92.22** (γ = 0.0005, G = 100)	**93.43** (γ = 0.00001, G = 50)	**91.74** (k = 20)	**94.15** (k = 19)
Science	**87.94** (γ = 0.00005, G = 1000)	**89.12** (γ = 0.00001, G = 100)	**86.79** (k = 20)	**91.17** (k = 17)
Talk	**76.87** (γ = 0.0001, G = 100)	**79.47** (γ = 0.00001, G = 50)	**77.87** (k =26)	**81.85** (k = 19)

Conclusion and Future Research

- Conclusion
  - A generic software platform is provided for an analyst to develop and test automated diagnostic procedures for any given system
  - An attempt is made to integrate disparate diagnostic techniques in the form of a toolbox
  - The developed software tools are tested on various real-world examples namely
    - ➔ For FDD in Engineering Systems
      - ➲ Flow tank system, Automotive Suspensions system and Engine
    - ➔ For Information Retrieval
      - ➲ Text Categorization
- Future Research
  - Enhance the usability of the integrated toolbox by testing it on various examples from disparate domains
  - Add more tools to the existing tooset
  - Develop an evaluation tool for comparing performance of the techniques
  - Utilize the developed tools for on-board applications and in the HIL testing during product development phase.