## ICINCO 2007

4th International Conference on Informatics in Control, Automation & Robotics, 9-12 May, 2007, Angers, France

*Keynote Lecture*

# Real Time Diagnostics, Prognostics, & Process Modeling

*Industrial Perspective*

**Dimitar P. Filev**
Senior Technical Leader
Intelligent Control & Information Systems
Ford Research & Advanced Engineering
dfilev@ford.com

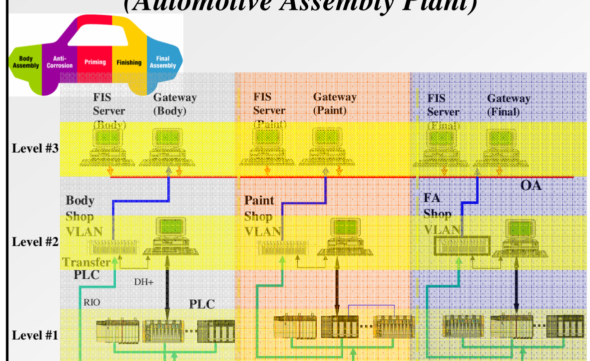
## Ford Motor Company


- Global automotive industry leader based in Dearborn, MI.
- Manufactures and distributes automobiles in 200 markets across six continents.
- 300,000 employees & 108 plants worldwide
- Ford Motor Company celebrated its 100th anniversary on June 16, 2003.

## Outline

- <u>Introduction</u>
- <u>Practical Process Modeling</u>
  - Applied Indirect Adaptive Control
  - Rule-Base Guided Adaptive Control
  - Opportunities for Process Optimization
- Diagnostics & Prognostics in Industrial Setting
  - Research Motivation for Condition-Based / Predictive Maintenance (CBM / PdM)
  - Autonomous Diagnostics & Prognostics – the Novelty Detection Approach
  - Estimation of Machine Health
  - Remaining Useful Life Prediction
  - Applications
- Evolving Systems
- Concluding Comments

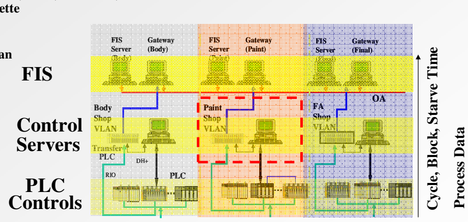
## Generic Plant Information Infrastructure
## *(Automotive Assembly Plant)*




Robots, CNC, Conveyor Drives, Weld Guns, Bells & Reciprocators, Pumps, Lifts, eWorkcells, etc.

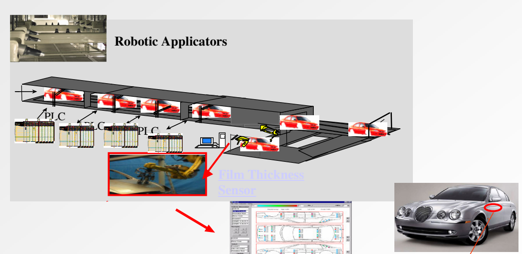
## Generic Plant Informatics Applications

- ActivPlant
  - Ford, Toyota, DCX, CocaCola, Bosch, Gillette
- Cimplicity
  - GM, Nissan

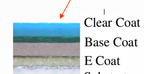



- Low, PLC level: local control algorithms (ladder logic, PID control)
- Intermediate level: information sharing and process performance enhancement
- Upper, FIS level: machine operating attributes for Overall Equipment Efficiency (OEE) assessment & limited set of process data
- **Opportunities:** ***machine health monitoring, process modeling, supervisory control, adaptive control, process optimization, autonomous agents, etc.***

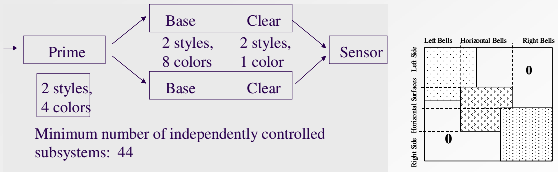
## Applied Intelligent Control: *Control of Automotive Paint Process*

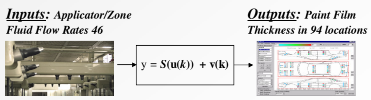



- Multiple Inputs and Outputs
- Nonlinear Behavior
- Strong Interactions
- Multiple Operating Modes
- Legacy Controls

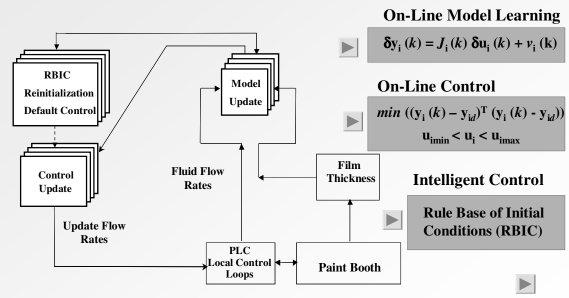


## System Decomposition:
*Independent Subsystems*



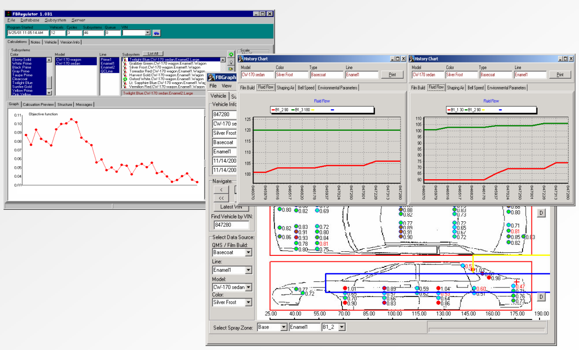




## Intelligent Adaptive Control
*Supervisory Control Architecture*

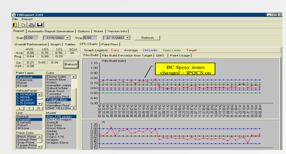



US Patents: 6 528 109, 6 502 059, 6 701 193, 6 484 121, 5 959 211

## Example: *Application of the Intelligent Control Algorithm*

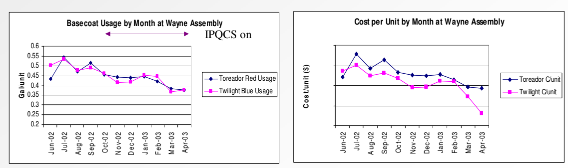


## Impact of Integrated Paint Quality Control on Material Cost

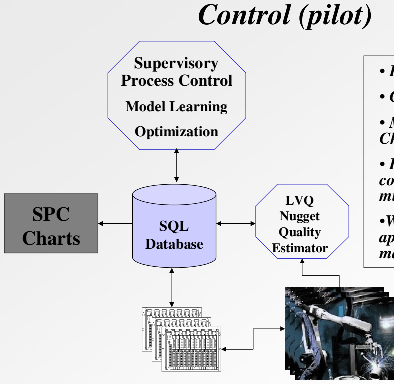


**Material Cost Reduction while IPQCS on:**

- **Toreador – 10.3%**
- **Twilight – 12.4%**

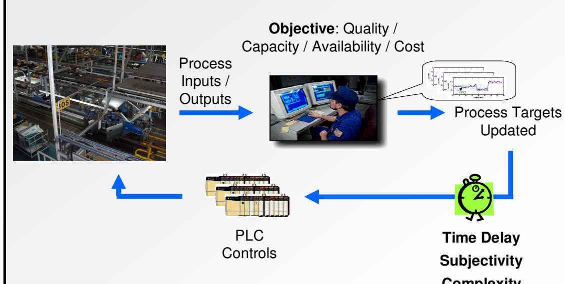



## Supervisory Control Architecture: *Weld Control (pilot)*



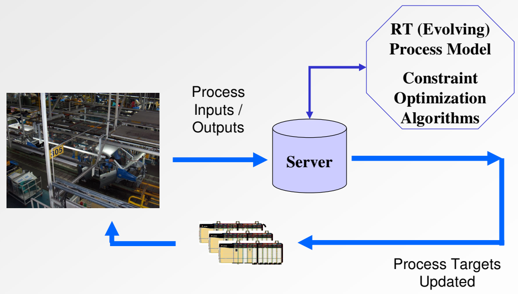

- *Legacy local controls*
- *Computerized sensors*
- *Monitoring & SPC Charts*
- *Improved conventional control solution at minimal risk*
- *Wide range of application in manufacturing*

## Challenges: *Manufacturing is Dominated by Manual Strategies for Process Improvement*




**Conventional Overall Equipment Efficiency (OEE) Scheme**

## Vision : *"Lights Out" (Autonomous) Process Optimization*

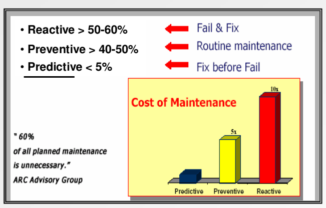



**Real Time Overall Equipment Efficiency (OEE) Optimization Scheme**

# Outline

- **Introduction**
- **Practical Process Modeling**
  - **Applied Indirect Adaptive Control**
  - **Rule-Base Guided Adaptive Control**
  - **Case Study: *Control of Automotive Paint Process***
- **<u>Diagnostics & Prognostics in Industrial Setting</u>**
  - **Research Motivation for Condition-Based / Predictive Maintenance (PdM)**
  - **Autonomous Diagnostics & Prognostics – the Novelty Detection Approach**
  - **Estimation of Machine Health**
  - **Remaining Useful Life Prediction**
  - **Applications**
- **Evolving Systems**
- **Concluding Comments**

## Predictive Maintenance (PdM): *Impact & Opportunities*



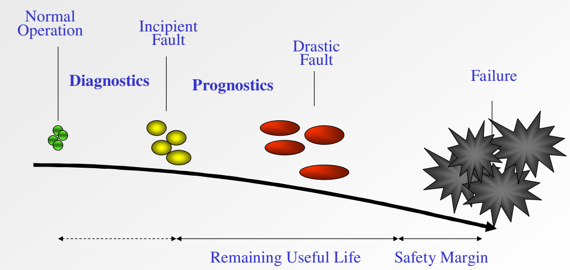

**<u>Current Status</u>**

- **Machine health monitoring is in its early stage**
- **Conventional diagnostics methods & systems are cost prohibitive**
- **Lack of reliable fault prognostics algorithms**
- **Great research opportunity with significant potential impact**

***"More than $1 trillion is spent each year to replace perfectly good equipment because no reliable and cost-effective method is available to predict the equipment's remaining life".*** *(McLean & Wolfe, Sensor Magazine Online, June, 2002)*

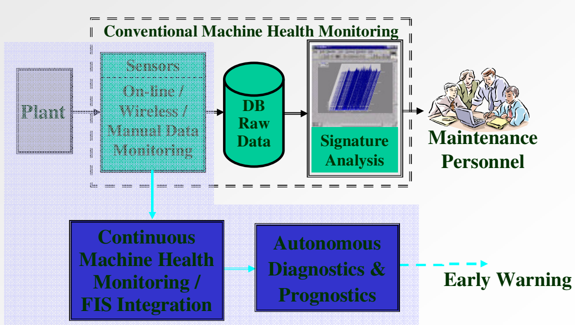
**PdM Systems – one of the main drivers for autonomous D & P**

## Machine Health Monitoring: *Diagnostics & Prognostics of Faults*

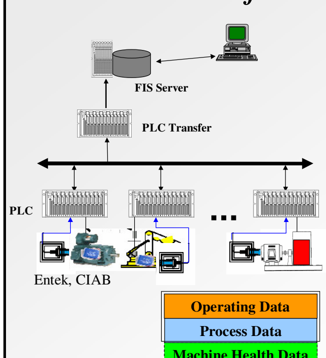



**Diagnostics** - detect and classify fault conditions in real time
**Prognostics** - provide early detection of the precursor to a failure condition

## Modern Machine Health Monitoring: *Synergy between FIS and Autonomous D&P*

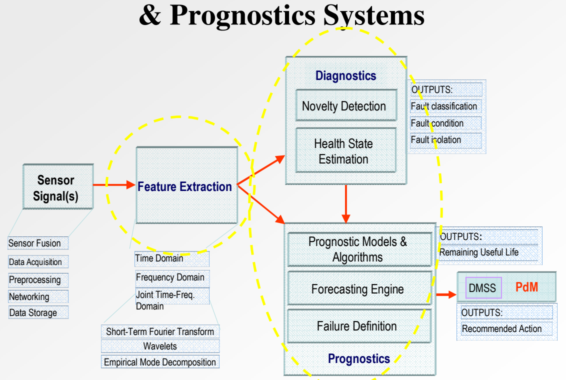



## FIS: *Continuous Machine Health Monitoring for PdM*




- **Continuous monitoring of machine health attributes (vibration, lubricant temperature, etc.) – enabler for novelty detection based fault prognostics**
- **Reduced installation and maintenance cost**
- **Enabler for integration with sensor networks**

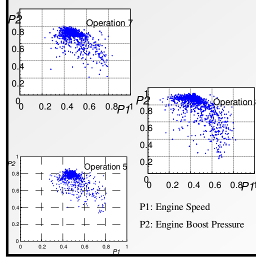
## Basic Elements of Autonomous Diagnostics & Prognostics Systems




## The Novelty Detection Challenge

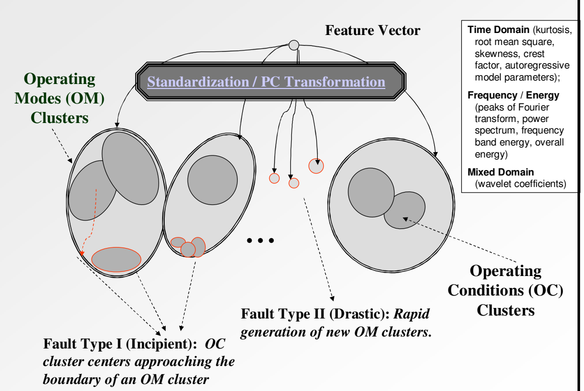
- How to recognize (and compare numerically) the **difference** between the different machine operations?
- How identify the **difference** between a normal machine operation and an anomaly?

Hydraulic Excavator of *Shin-Caterpillar Mitsubishi*




| | Code | Label | Operation Description |
|---|---|---|---|
| FAULTY CONDITIONS | 1 | F1 | Fuel Spray Nozzle deactivated |
| | 2 | F2 | TurboCharger Deterioration |
| | 3 | F3 | Valve Clearance Changed |
| | 4 | F4 | Air Filter Obstruction (High) |
| | 5 | F5 | Air Filter Obstruction (Low) |
| NORMAL | 6 | NH | Normal Operation (High Load) |
| | 7 | NM | Normal Operation (Medium L.) |
| | 8 | NL | Normal Operation (Light Load) |

## Novelty Detection Model




## Learning OM Clusters to Approximate Equipment Operating Modes

**If** {Feature vector is within the closest OM cluster}

$$(\underline{y} - \underline{y}_i^*)' S_{yi}^{-1} (\underline{y} - \underline{y}_i^*) < \chi^2_{p,\alpha}$$

**Then** {Update cluster parameters & transformation mapping}

$$y_i^*(k) = \alpha\, \underline{y}_i^*(k-1) + (1-\alpha)\, \underline{y}(k)$$

$$S_i^{-1}(k) := Q_i(k)$$

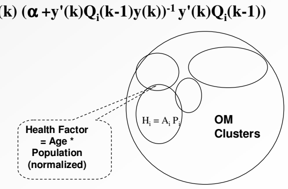
$$Q_i(k) = \alpha^{-1}(Q_i(k-1) - Q_i(k-1)y(k)\,(\alpha + y'(k)Q_i(k-1)y(k))^{-1}\, y'(k)Q_i(k-1))$$

$$T_i: \quad Q_i = T_i^{(u)} Q_i^{(s)} T_i^{(v)'}$$

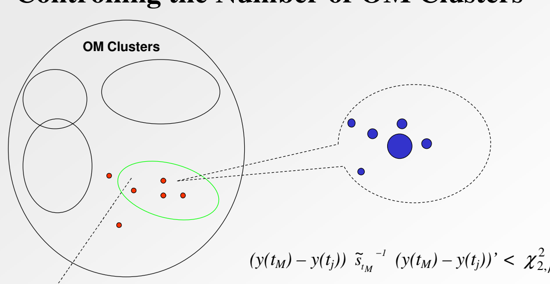
**Else** {Start a new OM cluster}

$$y_{i+1}^* = y(k)$$

$$S_{i+1} = Q_i(0)$$




## Controlling the Number of OM Clusters

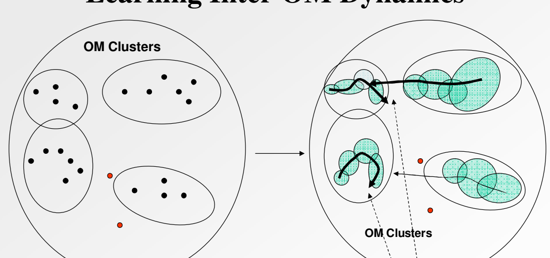



$$(y(t_M) - y(t_j))\ \tilde{s}_{t_M}^{-1}\ (y(t_M) - y(t_j))' < \chi^2_{2,\beta}$$

$T_j = 1 / \sum_{l=1}^{r} (||y(t_j) - y(t_l)||)^2$ ⟶ Mountain Potential (Mountain Clustering)

$\tilde{s}_{t_j} = \sum_{l=1}^{r} (y(t_j) - y(t_l)|)' (y(t_j) - y(t_l)|)$ $\qquad T_j = 1 / tr(\tilde{s}_{t_j}), j=[1, r]$

## Learning Inter OM Dynamics




Evolving Models (eTS) approximate local OM dynamics

$$y_i(k) = f_i(y_i(k-1), y_i(k-2), \ldots, y_i(k-q)),\ i=[1, 2]$$

$x_{ti} = [\,\bar{y}_i(k+T-1)\ \bar{y}_i(k+T-2)\ \bar{y}_i(k+T-3)]^T, i=[1, 2]$ ⟶ $\bar{y}(k+T)$

## Real Time Diagnostics

- Diagnostics is carried out based on three independent methods of analysis
  - **Diagnostics based on classification (PC space)**

$$(y(k) - y_j^*)\, s_j^{-1}\, (y(k) - y_j^*)' < \chi^2_{2,\beta} \qquad r_c$$

  - **Diagnostics based on feature/signal enveloping (feature space)**

$$| Y(k) - \overline{Y}_j | \leq c\, \overline{\sigma}_j \qquad r_{SPC}$$

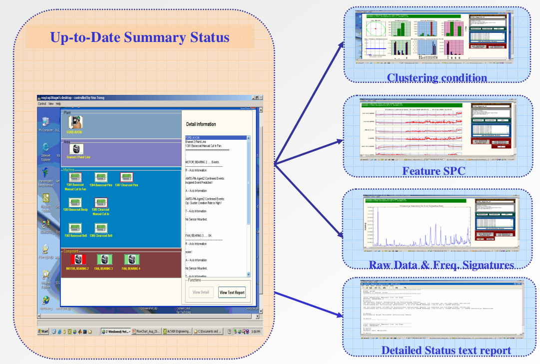
  - **Diagnostics based on velocity threshold**

$$(y(k+1) - y(k))'\,(y(k+1) - y(k)) < v^* \qquad r_v$$

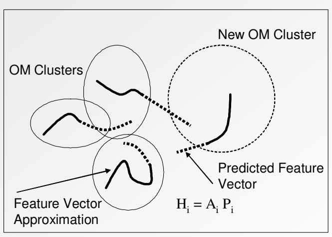
- Above three results are combined through a weighted voting rule to come up with a severity rating

$$S_R = w_c r_c + w_{spc} r_{spc} + w_v r_v$$

## Summarized Machine Health Report




## Prediction of Type 1 (Incipient) Equipment Faults

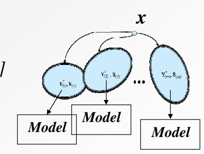



$$I = \arg\min_j \left(d_j(\bar{y}(k+T) - y_j)\, s_j^{-1} (\bar{y}(k+T) - y_j^*)'\right),\ j=[1, m].$$

$$(\bar{y}(k+T) - y_I^*)\, s_I^{-1}\, (\bar{y}(k+T) - y_I^*)' < \chi^2_{2,\beta}$$

**Failure Modes Type I: OM feature models approaching the boundary of an OM cluster with a low health factor**

## Real Time Prognostics: *eTS Model Based Prediction*



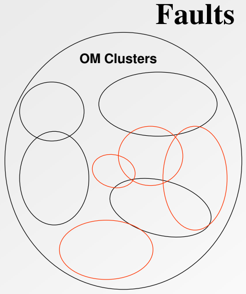

- Fuzzily blended sub-models:

$$\mathfrak{R}^i\text{: IF } (x_1 \text{ is } \aleph_1^i) \text{ AND } \ldots \text{ AND } (x_n \text{ is } \aleph_n^i)$$
$$\text{THEN } (y^i = x_e^T \pi^i) \quad i=[1, R]$$

- Pseudo-linear universal approximators
- Approximate non-linear dynamics
- Multiple operating modes
- Parameter and structure learning
- NN-like computational efficiency (ANFIS)

$$y = \Sigma_i v_i(x)\, y^i(x)$$

## Prediction of Type 2 (Drastic) Equipment Faults

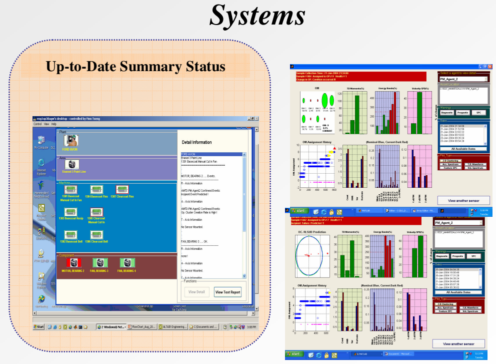



**Dramatic increase of the number of created clusters with low health factors**

$$(T^*_O(k) - T_O(k))^2 / V^*_O(k) < 9$$

$$T^*_O(k+1) = \alpha T^*_O(k) + (1 - \alpha)\, T_O(k)$$
$$V^*_O(k+1) = \alpha V^*_O(k) + (1 - \alpha)\, (T_O(k) - T^*_O(k))^2$$

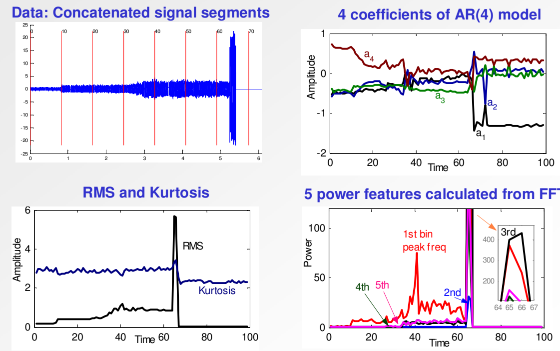
## Summarized Status of Monitored Equipment: *A Step Towards Immune Systems*



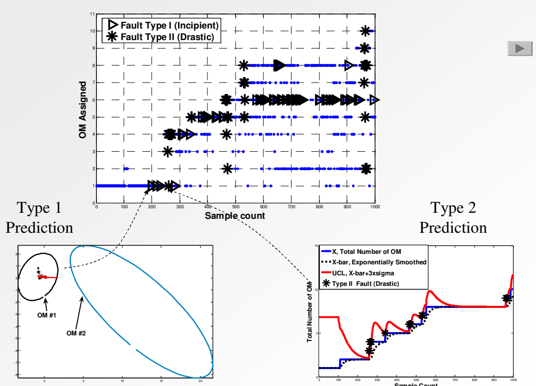
## Validation: *Accelerated Testing of Bearings under Severe Conditions*

- 6309 DGBB bearing
  SKF Condition Local Monitoring Unit (LMU).
- Bearing life
  - Normal Condition: Several years assuming 24 hours/day operation
  - Accelerated testing conditions (increasing load from 2.6 KN to 18 KN)
- Sensor: An accelerometer in the vertical axis
- Sampling: 213 points snapshot at 5 kHz at 15 min interval

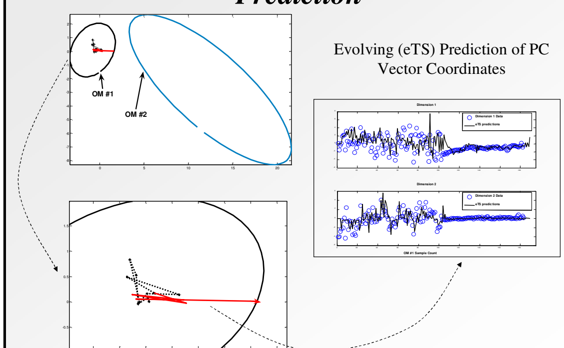
## Example: *Data and Features*



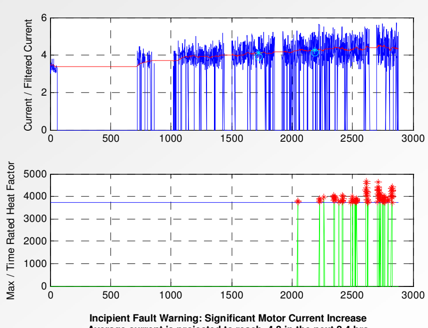

## Example: *Incipient & Drastic Fault Warnings*



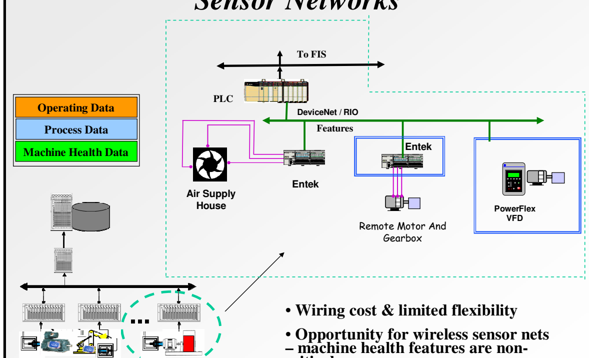

## Example: *Zoomed-In View of Incipient Fault Prediction*




## Example: *Single Feature (Current Monitoring of VFD Controlled Electric Motors)*

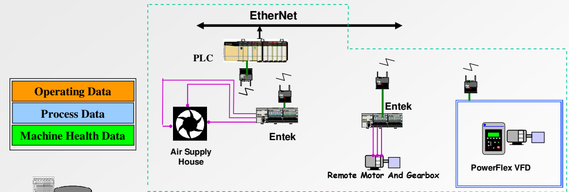


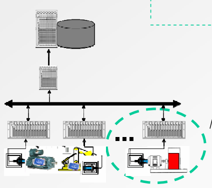

## Efficient FIS / PdM Integration: *Local Sensor Networks*




- Wiring cost & limited flexibility
- Opportunity for wireless sensor nets – machine health features are non-critical

## Efficient FIS / PdM Integration: *Local Wireless Sensor Networks*



**Cost – Wired vs. Wireless**

**Accelerometers:**

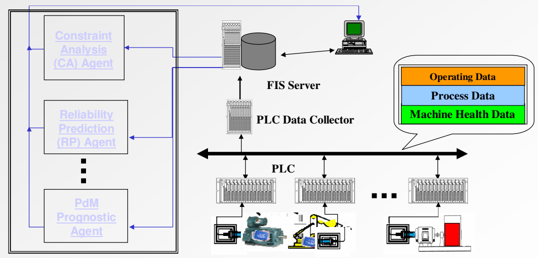
**$100 / sensor ≈ $600 - $1000 (installed)**

**$400 / wireless sensor ≈ $450 (installed)**

**Flexibility**

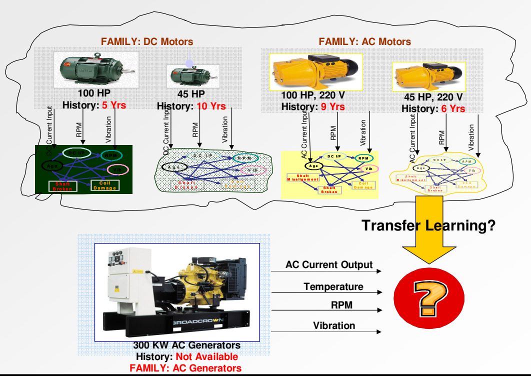
**Access to locations and devices that would be expensive to wire (air supply houses, cranes, turntables, etc.)**

## Vision: *Agent Based OEE Optimization*

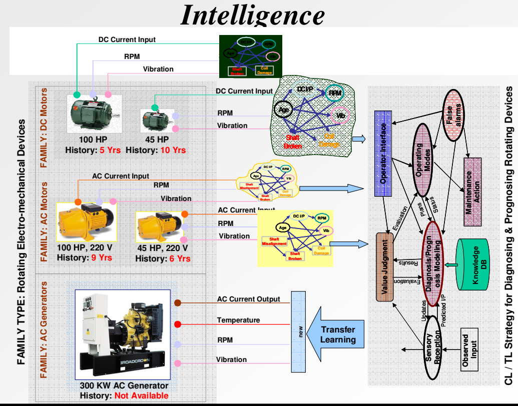



- **Autonomous CA & RP agents increase FIS effectiveness through automating some of the main tasks in constraint analysis and equipment reliability assessment & prediction**
- **Autonomous PdM agent identifies potential drastic or incipient faults and generates warnings before faulty state occurrence; PdM agent is based on Ford patented unsupervised learning algorithm requiring minimal spec information during setup**

## Vision: *Transfer Learning Approach*



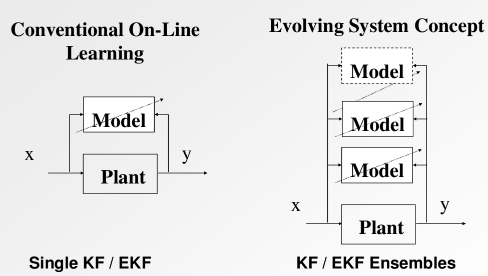

## Vision: *Transfer Learning & Collective Intelligence*




# Outline

- **Introduction**
- **Practical Process Modeling**
  - **Applied Indirect Adaptive Control**
  - **Rule-Base Guided Adaptive Control**
  - **Case Study: *Control of Automotive Paint Process***
- **Diagnostics & Prognostics in Industrial Setting**
  - **Research Motivation for Condition-Based / Predictive Maintenance (CBM / PdM)**
  - **Autonomous Diagnostics & Prognostics – the Novelty Detection Approach**
  - **Applications**
- **<u>Evolving Systems</u>**
- **Concluding Comments**

## Evolving Systems




- **Systems that autonomously develop their structure and parameters based on data.**
- **Adaptive systems with memory & multiple model structure**

## Fuzzy Systems That Can Evolve

↓

## Takagi-Sugeno (TS) Models

$\mathfrak{R}^i$: *IF* ($x_1$ *is* $\aleph_1^i$) *AND … AND* ($x_n$ *is* $\aleph_n^i$) *THEN* $y^i = f_i(x)$ $i=[1, R]$

- Real time parameter and structure learning
- Fuzzily blended local sub-models
- Open multimodel structure
- Universal approximators
- Multiple operating modes
- NN-like computational efficiency

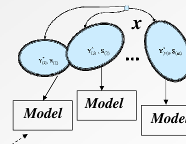
Applications:

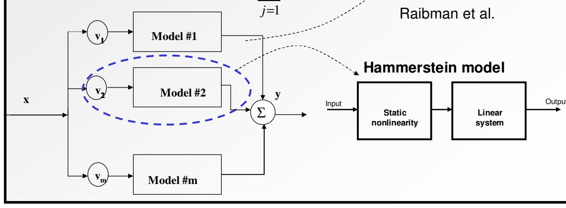
- Soft Sensors
- Prognostics
- Control

## Alternative Interpretations of TS

$\mathfrak{R}^i$: *IF* ($x_1$ *is* $\aleph_1^i$) *AND … AND* ($x_n$ *is* $\aleph_n^i$) *THEN* ($y^i = x_e^T \pi^i$) $i=[1, R]$

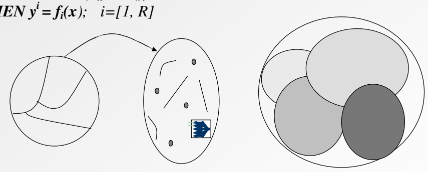
$y = \Sigma_i v_i(x)\ y^i(x)$ $\qquad v_i = \frac{\tau^i}{\sum_{j=1}^{N} \tau^j}$



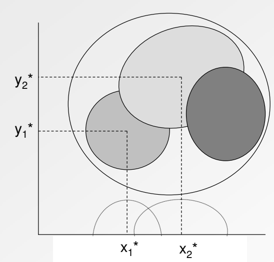

## Flexible TS: *Geometric Interpretation*

$\mathfrak{R}^i$: *IF* ($x_1$ *is* $\aleph_1^i$) *AND … AND* ($x_n$ *is* $\aleph_n^i$) *THEN* $y^i = f_i(x)$; $i=[1, R]$



- Fuzzy partitioning of the input space
- Rules are granules (bundles, clusters) in the input –output space
- Fuzzy model structure identification problem can be transformed to clustering (similarly to RBF NN)
- Granular structure implies system decomposition and simpler subsystem models

## Evolving Modeling Paradigm

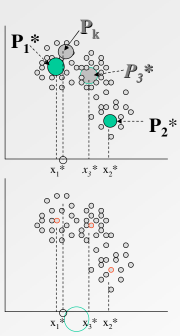



IF x is close to $x_1^*$ THEN y is $y_1^*$

IF x is close to $x_2^*$ THEN y = $a_0$ + $a_1$ x

- Structure Learning through Incremental Clustering the Input-Output Space (K-NN, SOM, K-NN-M, Evolving Mountain/Subtractive)
- Input-Output Cluster Covariance Defines the Consequent Subsystem Structure
- Parameter Learning (RLS, LMS, BP, EKF, Clustering)

## eTS: *On-Line Learning Model Structure Using Incremental Mountain Clustering*




1. *IF the potential of z(k) is significantly different than all cluster centers potentials*
   THEN
   *IF the new data input x(k) is close to the existing antecedent centers* $x_i^*$
   THEN *x(k) replaces this center* $x_i^*$
   ELSE *a new rule with center* $x^*_{M+1}$ *is created*
2. *IF the potential of z(k) is within the range of cluster centers potentials*
   THEN *the new data point z(k) is assigned to the closest center and model is updated*

## eTS: *Recursive Least Square (RLS, KF / EKF / EKF+) Model Learning*

$$\psi = [\lambda_1 x^T, \lambda_2 x^T, \dots, \lambda_N x^T]^T \qquad \theta_k = \left[\pi^T_{1(k-1)}, \pi^T_{2(k-1)}, \dots, \pi^T_{R(k-1)}, \pi^T_{(R+1)k}\right]^T$$

– **Parameter Learning (No New Rule)**

$$y_k = \psi_k^T \theta_{k-1}$$

$$\theta_k = \theta_{k-1} + C_k \psi_k (y_k - \psi_k^T \theta_{k-1}) \qquad \theta_0 = 0$$

$$C_k = C_{k-1} - \frac{C_{k-1}\psi_k\psi_k^T C_{k-1}}{1+\psi_k^T C_{k-1}\psi_k};\ k=[1, TD] \qquad C_0 = \Omega I$$

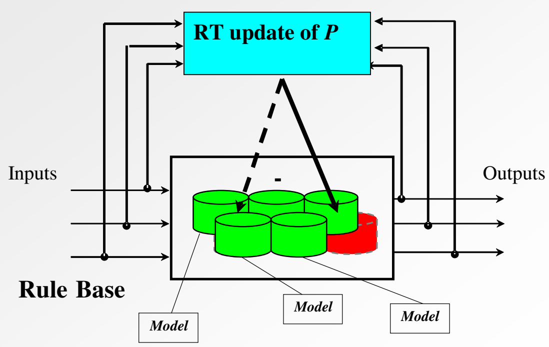
– **Structure Learning (New Rule Added):**

$$C_k = \begin{bmatrix} \rho\varsigma_{11} & \dots & \rho\varsigma_{1R(n+1)} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \rho\varsigma_{R(n+1)1} & \dots & \rho\varsigma_{R(n+1)R(n+1)} & 0 & \dots & 0 \\ 0 & 0 & 0 & \Omega & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & \Omega \end{bmatrix}$$

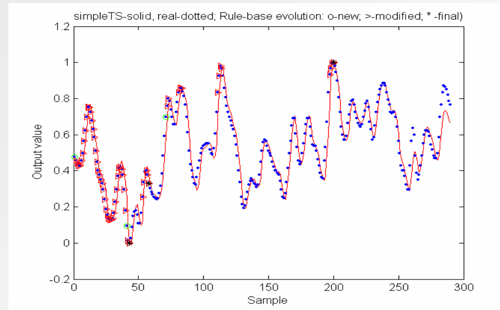
$$\pi_{R+1k} = \sum_{i=1}^{R} \frac{\frac{1}{1+(z_k - z_i^*)^2}}{\sum_{j=1}^{R} \frac{1}{1+(z_k - z_j^*)^2}} \pi_{ik-1}$$

$$\rho = \frac{R^2+1}{R^2}$$

## eTS: *Summary*




## Test Results: *Box-Jenkins Data Set*




- Data samples that originate new rule (circles)
- Data samples that replace existing centers (triangles)
- Final position of the focal point (asterisk)

## Real Time Diagnostics, Prognostics, & Process Modeling: *Industrial Perspective*

### Drivers

- **Advances in information technology, sensor development, and communications**
- **Network architectures, sensor nets, industrial web**
- **Exponentially increasing amount of information**

### Challenges

- **Need for real time autonomous algorithms for diagnostics, prognostic, process control & optimization to increase**
- **Fast, and cost effective development process**
- **Robustness**
- **Performance self-assessment capability**
- **Low cost of ownership and maintenance**

# Reference Material

## Adaptive Control Strategy

*On-Line Model Learning*

$y(t) = S(\mathbf{u}(t))$: $\boldsymbol{J}(k+1) = \boldsymbol{J}(k) + w(k)$

$\boldsymbol{\delta}\mathbf{y}(k) = \boldsymbol{J}(k)\, \boldsymbol{\delta}\mathbf{u}(k) + v(k)$

***LMS (Widrow-Hoff) Model Learning***

$$\boldsymbol{J}(k) = \boldsymbol{J}(k\text{-}1) + \alpha\, (\boldsymbol{\delta}\mathbf{y}(k) - \boldsymbol{J}(k\text{-}1)\, \boldsymbol{\delta}\mathbf{u}(k))\, \boldsymbol{\delta}\mathbf{u}^T(k) / (\boldsymbol{\delta}\mathbf{u}^T(k)\, \boldsymbol{\delta}\mathbf{u}(k))$$

***RLS (Kalman Filter) Model Learning***

$$\boldsymbol{J}^T_j(k) = \boldsymbol{J}^T_j(k\text{-}1) + \mathbf{L}_j(k\text{-}1)\, (\boldsymbol{\delta}\mathbf{y}_j(k) - \boldsymbol{J}^T_j(k\text{-}1)\, \boldsymbol{\delta}\mathbf{u}(k))$$

$$\mathbf{L}_j(k\text{-}1) = \mathbf{P}_j(k\text{-}1)\, \boldsymbol{\delta}\mathbf{u}(k)\, (\mathrm{R}_j + \boldsymbol{\delta}\mathbf{u}^T(k)\, \mathbf{P}_j(k\text{-}1)\, \boldsymbol{\delta}\mathbf{u}(k))^{-1}$$

$$\mathbf{P}_j(k\text{-}1) = \mathbf{L}_j(k\text{-}1)\, \boldsymbol{\delta}\mathbf{u}^T(k)\, \mathbf{P}_j(k\text{-}1) + \mathbf{Q}_j$$

## Adaptive Control Strategy

*On-Line Control (Constrained Optimization)*

$$\min\ ((\mathbf{y}(k) - \mathbf{y}_d)^T (\mathbf{y}(k) - \mathbf{y}_d)) \quad \mathbf{u}_{min} < \mathbf{u} < \mathbf{u}_{max}$$

***Levenberg-Marquardt (LM) Control Update***

$\mathrm{u}(k+1) = \mathrm{u}(k) + \mathrm{K}(k)\, (\mathrm{y_d} - \mathrm{y}(k))$

$\mathrm{K}(k) = \boldsymbol{J}^T(k)\, \mathbf{G}\, (\rho\, \mathrm{I} + \boldsymbol{J}(k)\, \boldsymbol{J}^T(k))^{-1}$

$\mathbf{u}(k+1) = \mathrm{sat}(\mathbf{u}(k) + \boldsymbol{J}^{+}(k)\, (\mathbf{y}_d - \mathbf{y}(k)))$

***Constrained Optimization Control Update***

$\mathrm{u}(k) = \mathit{argmin}\, (\|\mathrm{y_d} - \mathrm{y}(k)\|^2 + \alpha\, \|\mathrm{u}(k) - \mathrm{u}(k\text{-}1)\|^2)$

s.t.

$y(k) = \mathrm{y}(k\text{-}1) + \boldsymbol{J}(\mathrm{u}(k) - \mathrm{u}(k\text{-}1))$

## Stability of IPQC Control Algorithm

$$\Delta\mathbf{y}(k) = \mathbf{J}(k)\ \Delta\mathbf{u}(k)$$

$$\mathbf{J}_j(k+1) = \mathbf{J}_j(k) + \mathrm{w}_j(k)$$
$$\Delta\mathbf{y}_j(k) = \Delta\mathbf{u}^T(k)\ \mathbf{J}_j(k) + \mathbf{v}_j(k)$$

**IPQC Control Laws:**

- **Levenberg-Marquardt Control Update**
- **LQR Control Update**
- **Constraint Optimization Control Update**

---

- **LQR Control Update**

$$\mathrm{u}(k+1) = \mathrm{u}(k) + \mathrm{K}(k)\ (\mathrm{y}_d - \mathrm{y}(k))$$
$$\mathrm{K}(k) = (\hat{\mathbf{J}}^T(k)\ \overline{\mathbf{P}}\ \hat{\mathbf{J}}(k) + \overline{\mathbf{R}})^{-1}\ \hat{\mathbf{J}}^T(k)\ \overline{\mathbf{P}}$$

$$\mathbf{y}(k+1) = \mathbf{y}(k) + \hat{\mathbf{J}}(k)\ \Delta\mathbf{u}(k+1)$$

$$J = \sum_{k=1}^{\infty} (y_d - \mathbf{y}(k))^T\ \overline{\mathbf{Q}}\ (y_d - \mathbf{y}(k)) + \Delta\mathbf{u}^T(k+1)\overline{\mathbf{R}}\Delta\mathbf{u}(k+1)$$

$$\overline{\mathbf{Q}} = \overline{\mathbf{P}}\hat{\mathbf{J}}(\hat{\mathbf{J}}^T\overline{\mathbf{P}}\hat{\mathbf{J}} + \overline{\mathbf{R}})^{-1}\hat{\mathbf{J}}^T\overline{\mathbf{P}}$$

- **Closed Loop Dynamics: LQR Control Update**

$$\mathbf{y}(k+1) = \mathbf{y}(k) + \mathbf{J}(k+1)\Delta\mathbf{u}(k+1)$$
$$= (\mathbf{I} - \mathbf{J}(k+1)(\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}}\hat{\mathbf{J}}(k) + \overline{\mathbf{R}})^{-1}\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}})\mathbf{y}(k) +$$
$$(\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}}\hat{\mathbf{J}}(k) + \overline{\mathbf{R}})^{-1}\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}}\mathbf{y}_d$$

$$\mathbf{A} = (\mathbf{I} - \mathbf{J}(k+1)(\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}}\hat{\mathbf{J}}(k) + \overline{\mathbf{R}})^{-1}\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}})$$

---

## Stability Condition

- ***Closed Loop Dynamics: LM Control Update***

$$\mathbf{A} = \mathbf{I} - \mathbf{J}(k+1)\ \hat{\mathbf{J}}^T(k)\ \mathbf{G}\ (\rho\,\mathbf{I} + \hat{\mathbf{J}}(k)\ \hat{\mathbf{J}}^T(k))^{-1}$$

- ***Closed Loop Dynamics: LQR Control Update***

$$\mathbf{A} = (\mathbf{I} - \mathbf{J}(k+1)(\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}}\hat{\mathbf{J}}(k) + \overline{\mathbf{R}})^{-1}\hat{\mathbf{J}}^T(k)\overline{\mathbf{P}})$$

**Stability Condition**

$$\mathbf{J}(k+1) \approx \hat{\mathbf{J}}(k) \ \Rightarrow\ \mathbf{A}^T(k)\ \mathbf{I}\ \mathbf{A}(k) < \mathbf{I}$$

---

## Intelligent Control Strategy

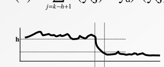
### *The Rule-Base of Initial Conditions*

*IF* x(k) is *close to* $x_i^*$
*THEN* $\tilde{J}(k)$ is $J_i^*$ ; $\tilde{u} = u_i^*$

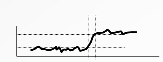
$$\tilde{J}(k) = \sum_{i=1}^{m} \tau_i J_i^* \Big/ \left(\sum_{i=1}^{m} \tau_i\right);$$

$$\tilde{u}(k) = \sum_{i=1}^{m} \tau_i u_i^* \Big/ \left(\sum_{i=1}^{m} \tau_i\right)$$

$$E(k) = \sum_{j=k-h+1}^{k} (y(j) - y_d)'\ (y(j) - y_d)$$




**Update the RBIC with the current operating point & model**



**Infer a new initial condition from the RBIC**

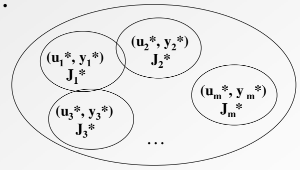
**Synergy Between Conventional & Intelligent Control**

---

## The Rule-Base of Initial Conditions (RBIC)

*IF* x(k) is *close to* $x_i^*$ *THEN* $\tilde{\mathbf{J}}(k)$ is $\mathbf{J}_i^*$ and $\tilde{u}$ is $u_i^*$

. . .




$\mathbf{x}(k) = (\mathbf{u}^T(k), \mathbf{y}^T(k))$ current operating point

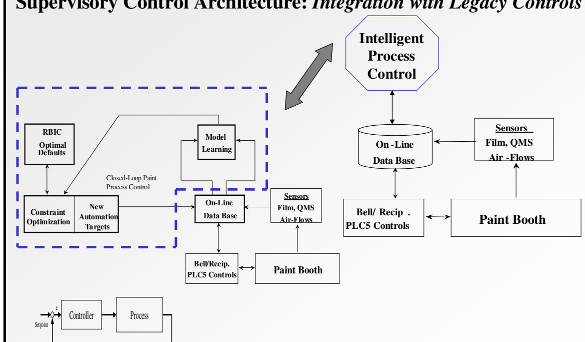
$\mathbf{x}_i^* = (\mathbf{u}_i^{*T}, \mathbf{y}_i^{*T})^T$ prototypical operating point

$$t_i = exp(-(\mathbf{x}(k) - \mathbf{x}_i^*)^T(\mathbf{x}(k) - \mathbf{x}_i^*)/\sigma^2),\ i=[1, m]$$

---

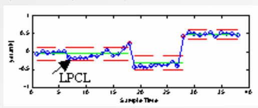
## Implementation of Intelligent Process Control Systems

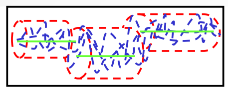
**Supervisory Control Architecture:** ***Integration with Legacy Controls***



## Novelty Detection: *Inspired from Process Monitoring*

- Statistical Process Control Charts
  - Statistical process control (SPC) involves using statistical techniques to measure and analyze the variation in processes
- Pattern Recognition Approach
  - Conventional classification methods require enough examples for all classes; problem – limited fault data
- **Novelty detection is the process of learning the *normality* of a system by fitting a model to the set of normal examples**

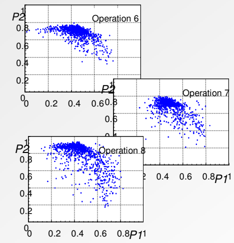


$|y - y_i^*| < 3\,\sigma_{yi}$  $(\underline{y} - \underline{y}_i^*)'\, S_{yi}^{-1}(\underline{y} - \underline{y}_i^*) < (1/n)\chi^2_{p,\alpha}$

## Novelty Detection Paradigm: *Main Problem #1*

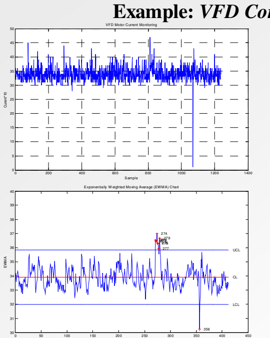
**The Problem:** *How* to recognize (and compare numerically) the **difference** between the different machine operations? *Once this problem is solved, we would be able to discover the slow (often: invisible) tendency of **performance deterioration** of the machine/system.*




| | Code | Label | Operation Description |
|---|---|---|---|
| FAULTY CONDITIONS | 1 | F1 | Fuel Spray Nozzle deactivated |
| | 2 | F2 | TurboCharger Deterioration |
| | 3 | F3 | Valve Clearance Changed |
| | 4 | F4 | Air Filter Obstruction (High) |
| | 5 | F5 | Air Filter Obstruction (Low) |
| NORMAL | 6 | NH | Normal Operation (High Load) |
| | 7 | NM | Normal Operation (Medium L.) |
| | 8 | NL | Normal Operation (Light Load) |

## Novelty Detection Paradigm: *Main Problem #2*

**Example:** ***VFD Control of Electric Motors***



**VFD Control:**

**$\Phi$ = V / f = const**

**Major Reasons for High Current:**

Time

1. Load increase
2. Voltage drop
3. Friction increase (jam, brake, gearbox / bearing failures)
4. Insulation degradation

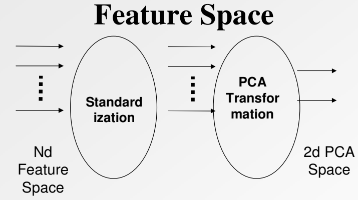
**VFD Time Rated Heat Factor Window:**

**$HF = I^2\,T = (1.5\,I_n)^2\,180$**

**HF Range:**

**$I = [102\% - 250\%\,I_n]$; $T = [1000s – 40s]$**

## Preprocessing & Transformation of Feature Space




Recursive Standardization

$$Y(k) = (Y^R(k) - \bar{Y})\,(diag(\bar{S}))^{-0.5}$$

$$\bar{Y}(k+1) = \alpha\bar{Y}(k) + (1 - \alpha Y^R(k+1))$$

$$\bar{S}(k+1) = \alpha\bar{S}(k) + (1 - \alpha)\,(Y_R(k+1) - \bar{Y}(k+1))'\,(Y^R(k+1) - \bar{Y}(k+1))$$

Recursive PCA Transformation

$$S(k+1) = \alpha S(k) + (1 - \alpha)\,Y(k+1)'\,Y(k+1)$$

$$S = T * S_o * T'$$

$$y(k) = Y(k) * T^{(2)}$$

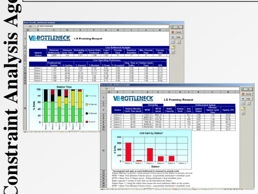
## Constraint Analysis and Reliability Prediction Agents

| Cycle Time | Block Time |
|---|---|
| Down Time | Starve Time |

FIS Database

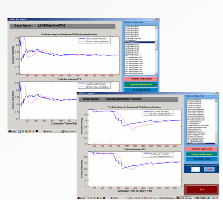
Constraint Analysis Agent

- Update MTBF / MTBR Distributions
- Identify Bottleneck Constraints

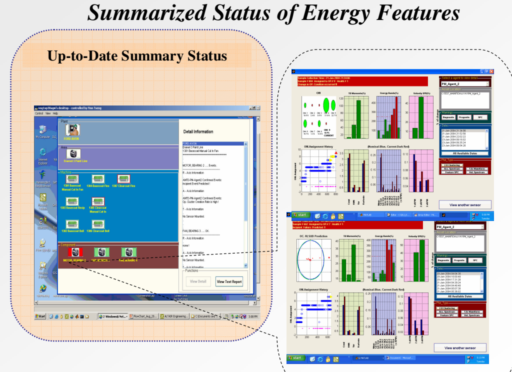


Reliability Prediction Agent

- Update Survival Function Models
- Predict Asset Fault Probabilities



## *Predicted Fault Warning is Combined with a Summarized Status of Energy Features*



**EKF⁺ Learning of Nonlinear Systems**

Consider a static nonlinear system with unknown parameters:

$$d_n = h(x_n)$$

If the analytical form (not the parameters) of the nonlinear input/output mapping *h* is known then assuming slow changing parameters the learning problem can be viewed as estimation of the state of the dynamic nonlinear system:

$$\begin{aligned} x_{n+1} &= x_n \\ d_n &= h(x_n) \end{aligned}$$

The result of this assumption is the EKF learning rule:

$$\begin{aligned} K_n &= P_n H_n (R_n + H_n^T P_n H_n)^{-1} \\ \hat{x}_n &= \hat{x}_{n-1} + K_n [d_n - h(\hat{x}_{n-1})] \\ P_{n+1} &= P_n - K_n H_n^T P_n + Q_n \end{aligned}$$

where $H_n^T = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}_n}$

In a more realistic setting the nonlinear input/output mapping *h* is not known. In this case we can use another Kalman filter (conventional) to learn the Jacobian *H* of the unknown mapping *h*:

$$H_n^T = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}_n}$$

To learn H we assume a linear system with a state vector formed by the elements of H:

$$\begin{aligned} H(k+1) &= H(k) + w(k) \\ \delta d(k) &= H(k)\, \delta h(k) + v(k) \end{aligned}$$

Then the conventional (linear) Kalman filter:

$$H(k) = H(k-1) + L(k-1)\, (\delta d(k) - H(k-1)\, \delta x(k))$$

$$L(k-1) = S(k-1)\, \delta x(k)\, (R + \delta x(k)\, S(k-1)\, \delta x(k))^{-1}$$

$$S(k-1) = L(k-1)\, \delta x^T(k)\, S(k-1)$$

Provides the real time estimation of H that is substituted in the EKF learning rule.