Soft Computing (SC) in the Design of Anomaly Detection Models (AD)

Piero P. Bonissone

Chief Scientist
GE Global Research, Schenectady, NY, USA
Anomaly Detection
Motivation: Prognostics and Health Management (PHM)

Soft Computing: Evolution of a Concept
History (1991-2007)

Applications of SC to Anomaly Detection
Anomaly Detection for Aircraft Engine
- Models Fusion (Categorical & Time-series Data) to reduce false alarms
- Use of EA +FS + AANN to improve model accuracy

Conclusions & References
(1) Anomaly Detection

Motivation: Prognostics and Health Management (PHM)
Why is Prognostics & Health Management (PHM) Important?

**Without PHM**

It’s difficult to know useful life left in an engine.

This leads to a large logistics footprint and higher operating costs.

Conservative estimates are necessary to ensure reliability. Parts replaced while they are still useful. Even with large spares inventory unforeseen problems can cause major fleet disruptions.

---

**With PHM**

Specific engine part conditions are known. Small problems can be addressed before they lead to larger more costly maintenance.

Smaller logistics footprint - lower costs.

Reduced spares inventory. Engine parts replaced only when necessary - reduced maintenance. Engines have longer life with better reliability. Fleet maintenance is more manageable with fewer disruptions.

---

PHM is a major enabler for Condition Based Maintenance (CBM)

**CBM Goals:** Unplanned → Planned Maintenance Events

**PHM Evolution:** Diagnostics → Prognostics → Optimization
Technical Synergy in PHM

Prognostics and Health Management (PHM)

On-board Sensors & Off board Inspections

Anomaly Detection, Diagnostics & Prognostics Alg.

On-board / Off board Optimization Alg.

Visualization & Multi-Criteria Decision-Making Sys.

Imaging devices
PACS servers/workstations

Engines / Aircrafts (fixed & rotary wings)

Turbines / Engines / Motors / Plants / towers

Turbines
Locomotives

GE Healthcare  GE Aviation  GE Energy  GE Oil & Gas  GE Rail

Dr. Piero P. Bonissone © All Rights Retained - KDD4Service 2011
PHM Capabilities and Enabling Technologies

Stage 1: AD + RM&D

Data Acquisition: Remote Monitoring of a fleet of assets legacy fleets use existing instrumentation

Anomaly Detection: Identification of assets deviating from the rest of the fleet (Anomaly detection)

Diagnostics: Root Cause Isolation for each asset exhibiting anomalous behavior
PHM Capabilities and Enabling Technologies

Stage 1: AD + RM&D

Data Acquisition: Remote Monitoring of a fleet of assets
legacy fleets use existing instrumentation

Anomaly Detection: Identification of assets deviating from the rest of the fleet (Anomaly detection)

Diagnostics: Root Cause Isolation for each asset exhibiting anomalous behavior

Stage 2: Prognostics

Prognostics: Prediction of each asset’s remaining useful life (RUL): the decision’s time-horizon.

Unplanned → Planned Maintenance
PHM Capabilities and Enabling Technologies

Stage 1: AD + RM&D

Data Acquisition: Remote Monitoring of a fleet of assets. Legacy fleets use existing instrumentation.

Anomaly Detection: Identification of assets deviating from the rest of the fleet (Anomaly detection).

Diagnostics: Root Cause Isolation for each asset exhibiting anomalous behavior.

Stage 2: Prognostics

Prognostics: Prediction of each asset’s remaining useful life (RUL): the decision’s time-horizon.

Unplanned $\rightarrow$ Planned Maintenance

Stage 3: Control & Optimization

Control / Advisory Generation: Real-time decisions: safety actions, fault accommodation & recovery.

Optimization: Offline decisions: Optimize maintenance actions, supply chain mgmt., production.

Feedback & Learning: Validation of maintenance cases, automated learning of patterns.
PHM Capabilities and Enabling Technologies

**Stage 1: AD + RM&D**

- **Data Acquisition**: Remote Monitoring of a fleet of assets. Legacy fleets use existing instrumentation.
- **Anomaly Detection**: Identification of assets deviating from the rest of the fleet (Anomaly detection).
- **Diagnostics**: Root Cause Isolation for each asset exhibiting anomalous behavior.

**Stage 2: Prognostics**

- **Prognostics**: Prediction of each asset’s remaining useful life (RUL): the decision’s time-horizon. Unplanned → Planned Maintenance.

**Stage 3: Control & Optimization**

- **Control / Advisory Generation**: Real-time decisions: safety actions, fault accommodation & recovery.
- **Optimization**: Offline decisions: Optimize maintenance actions, supply chain mgmt., production.
- **Feedback & Learning**: Validation of maintenance cases, automated learning of patterns.
Developed broad set of PHM algorithms for:

**Anomaly Detection**
- Bearing Anomaly Detection
- Cargo Aircraft

**Diagnostics**
- Trucks
- Aircrafts
- Fault Detection (Boeing)
- Automated Defect Analysis PII
- Smart Wires
- Electrostatic Percipitators

**Prognostics**
- Aircraft Engine
- Ball Bearings
- Evo Locomotives
- Paper Machines

**On-board Fault Accomodation**
- Rapid Fault Detection & Operability Restoration

**Offboard Decision Support System**
- Aircraft Engine Maintenance Optimization
- Logistics Decisions for Aircraft Engines (nominal data)
- Coal Burning Optimization
Developed broad set of PHM algorithms for:

**Diagnostics**
- Trucks
- Aircrafts
- Fault Detection (Boeing)
- Automated Defect Analysis PII
- Smart Wires
- Electrostatic Percipitators

**Prognostics**
- Aircraft Engine
- Ball Bearings
- Evo Locomotives
- Paper Machines

**On-board Fault Accomodation**
- Rapid Fault Detection & Operability Restoration

**Offboard Decision Support System**
- Aircraft Engine Maintenance Optimization
- Logistics Decisions for Aircraft Engines (nominal data)
- Coal Burning Optimization

**PHM Applications at GE**

- Automated interpretation of MRI calibration system test (SPT)
- Remote Locomotive diagnostics & repair recommendation (EOA)
- Automated failure signature extraction
- Diagnostics driven by anomaly detection (work in progress)
- Fusion of parametric and non-parametric fault information
- DSS for pipelines fault identification
- Condition assessment system for detecting and diagnosing ESPs or wires faults/defects
Developed broad set of PHM algorithms for:

- Anomaly Detection
  - Bearing Anomaly Detection
  - C130J
- Diagnostics
  - Trucks
  - Aircrafts
  - Fault Detection (Boeing)
  - Automated Defect Analysis PII
  - Smart Wires
  - Electrostatic Percipitators

**Prognostics**

- Aircraft Engine
- Ball Bearings
- Evo Locomotives
- Paper Machines

On-board Fault Accommodation
- Rapid Fault Detection & Operability Restoration

Offboard Decision Support System
- Aircraft Engine Maintenance Optimization
- Logistics Decisions for Aircraft Engines (nominal data)
- Coal Burning Optimization

**3rd Level Interpretation**

- RUL estimation for engine core components.
- RUL estimation for ball bearing
- Locomotive RUL estimation based on parametric model for health assessment
- Prediction of Time-to-break for paper web
Developed broad set of PHM algorithms for:

Anomaly Detection
  Bearing Anomaly Detection
  C130J

Diagnostics
  Trucks
  Aircrafts
  Fault Detection (Boeing)
  Automated Defect Analysis PII
  Smart Wires
  Electrostatic Percipitators

Prognostics
  Aircraft Engine
  Ball Bearings
  Evo Locomotives
  Paper Machines

**On-board Fault Accommodation**
  Rapid Fault Detection & Operability Restoration

Offboard Decision Support System
  Aircraft Engine Maintenance Optimization
  Logistics Decisions for Aircraft Engines (nominal data)
  Coal Burning Optimization

---

Control / Fault Accommodation

Real-time gain adjustment to restore stall margins in the presence of a fault

On-board Tactical Control
Developed broad set of PHM algorithms for:

- Anomaly Detection
  - Bearing Anomaly Detection
  - C130J

- Diagnostics
  - Trucks
  - Aircrafts
  - Fault Detection (Boeing)
  - Automated Defect Analysis PII
  - Smart Wires
  - Electrostatic Percipitators

- Prognostics
  - Aircraft Engine
  - Ball Bearings
  - Evo Locomotives
  - Paper Machines

- On-board Fault Accommodation
  - Rapid Fault Detection & Operability Restoration

**Offboard Decision Support System**

- Aircraft Engine Maintenance Optimization
- Logistics Decisions for Aircraft Engines
- Coal Burning Optimization

**Optimization**

- Maintenance cost Optimization by:
  - Adjusting workscope policy
  - Adjusting blade repair, inspection, rebuild, and scrap policies

- Logistics Optimization: Mission Reliability, cost, time to repair

- Optimization of Coal-fired boilers (HR/NOx tradeoffs)

**Off-board Strategic Planning**
(2) Soft Computing: Evolution of a Concept

History: 1991-2007
  Offline Meta to design Online Meta and Object models
“In contrast to traditional hard computing, soft computing exploits the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, low solution-cost, and better rapport with reality” (Zadeh 1991)
Soft Computing: Hybrid Probabilistic Systems

Approximate Reasoning

Functional Approximation/Randomized Search

Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms

BBN
D-S
CART
Rand. For.

Probability of Fuzzy Events
Belief of Fuzzy Events
Fuzzy Influence Diagrams
Fuzzy Random Forest
Evolved BBN
Soft Computing: Hybrid FL Systems

Approximate Reasoning

Probabilistic Models

Multivalued & Fuzzy Logics

Fuzzy Systems

Fuzzy Logic Controllers

Multivalued Algebras

Functional Approximation/Randomized Search

Neural Networks

Evolutionary Algorithms

HYBRID FL SYSTEMS

NN modified by FS (Fuzzy Neural Systems)

FLC Tuned by NN (Neural Fuzzy Systems)

FLC Generated and Tuned by EA

Dr. Piero P. Bonissone © All Rights Retained - KDD4Service 2011
Soft Computing: Hybrid NN Systems

Approximate Reasoning

Probabilistic Models
Multivalued & Fuzzy Logics

Functional Approximation/Randomized Search

Neural Networks
Evolutionary Algorithms

HYBRID NN SYSTEMS

NN parameters (learning rate $\eta$, momentum $\alpha$) controlled by FLC

NN topology &/or weights generated by EAs
Soft Computing: Hybrid EA Systems

- Probabilistic Models
- Multivalued & Fuzzy Logics
- Neural Networks
- Evolutionary Algorithms

Approximate Reasoning

Functional Approximation/Randomized Search

Evolution Strategies
Evolutionary Programs
Genetic Algorithms
Genetic Progr.

EA parameters (N, P_cr, P_mu) controlled by FLC
EA-based search inter-twined with hill-climbing
EA parameters (Pop size, select.) controlled by EA

HYBRID EA SYSTEMS

EA parameters (N, P_cr, P_mu) controlled by FLC
EA-based search inter-twined with hill-climbing
EA parameters (Pop size, select.) controlled by EA
Hybrid Soft Computing (H-SC): Two Level Modeling

**SC term is coined by Zadeh [1-2]**

**SC concept is formalized [3]**

**Synergy of Hybrid SC is illustrated [4]**

**Hybrid SC is presented as a 2-level modeling: meta & object level [5]**

**Two Level modeling: Meta- & Object- level**

**Representing Meta- & Object-level knowledge**

**Generalizing Meta-Heuristics to search for object-models**

**Family of Meta-Heuristics (MH): searching for object-models [7]**

**Offline MH (design) Online MH (control, fusion) Object models (probl. Solv.) [8]**


**2003**

**2006**

**2008**

**2010**

**Hybrid SC: Synergy between Reasoning & Search**

**Association of Components**

**Hybrid SC: Synergy between Reasoning & Search**

- **Performance of Object-level PS**
- **Parameters of Object-level PS**
- **Suite of Representative Problems**
- **Off-line Tuning**
- **Run-time Environment**
- **Run-time Parameters for Object-level PS**

**KB**

- **State Variables: Performance of Object-level PS**
- **Control Variables: Modified parameters for Object-level PS**

**Object-level Problem**

**Object-level Problem Solver (PS)**

**Off-line KB definition**

**Controller**

**Object-level PS**

**Run-time Environment**
Offline Meta-Heuristics:
EA generates Structure & Parameters

Object-level Problem Solver (PS)

Meta-Level PS

Parameters of Object-level PS

Performance of Object-level PS

Suite of Representative Problems

Off-line Tuning

Run-time Parameters for Object-level PS

Run-time Environment

Object-level Problem Solver (PS)

Object-level Problem
Examples of Offline Meta-Heuristics

**Meta- Level PS**

- **Control Problem**
  - EA
- **Prediction Problem**
  - NN
- **Control Problem**
  - EA
- **Optimiz. Problem**
  - EA
- **Optimiz. Problem**
  - EA
- **Classif. Problem**
  - EA
- **Classif. Problem**
  - EA

**Object- Level PS**

- **Control Problem**
  - FLC
  - Tuning FLC Parameters
- **Prediction Problem**
  - FS
  - Tuning FS Parameters
- **Control Problem**
  - Controller
  - Tuning Gain Schedule Parameters
- **Optimiz. Problem**
  - NN
  - Tuning NN Parameters
- **Optimiz. Problem**
  - EA
  - Tuning EA Parameters
- **Classif. Problem**
  - BBN
  - Evolving Tuning Bayesian Classifiers
- **Classif. Problem**
  - F-IBM
  - Tuning Fuzzy Instance Based Classifiers

**Generation and tuning of fuzzy rule-based systems for classification, control, fusion, etc.**

- EA: Pragmatics
- FS: Semantics

**Tuning of Evolutionary Algorithm Parameters**

- EA: Pragmatics
- FS: Semantics

**Generation and tuning of fuzzy instance- or case-based systems for classification, prediction, etc.**

- EA: Pragmatics
- FS: Semantics
Example: Use of EA to Design a Classifier [using a Wrapper Approach]

**Evolutionary Algorithm**

- **Individual Decoder**
- **Mutation**
  - Uniform Mutation
  - Gaussian Mutation
  - Original
- **Elitist** (best from Pop i)
- **Pop.(i+1)**

**Fitness Function:** Precision

\[ f = \frac{TP}{TP + FP} \]

**Classifier Evaluation**

- **Ground Truth**
  - T
  - F
- **Classifier Output**
  - TP
  - FP
  - FN
  - TN

- **Leave-One-Out Testing**
- **XML Config File**
- **Classifier Instance**
On-Line Meta-Heuristics: KB Controller for Object-Level Problem Solver

State Variables: Performance of Object-level PS

Control Variables: Modified parameters for Object-level PS

Off-line KB definition

Controller

Object-level PS

Object-level Problem
Hybrid Soft Computing (H-SC): Two Level Modeling

- **SC term** is coined by Zadeh in 1991
- **SC concept** is formalized in 1994
- **Hybrid SC** concept is illustrated in 1997
- **Hybrid SC** is presented as a 2-level modeling: meta & object level in 1999
- **Hybrid SC** is represented as a 2-level modeling: meta & object level knowledge in 2003

**Examples of On-Line Meta-Heuristics**

**Meta- Level PS**
- FLC
- Optimization

**Object- Level PS**
- FLC
- ANFIS
- EA

**Domain Knowledge**
- Transactional
- Diagnostic
- Prognostics
- Control

**Time Horizon**
- One Shoot
- Tactical
- Operational
- Strategic
- Lifecycle
Hybrid Soft Computing (H-SC): Family of MH’s

Approximate Reasoning
- Probabilistic Models
- Multi-valued & Fuzzy Logic

Functional Approximation / Randomized Search
- Neural Networks
- Meta-heuristics

Hybrid Models
- MH-1
- MH-2
- MH-3
- MH-4

MH-1 = Evolutionary MH
MH-2 = Relaxation MH
MH-3 = Search MH (Individual Search, Cooperative Multiple Search, Non-Cooperative Multiple Search)

Generalized MH’s to search for object-models

SC term is coined by Zadeh 1991
SC concept is formalized [1-2] 1994
Hybrid SC Concept is formalized [3] 1997
Synergy of Hybrid SC is illustrated [4] 1999
Hybrid SC is presented as a 2-level modeling: meta & object level [5] 2003

Two Level modeling: Meta- & Object- level
Representing Meta- & Object- level knowledge


Offline MH to design
Online MH + object-models
Offline MH (design)
Online MH (control, fusion)
Object models (probl. Solv.) [8] 2010
Soft Computing: Evolution of a Concept


Offline Meta to design Online Meta and Object models
Hybrid Soft Computing (H-SC):
On-line MH for Model Design; Off-line MH for Model Control; Object Model for Problem Solving

1991 SC term is coined by Zadeh
1994 SC concept is formalized [1-2]
1997 Hybrid SC Concept is formalized [3]
1999 Synergy of Hybrid SC is illustrated [4]
1999 Hybrid SC is presented as a 2-level modeling: meta & object level [5]
2003 Two Level modeling: Meta- & Object-level [1-2]
2006 Representing Meta- & Object-level knowledge [7]
2006 Generalizing Meta-Heuristics to search for object-models
2008 Offline MH to design Online MH + object-models
2008 Offline MH (design) Online MH (control, fusion) Object models (probl. Solv.) [8]
2010 Two Level modeling: Meta- & Object-level

Offline MH’s
- Design
  - Object-level (local) Problem Solver
- Run-time
  - Object-level (local) Problem Solver

Online MH’s
- Run-time
  - Design
  - Offline MH’s

Hybrid Soft Computing (H-SC):
On-line MH for Model Design; Off-line MH for Model Control; Object Model for Problem Solving
## SC Techniques for Offline MH’s, Online MH’s, and Object-level Models

<table>
<thead>
<tr>
<th>Problem Instance</th>
<th>Problem Type</th>
<th>Model Design (Offline MH’s)</th>
<th>Model Controller (Online MH’s)</th>
<th>Object-level models</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anomaly Detection (System)</td>
<td>Classification</td>
<td>Model T-norm tuning</td>
<td>Fuzzy Aggregation</td>
<td>Multiple Models: SVM, NN, Case-Based, MARS</td>
<td>[24]</td>
</tr>
<tr>
<td>Anomaly Detection (System)</td>
<td>Classification</td>
<td>Manual design</td>
<td>Fusion</td>
<td>Multiple Models: Kolmogorov Complexity, SOM. Random Forest, Hotteling T2, AANN</td>
<td>[25, 26]</td>
</tr>
<tr>
<td>Anomaly Detection (Model)</td>
<td>Classification &amp; Prediction</td>
<td>EA tuning of fuzzy supervisory termset</td>
<td>Fuzzy Supervisory</td>
<td>Multiple Models: Ensemble of AANN’s</td>
<td>[27, 28]</td>
</tr>
<tr>
<td>Insurance Underwriting: Risk management</td>
<td>Classification</td>
<td>EA</td>
<td>Fusion</td>
<td>Multiple Models: NN, Fuzzy, MARS,</td>
<td>[29, 30]</td>
</tr>
<tr>
<td>Load, HR, NOx forecast</td>
<td>Prediction</td>
<td>Multiple CART trees</td>
<td>Fusion</td>
<td>Multiple Models: Ensemble of NN’s</td>
<td>[31, 34]</td>
</tr>
<tr>
<td>Aircraft engine fault recovery</td>
<td>Control/Fault Accommodation</td>
<td>EA tuning of linear control gains</td>
<td>Crisp supervisory</td>
<td>Multiple Models (Loop): SVM + linear control</td>
<td>[14]</td>
</tr>
<tr>
<td>Power plant optimization</td>
<td>Optimization</td>
<td>Manual design</td>
<td>Fusion</td>
<td>Multiple Models (Loop): MOEA + NN’s</td>
<td>[32, 33, 34]</td>
</tr>
<tr>
<td>Flexible mfg. optimization</td>
<td>Optimization</td>
<td>Manual design</td>
<td>Fuzzy Supervisory</td>
<td>EA</td>
<td>[10, 35]</td>
</tr>
</tbody>
</table>
(3) Applications of SC to Anomaly Detection (for Aircraft Engine)

Fusion of Models (Categorical & Time-series Data) to reduce false alarms

- Use of EA +FS + AANN to improve model accuracy
Anomaly Detection (Fusion)

<table>
<thead>
<tr>
<th>Problem Instance</th>
<th>Problem Type</th>
<th>Model Design (Offline MH’s)</th>
<th>Model Controller (Online MH’s)</th>
<th>Object-level models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anomaly Detection</td>
<td>1-class Classification</td>
<td>Manual</td>
<td>Fusion</td>
<td>Multiple Models: Kolmogorov Complexity, SOM, Random Forest, Hotelling T2, AANN</td>
</tr>
</tbody>
</table>

Anomaly Detection (AD) using both Parametric and Categorical Data Sources

Categorical data sources

Parametric data sources

Leverage ALL the information you have BEFORE the flight

Source: Integrated System Health Management (ISHM) and Software Technologies Support, P. Bonissone, N. Eklund, N. Iyer, H. Qiu and A. Varma, 2007GRC928, November 2007, Public (Class 1)
Fusion within Anomaly Detection Modules

Categorical data sources

- CMC Control
- FADEC Control

Parametric data sources

- Platform #
- Download #
- Flight #
- Date & Time
- Fault codes

- Information Theory (Kolmogorov) AD
- Neural Net (SOM) AD
- Statistics (RF) AD

- Platform #
- Download #
- Flight #
- Date & Time
- Real-Valued timeseries

- Spearman Correlation AD
- Hotteling T2 AD
- Auto Associative NN (AANN) AD

Outputs Fusion
Anomaly Detection Using Information Theory

Detector based on Kolmogorov Complexity
- 2D Display obtained by projecting 84-dimensional object while minimize distorsion

Clear Identification of anomalies

FADEC Messages
Frequency extraction
String Encoding
NCD Matrix Computation
Labeling Normal and Anomalous Regions
Visualization: 1D or 2D Anomaly Projection

Information Theory Detector

NCD Matrix

Flight \( x \)

\[
NCD(x, y) = \frac{C(x,y) - \min(C(x),C(y))}{\max(C(x),C(y))}
\]

Distance Matrix of Pairwise NCD

Visualization:
1D or 2D Anomaly Projection

Labeling Normal and Anomalous Regions

NCD Matrix Computation

String Encoding

Frequency extraction

FADEC Messages
Anomaly Detection Algorithms Using Self-Organizing Maps (AI)

1. Train SOM on normal data to obtain normal operating envelope
2. Declare a case novel if its projection to the map falls outside the envelope

Flights known to have failures (Red circles) generate trajectories that pass through common region

Identification of “Precursor Zone”
Anomaly Detection Algorithms Using Random Forest (Statistics) [Breiman & Cutler]

- Random Forest Computation
- Multi Dimensional Scaling (MDS) projection
- Minimal Spanning Tree (MST)
- Distance threshold to segment MST

.Data:
2449 possible fault codes
131 active over 84 flights

RF:
30 forests with new synthetic data
500 trees per forest
OOB error rate (7%)
Anomaly Detection Algorithms Using Random Forest (Statistics) [Breiman & Cutler]

- Random Forest Computation
- Multi Dimensional Scaling (MDS) projection
- Minimal Spanning Tree (MST)
- Distance threshold to segment MST

Statistics Detector

DATA:
2449 possible fault codes
131 active over 84 flights

RF:
30 forests with new synthetic data
500 trees per forest
OOB error rate (7%)

Anomaly Detection Algorithms

Using Random Forest (Statistics)

Robust Identification of Anomalies

Units with Large Anomalies
Units with Medium Anomalies

Flights known to have failures (Red circles) aren’t connected to main body of Minimum spanning tree!
Fusion of Anomaly Detection Algorithms based on categorical data

Fusion of error-uncorrelated detectors increases robustness & accuracy

Source: Integrated System Health Management (ISHM) and Software Technologies Support, P. Bonissone, N. Eklund, N. Iyer, H. Qiu and A. Varma, 2007GRC928, November 2007, Public (Class 1)
Hotteling T2 Score

Definition

The Hotteling T-square statistic, $t^2$, is a generalization of Student's t statistic that is used in multivariate hypothesis testing.

Properties

The Hotteling T2 metric provides great sensitive to small drifts, at the price of increased computational complexity.

Computation

Given a group of variables $x = (x_1, x_2, \ldots, x_p)$ with mean $\mu = (\mu_1, \mu_2, \ldots, \mu_p)$, and covariance matrix $W = \sum(x - \mu)(x - \mu)'/(n-1)$

Then, the t-square statistic is computed as

$$t^2 = (x - \mu)' W^{-1} (x - \mu)$$

Note that $t^2$ is also closely related to squared Mahalanobis distance
AD Fusion: Output fusion for Categorical & Parametric Models

Normalized Raw Measurement

<table>
<thead>
<tr>
<th>Platform #</th>
<th>Download #</th>
<th>Date &amp; Time</th>
<th>Fault codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Theory (Kolmogorov) AD</td>
<td>Neural Net (SOM) AD</td>
<td>Statistics (RF) AD</td>
<td></td>
</tr>
</tbody>
</table>

Outputs Fusion

Real-Valued timeseries

<table>
<thead>
<tr>
<th>Platform #</th>
<th>Download #</th>
<th>Date &amp; Time</th>
<th>Real-Valued timeseries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spearman Correlation AD</td>
<td>Hotteling T2 AD</td>
<td>Auto Associative NN (AANN) AD</td>
<td></td>
</tr>
</tbody>
</table>

Sources: (1) Integrated System Health Management (ISHM) and Software Technologies Support, P. Bonissone, N. Eklund, N. Iyer, H. Qiu and A. Varma, 2007GRC928, November 2007, Public (Class 1)
AD Fusion: Output fusion for Categorical & Parametric Models

Sources:
1. Integrated System Health Management (ISHM) and Software Technologies Support, P. Bonissone, N. Eklund, N. Iyer, H. Qiu and A. Varma, 2007GRC928, November 2007, Public (Class 1)
Fusion of all Anomaly Detection Algorithms

FADEC / CMC Messages
Sensors
Utilization/Operations
Maintenance Actions
Parts Orders DB's

Information Theory Detector

AI Detector

Statistics Detector

Statistics
SRC or Hotteling T2 Detectors

Engineering - AI
Residual Analysis using
Physics-based models and AANN

FUSION OF ANOMALY DETECTORS

ROC Curves

Fusion of error-uncorrelated detectors increases robustness & accuracy

Source: Integrated System Health Management (ISHM) and Software Technologies Support, P. Bonissone, N. Eklund, N. Iyer, H. Qiu and A. Varma, 2007GRC928, November 2007, Public (Class 1)
(3) Applications of SC to Anomaly Detection
(for Aircraft Engine)

- Fusion of Models (Categorical & Time-series Data) to reduce false alarms

Use of EA +FS + AANN to improve model accuracy
Anomaly Detection (Model Improvement)

Operational Envelope ➔ Engine Physics-based Simulator ➔ Sensor Data ➔ Operational State Vector ➔ Run-Time Anomaly Detection Model

**Object Models: AANN’s**

**Fuzzy Supervisory Rule Set**

<table>
<thead>
<tr>
<th>RULES</th>
<th>Altitude</th>
<th>Amb. Temp.</th>
<th>Mach #</th>
<th>Model #</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>AANN-1</td>
</tr>
<tr>
<td>R2</td>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
<td>AANN-2</td>
</tr>
<tr>
<td>R3</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>AANN-3</td>
</tr>
</tbody>
</table>

**Fuzzy Supervisory Term Set**

**Offline MH: Evolutionary Algorithm**

- Individual in EA population defines Fuzzy Sup. Termset
- Fuzzy Supervisory interpolates among AANN’s using termset
- Compute residuals between nine simulated sensors & interpolated AANN’s output
- Compute Fitness Function based on aggregate of nine sensor residuals
- Evolutionary Algorithm based on Fitness Function

<table>
<thead>
<tr>
<th>Problem Instance</th>
<th>Problem Type</th>
<th>Model Design (Offline MH’s)</th>
<th>Model Controller (Online MH’s)</th>
<th>Object-level models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anomaly Detection</td>
<td>1-class Classification</td>
<td>EA tuning of fuzzy supervisory termset</td>
<td>Fuzzy Supervisory</td>
<td>Multiple Models: Ensemble of AANN’s</td>
</tr>
</tbody>
</table>

Sources of Anomalies

**Transients**
- Operator / Pilot
- Reference Generator
- Operational Log Generator

**Assuming perfect controls**
- Controller Design Process
- Controller
- Actuator
- State Estimator
- Event/Message Log Generator
- Event Log Generator Design Process

**System Failures**
- Sensor Failures
- Model Inadequacy

**Monitoring**
- Anomaly Detection Model
- Real-valued, time-stamped data
- Categorical, time-stamped data

**Operations**
- Operational Log Generator Design Process

**Control**
- Reference Generator Design Process

**Event Log Generator Design Process**
Physics-Based Simulation

– **CLM**: Component Level Model is a physics-based thermodynamic model widely used to simulate the performance of a commercial aircraft engines.

– **Flight Regime**: Flight conditions, such as altitude, Mach number, ambient temperature, and engine fan speed, and a large variety of model parameters, such as module efficiency and flow capacity are inputs to the CLM.

– **Outputs**: CLM’s outputs are the values for *pressures, core speed and temperatures* at various locations of engine, which simulate sensor measurements.

– **Noise**: Realistic values of sensor noise can be added after the CLM calculation.
Basic AD Model: Auto-Associative Neural Network

Rationale
The Auto-Associative Neural Network (AANN) leverages covariance information like other approaches (SRC and T2). The AANN also produces sensor estimated values to replace the ones generated by faulty sensors. This approach provides a better discrimination between sensor faults and system component faults.

Definition/Properties
AANN computes the largest Non-Linear Principal components (NLPCA) - the nodes in the intermediate layer – to identify and remove correlations among variables.

NLPCA uncover both linear and nonlinear correlations, without restriction on the type of the nonlinearities present in the data.

Computation
Traditional NN training with back-propagation

Variable Contribution
Residuals magnitude/distribution
Experiments with Simulated GE90 Aircraft Engines

- Experiment Setup
- Segmentation of the Operating Space
- Experiments
  1\textsuperscript{st} - 3 local models
  2\textsuperscript{nd} - 1 Global Model
  3\textsuperscript{rd} - 3 local Models + Supervisory Model
Experiment Setup

Control

Dynamic System

Sensors

Real-valued, time-stamped data

Monitoring/Simulation

Anomaly Detection

OK Abnormal

Type of Anomaly (system, sensor)

Time of Anomaly

Anomaly Severity

Raw Sensor Measurement

Sensor Estimation

Residual Analysis (Aggregate Measure)
Segmentation of the Operating Space
Three regions in the Flight Envelops

![Graphs showing three regions in the flight envelopes.](image)
Experiments

Experiments Settings
– We used a steady state CLM model for a commercial, high-bypass, twin-spool turbofan engine.
– We can manipulate flight conditions to simulate different operation regimes (i.e. flight envelopes of aircraft) and generate data corresponding to them

1st Experiment
Three AANN’ s: One for each region in the flight envelop (region)
Vary ALT, Mach and Tamb ->1000 normal operating pts for each region
Run each operation point through CLM to generate a 9x1 sensor vector
900 points for training (200 for validation); 100 points reserved for test
Each local model performs very well (better than global model) in region of competence, and performs poorly outside its limited scope

2nd Experiment
One Global AANN
Train on same 2700 training data points from experiment 1
Run each operation point through CLM to generate a 9x1 sensor vector
Test on the left 300 points
Global model performs fair across all three regions - shows higher variance than each local AANN operating within its scope

3rd Experiment
Three AANN’ s: One for each region in the flight envelop
Fuzzy Supervisory Model (FSM) to interpolate among local AANN’ s
Experiment 1

• Vary ALT, Mach and Tamb ->1000 normal operating pts for each flight envelop
• Run each operation point thru CLM to generate a sensor vector (9x1)
• Three AANN’s: One for each region in the flight envelop
• 900 points for training (200 for validation); 100 points reserved for test

Goal: Create three local models
Results: High performance when in scope inadequate performance when out of scope
Raw Data from Flight Env 1

- **ZPONG**: Variations from 98 to 101
- **P**: Variations from 120 to 140
- **T**: Variations from 5 to 30
- **T3**: Variations from 1140 to 1340
- **T4**: Variations from 1700 to 1740
- **T5**: Variations from 1140 to 1220
- **ZNGV**: Variations from 75 to 79
- **ZMDC**: Variations from 6000 to 9000
- **VBVPOS**: Variations from 12 to 18
Residuals: test set from FE1 on AANN1

Correct Scope of local model: Small Residuals!
Residuals: test set from FE2 on AANN1

Incorrect Scope of local model: Larger Residuals (10x)
Experiment 2

- One global AANN
- Train on the 2700 training data points from experiment 1
- Test on the left 300 points

Goal: Create **one Global model**
Results: Mediocre performance across entire space – better than worse performance of local models
Test data from FE1

Increased variance (3x) compared to experiment
3rd Experiment
- Three AANN’s: One for each region in the flight envelop
- Fuzzy Supervisory Model (FSM) to interpolate among local AANN’s

Simulate the change of flight conditions
  FE1: 200 pts
  FE1 → FE2: 200 pts
  FE2: 200 pts
  FE2 → FE3: 200 pts
  FE3: 200 pts

Test the simulated data on the Fuzzy Supervisory Model
+ AANN1, AANN2, AANN3

Intentionally making transitions in the space not covered by any pre-trained flight envelop
Hierarchical structure performs very well across all regions – including transitions

Goal: Create a Fuzzy Supervisory for three local models
Results: Higher performance across all regions
Flight Envelop Transitions

Operating Regime Transition

Operating Space

Crisp Model-Transition

Fuzzy Model-Transition
Transition Management Using Fuzzy Supervisory Model

AANN Interpolation by Fuzzy Supervisory

Residuals Generation
\[ R_i(AANN_j) = \sum_{j=1}^{3} w_j \times (I_i) \]

Network Implementation (ANFIS-like)

Figure Of Merit (FOM)

\[ FOM = \sqrt{\frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \left( \frac{R_{ij}}{X_i} \right)^2} \]

- \( n \) is the number of the variables (sensors)
- \( m \) is the number of data points (measurement)
- \( R_{ij} \) is the residual between true measurement and AANN estimation,
- \( X_i \) is the mean of the true measurement
Residuals for each AANN and for hierarchical system (with FSM)

FSM provides great transition management across regions
Design Tuning

- Design Choices in the Fuzzy Supervisory Model (FSM)
- Tuning the Fuzzy Supervisory Model
  - Manual tuning of FSM State Partitions
  - Automated tuning of FSM State Partitions
Manual tuning, extending AANN1’ s scope, lead to a 25% FOM improvement.

We could use FOM for gradient or evolutionary parametric tuning.
Automated FLS Tuning with an EA using a Wrapper Approach

Operational Envelope -> Engine Physics-based Simulator

Operational State Vector

Sensor Data

Run-Time Anomaly Detection Model

Fuzzy Supervisory System

Residual Analysis

Fuzzy Supervisory Rule Set

<table>
<thead>
<tr>
<th>RULES</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>High</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>R2</td>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>R3</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>

Fuzzy Supervisory Term Set

Evolutionary Algorithm

Tuning the FS termset in a wrapper approach

Individual in EA population defines Fuzzy Sup. Termset

Fuzzy Supervisory interpolates among AANN’s using termset

Compute residuals between nine simulated sensors & interpolated AANN’s output

Compute Fitness Function based on aggregate of nine sensor residuals

Evolutionary Algorithm based on Fitness Function
Automated FLS Tuning: Encoding Trapezoidal Membership functions

Encoding the abscissa of the slope intersections ($x_i$) and the lengths of the bases of each triangle ($L_i$) as an individual in the Evolutionary Algorithm population.
Evolutionary Search for Tuning a Fuzzy Supervisory System using a Wrapper Approach

\[ \sum_{i}^{n} \sum_{m}^{n} \frac{R_{ij}}{X_i} \]

Fitness Function: FOM

Evolutionary Algorithm

Pop Size = 500 individuals
GenMax = 1,000 generations
Automated FLS Tuning: Membership function parameters

Meta-Heuristic Tuning

Use **Global Tuning**
(based on FOM fitness function and Genetic Algorithm)
to further improve results

Note: Magnified scale to enhance comparison

- **FOM = 7.25**
- **FOM = 6.80**
Most residual errors occur in the [200, 600] interval, indicating a performance limit that cannot be addressed only by tuning the FLS. Rather it suggests the need for an additional AANN-4 to provide better coverage in that region.
Design Tradeoffs

Model Complexity

Model Accuracy (- FOM)

- Single Global AANN
- Multiple Local AANN’s - simple model switch
- Multiple Local AANN’s + FLS
- FLS Trapezoid parameters tuned by GA*
- Manual design of additional Local AANN (AANN-4) +
  GA tuning of FLS using GBF & T-norms** parameters

* Chromosome: \([x_1,\ldots,x_5,L_1,\ldots,L_5]\)

** Chromosome: \([([a_{11},b_{11},c_{11}],\ldots,([a_{13},b_{13},c_{13}],\ldots,([a_{n3},b_{n3},c_{n3}],p)]

Skip to end
Future Work

• Hierarchical Design (to Improve Accuracy and Extend Region of competence)
  + Used Offline Metaheuristics (EA) and Online Metaheuristics (FLS) with AANN model
  - Use a more complex encoding for the EA individual to evolve BOTH structure and parameters:
    # AANN Models
    Scope of AANN Models
    Evolve membership Functions (GBF) in FLS
    Evolve Aggregation operators (parameterized T-norms)

• Model Lifecycle (to maintain model Vitality)
  - Use Offline Metaheuristics (EA) to create/retune hierarchical design with updated data sets (e.g. reflecting more recent engine degradation)
(4) Conclusions & References
Summary

• Role of Anomaly Detection in PHM
• Modeling with SC: Combining Domain Knowledge with Field Data
• SC Evolution (1991-2010)
  – Association
  – Symmetric Hybrid Systems (Reasoning & Search)
  – Structured Hybrid Systems (Meta- & Object- Level Reasoning)
  – Offline MH and On-Line MH
  – Offline MH (model design); On-Line MH (models control or fusion); Object-models (problem solving)
• Model Lifecycle Management
  – Use Offline MH to design and update the On-line MH and Object-level models
• Applications of SC to Anomaly Detection for Aircraft Engine
  – Anomaly Detection (System): Kolmogorov, SOM, RF, Hotteling T2, AANN + Fusion
  – Anomaly Detection (Model): EA +FS + AANN
• Other SC Applications to Classification and Prediction (not covered)
  – Classification Digital Underwriting (EA +FS)
  – Prediction Power Plant Management (EA + CART + Fusion + NN)
• Hybrid SC allows to easily integrate a broad set of techniques for leveraging knowledge and data
Hybrid Soft Computing (H-SC): A Personal Timeline

REFERENCES


Questions?

Comments?