Combining Data-Driven and Model-Based Methods to Improve Diagnosis of Complex Systems*

Gautam Biswas

Cornelius Vanderbilt Professor of Engineering

Professor of Computer Science, Computer Engineering

gautam.biswas@vanderbilt.edu

EECS Dept

Institute for Software Integrated Systems (ISIS)

Vanderbilt University, Nashville, TN

* *Funded by NASA Aviation Safety Program* (NASA NRA NNL09AA08B), *STTR programs, NSF,* and *internal funds* (*Smart Cities*)

Acknowledge

- Past & Present Graduate Students
  - Daniel Mack
  - Hamed Khorasgani
  - Benjamin Stadnick
  - Avisek Naug
  - Ibrahim Ahmed
- All of my colleagues at NASA, Honeywell, Vanderbilt U, other academics

Outline of Talk

- Model and Data-driven methods for diagnosis & anomaly detection
  - o Pros and cons
- Data-driven methods
  - o Three case studies
    - Improving diagnosers using classifiers
    - Anomaly detection – discovering new faults
    - Combining model- and data-driven diagnosis

      Also called hybrid diagnosis
  - o What do we learn from these approaches?
    - How well do they generalize?
    - Can we come up with a systematic framework for combining hybrid (model and data driven) diagnosis?

Situating the presentation

- Model-based
  - o Physics-based models
  - o Equations/graphical representations
  - o Residual Analysis, Consistency-based methods

- Statistical Methods (Data-driven)
  - o PCA, PLS, ICA, and their variations

- AI-based methods (Data Driven)
  - o Learning new knowledge from the data
    - New features for classifying faults, better thresholds
    - Finding new faults ….

Model- versus Data-Driven Diagnosis Approaches

- Model-based Approaches
- Pros
  - Formal representations
  - Automated analysis (reasoners)
  - Verification & Validation
- Cons
  - Models necessarily incomplete, sometimes unavailable
  - Have finite shelf life
  - Typically don't account for decision making situations with humans in the loop

- Data-driven Approaches
- Address limitations of model-driven approaches
  - Augment models with historical data
  - Continued monitoring, data collection provides up-to-date behaviors of system
  - Can include human-system interaction data
  - Can account for changing environments & specific scenarios of interest

Context: Complex Systems (Vehicles, Industrial Plants, Power Plants)

Data Driven Approaches Challenges

- Data Acquisition
  - Heterogeneous
  - Synchronizing distributed data collected at different rates
- Data Storage & Retrieval, Curation, Preprocessing
  - Distributed or centralized?
  - Increasingly non relational (noSQL)
  - Problem-driven curation and pre-processing
- Analytics & Machine Learning
  - Fit Data to Problem
  - What is the right algorithm to use?
  - Offline versus online analysis
- Feedback & Control
  - How are they affected as we incorporate new patterns into monitoring system?
  - Human Machine interfaces

Real world systems & scenarios

My Work
Three Case Studies

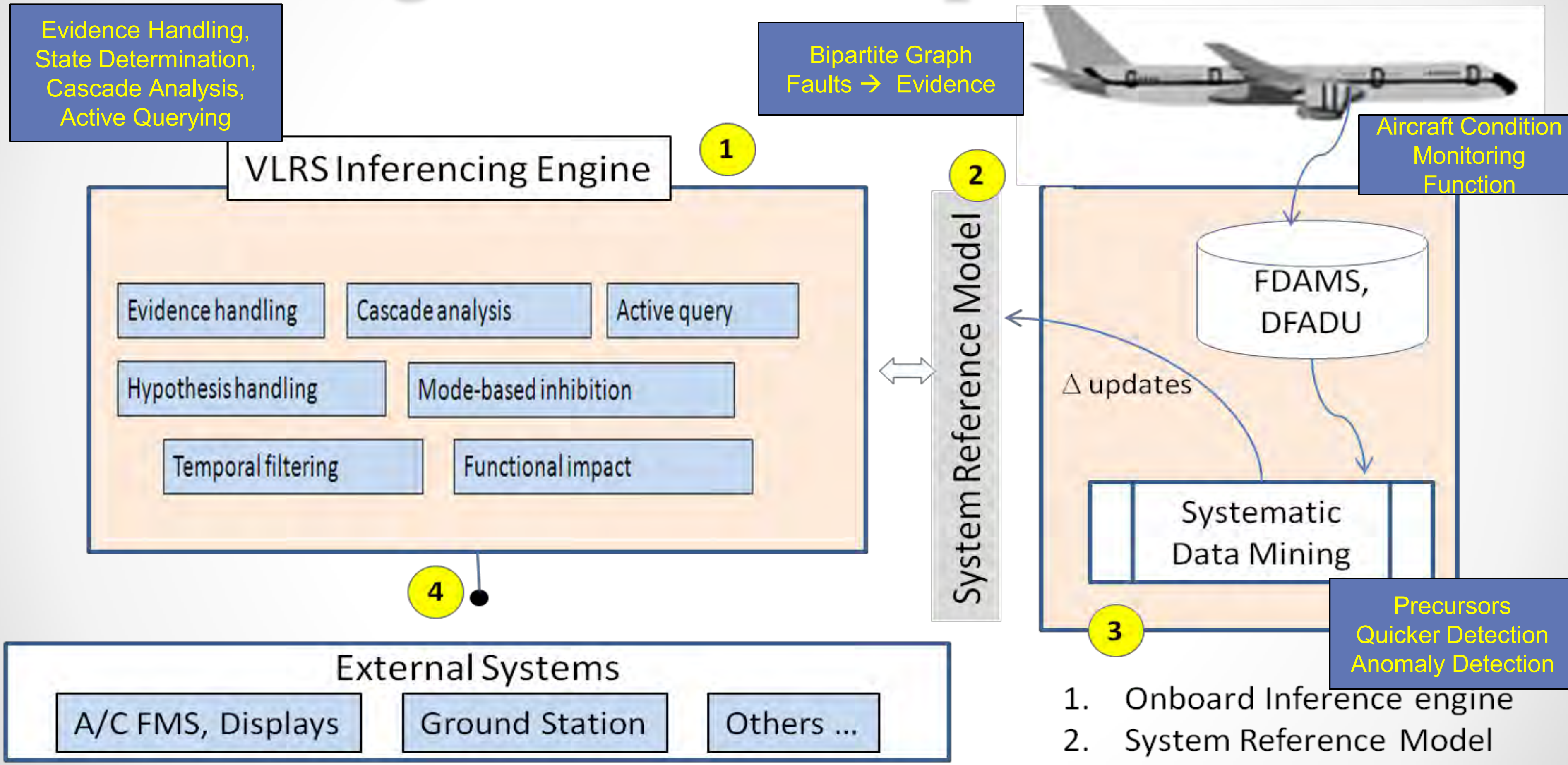
- Improving Diagnosers of Aircraft Systems
  - Original diagnoser – designed by human expert
- Anomaly Detection for discovering new faults in aircraft flight
  - Augmenting existing diagnosers with new faults
- Combined Model- and Data-Driven Diagnosis
  - Diagnosis from residuals – generated from physics models then augmented with additional residuals generated from data

All of these studies work with real data, and involve experts in the loop

Improving Diagnoser Functionality

Mack, D. L., Biswas, G., Koutsoukos, X. D., & Mylaraswamy, D. (2017). Learning Bayesian Network Structures to Augment Aircraft Diagnostic Reference Models. *IEEE Trans. Automation Science and Engineering*, *14*(1), 358-369.

Case Study 1
Single Aircraft Operations



1. Onboard Inference engine
2. System Reference Model
3. Offline learning loop
4. Comm. Interface protocols

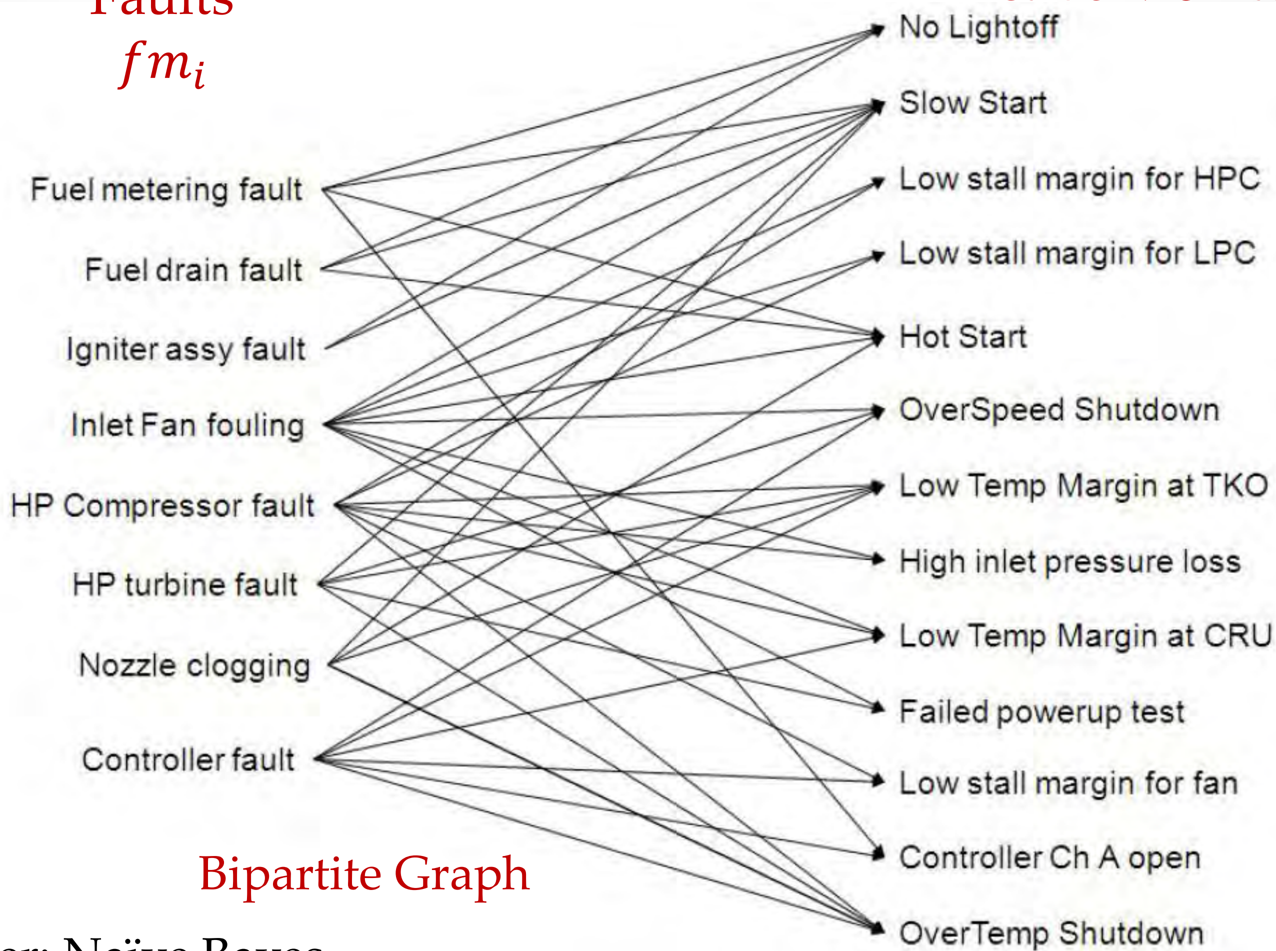
Learned Knowledge → Improved Detection & Isolation

Felke, 1994

Example Reference Model

Faults fm_i

Active Monitors (m_j)



Bipartite Graph

Reasoner: Naïve Bayes

$$P(fm_i \mid m_i, m_j, \ldots, m_k) = \alpha \times P(m_i \mid fm_i) \times P(m_j \mid fm_i) \times \ldots\ldots \times P(m_k \mid fm_i)$$

and $fm_1, fm_2, \ldots, fm_n$ do not interact

Improving Diagnoser

- Regional Airline Data
  - Several Aircraft over Several Years
  - 182 Sensors at Different Sampling Rates
  - Varying Flight Durations(Minutes to Hours)
  - In Binary form (DAR files): Up to 12MB per flight
  - Clean vs. Corrupt
- Trail of Data
  - Extract information into usable form (Data Warehouse)
    - Build Database (Data Warehouse)
    - 12 Tail Numbers for 4 engine aircraft
    - Flight times for each (up to 5 flights per day; include short hops)
    - Multiple years > 6000 Flights
    - Multiple Fault Annotations

Data Transformation

Sensors on Aircraft → Condition Indicators → Diagnostic Monitors

Raw Parameters
Engine 1 Speed
Engine 2 Speed
Engine 3 Speed
Engine 4 Speed
Core Speed Engine 1
Core Speed Engine 2
Core Speed Engine 3
Core Speed Engine 4
Air Temperature
Engine 1 Exaust Gas Temperature
Engine 2 Exaust Gas Temperature
Engine 3 Exaust Gas Temperature
Engine 4 Exaust Gas Temperature
Flight Phase
Altitude

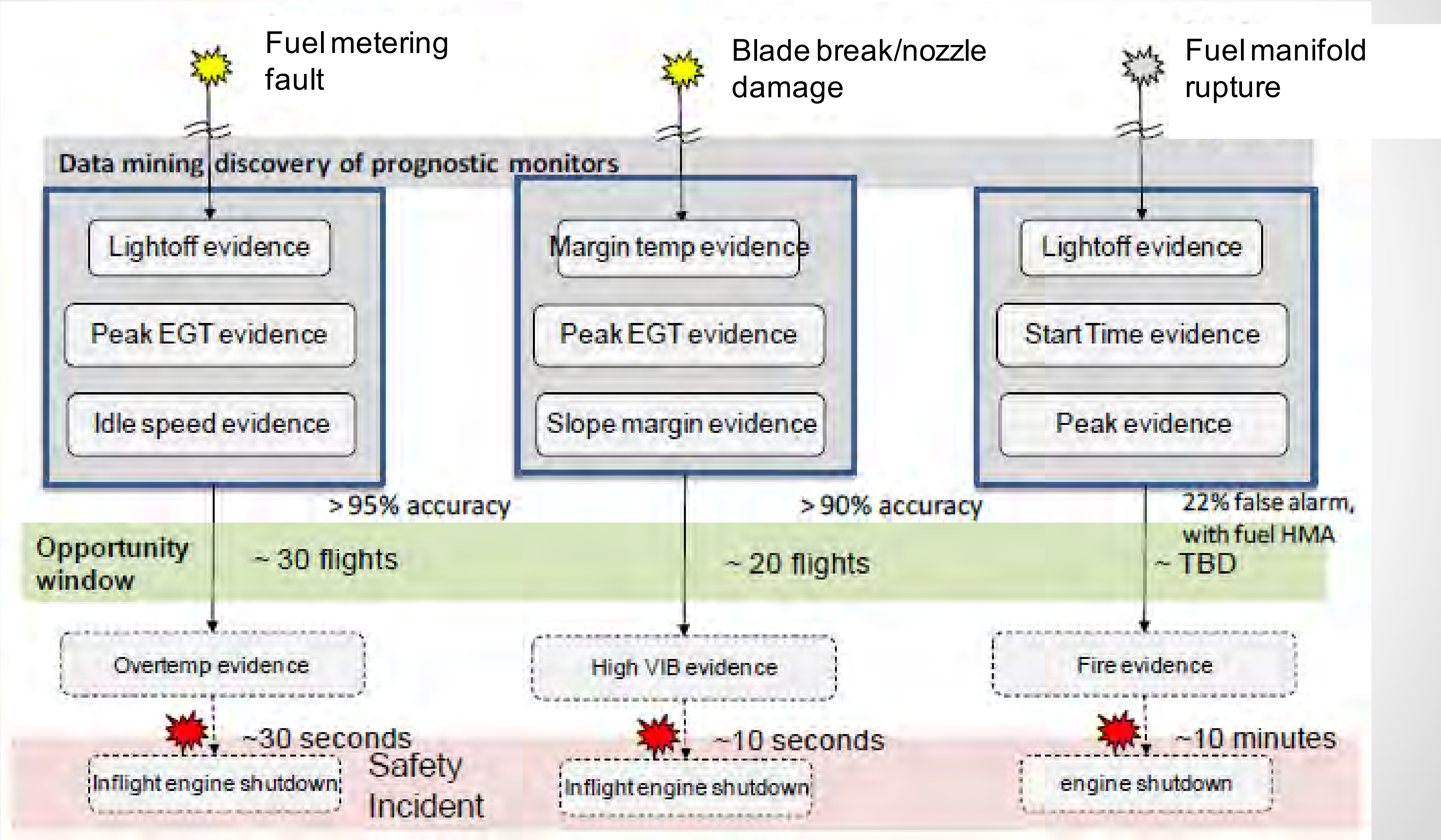
Startup Indicators
StartTime
IdleSpeed
peak Engine Temperature
Core Speed at Peak
StartSlope
StrtCutOff
LiteOff
prelit Engine Temperature
phaseTWO
timeToPeak

TakeOff Indicators
peak Core Speed
peak Engine Speed
peak Engine Temperature
takeoff Core Speed
takeoff Engine Speed
takeoff Air Temperature
takeoff Altitude
takeoff Engine Temperature
takeoff Margin

Rolldown Indicators
Rolltime
resdTemperature
dip Engine Temperature
Corespeed at Dip
Corespeed Slope
Corespeed Cutoff

no Start
slow Start
Hung Start
High Temp
multStart
phOneDwell
hotStart
medTempMargin
lowTempMargin
overSpeed
overTemp
abruptRoll
highRollEGT
rollBearing

Adverse Events Three Examples



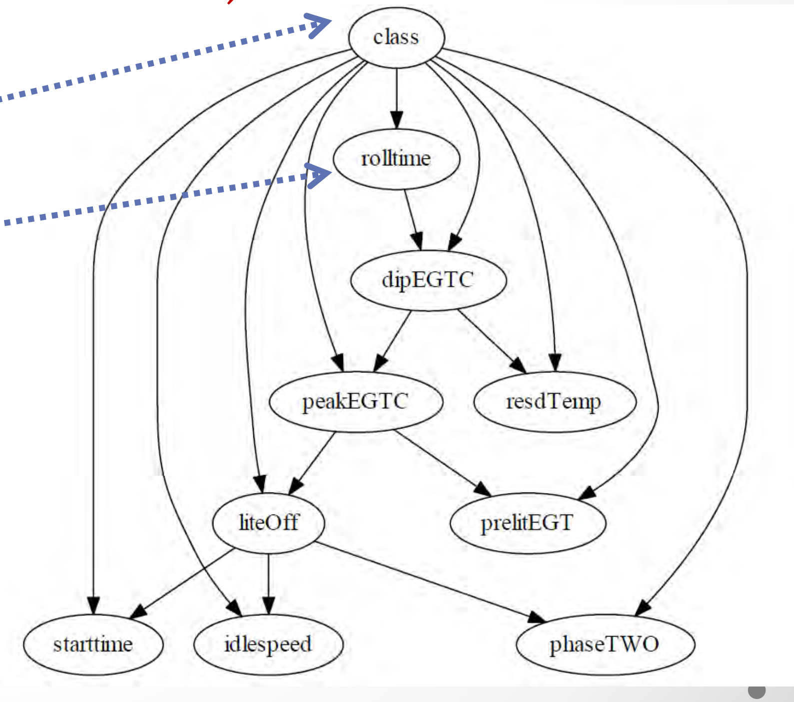
Machine Learning Approach

- Classifier methods to find additional/ refined features to classify faults more accurately
  - Used Tree-Augmented Naïve Bayesian (TAN) Classifier
  - Why TAN?
    - Simple extension to Naïve Bayes classifiers; not as a complex as full Bayesian network
  - Use *n-fold* cross validation to validate TAN classifier
  - Update Reference Model
  - Test, Validate, & Deploy

Working under constraint – Reasoner changes/updates require recertification
But updates/changes to reference model do not (data)

Introduction to TANs

- TAN structure
  - Naïve Bayes + capture some dependence between variables
  - TAN structures reduce this dependence information to capturing the dependence with one other variable (i.e., capture most important dependence)
- Examine TAN structure
  - Class note: Fault
  - Obs Root node
  - Rest of CIs + DMs



Dependencies – directional
Represent causal relations

Deriving TAN structures

- Can be created by Greedy search
  - Add/remove links
  - Example:
    - Find most correlated node to fault – make it the observational root node
    - Connect fault node to this node + all other evidence nodes
    - Order nodes by correlation value between them
    - For each of the remaining evidence nodes: pick the subset with highest correlations that satisfies TAN structure
- Our approach
  - Use Minimum Weighted Spanning Tree (MWST) using Mutual Information (MI) for edge weights
    - Connect fault node to all evidence nodes
    - Pick observational root node
    - Add directionality to edges by recursively directing all edges away from observational root node

Updating Reference Model

- Update DMs
  - Thresholds; e.g., FM2 $\rightarrow$ DM2

    Update: P(FM2|DM1,DM2)

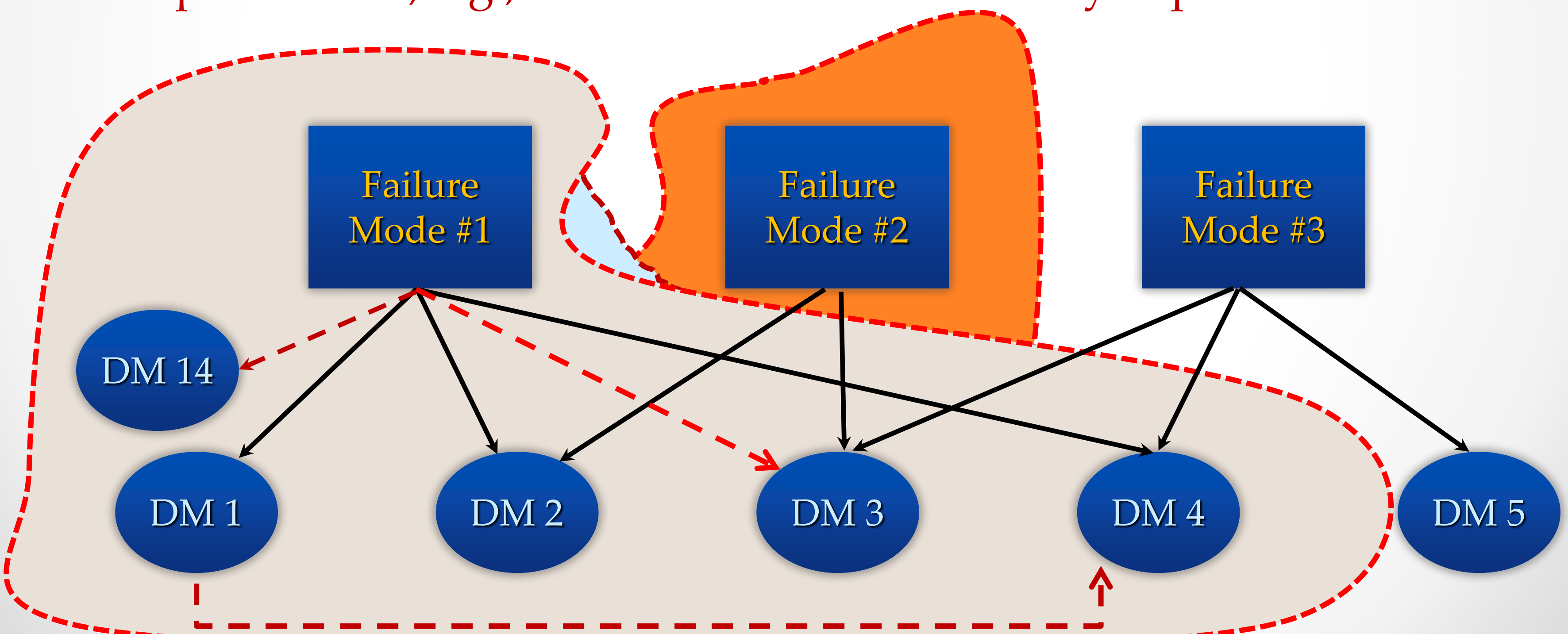
- Add new DMs?
  - New (FM1 $\rightarrow$ DM3)

    Update: P(FM1|DM1,DM2,DM3,DM4)

- Create "Super Monitor"

  P(FM1|DM1,DM4) > P(FM1|DM1)×P(FM1|DM4)

  - Dependencies, e.g., DM1 & DM4 substituted by super monitor DM14



Update: P(FM1|DM1,DM2,DM4)

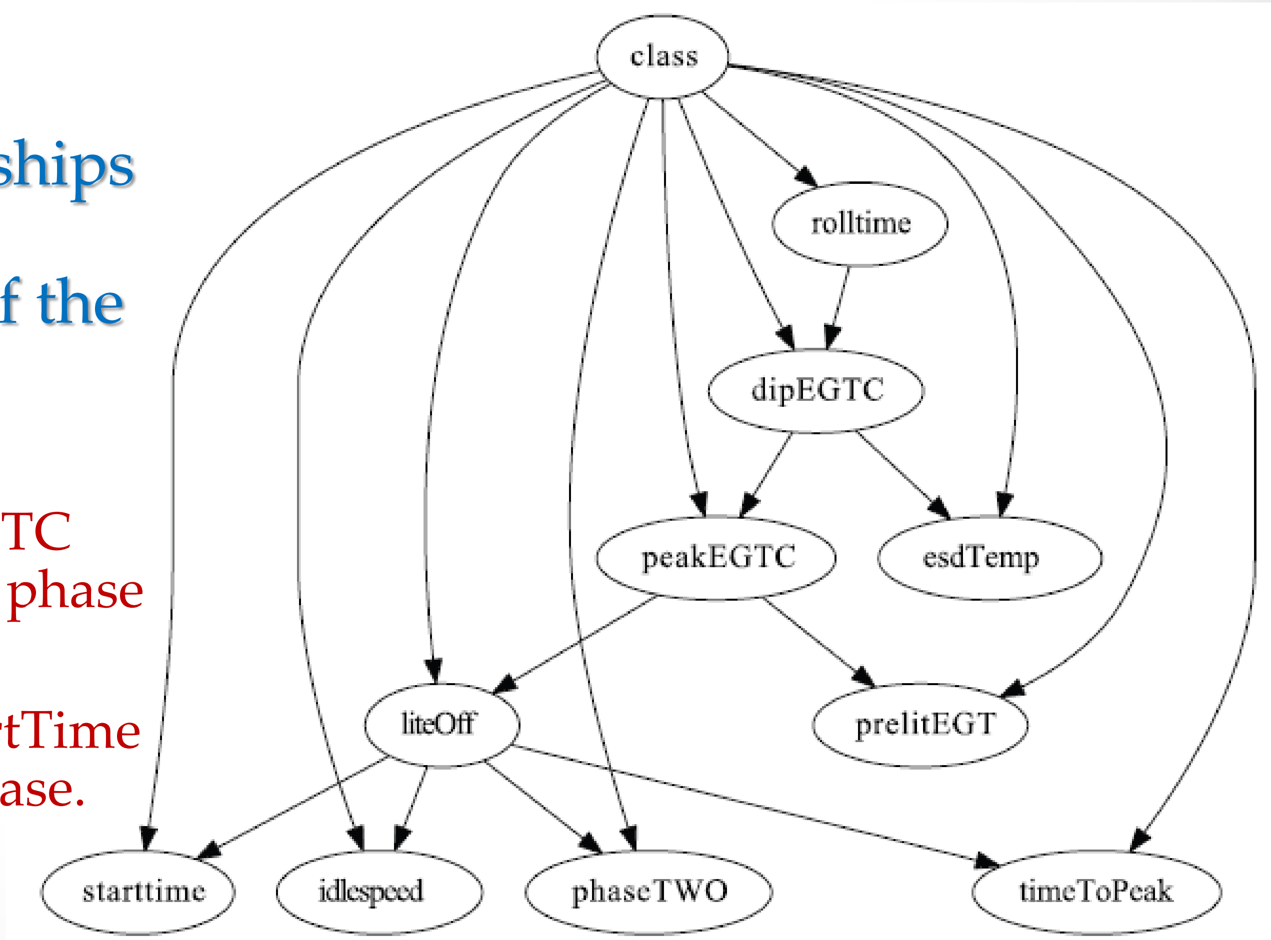
Case 1.1: Fuel metering fault

- Focus on incidents reported in FAA Aviation Safety Information Analysis & Sharing (ASIAS) database
  - Select incident associated with an aircraft – *"Overheated engine – Imminent Fire Hazard"* warning: *"Land immediately"*
- Assume 50 flights before actual incident occur, likely to indicate degrading behavior that led to incident
- Classifier results
  - 10-fold cross validation

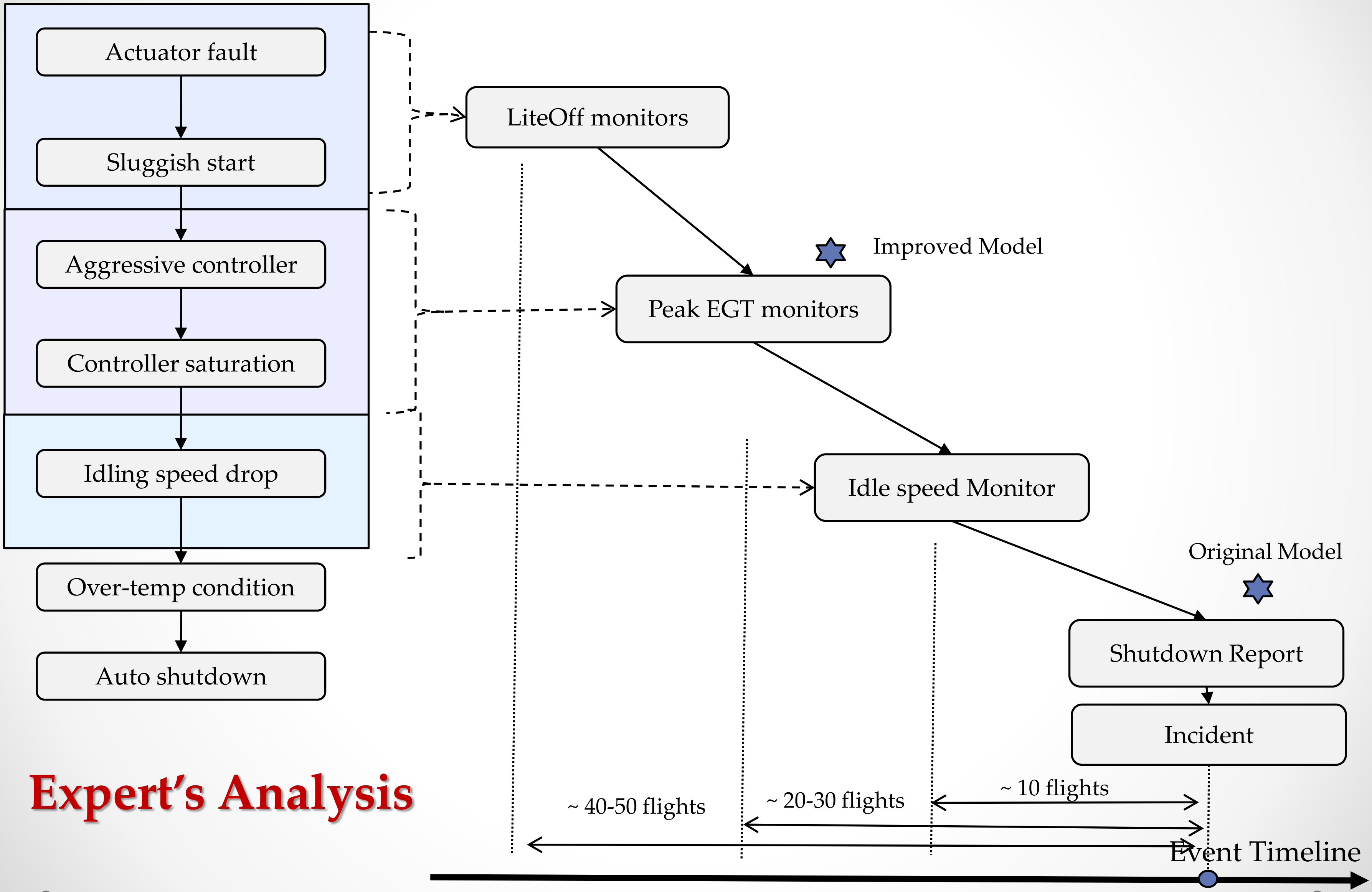
Accuracy	False Positives	False Negatives
99.6%	0.5%	0%

Generated TAN Structure

- Class = FuelHMA
- Expert's attention drawn to relationships between CI's for different phases of the flight
    - Rolltime & dipEGTC during shutdown phase
    - PeakEGTC & startTime during startup phase.



Anatomy of the FuelHMA incident & impact on Reasoner



Expert's Analysis

Case 1.1: Fuel HMA

- Results from Binning

Bin	Training Flights	Accuracy Holdout set	FP%	Obs Root node (ORN)	Children ORN	Notes
1	1—10	97.65%	2.30	IdleSpeed	StartTime	Thresholds chosen from this Bin due to low FP%
2	11—20	93.9%	5.70	peakEGTC	liteOff, dipEGTC	peakEGTC important node
3	21– 30	94.65%	5.30	peakEGTC	liteOff, dipEGTC	peakEGTC important node
4	31 – 40	96.62%	3.50	startTime	peakEGTC	Links startTime & PeakEGTC
5	41 – 50	96.06%	4.10	liteOff	Phase Two RollTime	Links Startup & Rolldown CI

Results: Fuel Metering Fault

- First Case Study - Fuel Metering
  - Mid-Air Emergency
  - Results
    - High Accuracy - 99.7%
  - Expert Analysis
    - Improved Detection Thresholds
    - Added New Evidence for Engine Temperature
    - Added New Evidence for Engine Speed and Temperature

Original Reference Model

	Event Minus 30 Flights	Event Minus 20 Flights	Event Minus 10 Flights
HPT Degradation	0.15	0.15	0.15
Fuel Metering	1.31	1.31	1.31
Fuel Delivery			
Turbine Nozzle	3.23	3.23	3.23
Bearing			
Duct Rupture			
Igniter Fault	2.29	2.29	2.29

Augmented Reference Model:

	Event Minus 30 Flights	Event Minus 20 Flights	Event Minus 10 Flights
HPT Degradation	0.15	0.15	0.15
Fuel Metering	13.29	13.29	8.52
Fuel Delivery	2.08	2.08	0.45
Turbine Nozzle	2.07	2.07	2.07
Bearing	2.40	2.40	2.40
Duct Rupture	3.69	3.56	3.56
Igniter Fault	2.29	2.29	2.29

Case 1.3: System-level fault

- Event Information
  - Engine 1 Fire warning illuminated
- What classifier told us ?
  - Both engine 1 & 3 showed fault manifestations
  - Fault Manifold was leaking (supplies multiple engines)
  - Manifestation time?
- Results
  - Accuracy = 90.3%; FP rate = 5.4% (one class classifier)
  - What about two class classification?
    - Fuel HMA & Fuel Manifold failure
    - Fuel Manifold accuracy drops to 77.5% & FP rate 22.5%

Issues

- Generalizing from 1-class classifier to multi-class classifiers
  - What are the consequences?
  - How do we evaluate?

- Possible solution: Feed output from classifiers into Bayes net to resolve dependencies and rank hypotheses
  - Related past work:
    - combine PCA + SDG (Tidriri, et al., 2016)
    - PCA + observer methods (Wang & Qin, 2002)
    - Linear model predicted by KF into ANN (Siswantoro, et al., 2016)

More General approach: Develop fusion methods?

Anomaly Detection
Finding unknown faults

• • •

Mack, D. L., Biswas, G., Khorasgani, H., Mylaraswamy, D., & Bharadwaj, R. (2018). Combining expert knowledge and unsupervised learning techniques for anomaly detection in aircraft flight data. *at-Automatisierungstechnik*, *66*(4), 291-307.

Biswas, G., Khorasgani, H., Stanje, G., Dubey, A., Deb, S., & Ghoshal, S. (2016). An application of data driven anomaly identification to spacecraft telemetry data. In *Prognostics and Health Management Conference*.

Anomaly Detection

- Finding patterns in data that do not correspond to expected (normal) behavior
  - Also called outliers
  - Anomaly detection related to Novelty detection
- Types of Anomalies
  - Point anomalies
    - Credit card fraud detection
  - Contextual anomalies
    - Patterns extracted from a spatial region or a time sequence
      - Need contextual + behavioral attributes
    - Used a lot in time series applications
      - Fault detection
  - Collective anomalies
    - Collection of data points represents an anomaly with respect to the entire data set
      - Example, a decreasing trend in time series data – each point is within bounds but the data points over time should be steady or increasing gradually

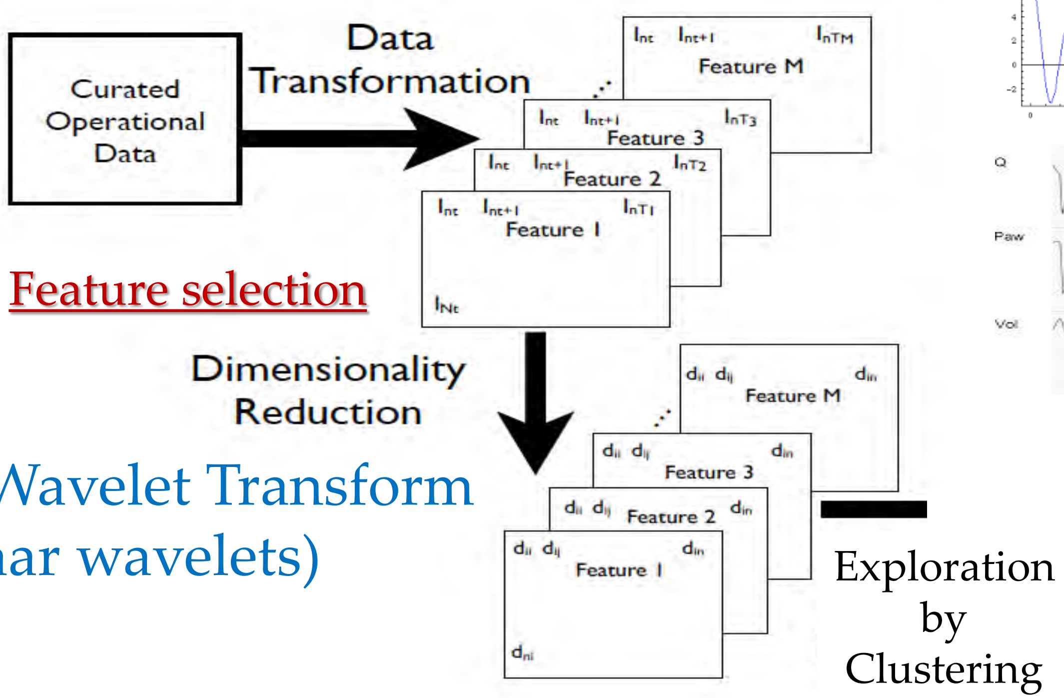
Chandola, Banerjee, & Kumar (2009). "Anomaly Detection: A Survey," ACM Computing Surveys, 41(3): 15-58.

Case Study 2
Unsupervised Anomaly Detection

- Exploratory, unsupervised learning
  - Look for Previously Undetected, and Unknown Anomalies
- Data unlabeled, but work with entire data set – big data problem
- Approach
  - Start with flight segments (contextualized – take off segments)
  - Reduction and Discrete Feature Generation across Time Series
  - Generate Dissimilarity Measures to compute pairwise dissimilarities among flights
  - Find anomalous flight segments

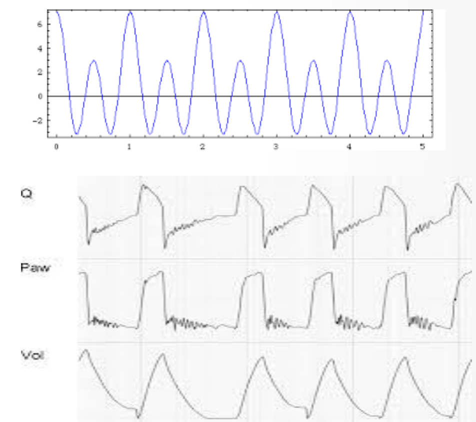
Step 1: Preprocessing

- <u>Transform</u> curated data to produce a multi-dimensional data structure $(flights \times signals \times signal\ waveforms)$ for exploration
- <u>Feature reduction</u>: continuous signals, multiple sampling rates to discrete features



<u>Feature selection</u>

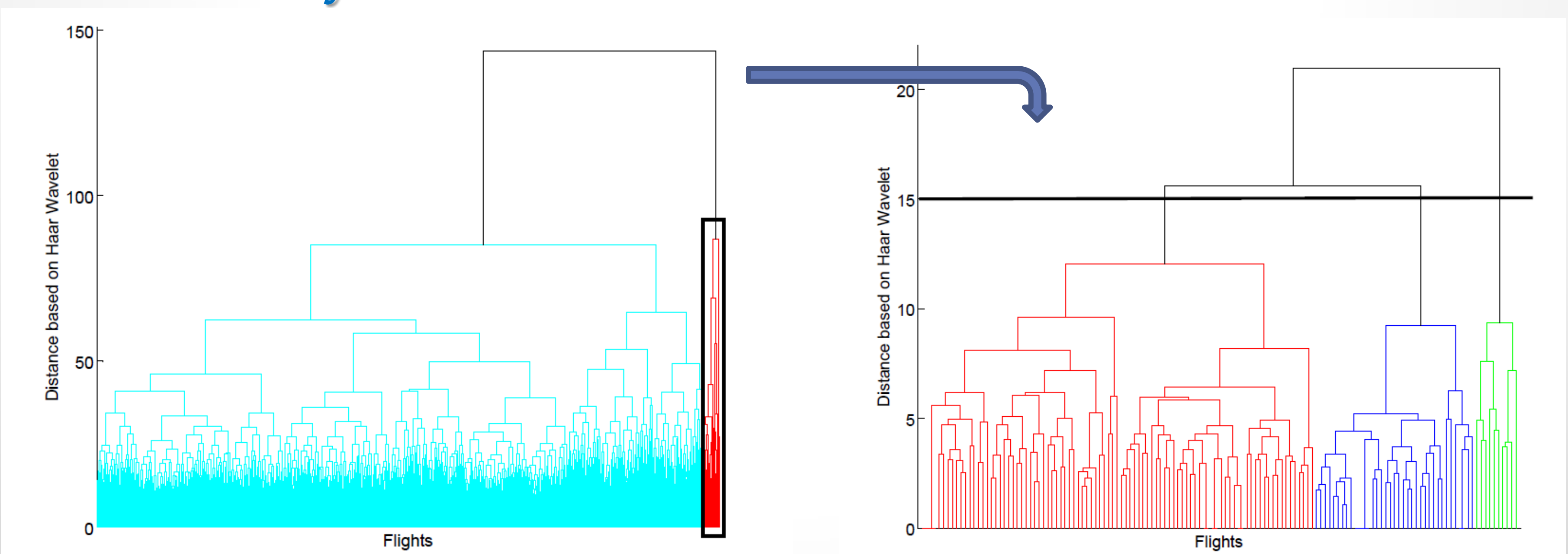
Discrete Wavelet Transform (Haar wavelets)



$$\begin{pmatrix} a_{11} & a_{12} & \bullet & a_{1m} \\ a_{21} & a_{22} & \bullet & a_{2m} \\ \bullet & \bullet & \bullet & \bullet \\ a_{M1} & a_{M2} & \bullet & a_{Mm} \end{pmatrix}$$

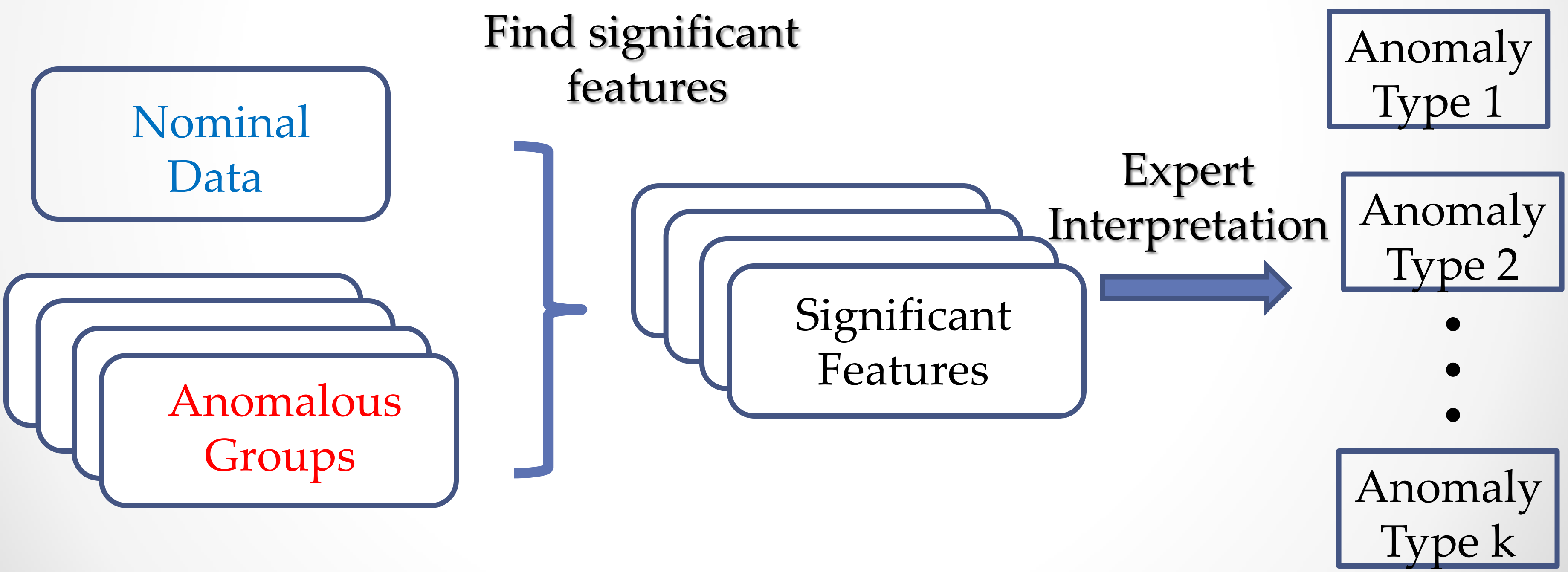
Step 2: Unsupervised Learning

- Dissimilarity matrix (pairwise distance between every object pair)
- Hierarchical Clustering algorithm (UPGMA)
- Outliers – individual objects or small groups sufficiently different from nominal clusters



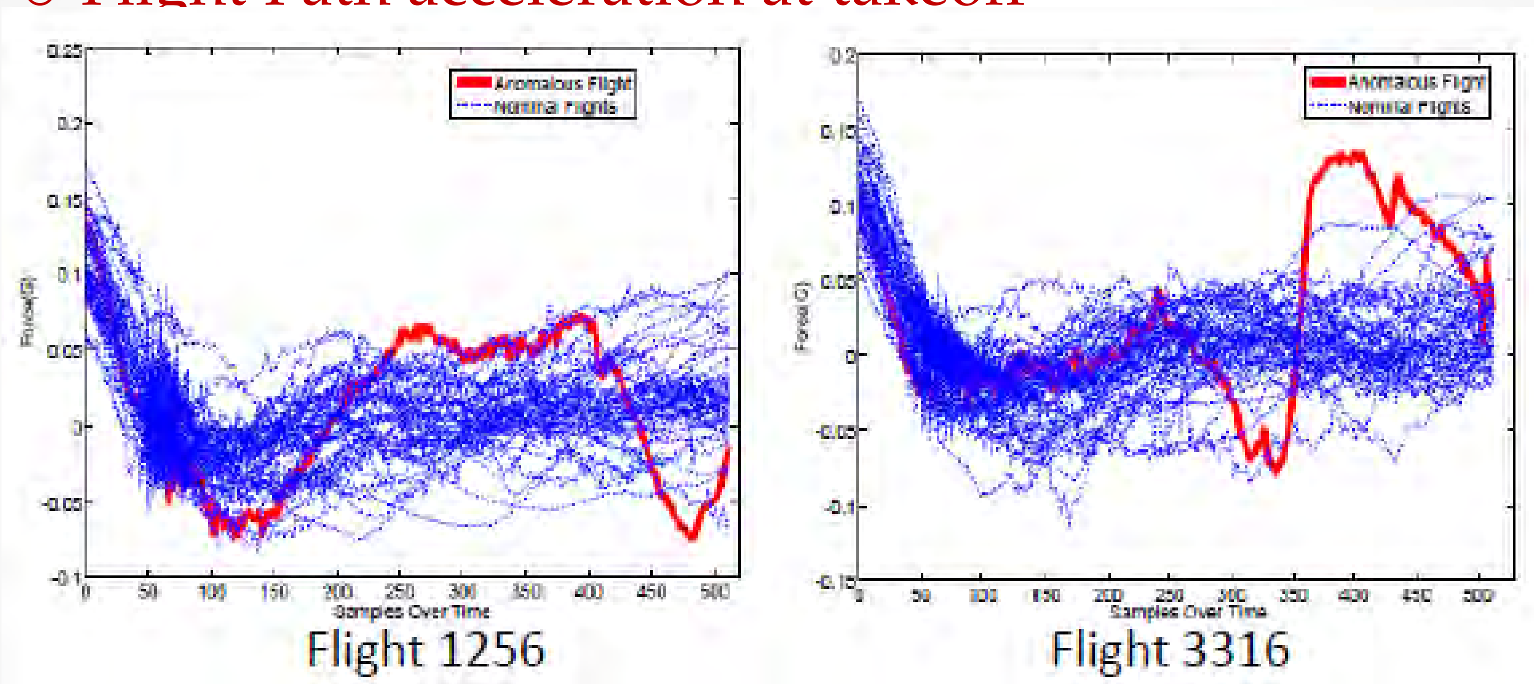
Step 3: Anomalies

- Characterizing Anomalies
  - Extract significant features and consult experts to characterize anomalies

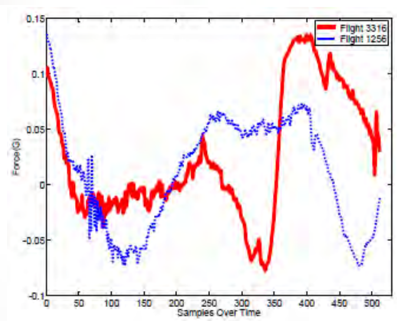


Example: Anomalous Group 3 Steep Takeoffs

- Flight Path acceleration at takeoff



Flight # 1256 not unusual
Acceleration slowed down: unusual,
but Autopilot in control



Flight # 3316: Near stall condition: confirmed by auto thruster setting
Auto thruster disengaged

Combining Model + Data Driven Methods: Hybrid Approaches …

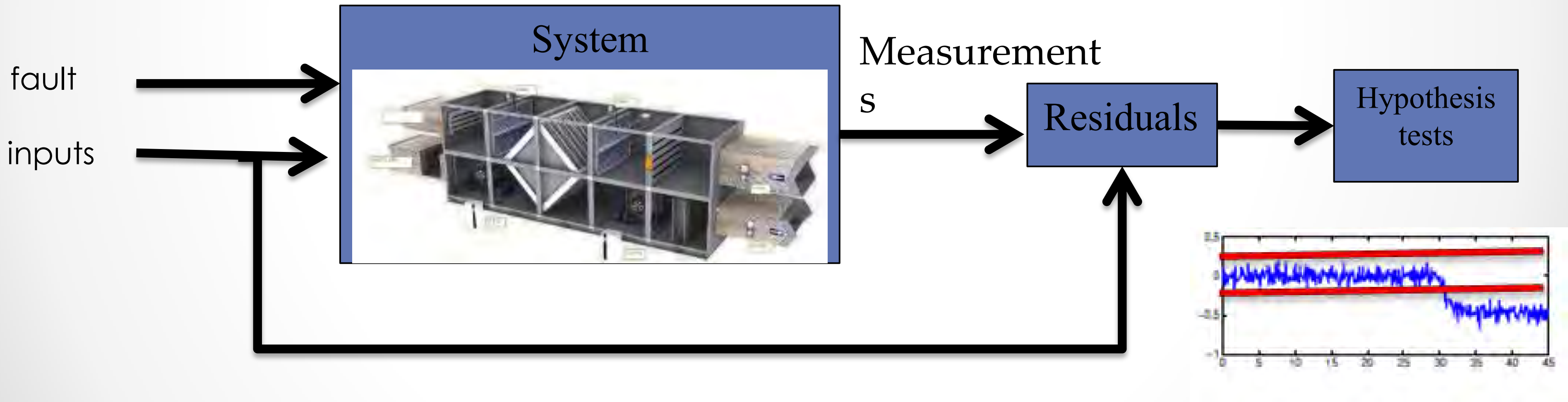
Khorasgani, H., & Biswas, G. (2018). A methodology for monitoring smart buildings with incomplete models. *Applied Soft Computing*, *71*, 396-406.

Case Study 3:
Model + Data-driven diagnosis
Smart Buildings

- It is not feasible to generate an accurate and complete model for smart buildings
  - Especially difficult because highly precise and accurate spatio-temporal models very expensive to create
  - But models of components and subsystems – possible
    - Outdoor air unit (OAU)
      - Relationship between a fan's static pressure and airflow is nonlinear and a function of the fan's rotational speed..
      - The performance of the exhaust fan and the output fan are <u>not independent</u> but the dependency is not modeled.
      - <u>Unknown parameters</u> such as wind speed, and the air filter's resistance affect the model.
- May not have training data for all the operation modes and fault modes

Model-based Fault Detection and Isolation

- Model-based Approaches:
  - Use a physics-based model that defines nominal/faulty behavior of a dynamic system to detect faulty behaviors.



Fault detection

Residual: A fault indicator, based on a deviation between measurements and model-equation based computations.
Hypothesis test: determines when change in a residual values are statistically significant.

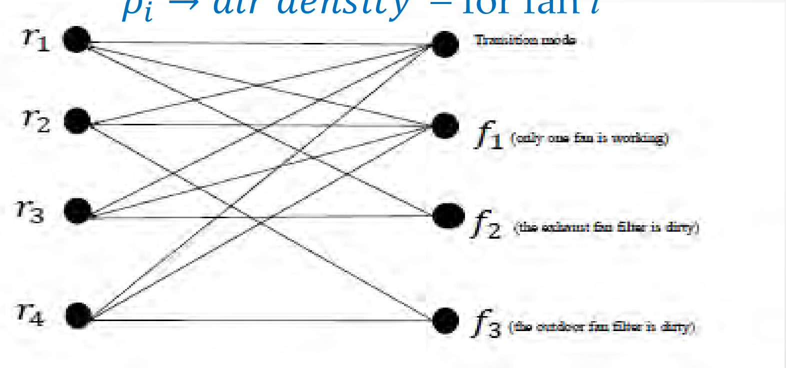
Model-based Fault Detection and Isolation in OAU

- **Faults**
  - Only one fan is operating (in normal situation they are both on or off)
  - Exhaust fan or outdoor fan filters are dirty/blocked
- **Diagnoser design:**
  - The complete model was not available
    - Used laws of physics to derive relationships between fan speed, static pressure, and airflow
    - Developed a maximum likelihood estimator (MLE) to estimate the parameters
  - Analytical redundancy relationship (ARR) approach to generate the residuals
  - Z-test [Biswas et al.,2003] as the hypothesis test

- Physical laws to derive relations between exhaust fan, outside fan speed, static pressure and airflow

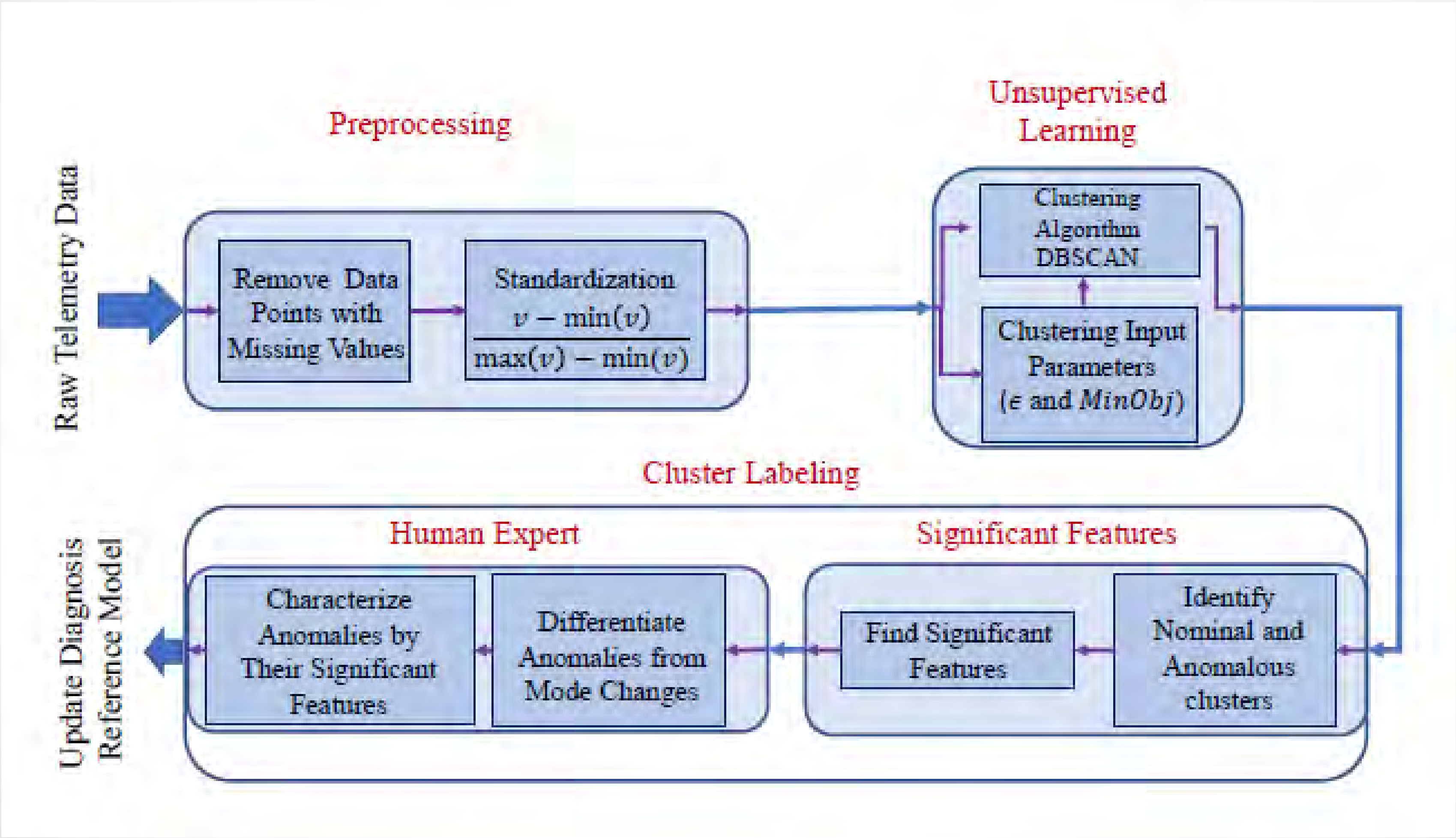
$$P_2 = P_1\left(\frac{D_2}{D_1}\right)^2\left(\frac{N_2}{N_1}\right)^2\left(\frac{\rho_2}{\rho_1}\right);\ Q_2 = Q_1\left(\frac{D_2}{D_1}\right)^3\frac{N_2}{N_1}$$

$Q_i \rightarrow airflow;\ D_i \rightarrow diameter;$
$N_i \rightarrow rotational\ speed;\ P_i \rightarrow static\ pressure;$
$\rho_i \rightarrow air\ density$ – for fan i



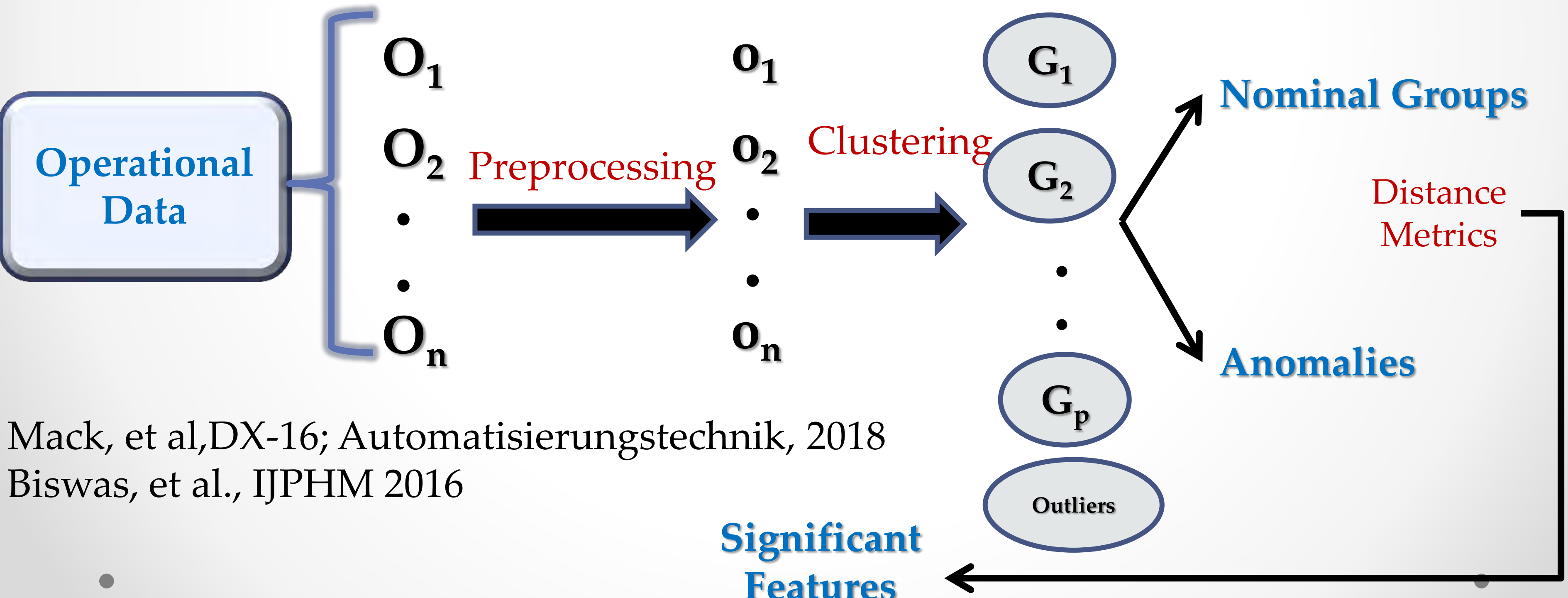
Diagnosis approach	Accuracy	False positive rate
Model-based approach	87.1%	12.7%

Data-Driven Approach
Updating Diagnosis Model



Unsupervised Data-driven Feature Extraction

- Preprocessing
  - Standardizes the time series variables (10, 396 training samples)
- Clustering
  - Extracts the clusters in the data set (used dbscan2 – 5 anomalous groups)
- Significant Features
  - Set of features that best distinguish an anomalous cluster from nominal operations



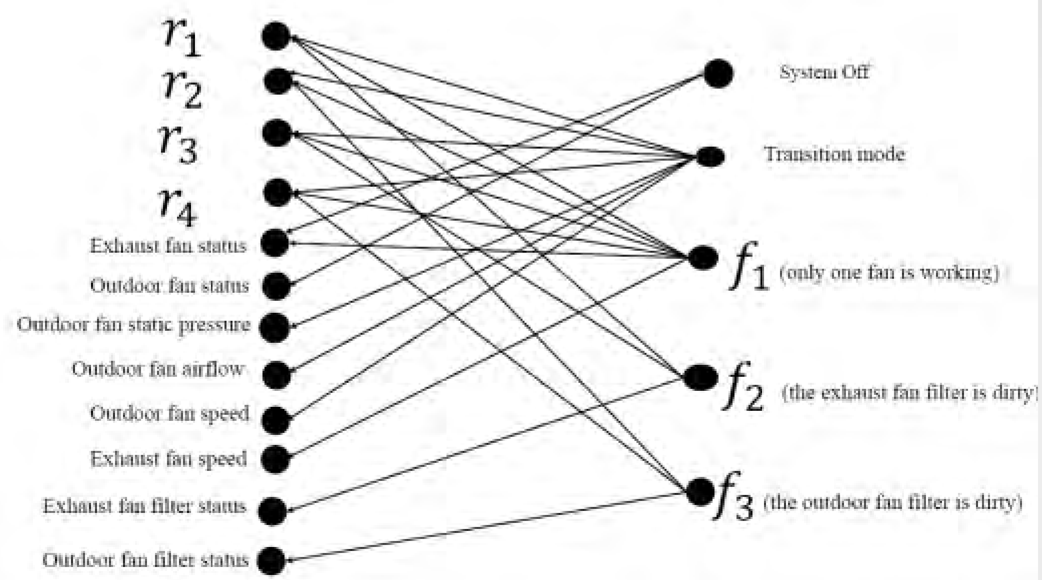
Mack, et al,DX-16; Automatisierungstechnik, 2018
Biswas, et al., IJPHM 2016

The Operating Modes & Their Significant Features

Cluster	Detected Mode or Anomaly	Significant Features	Description
1	Normal operation mode		
2	Mode: the system is off	• Exhaust fan status • Outdoor fan status	• The OAU is off in this mode
3	Mode: transition	• Outdoor fan static pressure • Outdoor fan airflow • Outdoor fan speed command	• Low pressure and airflow when the system starts
4	Fault: the outdoor fan filter is dirty	• Outdoor fan filter status	• The outdoor fan filter has to be changed.
5	Fault: the exhaust fan filter is dirty	• Exhaust fan filter status	• The exhaust fan filter has to be changed.
6	Fault: only one fan is working	• Exhaust fan speed command • Exhaust fan status	• Exhaust fan off and outdoor fan on

Integrated Model + Data driven Fault Diagnosis

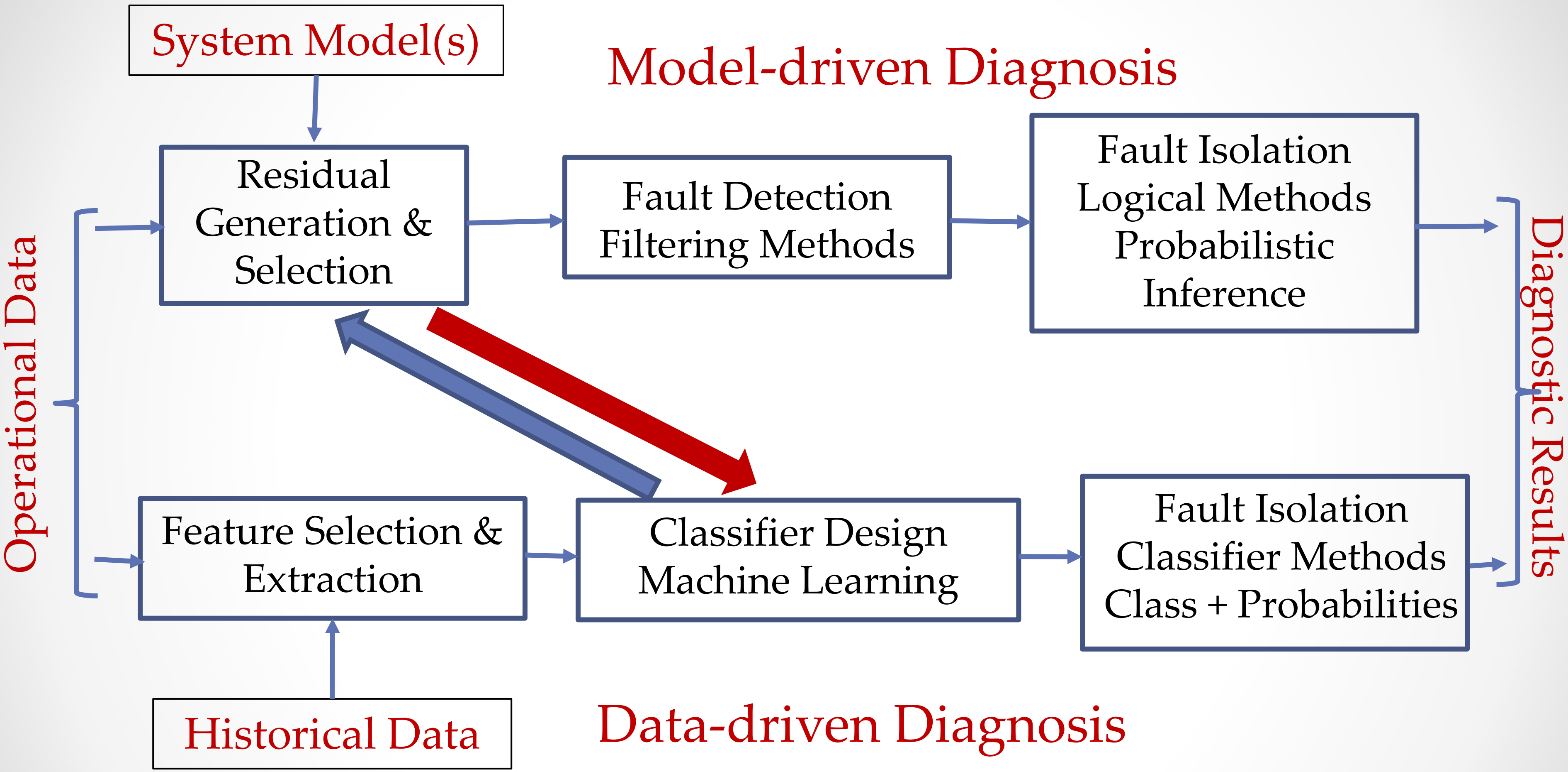
- Model-based diagnosis:
  - Monitors: outputs of the hypothesis tests
- Data-driven diagnosis:
  - Monitors: selected features
- Integrated approach
  - Monitors: residuals+ significant features



Hybrid diagnosis reference model

Diagnosis approach	Accuracy	False positive rate
Model-based approach	87.1%	12.7%
Hybrid approach	92.5%	2.5%

Discussion



Our work thus far: Classifier output → Residuals → Combined Fault Isolation

What about: Residual Analysis output → Classifier → Classifier-based diagnosis?

- Currently working on this approach …..

Next Steps

- Fault Detection Refinement & Anomaly detection methods apply
- What about multiple manufacturing/vehicle processes?
  - Equivalent to a fleet of aircraft
  - Projects working on: Prognostic Scheduling
- Impact on Control
  - What about fault tolerance and fault adaptivity?
  - Current project: Reinforcement learning for Fault Adaptive control (better performance than model predictive control)
- Area not explored – virtualization of complex processes; use of cloud computing architectures
  - Optimization of manufacturing operations – exploiting redundancy
- Area not explored – Cybersecurity