5-14-2010

Fault Detection for Systems with Multiple Unknown Modes and Similar Units

Anwer Bashi
University of New Orleans

Fault Detection for Systems with Multiple Unknown Modes and Similar Units

A Dissertation

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy
in
Engineering and Applied Science

by

Anwer S. Bashi

B.S. University of New Orleans, 1996
M.S. University of New Orleans, 1997

May, 2010

Dedication

To my wife Jenny, and our children, Sabah and Junaid

To my mother, Mona, and my father, Sami

Acknowledgments

I wish to convey my deep gratitude to my advisors, Dr. Rong Li and Dr. Vesselin Jilkov for their support and guidance. Their remarkable acumen in research was inspirational and it was their invaluable comments which helped transform raw ideas into proper scholarship. Over the years, I have appreciated the many fruitful opportunities that long-term collaborations with both Dr. Li and Dr. Jilkov have provided.

I also want to thank Dr. Edit Bourgeois, Dr. Juliette Ioup, and Dr. Golden Richard for serving on my dissertation committee, and for their constructive and valuable comments on the dissertation, as well as their cheerful outlook. I want to express a very special appreciation for Dr. Bourgeois, who was my advisor for my Masters degree, and who, in my experience, has always put the student's interest foremost.

Furthermore, I would also like to express my appreciation for all my colleagues in the Information and Systems Laboratory, most of all Ryan Pitre and Jifeng Ru, for many fruitful discussions and collaborations, and for all the members and staff of the Department of Electrical Engineering.

On a very personal note, I want to thank God. I pray that He helps me be grateful for all the blessings I have been given. I also want to express my heartfelt gratitude to my parents for their support, guidance and sacrifice throughout my life, and for allowing the wellbeing of their children to plot their life's course.

As for my wife, Jenny; I wouldn't have had the will to continue my course of study were it not for her encouragement and comfort, and I wouldn't have had the ability were it not for her willingness to take on more than her share.

Contents

Abstract

This dissertation considers fault detection for large-scale practical systems with many nearly identical units operating in a shared environment.

A special class of hybrid system model is introduced to describe such multi-unit systems, and a general approach for estimation and change detection is proposed. A novel fault detection algorithm is developed based on estimating a common Gaussian-mixture distribution for unit parameters whereby observations are mapped into a common parameter-space and clusters are then identified corresponding to different modes of operation via the Expectation-Maximization algorithm. The estimated common distribution incorporates and generalizes information from all units and is utilized for fault detection in each individual unit.

The proposed algorithm takes into account unit mode switching, parameter drift, and can handle sudden, incipient, and preexisting faults. It can be applied to fault detection in various industrial, chemical, or manufacturing processes, sensor networks, and others. Several illustrative examples are presented, and a discussion on the pros and cons of the proposed methodology is provided.

The proposed algorithm is applied specifically to fault detection in Heating Ventilation and Air Conditioning (HVAC) systems. Reliable and timely fault detection is a significant (and still open) practical problem in the HVAC industry – commercial buildings waste an estimated 15% to 30% ($20.8B - $41.61B annually) of their energy due to degraded, improperly controlled, or poorly maintained equipment.

Results are presented from an extensive performance study based on both Monte Carlo simulations as well as real data collected from three operational large HVAC systems. The results demonstrate the capabilities of the new methodology in a more realistic setting and provide insights that can facilitate the design and implementation of practical fault detection for systems of similar type in other industrial applications.

Keywords: Fault detection (FD), heating ventilation and air-conditioning (HVAC), hybrid system, multiple model, estimation, expectation-maximization (EM).

List of Figures

List of Tables

1

Introduction

Nomenclature

k	Sample, $1 \dots L$.
m	System mode, $1 \dots M$.
s	Unit, $1 \dots S$.
$X_{k\|n}^{(m,s)}$	X at time k for mode m in unit s, given data up to time n.
$\|X\|$	Denotes the determinant of X for a matrix, or the cardinality of X for a set.
$[x, y, z]_k$	x, y, z at time k.
$\{x\}$	The set of all x.
$\hat{x}$	Estimate of x.
$\bar{x}$	Mean of x.
$\tilde{x}$	Error of x.
$\mathring{x}$	Global x.
$\check{x}$	Local x.
$\mathcal{X}, \mathcal{S}$	Calligraphic font denotes a set.
`FNC`	Typewriter font denotes a function.
$p\left(x\right)$	The probability distribution function for x.
$P\left\{x\right\}$	The probability of x occurring.
$\mathcal{L}\left\{x\right\}$	The likelihood of x.

1.1 How to Read This Dissertation

An executive summary of sorts is given in Chapter 10 (Conclusion). However for those wishing to get a general overview of the developments documented in this dissertation, we suggest starting with Chapters 2 (Overview) and 6 (Algorithms), glancing at the figures in Chapters 8 (Experiments) and 9 (Performance), and then reading Chapter 10 (Conclusion).

Two conference papers [5, 6] and a journal article [4] have been authored on the work contained in this dissertation.

2

Overview

The Heating, Ventilation, and Air-Conditioning (HVAC) industry focuses on control and monitoring of equipment in a facility to ensure acceptable occupancy conditions. This includes temperature, pressure, and air-quality regulation of the hundred or thousands of individual units at a facility. There is sometimes an overlap between HVAC and regulatory oversight (such as minimal fresh-air requirements per tenant or square foot), or life-safety systems (maintaining a negative pressure in a tuberculosis ward, or controlling temperature in a surgery room). A large site can collect and process several 100 million data samples per day; however most of this information is not retained due to storage or bandwidth considerations.

This project focuses on fault detection for systems with many model-correlated units, and uses the HVAC industry as a detailed example. Currently, the industry does not utilize complex mathematical models or inference aside from those required for controlling analog systems. Often, the most mathematically complex components of an HVAC system are the PID (proportional-integral-derivative) controllers.

One of the reasons for this is the level of expertise and customization required in setting up complex solutions for control, optimization, or fault detection. Even if a technology promises better performance in some way than existing solutions, it is unlikely to become widespread unless it is “plug-and-play” since the current logic- and schedule-based techniques are generally considered sufficient.

Currently, almost all automatic fault detection is of the model-free, non-parametric form, such as alarming on out-of-range sensors. Model-based fault detection is very hard to make generally applicable since detailed equipment models tend to be manufacturer and model dependent.

While there have been many approaches suggested for fault detection in HVAC, many start with a very detailed model of the equipment in question, or spend considerable time generating such a model. Even those which do not use a transfer-function-based model (for example, using a neural network, or probability distribution) require significant training before-hand and may only be applicable to the specific piece of equipment they were trained on. Other, rule-based solutions (fuzzy-logic, expert system, or Bayesian reasoning) must be keyed to a specific equipment type and may require more detailed information than is financially practical.

2.1 Complications

One of the main problems with these methods is that parameters for simple models often drift, and sometimes quite dramatically, given different externalities. While detailed models can account for this, expecting a building engineer to develop a non-linear model for a variable-air-volume box, for example, which properly accounts for disturbances due to incident sunlight, outside air enthalpy effect on cold-deck reset temperature, leakage of cold or hot air from adjacent spaces, etc., is unrealistic. Also, such a model would require data that is almost surely not readily available to "key" the model to a particular unit (window size, current occupancy of space, configuration and thermal conductivity of bounding walls, etc.)

Another issue is that units are often in different modes, where they operate based on models which are different from each other. A very basic example of different modes is cooling mode as opposed to heating mode; however, as discussed later, modes can be, and often are, more subtle than this. We can consider each mode to be representative of a different model for operation. Units may change modes either suddenly or gradually.

Finally, sudden failures are often the simplest form to diagnose in HVAC since they usually have a dramatic impact on performance which can be detected at a number of downstream sensors. However, incipient or gradual failures are difficult to detect since they usually present as a slow degradation in performance which can only be seen over the course of a number of months. Techniques based on identifying a model and using performance or error metrics to detect failure will find it very difficult to identify a failure which occurs at a rate slower (often, much slower) than the model drifts under normal conditions.

Also, we address another class of failure which has largely been overlooked in fault detection literature – pre-existing failure. This is because it is impossible to identify a model of correct operation simply by looking at a unit after it has already failed.

2.2 Proposal

We propose a novel approach for fault detection in large HVAC systems. A typical "Class-A" HVAC installation is composed of many identical pieces of equipment: for example, 8 cooling towers, 6 chillers, 140 air-handling-units (AHU's), 700 variable-air-volume boxes (VAV's), and 6000 assorted liquid and gas temperature, flow, and pressure sensors. At most sites, all AHU's are of the same model from the same manufacturer (or possibly from a very small set of manufacturer-models), as are VAV's, chillers, and cooling towers. This means that gathering data from all identical units should provide more information as to whether any particular unit has failed.

We propose a method where models are identified for identical units and, based on this, a probability distribution for unit parameters (or models) is generated. Failed units are those which do not seem to operate according to the identified distribution.

This is achieved by abstracting a model as a structure and set of parameters. Based on the data, units are mapped into parameter-space (or parameter-structure space in the case that different modes are represented by different structures) by using linear (for linear uniform-structure mode-models), non-linear (for non-linear uniform-structure mode-models), or hybrid (for non-uniform-structure mode-models) parameter estimators.

We assume that the type of parameter distribution is known, and consider the multivariate Gaussian case in particular. This is in part due to simplicity, but some justification may be found in the Central Limit Theorem.

Given this information, clusters are identified corresponding to the different modes. A technique based on the expectation-maximization algorithm is used to cluster the data, while taking into consideration parameter drift, unit-mode probabilities, and transition probabilities. The algorithm is presented in an approximate form, since mode-transitions make evaluation of the full hypothesis tree infeasible.

In addition to detection of conventional faults, our method should prove useful for detection of incipient and pre-existing failures also. This is because information from other units can be used to identify acceptable model boundaries at any point in time, without restricting the estimation process to data collected from a single unit.

3

Literature Survey

When accurate models of a single-mode system are available, fault detection is relatively simple. Measurement residuals (i.e., the difference between the predicted and actual system measurements) can be statistically tested for consistency.

However, when the system in question has more than one mode, i.e. is a multi-mode or hybrid system [40], the situation becomes considerably more complicated since both the discrete mode and the continuous state must be estimated. Despite this, several approaches have been suggested to tackling the hybrid fault detection problem (Section 3.1).

The unit distribution is assumed to be a Gaussian mixture (Section 3.2). In order to be able to look at the statistics of the Gaussian mixture without the contribution of failed units, a Truncated Normal Distribution (Section 3.2.1) is used. This allows for detection of small faults without degrading the estimate through measurements originating from the faulty unit.

A concrete example of our generic fault detection approach is given by applying the developed algorithm to large building heating, ventilation, and air-conditioning systems. The field has a rich body of fault detection literature (Section 3.3) however none of the approaches previously suggested are well suited to handle unknown hybrid systems.

3.1 Multiple-Model Fault Detection

As previously mentioned, the parametric uncertainty aspect of this problem introduces significant complications since both the discrete mode and the continuous state must be estimated.

Despite this, several approaches have been suggested to tackling the hybrid system fault detection problem [56], such as the multiple-model adaptive estimation algorithm (MMAE) approach [45,47,49], and the interacting multiple-model fault detection (IMM-FDI) approach [46, 53, 58, 68]. The difference between these two approaches is in the interaction between filters. While the MMAE uses a bank of independent ("autonomous") filters, the newer IMM-FDI uses interacting filters, generally resulting in more accurate and faster detection. Interacting multiple model (IMM) estimation has been shown to be a more cost-effective hybrid estimation technique than autonomous multiple model estimation [40, 52].

Unfortunately, these approaches require a model set [57], with detection probability directly dependent on how well the model set is designed. Specifically, the failure mode must be represented within the designed model set. While it is possible to automatically design models for failed sensors and actuators, it is impossible or unrealistit to provide complete coverage of all possible failures. Systems often fail partially, resulting in degraded performance, or may fail in a gradual fashion. Additionally, insufficient coverage of normal operating conditions results in false detections when the mode changes to one which is not well represented within the model set. Inversely, broadening coverage (for example, through an increased process or measurement noise, or by adding models) to compensate for modelling deficiencies may result in missed or delayed detection since the failure mode is more likely to be similar to one of the models in the set.

Another approach is to "mix" models to generate a "best" candidate model. In [26] and [25], the authors generate an estimator model through a convex mixing of candidate models based on the relative model probabilities. A single filter is used to propagate the state estimate, so only a single state estimate exists. In contrast to the more "conventional" multiple model fusion techniques, they show how the *models* can be mixed, rather than the estimates. This approach is similar to that taken to generate the "expected mode" in the expected mode augmentation algorithm [37, 41, 42]. Hypothesis testing can be performed against the expected mode, which simplifies the problem by allowing fault detection to be treated as a binary hypothesis test. Nevertheless, the expected mode is a convex mixture and, as such, will suffer if the model set is designed poorly or has inadequate coverage.

Fault detection and target maneuver onset detection [55] have a lot in common. Maneuver onset is often detected by observing a deviation in the model of the target from some nominal "steady-state". Similarly, faults can be considered a deviation from "normal-operation". For this reason, many of the methodology utilized in maneuver onset detection can be applied to fault detection.

3.1.1 Change Detection

In general, given an appropriate test statistic, the Shiryaev Sequential Probability Ratio Test (SSPRT), cumulative sum (CUSUM), or a similar test, can be used for change detection [8, 54].

Wald's Sequential Probability Ratio Test (SPRT) is optimal in the sense that it makes the quickest possible decision for either $\mathcal{H}_0$ or $\mathcal{H}_1$ for specified false-alarm and missed-detection thresholds. The Shiryaev SPRT uses the change in the probability density function to provide the quickest detection of a change in a sequence of conditionally independent measurements [44]. Since SSPRT is a Bayesian approach, it requires the a priori probabilities for both hypotheses and transition probabilities to be defined.

First proposed by Page [50], the CUSUM algorithm is a sequential probability ratio test which is restarted as long as the decision taken is $\mathcal{H}_0$. Page suggested (and it was later proven [43, 59]) that zero is the optimal lower bound if the goal is to reject the null hypothesis.

CUSUM is optimal for simple hypotheses, where the distribution is completely known, before and after the fault. Since this is not the case in most fault detection scenarios, a composite hypothesis must be assumed, in which case likelihood marginalization may be used as in [55].

Two different classes of statistical tests are commonly used for maneuver onset detection algorithms (surveyed in [38]): Chi-squared significance tests such as Measurement Residual (MR) and Input Estimation (IE), and Likelihood Ratio tests such as Generalized Likelihood (GLR) and Marginalized Likelihood Ratio (MLR). Since many of the detection tests provide, or can be modified to provide, a maneuver or failure onset time, smoothing may be performed to improve the estimate. Given the hybrid nature of the problem, multiple-model smoothing techniques must be used, as in [10, 13, 14, 31].

3.2 Gaussian Mixtures

Finite mixture models (models expressed as convex combinations of other probability density functions) have been popularly used to model complex probability distributions. One particularly important mixture model is the Gaussian mixture model (GMM), which utilizes the easily represented and versatile Gaussian distribution. They are among the most statistically mature methods for both density estimation and clustering, and have been the subject of theoretical and experimental study for over a century [61].

A Gaussian mixture model can be used to represent any given probability density with an arbitrary degree of accuracy if enough mixture components are used [60]. Many approaches to estimation of mixture density are available [15, 62, 70]. Different approaches of using Gaussian mixtures to identify failures are looked at in Appendix D, as well as methods of calculating the distance between different mixtures.

3.2.1 Truncated Normal Distribution

The calculation of parameters for a truncated normal distribution (TND) is a recurring problem in real-world testing. A Gaussian distribution assumes limits of $\pm\infty$. In practice, there are usually upper and lower limits imposed on an experiment, so that the actual distribution being measured is a truncated normal distribution (for example, [20, 28, 29]).

A truncated distribution is used for the purpose of outlier rejection in Chapter 7. The statistics for the original distribution are then inferred from the statistics of the truncated distribution.

For univariate TND's (i.e. distribution of 1-dimensional random variable), the relationship between the mean and covariance of the truncated sample and the original (untruncated) distribution is well known [20, 23], as are the statistics of bivariate [9, 18, 48], and trivariate [63] distributions. However no closed-form solution has been presented for the general multivariate case. In Chapter 7, numerical methods are used to generate a lookup-table for truncated statistics.

3.3 HVAC Fault Detection

Fault detection research in the HVAC industry started in the late 1980's and early 1990's, when the International Energy Agency (IEA) commissioned the Annex 25 research project on real-time simulation of heating, ventilation, air conditioning, and refrigeration systems for building optimization, fault detection, and diagnosis ([19, 30] contain almost 1000 pages of resulting papers and research). A survey of earlier work is found in [22], while a survey of recent approaches to FDD in HVAC appears in [33] and [34].

Numerous approaches have been suggested for fault detection in HVAC or industrial processes. Currently, the most common type of fault detection in production use is of the model-free, non-parametric form, such as issuing alarms on out-of-range sensors. Model-free approaches to fault detection and isolation can broadly be classified into: limit checking, installation of special sensors, installation of multiple sensors, and frequency analysis of plant measurements [22].

Model-based fault detection is rarer since it often requires detailed equipment models (as in [11]), which tend to be manufacturer and model dependent; manufacturers do not generally provide reference models. Model-based fault detection will often require significantly more expertise by the installer, and a longer setup time.

An expert systems approach is taken to FDD in [24], where a prototype FDD system was developed for use in compression refrigeration plants. The expert rules were designed by combining results from a failure mode and effect analysis (FMEA) study with knowledge obtained from interviews with system designers and service engineers. The expert rules analyzed deviations of some variables from their ideal estimated values (as given by a predefined non-linear statistical model).

In [16], Dexter designs a fuzzy rule set based on expert knowledge or computer simulations which can be said to represent normal operation and operation under different failure modes. Another partial fuzzy model is trained using operating data collected from the actual plant in question and compared to the reference models. Failures are identified by closeness of match between the trained model and the reference models. A similar approach is taken in [17], where a semi-qualitative model-based FDD methodology is used. Nonlinear reference models of normal operation and several different faults are provided and compared to a model generated on-line using a fuzzy identification scheme.

Separate black-box models were used for system and component failures in [51]. Autoregressive exogenous (ARX) and multi-layer back-propagation neural network models were used. Failure was detected by comparing the actual system output with the output predicted by the models.

Generally speaking, model-based fault detection often starts with a detailed model of the equipment in question, or spends considerable time generating such a model. Even those which do not use a transfer-function-based model (as in [51], where a neural network is used) require significant training beforehand and may only be applicable to the piece of equipment they were trained on. Other, rule-based, solutions (e.g., fuzzy-logic [16, 24, 65] or semi-qualitative [17]) must be keyed to a specific equipment type and might require more detailed information than is financially practical to collect in some situations.

4

Problem Domain

This chapter introduces the problem domain that the fault detection algorithm (to be developed later) will be applied to. Heating, Ventilation, and Air-Conditioning (HVAC) system architecture, unit modes, and fault categories are discussed in very brief detail.

The HVAC industry focuses on control and monitoring of equipment in a facility to ensure acceptable occupancy conditions. This includes temperature, pressure, and air-quality regulation of the hundreds or thousands of individual units at a facility. There is sometimes an overlap between HVAC and regulatory oversight (such as minimal fresh-air requirements per tenant or square foot), or life-safety systems (maintaining a negative pressure in a tuberculosis ward, or controlling temperature in a surgery room). A large site can collect and process several 100 million data samples per day however most of this information is not retained due to storage or bandwidth considerations.

4.1 Economic Considerations

According to the Department of Energy[1] about 53% of building energy end-use in America is spent on HVAC and lighting for a total expenditure of \$192.5B in 2006 dollars (Table 4.1). The total energy usage when lighting is excluded is 35.3%, (\$138.7B).

Some types of fault detection can make a big difference in the efficiency of a building.

[1] http://buildingsdatabook.eren.doe.gov

	Energy (Quad. BTU)	Energy (%)	Cost (2006-$)	Cost (%)
Space Heating	7.66	19.8%	$94.5B	22.8%
Lighting	6.86	17.7%	$53.8B	13.0%
Space Cooling	4.91	12.7%	$38.9B	9.4%
Ventilation	1.10	2.8%	$5.3B	1.3%

Table 4.1: DOE Buildings Energy End-Use Expenditure and End-Use Splits

While abrupt faults are often safety issues, incipient faults are more of a maintenance or efficiency concern [21]. This is because, at the early stages of failure, units with incipient faults will still be able to maintain set-point targets however will have to work harder to do so. According to [33], commercial buildings waste an estimated 15% to 30% ($20.8B – $41.61B annually) of their energy due to poorly maintained, degraded, or improperly controlled equipment.

Additionally, buildings which do not necessary have any incipient or abrupt faults, but which have been optimized for HVAC can see performance improvements of 5% to 32% ($6.94B to $44.38B annually) in annual energy consumption for lighting and HVAC [1, 64]. Some of these suboptimal design decisions or miscalibration issues may be considered "pre-existing" faults.

4.2 HVAC Architecture

An HVAC system is composed of many mechanical units that are nearly-identical. Each unit may be in one of several modes, with the mode dynamics for similar units being very similar. These dynamics drift slowly based on global system changes such as the slow change in the chill-water temperature (from, for example, a chill-water reset energy-savings algorithm), or static pressure in a duct (for example, gradual pressure increase when all space temperatures being fed by the duct meet set-point).

Different classes of units exist in an HVAC system (for example, cooling towers, chillers,

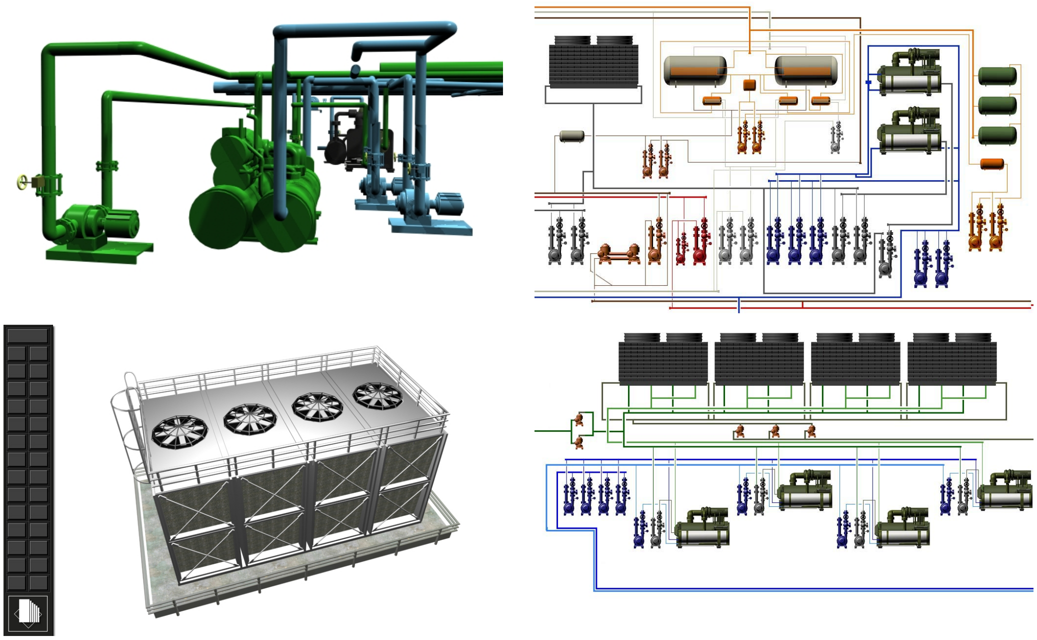


Figure 4.1: Chillers (top left) in the Central Plant (right) release heat from cooling towers (bottom left), which feed cold water to the AHU's.

air handling units, and variable or constant air-volume boxes); however, in this treatment, fault detection will be applied to a single class at a time.

4.3 Modes

We describe a mode as being representative of a unique model for operation. Units with dynamic modes may change modes either suddenly or gradually, while those with static modes are mode-fixed and do not change modes during the time in consideration.

4.3.1 Dynamic Modes

As an example of a system with dynamic modes, we consider a chill-water system. Plant chillers cool supply water to a temperature of 45^{o}F to 55^{o}F, based on outside air temperature. The supply water feeds many (e.g. 20-200) air handling units (Figure 4.3), some of which

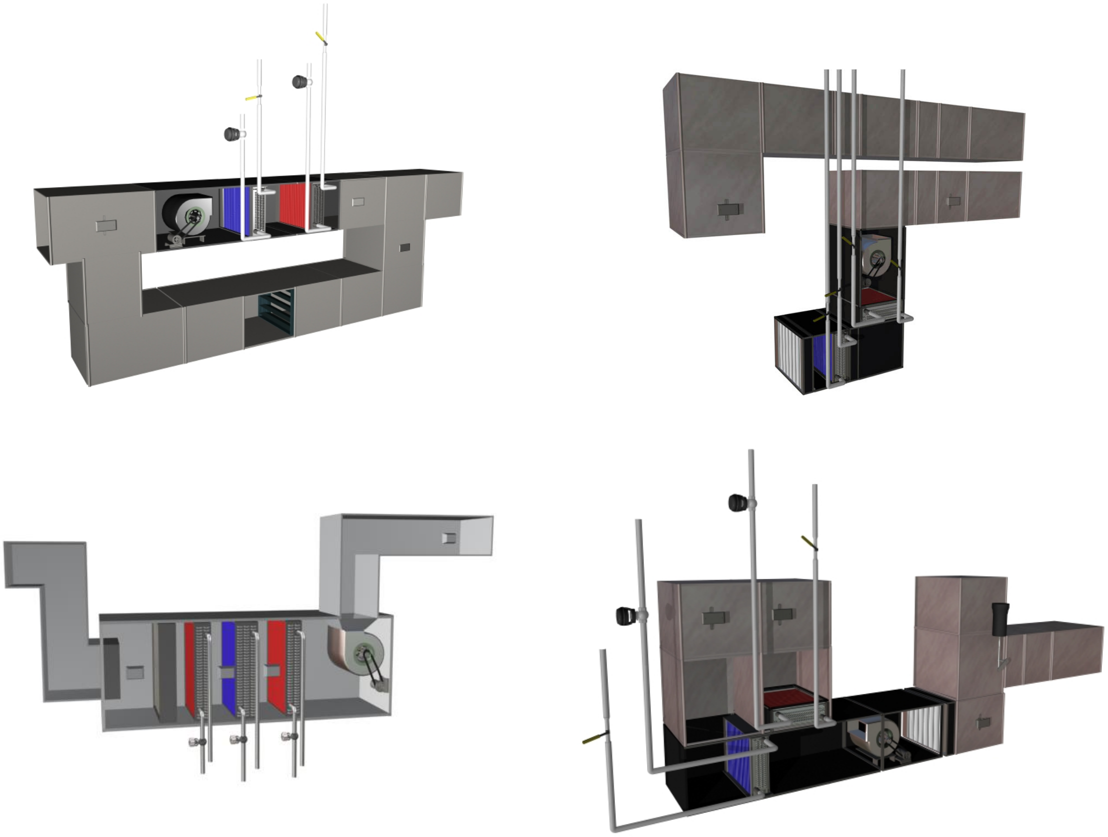


Figure 4.2: Despite similar mechanical descriptions, AHU's may come in a variety of configurations.

may have outside air "economizer-mode" enabled based on a combination of the outside-air enthalpy and the load requirements for the floor that the unit feeds. The two modes in this case are normal-cooling and economizer-mode.

Another example of dynamic modes can be found in combination cooling/heating valves. Such valves could be represented with two (cooling and heating) or three (cooling, dead-zone, and heating) modes.

4.3.2 Static Modes

Static modes represent operating dynamics that are different due to a permanent condition. While we may allow static modes to change, we assume that the likelihood of this happening

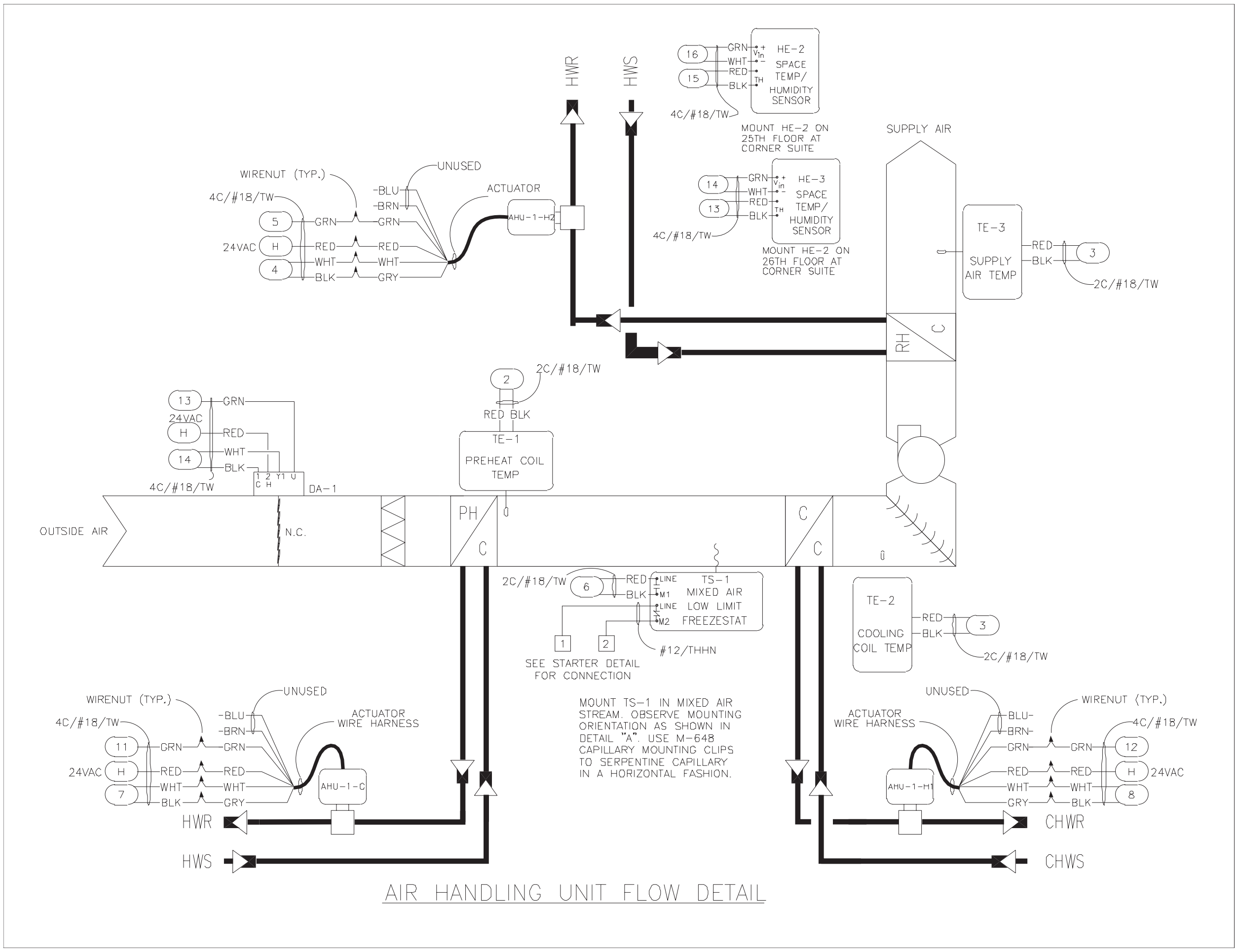
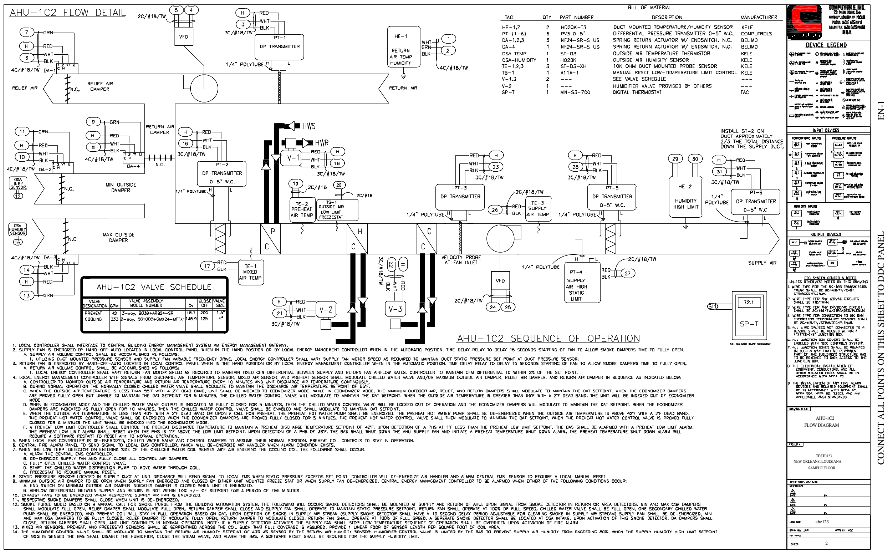


Figure 4.3: There are generally one or more Air Handling Units (pictured above) for each floor in a tall building.

during data collection for a single run is vanishingly small.

An example of a static mode is a system where there are two different sized ducts or pipes that feed a particular class of unit. If there are two different duct sizes used throughout the installation, then units connected to each duct could be considered to be in different modes.

Also, units by different manufacturers often operate under different dynamics. So, for a campus installation, some buildings may have air handling units designed by one manufacturer while others may have units designed by a different manufacturer (possibly because they were constructed at a later date). The permanent difference in dynamics between units could be represented by different modes of operation.



BILL OF MATERIAL

TAG	QTY	PART NUMBER	DESCRIPTION	MANUFACTURER
HE-1,2	2	HD20K-T3	DUCT MOUNTED TEMPERATURE/HUMIDITY SENSOR	KELE
PT-(1-6)	6	PV3 0-5"	DIFFERENTIAL PRESSURE TRANSMITTER 0-5" W.C.	COMPUTROLS
DA-1,2,3	3	NF24-SR-S US	SPRING RETURN ACTUATOR W/ ENDSWITCH, N.C.	BELIMO
DA-4	1	NF24-SR-S US	SPRING RETURN ACTUATOR W/ ENDSWITCH, N.O.	BELIMO
OSA TEMP	1	ST-03	OUTSIDE AIR TEMPERATURE THERMISTOR	KELE
OSA-HUMIDITY	1	H020K	OUTSIDE AIR HUMIDITY SENSOR	KELE
TE-1,2,3	3	ST-D3-XH	10K OHM DUCT MOUNTED PROBE SENSOR	KELE
TS-1	1	A11A-1	MANUAL RESET LOW-TEMPERATURE LIMIT CONTROL	KELE
V-1,3	2	---	SEE VALVE SCHEDULE	---
V-2	1	---	HUMIDIFIER VALVE PROVIDED BY OTHERS	---
SP-T	1	MN-S3-700	DIGITAL THERMOSTAT	TAC

AHU-1C2 VALVE SCHEDULE

VALVE DESIGNATION	GPM	VALVE ASSEMBLY MODEL NUMBER	Cv	CLOSE OFF	VALVE SIZE
PREHEAT	42	3-way, B339+ARB24-SR	18.7	200	1.5"
COOLING	333	2-Way, G6100C+GMX24-MFTX1	148.9	125	4"

AHU-1C2 SEQUENCE OF OPERATION

1. LOCAL CONTROLLER SHALL INTERFACE TO CENTRAL BUILDING ENERGY MANAGEMENT SYSTEM VIA ENERGY MANAGEMENT GATEWAY.
2. SUPPLY FAN IS ENERGIZED BY HAND-OFF-AUTO LOCATED IN LOCAL CONTROL PANEL WHEN IN THE HAND POSITION OR BY LOCAL ENERGY MANGEMENT CONTROLLER WHEN IN THE AUTOMATIC POSITION. TIME DELAY RELAY TO DELAY 15 SECONDS STARTING OF FAN TO ALLOW SMOKE DAMPERS TIME TO FULLY OPEN.
   A. SUPPLY AIR VOLUME CONTROL SHALL BE ACCOMPLISHED AS FOLLOWS:
      1. UTILIZING DUCT MOUNTED PRESSURE SENSOR AND SUPPLY FAN VARIABLE FREQUENCY DRIVE, LOCAL ENERGY CONTROLLER SHALL VARY SUPPLY FAN MOTOR SPEED AS REQUIRED TO MAINTAIN DUCT STATIC PRESSURE SET POINT AT DUCT PRESSURE SENSOR.
3. RETURN FAN IS ENERGIZED BY HAND-OFF-AUTO LOCATED IN LOCAL CONTROL PANEL WHEN IN THE HAND POSITION OR BY LOCAL ENERGY MANAGEMENT CONTROLLER WHEN IN THE AUTOMATIC POSITION. TIME DELAY RELAY TO DELAY 15 SECONDS STARTING OF FAN TO ALLOW SMOKE DAMPERS TIME TO FULLY OPEN.
   A. RETURN AIR VOLUME CONTROL SHALL BE ACCOMPLISHED AS FOLLOWS:
      1. LOCAL ENERGY CONTROLLER SHALL VARY RETURN FAN MOTOR SPEED AS REQUIRED TO MAINTAIN FIXED CFM DIFFERENTIAL BETWEEN SUPPLY AND RETURN FAN AIRFLOW RATES. CONTROLLER TO MAINTAIN CFM DIFFERENTIAL TO WITHIN 2% OF THE SET POINT.
4. LOCAL ENERGY MANAGEMENT CONTROLLER WITH SIGNALS FROM DISCHARGE AIR TEMPERATURE SENSOR, MIXED AIR SENSOR, AND PREHEAT SENSOR SHALL MODULATE CHILLED WATER VALVE AND/OR MAXIMUM OUTSIDE AIR DAMPER, RELIEF AIR DAMPER, AND RETURN AIR DAMPER IN SEQUENCE AS INDICATED BELOW:
   A. CONTROLLER TO MONITOR OUTSIDE AIR TEMPERATURE AND RETURN AIR TEMPERATURE EVERY 10 MINUTES AND UNIT DISCHARGE AIR TEMPERATURE CONTINOUSLY.
   B. DURING NORMAL OPERATION THE NORMALLY CLOSED CHILLED WATER VALVE SHALL MODULATE TO MAINTAIN THE DISCHARGE AIR TEMPERATURE SETPOINT OF 55°F.
   C. WHEN THE OUTSIDE AIR TEMPERATURE IS LESS THAN 55°F WITH A 2°F DEAD BAND, UNIT SHALL BE INDEXED TO ECONOMIZER MODE. WHILE IN ECONOMIZER MODE, THE MAXIMUM OUTDOOR AIR, RELIEF, AND RETURN DAMPERS SHALL MODULATE TO MAINTAIN THE DAT SETPOINT. WHEN THE ECONOMIZER DAMPERS ARE PROVED FULLY OPEN BUT UNABLE TO MAINTAIN THE DAT SETPOINT FOR 5 MINUTES, THE CHILLED WATER CONTROL VALVE WILL MODULATE TO MAINTAIN THE DAT SETPOINT. WHEN THE OUTSIDE AIR TEMPERATURE IS GREATER THAN 55°F WITH A 2°F DEAD BAND, THE UNIT WILL BE INDEXED OUT OF ECONOMIZER MODE.
   D. WHEN IN ECONOMIZER MODE AND THE CHILLED WATER VALVE OUTPUT IS INDICATED AS FULLY CLOSED FOR 5 MINUTES, THEN THE CHILLED WATER CONTROL VALVE WILL BE LOCKED OUT OF OPERATION AND THE ECONOMIZER DAMPERS WILL MODULATE TO MAINTAIN THE DAT SETPOINT. WHEN THE ECONOMIZER DAMPERS ARE INDICATED AS FULLY OPEN FOR 10 MINUTES, THEN THE CHILLED WATER CONTROL VALVE SHALL BE ENABLED AND SHALL MODULATE TO MAINTAIN DAT SETPOINT.
   E. WHEN THE OUTSIDE AIR TEMPERATURE IS LESS THAN 40°F WITH A 2°F DEAD BAND OR UPON A CALL FOR PREHEAT, THE PREHEAT HOT WATER PUMP SHALL BE ENERGIZED. THE PREHEAT HOT WATER PUMP SHALL BE DE-ENERGIZED WHEN THE OUTSIDE AIR TEMPERATURE IS ABOVE 42°F WITH A 2°F DEAD BAND. THE PREHEAT HOT WATER CONTROL VALVE SHALL BE ENERGIZED WHEN THE ECONOMIZER DAMPERS ARE PROVED FULLY CLOSED FOR 5 MINTUES. THE PREHEAT HOT WATER CONTROL VALVE SHALL THEN MODULATE TO MAINTAIN THE DAT SETPOINT. WHEN THE PREHEAT HOT WATER CONTROL VALVE IS PROVED FULLY CLOSED FOR 5 MINTUES THE UNIT SHALL BE INDEXED INTO THE ECONOMIZER MODE.
   F. A PREHEAT LOW LIMIT CONTROLLER SHALL CONTROL THE PREHEAT DISCHARGE TEMPERATURE TO MAINTAIN A PREHEAT DISCHARGE TEMPERATURE SETPOINT OF 42°F. UPON DETECTION OF A PHS AT 1°F LESS THAN THE PREHEAT LOW LIMIT SETPOINT, THE BAS SHALL BE ALARMED WITH A PREHEAT LOW LIMIT ALARM. THE PREHEAT LOW LIMIT ALARM SHALL RESET WHEN THE PHS IS 1°F ABOVE THE LOW LIMIT SETPOINT. UPON DETECTION OF A PHS OF 38°F, THE BAS SHALL SHUT DOWN THE AHU SUPPLY FAN AND INITIATE A PREHEAT TEMPERATURE SHUT DOWN ALARM. THE PREHEAT TEMPERATURE SHUT DOWN ALARM WILL REQUIRE A SOFTWARE RESTART TO RESET AHY TO NORMAL OPERATION.
5. WHEN LOCAL EMS CONTROLLER IS DE-ENERGIZED, CHILLED WATER VALVE AND CONTROL DAMPERS TO ASSUME THEIR NORMAL POSITION. PREHEAT COIL CONTROLS TO STAY IN OPERATION.
6. CENTRAL FIRE ALARM PANEL TO SEND SIGNAL TO LOCAL EMS CONTROLLER, WHICH WILL DE-ENERGIZE AIR HANDLER WHEN ALARM CONDITION EXISTS.
7. WHEN THE LOW TEMP. DETECTOR ON ENTERING SIDE OF THE CHILLDER WATER COIL SENSES 36°F AIR ENTERING THE COOLING COIL THE FOLLOWING SHALL OCCUR.
   A. ALARM THE CENTRAL EMS CONTROLLER.
   B. DE-ENERGIZE SUPPLY FAN AND FULLY CLOSE ALL CONTROL AIR DAMPERS.
   C. FULLY OPEN CHILLED WATER CONTROL VALVE.
   D. START THE CHILLED WATER DISTRIBUTION PUMP TO MOVE WATER THROUGH COIL.
   E. FREEZESTAT TO REQUIRE MANUAL RESET.
8. STATIC PRESSURE SENSOR LOCATED IN SUPPLY DUCT AT UNIT DISCHARGE WILL SEND SIGNAL TO LOCAL EMS WHEN STATIC PRESSURE EXCEEDS SET POINT. CONTROLLER WILL DE-ENERGIZE AIR HANDLER AND ALARM CENTRAL EMS. SENSOR TO REQUIRE A LOCAL MANUAL RESET.
9. MINIMUM OUTSIDE AIR DAMPER TO BE OPEN WHEN SUPPLY FAN ENERGIZED AND CLOSED BY EITHER UNIT MOUNTED FREEZE STAT OR WHEN SUPPLY FAN DE-ENERGIZED. CENTRAL ENERGY MANAGEMENT CONTROLLER TO BE ALARMED WHEN EITHER OF THE FOLLOWING CONDITIONS OCCUR:
   A. END SWITCH ON MINIMUM OUTSIDE AIR DAMPER INDICATES DAMPER IS CLOSED WHEN UNIT IS ENERGIZED.
   B. AIRFLOW DIFFERENTIAL BETWEEN SUPPLY AND RETURN IS NOT WITHIN 10% +/- OF SETPOINT FOR A PERIOD OF FIVE MINUTES.
10. EXHAUST FANS TO BE ENERGIZED WHEN RESPECTIVE SUPPLY AIR FAN IS ENERGIZED.
11. RESPECTIVE SMOKE DAMPERS SHALL CLOSE WHEN UNIT IS DE-ENERGIZED.
12. SMOKE PURGE MODE: BASED ON A MANUAL CALL FOR SMOKE PURGE FROM THE BUILDING AUTOMATION SYSTEM, THE FOLLOWING WILL OCCUR: SMOKE DETECTORS SHALL BE MOUNTED AT SUPPLY AND RETURN OF AHU, UPON SIGNAL FROM SMOKE DETECTOR IN RETURN OR AREA DETECTORS, MIN AND MAX OSA DAMPERS SHALL MODULATE FULL OPEN, RELIEF DAMPER SHALL MODULATE FULL OPEN, RETURN DAMPER SHALL CLOSE AND SUPPLY FAN SHALL OPERATE TO MAINTAIN STATIC PRESSURE SETPOINT, RETURN FAN SHALL OPERATE AT 100% OF FULL SPEED, CHILLED WATER VALVE SHALL BE FULL OPEN, ONE SECONDARY CHILLED WATER PUMP SHALL BE ENERGIZED, AND PREHEAT COIL WILL STAY IN FULL OPERATION BASED ON OAT. UPON DETETION OF SMOKE IN SUPPLY AIR STREAM (SUPPLY SMOKE DETECTOR SHALL HAVE A 10 SECOND DELAY PERIOD ADJUSTABLE FOR CLEARING SMOKE IN SUPPLY AIR STREAM) SUPPLY FAN SHALL BE DE-ENERGIZED, MIN AND MAX OSA DAMPERS TO BE FULLY CLOSED, RELIEF DAMPER TO MODULATE FULLY OPEN, RETURN DAMPER TO MODULATE CLOSED, RETURN FAN SHALL OPERATE AT 100% OF FULL SPEED. A SEPERATE SMOKE DETECTOR SHALL BE LOCATED AT OSA INTAKE. UPON ACTIVATION OF THIS SMOKE DETECTOR, DA DAMPERS SHALL CLOSE, RETURN DAMPERS SHALL OPEN, AND UNIT CONTINUES IN NORMAL OPERATION. NOTE: IF A SUPPLY DETECTOR ACTIVATES THE SUPPLY FAN SHALL STOP. LOW TEMPERATURE SEQUENCE OF OPERATION SHALL BE OVERRIDEN UPON ACTIVATION OF FIRE ALARM.
13. MIXED AIR SENSORS, PREHEAT, AND FREEZESTAT SENSORS SHALL BE SERPENTINED ACROSS THE COIL SUCH THAT FULL COVERAGE IS ASSURED. PROVIDE 1 LINEAR FOOT OF SENSOR LENGTH PER SQUARE FOOT OF COIL AREA.
14. THE HUMIDIFIER CONTROL VALVE SHALL BE MODULATE TO MAINTAIN THE RETURN AIR HUMIDITY SETPOINT OF 40% AS SENSED BY THE RETURN AIR HUMIDITY SENSOR. HUMIDIFIER CONTROL VALVE IS LIMITED BY THE BAS TO PREVENT SUPPLY AIR HUMIDITY FROM EXCEEDING 80%. WHEN THE SUPPLY HUMIDTY HIGH LIMIT SETPOINT OF 95% IS SENSED THE BAS SHALL DISABLE THE HUMIDIFIER, CLOSE THE STEAM VALVE, AND ALARM THE BAS. A SOFTWARE RESET SHALL BE REQUIRED FOR THE SUPPLY HUMIDITY LIMIT.

DDC SYSTEM CONTROLS NOTES
UNLESS OTHERWISE NOTED ON THIS DRAWING

1. WIRE TYPE FOR THE RS-485 TRANSMISSION TRUNK SHALL BE 2C/#18/TW/SHD/STRANDED/PLENUM.
2. WIRE TYPE FOR ANY 120VAC CIRCUITS SHALL BE #12/THHN.
3. WIRE TYPE FOR ANY 24VDC/AC CIRCUIT SHALL BE 2C/#16/TW/STRANDED/PLENUM.
4. WIRE TYPE FOR CONNECTION TO 10K OHM THERMISTOR TEMPERATURE SENSORS SHALL BE 2C/#18/TW/STRANDED/PLENUM.
5. ALL WIRE SPLICES NOT CONNECTED TO A DEVICE SHALL BE HOUSED WITHIN A 4"X4"X2-1/8" JUNCTION BOX.
6. ALL JUNCTION BOX COVERS SHALL BE LABELED WITH "DDC CONTROLS SYSTEM".
7. ALL JUNCTION BOXES SHALL BE MOUNTED IN SUCH A WAY THAT NO PERMANENT PART OF THE BUILDING'S STRUCTURE HAS TO BE REMOVED TO GAIN ACCESS TO THE JUNCTION BOX.
8. THE ELECTRICAL INSTALLATION OF EQUIPMENT, CONDUCTORS, AND ALL OTHER RELATED ITEMS SHALL BE IN ACCORDANCE WITH NFPA 70.
9. THE INSTALLATION OF ANY FIRE ALARM DEVICES AND RELATED EQUIPMENT SHALL BE IN ACCORDANCE WITH NFPA 72, NFPA 90A, NFPA 101, SICC), AND ANY APPLICABLE ANSI STANDARDS.

DRAWING TITLE
AHU-1C2
FLOW DIAGRAM

FACILITY
TEST0123
NEW ORLEANS, LOUISIANA
SAMPLE FLOOR

ISSUE DATE: 05/13/08
REVISIONS
JOB NO: abc123
DRAWN BY: JSR APP'D BY: MDC
SHEET: 2

Figure 4.4: More complex Air Handling Units may have additional features.

Note that units with different static modes may also have dynamic modes as in the case of differently manufactured air handling units with economizer-mode outside air dampers.

4.3.3 Combination Modes

Sometimes, units will have a combination of static and dynamic modes. For example, a system which has two types of variable air volume boxes (one with heating strips and the other with heating coils). We might describe the modes as strip-heating, strip-cooling, coil-heating, and coil-cooling.

4.4 Types of Failure

While it is impossible to enumerate all possible failures which could occur in an HVAC system, we can broadly categorize them by expected effect and onset characteristics.

Note that some of the following failures exist in all HVAC equipment to a degree (for

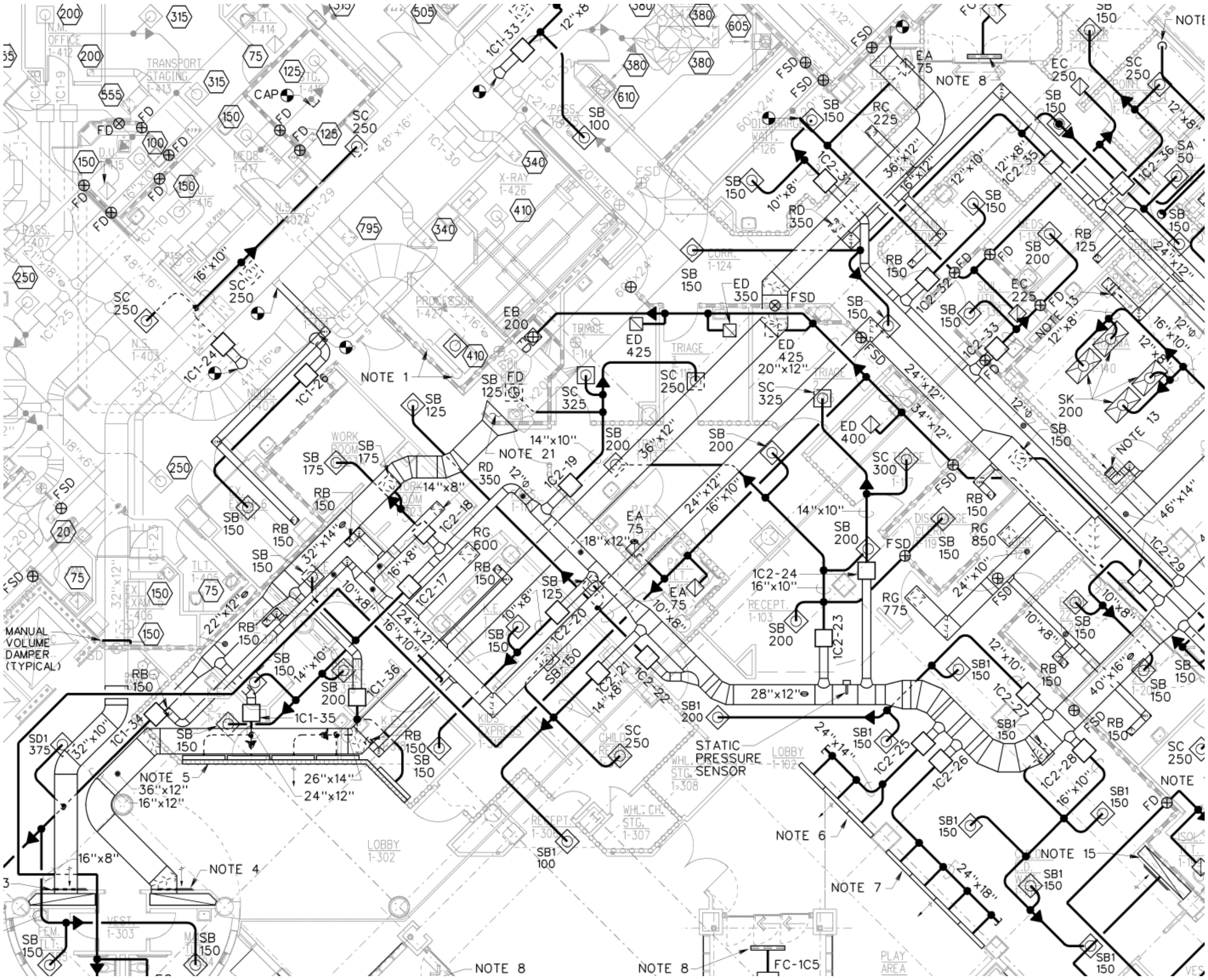


Figure 4.5: HVAC systems extend throughout large buildings. Pictured above is a small section of a floor plan.

example, sensor bias) however we consider them failures only if they are present to a greater than normal degree (where "normal" is described by somewhat arbitrarily chosen likelihood bounds).

4.4.1 Effect

Additive Measurement Failure Sensor bias normally presents as an additive error. The specification sheets of sensors normally provide error limits, which generally present as sensor bias, rather than zero-mean additive noise. Sensor bias is generally a much greater problem than additive noise since it cannot be integrated out of the measurements.

Multiplicative Measurement Failure A dead or reduced-effect sensor can be modelled as a multiplicative error.

Additive Process Failure Disturbances, leaks, or unknown/unmeasured inputs in the system can be modelled as additive process failures.

Multiplicative Process Failure Deterioration of equipment, reduction of efficacy due to calcification, coil freeze, pipe clogging, etc. They can be represented through a transformation of the state transition function.

Structural Failure Failures which result in a change in performance, but cannot be classified as one of the above categories. Examples might include assigning an incorrect feedback point for temperature control, measurements being "stuck" due to high network traffic, or incorrect programming on a unit.

4.4.2 Onset

Similarly, the different types of failure we expect to encounter can be classified as follows:

Sudden Failure The unit fails in a way that is immediately noticeable. This is considered an easy fault to diagnose, but ultimately depends on the magnitude of failure. Examples include:

- Sudden sensor failure.
- Rupture of an actuator diaphragm.
- The severing of sensor or actuator signal/power lines due to construction or human error.

- The failure or incorrect programming of a controller responsible for some of the internal processes of the unit.

Incipient Failure The unit fails slowly such that the change of dynamics is not noticeable over a short time-frame. A failure may be considered slow if the dynamics change due to the fault occurring slower than the normal (acceptable) drift in system dynamics. This is considered a difficult fault to diagnose. Examples include:

- Shortage of coolant in a heat-exchange system due to a slow leak.
- Development of hysteresis or stiction in valves.
- Difficulty for a fan to generate duct pressure due to a dirty filter.

Pre-existing Failure The unit does not operate similar to other units in its mode due to a pre-existing and possibly permanent fault. This is generally considered a difficult problem and is not usually considered by FDD techniques that rely on on-line parameter estimation or model fitting — since other FDD methodologies do not consider unit-groups, there is no way to obtain a baseline. Examples include:

- A variable air-volume box that has to reheat the air while others on the floor are cooling due to it being located too close to a fan or the incorrect use of reduction sleeves.
- The mislabelling or misconnection of sensor outputs during installation, where a sensor intended to be connected to one unit is in fact connected to a different unit.
- A valve connected in reverse. The valve will still work, however the operating characteristics will be sub-optimal, or the valve may fail to fully close when under pressure.

- Disconnected actuator pressure lines. It sometimes happens that a building technician will forget to reconnect a pressurized air tube or wire to an actuator after working on it. This failure could, under the correct conditions, go unnoticed for months. For example, if the valve defaults to open in a chronically underpowered situation, i.e. where the valve needs to be open most of the time, it is possible that there will be no deviation in performance until outside temperatures change significantly. While this is not a permanent fault, it can be classified as a pre-existing one if it occurred before data acquisition started.

4.5 Model Structure

This section discusses a number of different models which may be encountered in HVAC systems. A survey of model-based fault detection is given in [22], in addition to the many HVAC fault detection papers found in [30] and [19].

4.5.1 Model-free Fault Detection

The most common technique used to implement fault detection in HVAC Building Automation Systems (BAS) is the model-free approach, the most common being setting limits on acceptable value ranges:

Limit checking Simple, hard-coded limits for sensors. For example, temperature sensor limits, current limits.

Special Sensors Perform fault detection on behalf of system.

Redundancy For sensor failure only.

Expert Rules Domain-specific logical rules.

4.5.2 Parametric Models versus Physical Models

While there are reference models available by different manufacturers and organizations [3, 27, 35], most abstract away the complex internal behavior that may not be necessary to represent the input-output dynamics of the unit. Observation of some states needed to make the complex models useful may require sensors readings that are only available to the manufacturer. Furthermore, each manufacturer might have a different model, and most actual installations do not include the types of sensors that might be required to use complex models.

Simpler models are usually represented as a transfer function of the form [36],

$$G\left(s\right) = \frac{K_s}{1 + T_c s} \exp^{-T_D s}$$

where K_s is the system gain, T_c is the time constant, T_D is the dead-time, and s is the Laplace variable.

And despite the fact that most HVAC subsystems are most accurately modelled as non-linear distributed-parameter systems, the most commonly used model is given by [12]:

$$G\left(s\right) = \frac{K_s}{\left(1 + T_1 s\right)\left(1 + T_2 s\right)} \exp^{-T_D s}$$

These simpler models can be considered to be black-box models, since most of the more complicated dynamics that result in the observed behavior have been abstracted out.

4.5.3 Linear models

Auto-Regressive Moving Average with Exogenous Input (ARMAX) Model

Simple single-input, single-output linear transfer functions can be represented using an ARMAX model,

$$\begin{aligned}
A &= \left[\begin{array}{ccc} a_1 & a_2 & \dots \end{array} \right]^T \\
B &= \left[\begin{array}{ccc} b_0 & b_1 & \dots \end{array} \right]^T \\
Z_k &= \left[\begin{array}{ccc} z_{k-1} & z_{k-2} & \dots \end{array} \right]^T \\
U_k &= \left[\begin{array}{ccc} u_k & u_{k-1} & \dots \end{array} \right]^T \\
z_k &= A^T Z_k + B^T U_k + e_k
\end{aligned}$$

where the lengths of A and B are not necessarily the same. U_k and Z_k are input and observation vectors and are sized accordingly, and e_k is an Exogenous random input, which we will assume without loss of generality to be zero-mean.

Non-Linear Auto-Regressive Moving Average with Exogenous Input (NARMAX) Model

Some transfer functions that are non-linear only in the inputs and/or outputs can be represented using a NARMAX model:

$$\begin{aligned}
A &= \begin{bmatrix} a_1 & a_2 & \dots \end{bmatrix}^T \\
B &= \begin{bmatrix} b_0 & b_1 & \dots \end{bmatrix}^T \\
Z_k &= \begin{bmatrix} z_{k-1} & z_{k-2} & \dots \end{bmatrix}^T \\
U_k &= \begin{bmatrix} u_k & u_{k-1} & \dots \end{bmatrix}^T \\
z_k &= A^T f_z\left(Z_k\right) + B^T f_u\left(U_k\right) + e_k
\end{aligned}$$

where the lengths of A and B are not necessarily the same. U_k and Z_k are input and observation vectors and are sized accordingly, and e_k is an exogenous random input, which we will assume without loss of generality to be zero-mean. $f_z(Z_k)$ and $f_u(U_k)$ are non-linear functions that return vector results.

A NARMAX model may be solved using the same techniques as for an ARMAX model, since they are essentially the same after a transformation in the given data, U^k and Z^k.

While a NARMAX model uses non-linear functions of the inputs and observations, the parameters are multiplied by the transformed observations in a linear fashion. For this reason, NARMAX models are included in the linear models sections since standard linear techniques may be used to solve this class of model.

Exogenous Input

While both ARMAX and NARMAX have an exogenous random input whose covariance may assist in detection, we will only refer to *ARMA (shorthand notation for both ARMA and

NARMA) models below. This is because unknown inputs will be handled as process and measurement noise. For the problem domain under consideration, measurement noise is an easily obtained parameter, while process noise can plausibly be given or estimated.

ARMA State-space Representation

The parameter identification problem can be framed as a state-space formulation,

$$\begin{aligned}\theta_{k|k-1}^{(m,s)} &= F_{k-1}^{(m,s)}\theta_{k-1}^{(m,s)} + G_{k-1}^{(m,s)}w_{k-1}^{(m,s)}\\ z_k^{(m,s)} &= H_k^{(s)}\theta_{k|k-1}^{(m,s)} + v_k^{(m,s)}\end{aligned}$$

where $w_{k-1}^{(m,s)}$ and $v_k^{(m,s)}$ are the process and measurement noise with given distributions, $\theta^{(m,s)}$ is the parameter vector and $H_k^{(s)}$ is a unit-specific input and observation vector:

$$\begin{aligned}w_{k-1}^{(m,s)} &\sim \mathcal{N}\left(\bar{w}_{k-1}^{(m,s)}, Q_{k-1}^{(m,s)}\right)\\ v_k^{(m,s)} &\sim \mathcal{N}\left(\bar{v}_k^{(m,s)}, R_k^{(m,s)}\right)\\ \hat{\theta}^{(m,s)} &\triangleq \left[A^T|B^T\right]^T\\ H^{(s)} &\triangleq \left[f_z\left(Z_k^{(s)}\right)|f_u\left(U_k^{(s)}\right)\right]\end{aligned}$$

4.5.4 Non-linear Models

In general, any non-linear model may be used since there are well-known and effective ways to estimate the parameters (we will consider the Unscented Transform Filter in Section A.3). If the model has internal, unmeasurable, states, then the parameter estimation algorithm must take these into consideration by performing joint parameter-state estimation.

5

General Formulation

Before the fault detection problem may be addressed, the problem must be formally stated. This chapter attempts to produce a general formulation for the problem at hand, and provides concrete examples to illustrate possible choices of models, states, and parameters. Finally, a general approach is proposed to solve the fault detection problem.

5.1 Problem Statement

We consider a class of large-scale dynamic systems incorporating (a large number of) subsystems that have similar (or possibly, identical) structure and operate in a coupled manner as explained below.

Each subsystem, referred to as a unit, is described by the following model:

$$\xi_{k+1}^{(s)} = f_k^{(s)}\left(\xi_k^{(s)}, \theta_k^{(s)}, w_{\xi_k}^{(s)}\right) \tag{5.1}$$

$$\theta_{k+1}^{(s)} = g_k\left(\theta_k^{(s)}, m_k^{(s)}, w_{\theta_k}^{(s)}\right) \tag{5.2}$$

$$z_k^{(s)} = h_k^{(s)}\left(\xi_k^{(s)}, \theta_k^{(s)}, v_k^{(s)}\right) \tag{5.3}$$

where $k = 0, 1, 2, \ldots, L$ is the time index, $s = 1, 2, \ldots, S$ is the unit index, $x_k^{(s)} = \left[\xi_k^{(s)\prime}, \theta_k^{(s)\prime}\right]'$ is the base-state vector of unit s, $w_{\xi_k}^{(s)} \sim \mathcal{N}\left(0, Q_{\xi_k}^{(s)}\right)$, $w_{\theta_k}^{(s)} \sim \mathcal{N}\left(0, Q_{\theta_k}\right)$ and $v_k^{(s)} \sim \mathcal{N}\left(0, R_k^{(s)}\right)$ denote white and mutually independent Gaussian process and measurement noises, respectively, and $f_k^{(s)}$, g_k, $h_k^{(s)}$ are known state and measurement functions.

The mode of operation of unit s is modeled through a Markov chain $\left\langle m_k^{(s)} \right\rangle_{k=0,1,\ldots}$ with states $m_k^{(s)} \in \mathbb{M} = \{1, 2, \ldots, M\}$ and initial and transition probabilities as given by

$$P\left\{m_0^{(s)} = i\right\} = \mu_0 \tag{5.4}$$

$$P\left\{m_{k+1}^{(s)} = j | m_k^{(s)} = i\right\} = \pi_{ij} \tag{5.5}$$

for $i, j = 1, \ldots, M$.

The ultimate aim is to detect whether a fault has occurred in one or more of the units, $s = 1, 2, \ldots, S$, given observation data $Z_k = \{z_\kappa^{(s)} : s = 1, 2, \ldots, S\}_{\kappa=0}^{k}$.

The special structure and features of the above multi-unit hybrid system model can be exploited to achieve this fault detection goal. First, note that the state $x_k^{(s)} = \left[\xi_k^{(s)\prime}, \theta_k^{(s)\prime}\right]'$ of each unit is separated into two parts $\xi_k^{(s)}$ and $\theta_k^{(s)}$, the latter of which does not depend on the former.

While the process $\xi_k^{(s)}$ is determined through the unit-specific state transition model $f_k^{(s)}$ in (5.1), the process $\theta_k^{(s)}$ is determined through the state transition model g_k which is common for all units. Thus g_k models the common part in the structures of all units, meaning that for all units this part operates in the same manner. Furthermore, as seen from (5.4)–(5.5) the Markov transition model for the modal state $m_k^{(s)}$, as well as the initial probabilities, is common for all units, and the statistical properties and parameters of $\theta_0^{(s)}$ and $w_{\theta_k}^{(s)}$ are also assumed to be the same for all units. Under these circumstances the processes $\theta_k^{(s)}$, $s = 1, 2, \ldots, S$, will have the same statistical properties.

5.1.1 Assumptions

We assume that the state and measurement functions $\left(f_k^{(s)}, g_k, h_k^{(s)}\right)$ are known, as is the transition probability matrix $\left(P\left\{m_{k+1}^{(s)} = j | m_k^{(s)} = i\right\}\right)$, while the mode (m), states $\left(\xi_k^{(s)}\right.$

and $\theta_k^{(s)}\Big)$, and initial mode probabilities $\left(P\left\{m_0^{(s)} = i\right\}\right)$ are unknown.

We also place the restriction that no two modes may have $f_k^{(s)}$ and $h_k^{(s)}$ such that the observable distribution is identical for the two different modes. This can be considered a "mode observability" requirement.

Finally, we assume that if a fault occurs in a unit, s', it will inevitably exhibit itself in changed statistical properties of $\theta_k^{(s\prime)}$ as compared to the remaining processes, $\theta_k^{(s)}$, $s \neq s'$.

5.2 Examples

A number of examples are given to illustrate possible choices of global parameters and internal states. All examples assume:

$$w_k, v_k \sim \mathcal{N}\left(0, 1^2\right)$$

A superscript of s denotes a value that is related to unit $s \in \{1, 2, \ldots, S\}$, while a superscript of m denotes a value that is related to mode $m \in \{1, 2, \ldots, M\}$, ΔT is the sampling interval, Q and R are process and measurement noise covariance matrices, which are selected appropriately.

5.2.1 Dynamic ARMA: Three-Way Valve

A set of simple 2nd-order auto regressive moving average (ARMA) models may be used to describe the behavior of a three-way combination heating-cooling valve with a dead-band (illustrated in Figure 5.1). We designate cooling mode as mode 1, dead-band as mode 2, and heating mode as mode 3.

We define a, b to be the autoregressive and moving average parameters, respectively. Valve position and temperature are given by s and t. $q^{-i}x_k$ is the delay operator which

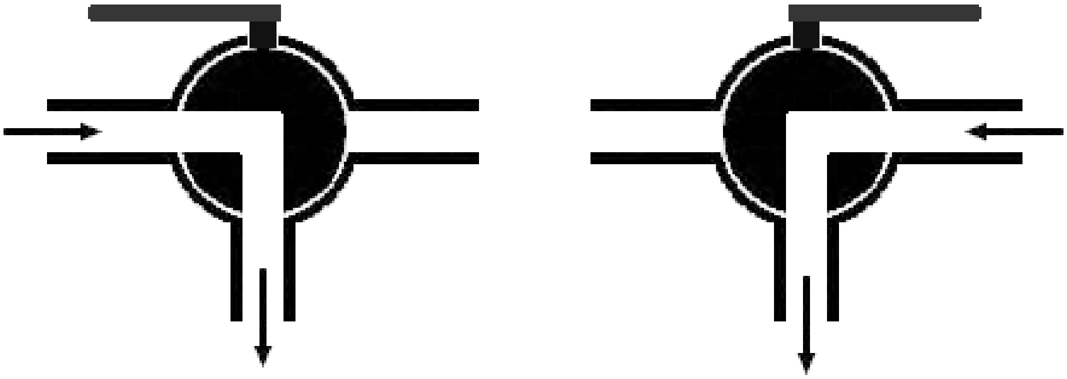


Figure 5.1: A simple three-way ball valve to select cold or hot water.

returns x_{k-i}. The output we are interested in is temperature z. We have:

$$x \triangleq \begin{bmatrix} a_1 & a_2 & b_0 & b_1 \end{bmatrix}^T \tag{5.6a}$$

$$F_k = I_{4,4}, \quad H_k = \begin{bmatrix} q^{-1}z & q^{-2}z & q^0 s & q^{-1}s \end{bmatrix}_k \tag{5.6b}$$

$$x_k = F_k x_{k-1} + \sqrt{Q} w_k \tag{5.6c}$$

$$z_k = H_k x_k + \sqrt{R} v_k \tag{5.6d}$$

The parameters and local states for modes 1 and 3 are given by

$$\xi_k^{(m,s)} \triangleq \text{[empty]} \tag{5.7a}$$

$$\theta_k^{(m,s)} \triangleq \left(\begin{bmatrix} a_1 & a_2 & b_0 & b_1 \end{bmatrix}_k^{(m,s)} \right)^T \tag{5.7b}$$

For mode 2 (dead-band), we fix the parameters as:

$$\theta_k^{(2,s)} \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T \tag{5.8a}$$

5.2.2 Dynamic NARMA – Supply Air Temperature

A 2nd-order NARMA model may be used to represent the relationship between valve position and differential pressure to chill water supply temperature.

We can define a, b, c to be the autoregressive, valve position, and flow rate parameters, respectively. Valve position and differential pressure measurements are given by s and Δp. $q^{-i}x_k$ is the delay operator which returns x_{k-i}. The output we are interested in is temperature, z. We have:

$$
\begin{aligned}
x &\triangleq \begin{bmatrix} a_1 & a_2 & b_0 & b_1 & c_0 & c_1 \end{bmatrix}^T \\
F_k &= I_{6,6}, \;\; H_k = \begin{bmatrix} q^{-1}z & q^{-2}z & q^0 s & q^{-1}s & q^0\sqrt{\Delta p} & q^{-1}\sqrt{\Delta p} \end{bmatrix}_k \\
x_k &= F_k x_{k-1} + \sqrt{Q}w_k \\
z_k &= H_k x_k + \sqrt{R}v_k
\end{aligned}
$$

Note that any number of inputs (we use 2-input here) and/or outputs can be stacked in a similar fashion to create a MIMO-system while retaining the general ARMA structure.

The parameters and local states are given by,

$$
\begin{aligned}
\xi_k^{(m,s)} &\triangleq [\text{empty}] \\
\theta_k^{(m,s)} &\triangleq \left(\begin{bmatrix} a_1 & a_2 & b_0 & b_1 & c_0 & c_1 \end{bmatrix}_k^{(m,s)} \right)^T
\end{aligned}
$$

5.2.3 Static Nonlinear – Valve Stiction

Stiction is the resistance to the start of motion caused by static friction and is a common problem in real-world valves. Excessive stiction can be considered a fault condition.

Given an applied force F_u, a static resistance force F_s, and the current velocity v, we can model the resulting velocity after stiction as:

$$v_s\left(v, K, F_u, F_s\right) = \begin{cases} 0, & v = 0 \text{ and } |F_u| < F_s \\ v + KF_u, & \text{otherwise} \end{cases}$$

If we define our state, measurement, and input vectors as:

$$x \triangleq \begin{bmatrix} s \\ v \\ F_s \\ K \end{bmatrix} = \begin{bmatrix} \text{Valve position} \\ \text{Valve velocity} \\ \text{Static resistance} \\ \text{Action Gain} \end{bmatrix}$$
$$z \triangleq \begin{bmatrix} s \end{bmatrix} = \begin{bmatrix} \text{Valve position} \end{bmatrix}$$
$$u \triangleq [F_u] = [\text{Applied force}]$$

we can describe the system dynamics using the following equations:

$$x_k = \begin{bmatrix} s + \Delta T v \\ v_s\left(v, K, F_u, F_s\right) \\ F_s \\ K \end{bmatrix}_{k-1} + \sqrt{Q} w_k$$

$$z_k = \begin{bmatrix} s \end{bmatrix}_k + \sqrt{R} v_k$$

Here, the global parameters are the static resistance and action gain, while the local states are valve position and velocity:

$$\theta \triangleq [F_s, K]\,, \;\; \xi \triangleq [s, v]^T$$

For this example, we assume an installation with 300 pneumatic valves of a particular size, from 4 different manufacturers. In our example, the valves from the different manufacturers have slightly different dynamic statistics, which we represent using 4 different static modes.

Note that HVAC installations generally do not provide actuator position feedback; however assuming a measurement of the position more succinctly illustrates fault detection based on stiction. To realistically make use of stiction-based fault detection, further dynamics should be introduced to correlate the input position with the actual measurements available (this is generally achieved via the measurement function).

5.2.4 Dynamic Nonlinear with Internal Parameters – VAV Box

Normally, a pressure sensor is used to measure air flow in the building automation industry. The flow through a duct is given by [12]:

$$f_a = A f C_v \sqrt{\frac{\Delta p_v}{g_s}}$$

where f_a is flow, A is the cross-sectional area of the duct, f is the flow characteristic of the valve, C_v is the flow coefficient, Δp_v is the impact pressure on the sensor (or differential pressure for the flow-cross class of pressure sensors), and g_s is the specific gravity of the fluid. Lumping the unknowns into C_v, we get a simple relationship for pressure-flow,

$$\Delta p_v = C_v f_a^2 = R_f^2$$

where we absorb $\sqrt{C_v}$ into R_f, since we are not interested in the actual flow rate, but only need a proportional flow rate.

If we define our state, measurement, and input vectors as:

$$x \triangleq \begin{bmatrix} T_r \\ R_f \\ V \\ D \end{bmatrix} = \begin{bmatrix} \text{Room (return-air) temperature} \\ \text{Duct air flow} \\ \text{Space volume} \\ \text{Disturbance} \end{bmatrix}$$

$$z \triangleq \begin{bmatrix} T_r \\ \Delta p_v \end{bmatrix} = \begin{bmatrix} \text{Room (return-air) temperature} \\ \text{Impact pressure} \end{bmatrix}$$

$$u \triangleq [T_s] = [\text{Supply air temperature}]$$

where we assume that duct air flow, space volume, and disturbance are all in scaled by unknown constants, we can describe the system dynamics using the following equations:

$$x_k = \begin{bmatrix} T_r - \frac{\Delta T R_f (T_r - T_s)}{V} + D \\ R_f \\ V \\ D \end{bmatrix}_{k-1} + \frac{\Delta T}{60} \begin{bmatrix} 0.2 & 0 & 0 & 0 \\ 0 & 25 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} w_k$$

$$z_k = \begin{bmatrix} T_r \\ R_f^2 \end{bmatrix}_k + \begin{bmatrix} 2 & 0 \\ 0 & 0.1 \end{bmatrix} w_k$$

The (square root of the) process noise covariance matrix is designed using typical values of the amount of change in a minute. Note that, while volume does not change, opening a door to an adjacent space does have the effect of changing the apparent volume. Also, the absolute value of volume is generally large compared to the other states, so a value of 10 does not constitute a large variance.

In this case, the global parameter would be the disturbance, D since, even though volume, V, is technically a parameter, it is not shared among units and so will be treated as a state:

$$\theta \triangleq [D] , \;\; \xi \triangleq [T_r, T_s, R_f, V]^T$$

6

Algorithms

The expectation maximization (EM) algorithm for a static multi-unit system with no parameter drift is derived in Appendix B. For systems with parameter drift, it does not help to batch-process the data since the optimal parameters at the end of the data are not the same as those at the beginning.

One possible solution is to update the EM estimator sequentially, i.e. perform a single iteration for the data at a single time, using the final condition of the last sequential iteration to initialize the starting point of the current one.

As the equivalent of one batch iteration, this sequential form of EM can be performed forwards or backwards (for invertible models) in time for all available time samples. The sequential EM algorithm can then be iterated, alternating forwards and backwards, until it converges.

For dynamic systems, we could make the assumption that no jumps occur during the period in question however this could seriously impact performance and correctness. A better mechanism, and one which we will make use of, is to explicitly account for jumps via a transition probability matrix.

6.1 System Model

The structure we will consider is a state-space system with additive Gaussian process and measurement noise,

$$x_k = f_{k-1}\left(x_{k-1}, u_k\right) + w_k$$
$$z_k = h_k\left(x_k, u_k\right) + v_k$$

where u_k is the input,

$$x_k \triangleq \left[\theta_k^T | \psi_k^T\right]^T$$

and θ_k, ψ_k are the unit- and mode-specific parameter vector and internal state respectively.

We denote an estimator suitable for this system with the following function:

$$\left[\hat{x}_k, \Sigma_k, L_k\right] = \texttt{Filt}\left(z_k, u_k; x_{k-1}, \Sigma_{k-1}\right)$$

where $\hat{x}_k$ is the estimate of $[\theta_k, \psi_k]$, Σ_k is a finite-state statistic describing the distribution of $\hat{x}_k$ such that $(\hat{x}_k, \Sigma_k)$ is a sufficient statistic (for the Gaussian case, this would be covariance), and the likelihood of the measurement, z_k, is

$$L_k \triangleq \mathcal{L}\left(z_k | u_k, x_{k-1}, \Sigma_{k-1}\right)$$

As shorthand notation, we define:

$$\check{x}_k^{(m,s)} \triangleq \left[\check{\xi}_k^{(m,s)\prime}, \check{\theta}_k^{(m,s)\prime}\right]' = [\text{local, local}]$$
$$\mathring{x}_k^{(m,s)} \triangleq \left[\check{\xi}_k^{(m,s)\prime}, \mathring{\theta}_k^{(m,s)\prime}\right]' = [\text{local, global}]$$
$$\check{x}_k^{(0,s)} \triangleq \text{active mode matched estimator for unit } s$$

6.2 Static Mode Systems

If we assume that the mode does not change during the estimation process, i.e., the mode transition probability matrix $\Pi = \mathrm{I}$, then only the current estimate is needed for estimating the mixture distribution at the current time, and since the current estimate includes information from all measurements up to that point in time, earlier estimates can be ignored.

This simply means that some initialization period is required before the parameter estimates can be considered accurate. The estimation of the common distribution depends only on information contained in $\{\hat{\theta}_k^{(s)}\}_{s=1}^{S}$ although the estimate of the common process contains information which could improve local estimates.

Note that, since the mode does not jump, we only need to run one estimator for each unit to obtain $\check{x}_k^{(s)}, \check{\Sigma}_k^{(s)}$, however to compute the likelihood that $z_k^{(s)}$ came from mode m, we will still need one filter per mode for each unit to generate the measurement residual used in the likelihood calculation.

- **Initialization**: In the absence of any relevant data, random initial parameter and state starting points and sufficiently diffuse covariances are chosen. Each unit is assumed to be in a mode, sampled according to the initial probability masses, until later reassigned by the mode change hypothesis test.

- **Estimation:** For each unit $s = 1, \ldots, S$ perform

  - *Mode Matched Estimation:* For each mode $m = 1, \ldots, M$ perform one-step filter update using the global (estimated mixture) parameters, and evaluate the likelihood $L_k^{(m,s)}$

$$\left[L_k^{(m,s)}, \check{x}_k^{(m,s)}, \check{\Sigma}_k^{(m,s)}\right] = \texttt{Filt}\left[z_k^{(s)}; \mathring{x}_{k-1}^{(m,s)}, \mathring{\Sigma}_{k-1}^{(m,s)}\right] \tag{6.2}$$

  - *Active Mode Matched Estimation:* Perform one-step filter update of the active mode estimator

$$\left[\breve{x}_k^{(0,s)}, \breve{\Sigma}_k^{(0,s)}\right] = \texttt{Filt}\left[z_k^{(s)}; \breve{x}_{k-1}^{(0,s)}, \breve{\Sigma}_{k-1}^{(0,s)}\right] \tag{6.3}$$

- **Active Mode Change Detection:** For each unit $s = 1, \ldots, S$ perform a statistical test using the model likelihoods $\{L_k^{(m,s)} : m = 1, \ldots, M\}$ to establish whether a mode change has occurred

$$\texttt{Detect}\left[\{L_k^{(m,s)} : m = 1, \ldots, M\}\right] \tag{6.4}$$

  Upon detection of an active mode change from $m_\kappa^{(s)}$ to m_κ^* that occurred at time $\kappa \leq k$, reinitialize:

$$\left[\breve{x}_k^{(0,s)}, \breve{\Sigma}_k^{(0,s)}\right] = \texttt{Reinit}\left[\mathring{x}_\kappa^{(m,s^*)}, \mathring{\Sigma}_\kappa^{(m,s^*)}; \kappa, k\right] \tag{6.5}$$

- **Common Distribution Estimation:** Compute the common Gaussian-mixture parameters $\mathring{\mu}_k^{(m)}, \mathring{\theta}_k^{(m)}, \mathring{\Sigma}_k^{(m)}$, $m = 1, 2, \ldots M$ via one iteration of the EM algorithm

$$\mathring{\mu}_k^{(m)} = \frac{1}{S} \sum\nolimits_{s=1}^{S} \mu_k^{(m,s)} \tag{6.6}$$

$$\mathring{\theta}_k^{(m)} = \frac{\sum_{s=1}^{S} \mu_k^{(m,s)} \breve{\theta}_k^{(0,s)}}{\sum_{s=1}^{S} \mu_k^{(m,s)}} \tag{6.7}$$

$$\mathring{\Sigma}_k^{(m)} = \frac{\sum_{s=1}^{S} \mu_k^{(m,s)} \left(\breve{\theta}_k^{(0,s)} - \mathring{\theta}_k^{(m)}\right) \left(\breve{\theta}_k^{(0,s)} - \mathring{\theta}_k^{(m)}\right)'}{\sum_{s=1}^{S} \mu_k^{(m,s)}} \tag{6.8}$$

  where the mode probabilities are given by

$$\mu_k^{(m,s)} = \frac{\mu_{k-1}^{(m,s)} L_k^{(m,s)}}{\sum_{m=1}^{M} \mu_{k-1}^{(m,s)} L_k^{(m,s)}} \tag{6.9}$$

- **Fault Detection:** For each unit $s = 1, \ldots, S$ through the active mode estimator compute the likelihood

$$L_k^{(s)} = f\left(z_k^s \middle| \left\{ \mathring{x}_k^{(m,s)}, \mathring{\Sigma}_k^{(m,s)}, \mu_k^{(m)} \right\}_{m=1,\ldots,M} \right) \tag{6.10}$$

to generate a sliding window test statistic which can be tested against a threshold.

6.3 Dynamic Mode Systems

In reality, the mode of a unit can change either gradually or suddenly. For the purpose of this discussion, we define a gradual change as one in which the parameters can successfully be tracked by the dynamic system model being used to represent the unit. A sudden change is one in which the period of adaptation for the parameters is not insignificant.

For gradual mode changes, parameter estimation can progress as described above for a static mode system with the understanding that mode changes are simply represented as a directed parameter drift. However, the mode probabilities used for EM estimation should explicitly take into account the mode transition probabilities (using the algorithm described below).

If the mode switches suddenly, there will be a period of elevated error as the unit's parameters adapt to those of the new mode. In many cases, multiple-model estimators might be suitable solutions. However if the only difference between modes is parameters and, more importantly, we do not know before-hand how to separate the modes, then the parameter estimates for each mode, given the same $z_k^{(s)}, u_k^{(s)}$ will converge.

For sudden mode change, some technique should be used to properly initialize the parameter for the new mode, otherwise the period of sudden parameter mismatch might be interpreted as a fault. Using standard hybrid estimation techniques may be problematic since almost all of them assume that the new mode is somehow represented within the model set, or by a convex combination of the models as in [41]. To avoid this problem a different unit parameter reinitialization is implemented in the proposed algorithm. We try to retain the fast reinitialization of states and probabilities present in an interacting multiple model estimator, but still use a single estimator for $\hat{x}_k^{(s)}, P_k^{(s)}$.

- **Initialization**: In the absence of any relevant data, random initial parameter and state starting points and sufficiently diffuse covariances are chosen. Each unit is assumed to be in a mode, sampled according to the initial probability masses, until later reassigned by the mode change hypothesis test.

- **Estimation:** For each unit $s = 1, \ldots, S$ perform

  – *Mode Matched Estimation:* For each mode $m = 1, \ldots, M$ perform one-step filter update using the global (estimated mixture) parameters, and evaluate the likelihood $L_k^{(m,s)}$

  $$\left[L_k^{(m,s)}, \check{x}_k^{(m,s)}, \check{\Sigma}_k^{(m,s)}\right] = \texttt{Filt}\left[z_k^{(s)}; \mathring{x}_{k-1}^{(m,s)}, \mathring{\Sigma}_{k-1}^{(m,s)}\right] \tag{6.11}$$

  – *Active Mode Matched Estimation:* Perform one-step filter update of the active mode estimator

  $$\left[\check{x}_k^{(0,s)}, \check{\Sigma}_k^{(0,s)}\right] = \texttt{Filt}\left[z_k^{(s)}; \check{x}_{k-1}^{(0,s)}, \check{\Sigma}_{k-1}^{(0,s)}\right] \tag{6.12}$$

- **Active Mode Change Detection:** For each unit $s = 1, \ldots, S$ perform a statistical test using the model likelihoods $\{L_k^{(m,s)} : m = 1, \ldots, M\}$ to establish whether a mode change has occurred

  $$\texttt{Detect}\left[\{L_k^{(m,s)} : m = 1, \ldots, M\}\right] \tag{6.13}$$

  Upon detection of an active mode change from $m_\kappa^{(s)}$ to m_κ^* that occurred at time $\kappa \leq k$, reinitialize:

  $$\left[\check{x}_k^{(0,s)}, \check{\Sigma}_k^{(0,s)}\right] = \texttt{Reinit}\left[\mathring{x}_\kappa^{(m,s^*)}, \mathring{\Sigma}_\kappa^{(m,s^*)}; \kappa, k\right] \tag{6.14}$$

- **Common Distribution Estimation:** Compute the common Gaussian-mixture parameters $\mathring{\mu}_k^{(m)}, \mathring{\theta}_k^{(m)}, \mathring{\Sigma}_k^{(m)}$, $m = 1, 2, \ldots M$ via one iteration of the EM algorithm

$$\mathring{\mu}_k^{(m)} = \frac{1}{S} \sum\nolimits_{s=1}^{S} \mu_k^{(m,s)} \tag{6.15}$$

$$\mathring{\theta}_k^{(m)} = \frac{\sum_{s=1}^{S} \mu_k^{(m,s)} \breve{\theta}_k^{(0,s)}}{\sum_{s=1}^{S} \mu_k^{(m,s)}} \tag{6.16}$$

$$\mathring{\Sigma}_k^{(m)} = \frac{\sum_{s=1}^{S} \mu_k^{(m,s)} \left(\breve{\theta}_k^{(0,s)} - \mathring{\theta}_k^{(m)}\right) \left(\breve{\theta}_k^{(0,s)} - \mathring{\theta}_k^{(m)}\right)'}{\sum_{s=1}^{S} \mu_k^{(m,s)}} \tag{6.17}$$

where the mode probabilities are given by

$$\mu_k^{(m,s)} = \frac{\bar{\mu}_k^{(m,s)} L_k^{(m,s)}}{\sum_{m=1}^{M} \bar{\mu}_k^{(m,s)} L_k^{(m,s)}} \tag{6.18}$$

$$\bar{\mu}_k^{(m,s)} = \sum\nolimits_{i=1}^{M} \mu_{k-1}^{(s,i)} \pi_{im} \tag{6.19}$$

- **Fault Detection:** For each unit $s = 1, \ldots, S$ through the active mode estimator compute the likelihood

$$L_k^{(s)} = f\left(z_k^s \middle| \left\{ \mathring{x}_k^{(m,s)}, \mathring{\Sigma}_k^{(m,s)}, \mu_k^{(m)} \right\}_{m=1,\ldots,M}\right) \tag{6.20}$$

to generate a sliding window test statistic which can be tested against a threshold.

6.4 Remarks

Looking at the above algorithms, it is clear that many details are missing. We discuss some of these below, while noting that there are obviously other methods for estimation, change detection, reinitialization, etc.

Estimation (`Filt`) For linear systems, we use a Kalman Filter to compute the maximum likelihood estimates (described in Sec. A.2) however most parameter estimation problems which cannot be stated in (N)ARMA(X) form tend to be non-linear, so an efficient and effective treatment of non-linear systems is critical.

For non-linear systems, we will use an unscented transform filter to compute the maximum likelihood estimates due to the UTF's simple and effective treatment of non-linear systems, as well as the relatively low computational burden. The UTF also provides estimates of the measurement covariance which is used for likelihood calculation. The UTF algorithm is described in Sec. A.3.

For systems where different modes are represented using different structures, and the structures are separable based on the measurements, a hybrid estimator may be used. In that case, we make use of the interacting multiple model (IMM) estimator described in Sec. A.4.

In the case that the likelihood of a measurement is less than some small number, ϵ, it is increased to ϵ to avoid underflow and division-by-zero problems which may result when calculating probabilities. Unless otherwise stated, $\epsilon = 10^{-9}$.

Mode Change Detection (`Detect`) Function `Detect` detects a change in the mode using likelihood and/or probability information about the unit. An SSPRT or CUSUM algorithm may be used [44, 50, 57].

Alternately, one can use a simpler change detector such as a simple probability test: A mode change is declared when the probability of one of the (non-active) modes becomes greater than all others by some threshold value, i.e., iff

$$\mu_k^{(m,s^*)} > T_s + \mu_k^{(m,s)} \quad \forall m \neq m^*$$

A threshold is used to avoid switching "chatter" for poorly mode-differentiated units (those which are nearly equidistant, in terms of likelihood, from more than one mode) in the presence of noise.

Reinitialization (`Reinit`) When a mode change occurs, we need to use our best estimate of parameters and state to reinitialize the active mode matched estimator. For systems where units gradually drift between modes, we can simply identify the new mode and allow the mode matched estimators to identify the parameters normally. For these systems, the reinitialization algorithm can be left blank since the active mode matched estimator is likely better than any of the mode matched estimators.

However, for systems where the mode jumps suddenly, we should initialize the parameters to our best estimate for that mode. One way of doing this is to use the parameter estimates from the mode-matched ML-estimator for the new mode, $\left(\breve{x}_k^{(m^*,s)}, \breve{\Sigma}_k^{(m^*,s)}\right)$, to reinitialize the active filter:

$$m^{(s)} \leftarrow m^*$$

$$\breve{\theta}_{k-1}^{(0,s)} \leftarrow \mathring{\theta}_{k-1}^{(m^*)}$$

$$\breve{\Sigma}_{k-1}^{(0,s)} \leftarrow \mathring{\Sigma}_{k-1}^{(m^*)}$$

$$\left[\breve{x}_k^{(0,s)}, \breve{\Sigma}_k^{(0,s)}\right] = \texttt{Filt}\left[z_k^{(s)}; \breve{x}_{k-1}^{(0,s)}, \breve{\Sigma}_{k-1}^{(0,s)}\right]$$

In some cases, local states exist which are mode-related, but not globally mode-correlated (i.e., they change with respect to mode but are not useful in clustering modes globally). For these cases, the relevant local states (which act more like parameters) are also reinitialized from the appropriate running ML-estimator.

Optionally, if the mode-switching algorithm provides a jump-time, smoothing may be performed to improve the estimate. Given the multi-mode nature of the problem, multiple-model smoothing techniques should be used, as in [10, 13, 14, 31].

Fault Detection The no-fault hypothesis $\mathcal{H}_0$ is represented by the Gaussian-mixture distribution:

$$L\left(z_k^{(s)}\right) = \sum_{m=1}^{M} \mu_{k-1}^{(m,s)} \mathcal{N}\left(z_k^{(s)}; \mathring{x}_k^{(m,s)}, \mathring{\Sigma}_k^{(m,s)}\right)$$

However, since we have no model for $\mathcal{H}_1$, we cannot use detection algorithms based on likelihood ratios, such as the Neyman-Pearson (NP) or Generalized Likelihood Ratio (GLR) tests. While we could say that the $\mathcal{H}_1$ model is given by the likelihood that $\tilde{z}_k^{(0,s)}$ is not in $\mathcal{H}_0$, i.e., $1 - L\left(\tilde{z}_k^{(0,s)}\right)$, this would contain no new information. In such a situation our test statistic is simply the log likelihood of the measurement residual with an exponential window of size W:

$$T_k^{(s)} = \frac{W-1}{W} T_{k-1}^{(s)} + \frac{1}{W} \log\left(L\left(z_k^{(s)}\right)\right)$$

with $T_0^{(s)}$ set to a value large enough to cause the initially poor parameter estimates to not register as failures.

If $T_k^{(s)} < \gamma_f$, where γ_f is the failure threshold, a fault is declared.

Unit Exclusion When attempting to detect faults based on set statistics, it is useful to be able to look at the statistics without the contribution of failed units. This allows for detection of small faults without degrading the estimate through measurements originating from the faulty unit. Unit exclusion aids in incipient and preexisting failure detection.

Units with a total likelihood of less than $T_e(\dots)$ are excluded from the cluster statistics calculation (Eqs. 6.15-6.19). $T_e(\dots)$ may be set to a small number (e.g. 10^{-4}) for simplicity, or for a more sophisticated treatment, assume different threshold values depending on the number of units estimated to be in a particular mode, their likelihoods, and the desired number of units to exclude.

Obviously, the selective exclusion of some data points changes the sample mean and variance of the cluster. In order to regenerate the original statistics, the covariance can be boosted as noted in Chapter 7. If the truncation is symmetric (as it is when truncating based on likelihood), then the mean is not changed.

Missing Measurements While the algorithm is centralized in nature, it nevertheless consolidates information from many different sources at once. Often, dozens or hundreds of different data acquisition and control systems (often called controllers) must report on the data for each time sample. This raises the possibility that some data will be lost or delayed. Despite this, the coordination required is distributed cooperation, rather than parallel cooperation, allowing looser and more forgiving synchronization techniques [7].

In the absence of updates, the state covariance can be boosted for the units failing to report:

$$P_\kappa = P_\kappa + kQ$$

When measurements finally arrive, the out-of-sequence measurements can be dealt with as explained in [2, 66].

In the case of only partial measurements arriving for a unit, the data can still be used by appropriate adjustment of the measurement covariance matrix. For example, if measurement i in the measurement vector is missing, we can set

$$[R_k]_{ii} = \infty$$

Labelling Since there is no mechanism to ensure that identified clusters are labelled identically to the models used to generate the data, cluster labels must be reassigned if the probability of correct identification is be measured.

This is done by relabelling identified clusters using the closest mode (true) center. Obviously, this is only an issue in simulations since the true modes are normally not available when using real data.

6.5 Limitations and Capabilities

The assumptions underlying the presented algorithm are novel. Because of this, the algorithm has a number of interesting limitations and capabilities.

6.5.1 Limitations

Ensemble Requirement The algorithm is dependent on the availability of multiple correlated units. It is completely inapplicable to systems with a single unit, and suffers performance degradation for systems with few units (e.g., less than twenty units for a 2-mode system).

Mode Representation Requirement In order to detect jumps from one mode to another, the new mode must have some representation, i.e., one or more units must already be in that mode, otherwise the jump could be considered a fault. Obviously, this is not an issue with single-mode systems however one of the more significant strengths of the algorithm is the ability to effortlessly adapt as units jump from mode to mode.

Slower Detection Than Uni-Mode, No-Drift Algorithms Since the distribution of the global parameters may have a larger covariance than that of a local parameter estimate, algorithms that address only a single mode and assume that parameter drift is either non-existent, or can be ignored over the time in which a fault occurs, will generally perform better (if the stated assumptions are true). Effectively, this limits applicability of single-mode slow-drift parameter algorithms to detection of sudden failures only, which may be sufficient for many problem domains. However, it should be noted that the proposed solution can be used to augment single-mode slow-drift algorithms, thereby allowing fast detection using such algorithms, while providing for detection of incipient or preexisting failures.

Central Processing Requirement As presented, the solution assumes a central location where information must be collected for analysis, which may be problematic for sensor networks. However, the information transfer requirement is minimal, consisting not of actual sample data but rather parameter statistics estimates (such as mean and covariance). Additionally, complete system synchronization is not an absolute requirement, but information from most nodes should be collected at a rate faster than parameter drift is expected to occur.

6.5.2 Capabilities

Multi-Mode Capable The proposed solution excels at addressing multi-mode systems, even when models are initially unavailable for all the modes.

Incipient and Preexisting Fault Parameter estimation explicitly accounts for parameter drift, and since fault detection does not depend on sudden deviation of local parameters, incipient and preexisting faults can be detected. This differs from techniques that account for gradual parameter drift, and assume that a fault is detectable by a sudden parameter change.

Can be used with other FD algorithms The proposed solution can be used to supplement other fault detection methodologies (such as the single-mode slow-drift algorithms mentioned above), thereby allowing for the positive features of both, and providing for detection of incipient or preexisting failures using the new algorithm.

Can Include Additional Mode Information If information is available about which mode is active at a particular unit, this information can be incorporated into the algorithm through the transition probability matrix or the initial mode probabilities.

Fast Parameter Initialization After Jumps The probability distribution for global parameters is being constantly estimated using data from every unit at the site. So parameter estimation after the sudden jump of a unit has the advantage of being able to use the global information to set its initial values, even if that unit has never before been seen in that mode. This reduced the "improbability"-spike which would occur otherwise, greatly reducing false positives.

7

Exclusion of Units in Fault Detection

When attempting to detect faults based on set statistics, it's useful to be able to look at the statistics without the contribution of failed units. This allows for detection of small faults without degrading the estimate through measurements originating from the faulty unit.

7.1 Truncated Normal Distribution

The calculation of parameters for a truncated normal distribution (TND) is a recurring problem in real-world testing. A Gaussian distribution assumes limits of $\pm\infty$. In practice, there are usually upper and lower limits imposed on an experiment, so that the actual distribution being measured is a truncated normal distribution (see, for example, [20], [29]).

Consider the multivariate normal distribution given by:

$$\mathcal{N}\left(z;\bar{z},P_z\right) \triangleq \frac{1}{\left|2\pi P_z\right|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\left(z-\bar{z}\right)^T P_z^{-1}\left(z-\bar{z}\right)\right]$$

or, if we set $\tilde{z} = z - \bar{z}$ and assume $P_z = \mathbf{I}$,

$$\mathcal{N}\left(\tilde{z}\right) \triangleq \frac{1}{\left(2\pi\right)^{\frac{N}{2}}} \exp\left[-\frac{1}{2}\tilde{z}^T\tilde{z}\right]$$

If we condition the observation on $z^T z \leq t$, the total probability contained in this region can be denoted by

$$\Phi(t) = \underset{\tilde{z}^T\tilde{z}\leq t}{\int\cdots\int} \mathcal{N}(\tilde{z})\,d\tilde{z}$$

Hence, after scaling the probability distribution becomes:

$$\mathcal{T}(\tilde{z};t) = \begin{cases} \frac{\mathcal{N}(\tilde{z})}{\Phi(t)} & z^T z \leq t \\ 0 & \text{elsewhere} \end{cases}$$

7.1.1 Using Likelihood as a Gating Threshold

The focus of most TND work considers arbitrary thresholds described in terms of left- and right-cutoff values. For multi-modal TND's, a matrix of cutoff values describes an N-dimension box outside of which observations are rejected.

For our purposes, we wish to define the threshold in terms of a probability boundary, beyond which observations are rejected. This has some very handy properties:

- It is mean- and covariance-invariant, meaning that the observation does not have to be pre-whitened or translated into a zero-mean, unit covariance form before truncation since the likelihood explicitly considers the distribution statistics.

- Because of the above point, correlation and skew can be safely ignored.

- Using the likelihood circumvents the "curse of dimensionality" for numerical integration. As the number of dimensions increases, a numerical integration needs exponentially more points to obtain an accurate result. Each random sample produces a result that may represent every point with that likelihood on the surface of an N-dimensional hypersphere.

- Since the Gaussian distribution is symmetric, the mean of the original distribution is equal to the mean of the truncated distribution if likelihood is chosen as the truncation threshold measure (meaning that the truncation is symmetric). Additionally, the covariance of the original distribution is equal to a constant matrix times the covariance of the truncated distribution if likelihood is chosen as the truncation threshold measure.

7.1.2 Threshold Selection

Consider the set of units where a particular mode, m, is the most probable mode for that unit, i.e.,

$$\mathcal{S}_-^{(m)} = \{s : \mu_k^{(m,s)} \geq \mu_k^{(n,s)} \;\; \forall m \neq n\}$$

where the number of units in $\mathcal{S}_-^{(m)}$ is,

$$S^{(m)} \triangleq |\mathcal{S}_-^{(m)}|$$

$\mathcal{S}_-^{(m)}$ is sorted in order of ascending likelihood,

$$s = \{\mathcal{S}_-^{(m)}\}_j, \;\; r = \{\mathcal{S}_-^{(m)}\}_i, \;\; j > i \Rightarrow$$

$$L_k^{(m,s)} > L_k^{(m,r)}$$

Assume that we want to exclude e units,

$$e = N_e\left(S^{(m)}\right)\Bigg\rceil, \;\; N_e\left(S^{(m)}\right) \geq 0$$

where N_e is a function chosen to allow for exclusion of more units as the number of units in mode m increases. For example, the following are possible choices:

$$N_e\left(S^{(m)}\right) = \frac{1}{N} S^{(m)}$$

$$N_e\left(S^{(m)}\right) = \sqrt{\frac{S^{(m)}}{R}}$$

$$N_e\left(S^{(m)}\right) = \max\left(0, \frac{S^{(m)}}{R} - N\right)$$

with typical values for $N = 10$, $R = 2$.

To assist in the description of the threshold selection, we know that the e-th and $e+1$-th smallest units are s_e and s_{e+1},

$$s_e = \{\mathcal{S}_-^{(m)}\}_e$$

$$s_{e+1} = \{\mathcal{S}_-^{(m)}\}_{e+1}$$

The threshold can be selected either as $L_k^{(m,s_e)} + \epsilon$ or $L_k^{(m,s_{e+1})} - \epsilon$. Both have philosophical justifications, but selecting one over the other will have an impact on the estimated covariance; the first option results in a smaller covariance, while the second option results in a larger estimate. Alternately, a middle point may be chosen,

$$LT_k^{(m)} \triangleq \frac{1}{2}\left(L_k^{(m,s_e)} + L_k^{(m,s_{e+1})}\right)$$

Finally, we designate the truncated set of units as $\mathcal{S}^{(m)}$,

$$\mathcal{S}^{(m)} \triangleq \{s \in \mathcal{S}_-^{(m)} : L_k^{(m,s)} \geq LT_k^{(m)}\}$$

7.1.3 Table Lookup of Covariance Ratio

The analytical solution for the mean and covariance of arbitrarily dimensioned TND's is very difficult to calculate.

For univariate TND's (where the dimension of the random variable is one), the relationship between the mean and covariance of the truncated sample and the original (untruncated) distribution is well known [20, 23]), as are the statistics of bivariate [9, 18, 48], and

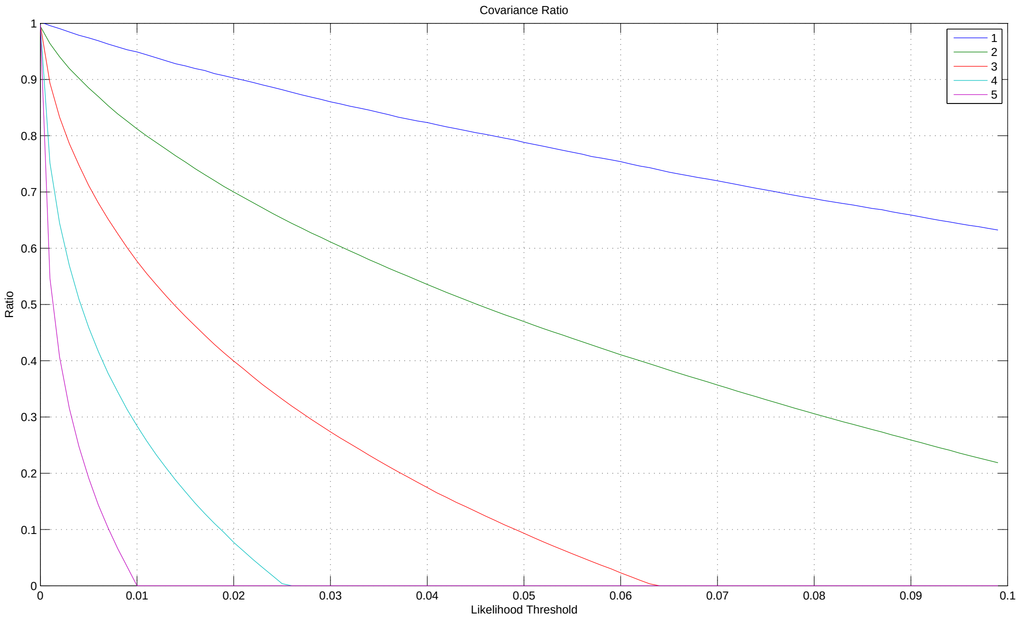


Figure 7.1: Lookup table of the ratio of the covariances of truncated normal and normal distributions for 1-5 dimensional distributions at 100 likelihood thresholds between 0 and 0.1. 100,000 Monte-Carlo iterations were used to generate each curve.

trivariate [63] distributions however this does not help in the general multivariate case.

Where no simple closed form solution exists, Monte-Carlo simulations can be used to generate a lookup table. The ratio of the covariance of the original distribution to the covariance of the resulting truncated distribution can be calculated from this lookup table,

$$GT_k^{(m)} = \texttt{TNDstat}\left[LT_k^{(m)}\right]$$

$\texttt{TNDstat}\,[L]^{-1}$ is plotted in Figs. 7.1 and 7.2 (corresponding to 1-5 unknown parameters in Fig. 7.1, and 6-10 unknowns in Fig. 7.2). The inverse of the truncated multiplier is plotted (i.e., the ratio of the covariance of the TND to that of the original distribution) in order to better show the falloff for different dimensions on the same plot. Note that the lookup table is independent of the actual covariance of the original distribution – the table

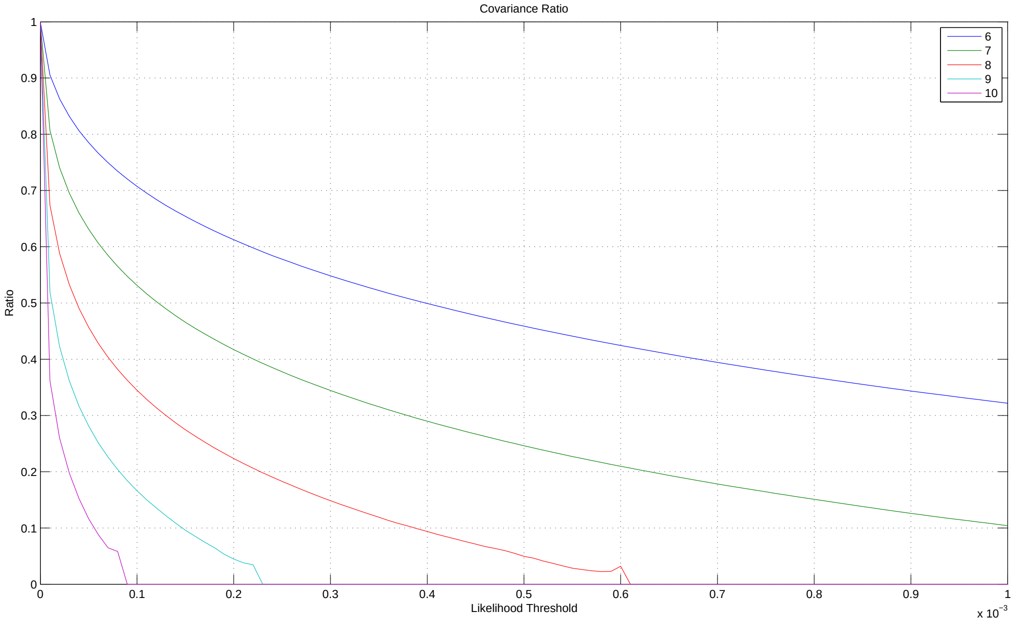


Figure 7.2: Lookup table of the ratio of the covariances of truncated normal and normal distributions for 6-10 dimensional distributions at 100 likelihood thresholds between 0 and 0.001. 100,000 Monte-Carlo iterations were used to generate each curve.

applies to any positive definite symmetrical covariance matrix.

8

Experiments

In order to better understand the performance of the proposed algorithms, we will be using parameter-space plots of the units. The global mode clusters will be shown using concentric rings at 1-σ, 2-σ, and 3-σ boundaries.

For simulations, the correct-detection probability can be shown, which we define as "the sample probability that the identified mode (at time k) for a particular unit is the true mode for that unit (at time k)". Detection delay is defined as the time taken, once a unit changes modes, for the (correct) mode to be detected.

8.1 Simple Reference Models

8.1.1 Simple Static Parameter Simulation

Initially, to test the performance of that algorithm, we will attempt to classify the mode statistics for units in one of two modes. The measurements are distributed according to

$$z_k \sim \mathcal{N}\left(\begin{bmatrix} A \\ B \end{bmatrix}, \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}\right)$$

The global parameters, A and B, constitute $\mathring{\theta}$:

$$
\begin{aligned}
\mathring{\theta}^{(1)} &\sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\mathring{\theta}^{(2)} &\sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)
\end{aligned}
$$

We will ignore mode-switching for now, and assume that the system is static, i.e., the probability of mode transition is zero. The performance can be seen in Fig. 8.1. Unit parameter-distribution is shown at different times in Fig. 8.2.

The experiment is repeated using four modes, with the results shown in Figs. 8.3 and 8.4:

$$
\begin{aligned}
\mathring{\theta}^{(1)} &\sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\mathring{\theta}^{(2)} &\sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\mathring{\theta}^{(3)} &\sim \mathcal{N}\left(\begin{bmatrix} -1 \\ -1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\mathring{\theta}^{(4)} &\sim \mathcal{N}\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)
\end{aligned}
$$

8.1.2 Simple Dynamic Parameter Simulation

We repeat the experiment using the same model as above while allowing for the units to switch modes 5% of the time, i.e.

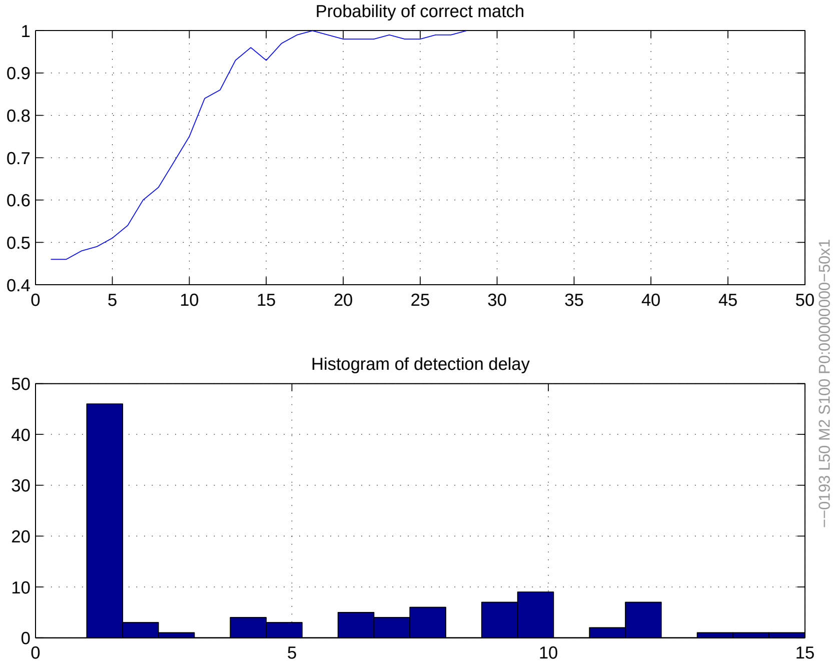


Figure 8.1: Performance of simple static 2-mode parameter estimation (100 units).

$$
\begin{aligned}
z_k &\sim \mathcal{N}\left(\begin{bmatrix} A \\ B \end{bmatrix}, \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}\right) \\
\mathring{\theta}^{(1)} &\sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\mathring{\theta}^{(2)} &\sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right) \\
\Pi &= \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}
\end{aligned}
$$

The performance can be seen in Fig. 8.5. Unit parameter-distribution is shown at different times in Fig. 8.6.

The experiment is repeated using four modes, with the results shown in Figs. 8.7 and

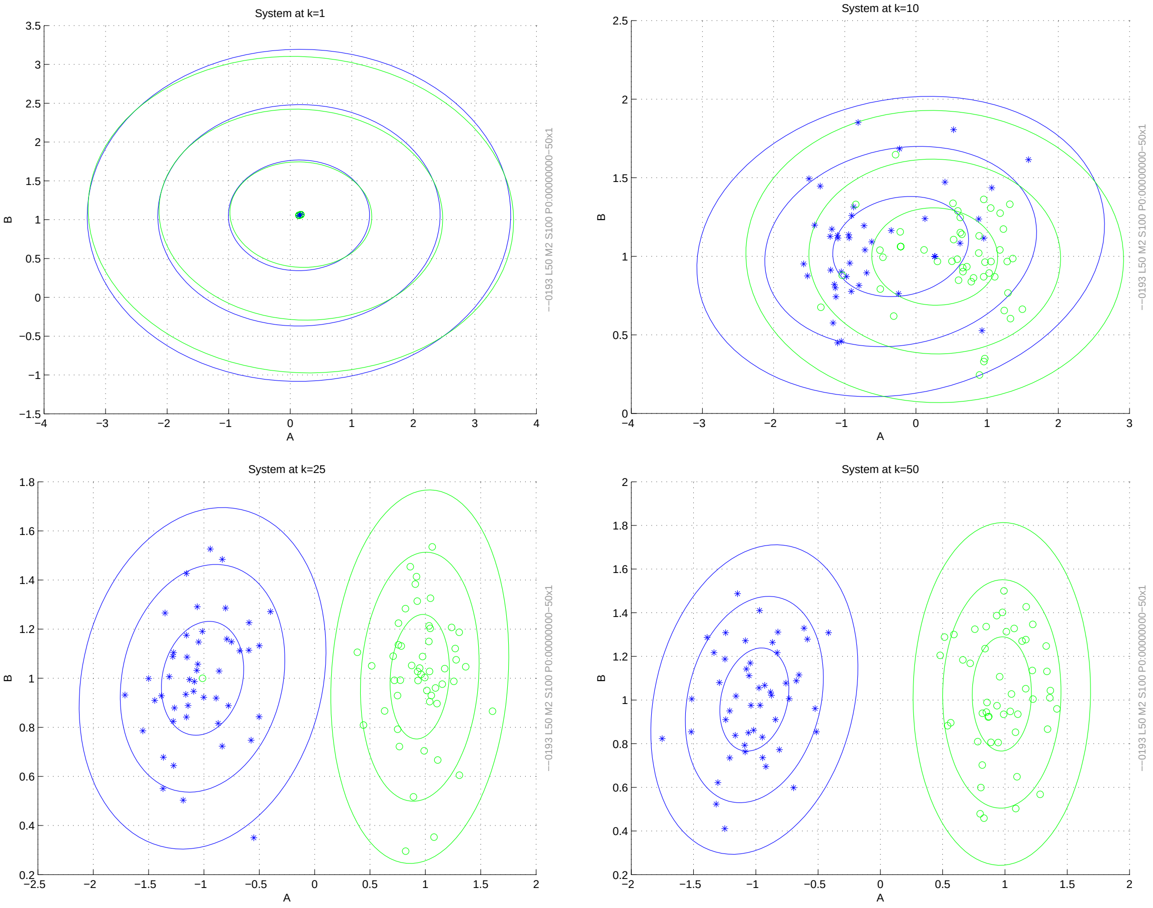


Figure 8.2: Snapshots of unit parameter-distribution at $k = 1, 10, 25, 50$.

8.8:

$$\mathring{\theta}^{(1)} \sim \mathcal{N}\left(\begin{bmatrix}-1\\1\end{bmatrix}, \begin{bmatrix}0.25^2 & 0.0\\0.0 & 0.25^2\end{bmatrix}\right)$$

$$\mathring{\theta}^{(2)} \sim \mathcal{N}\left(\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}0.25^2 & 0.0\\0.0 & 0.25^2\end{bmatrix}\right)$$

$$\mathring{\theta}^{(3)} \sim \mathcal{N}\left(\begin{bmatrix}-1\\-1\end{bmatrix}, \begin{bmatrix}0.25^2 & 0.0\\0.0 & 0.25^2\end{bmatrix}\right)$$

$$\mathring{\theta}^{(4)} \sim \mathcal{N}\left(\begin{bmatrix}1\\-1\end{bmatrix}, \begin{bmatrix}0.25^2 & 0.0\\0.0 & 0.25^2\end{bmatrix}\right)$$

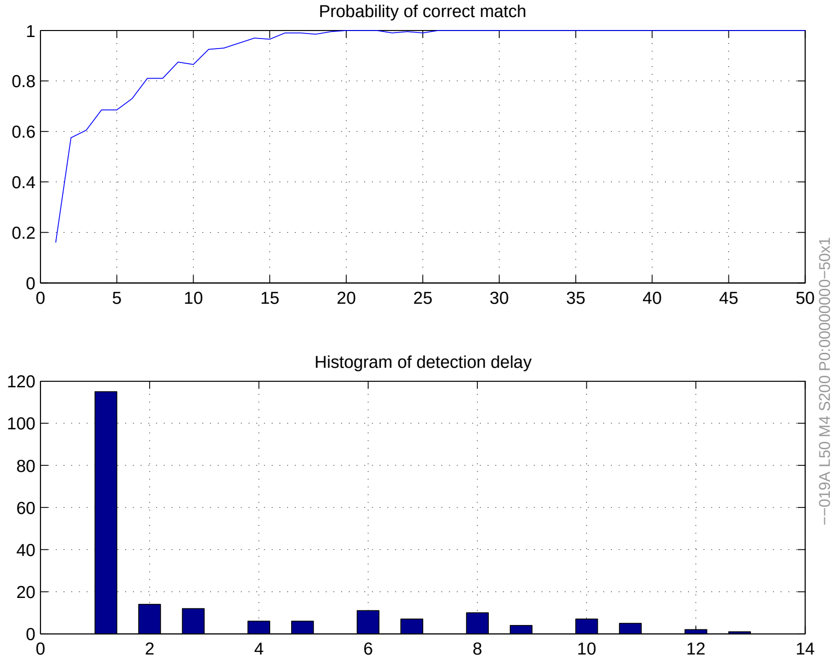


Figure 8.3: Performance of simple static 4-mode parameter estimation (200 units).

8.1.3 Observations

It can be seen that, as long as the separation between modes is clear, static systems seem to converge cleanly and quickly. However, if the rate of mode switching is too high, as in Figs. 8.5, 8.6, 8.7, and 8.8, then the identified mode statistics may not have a chance to converge before the units jump. For a 5%, 200 unit system, it is very likely that several units (10 on average) are changing modes at any point in time.

8.2 AHU Models

Unless otherwise stated,

$$w_k \sim \mathcal{N}(0, Q)$$

$$v_k \sim \mathcal{N}(0, R)$$

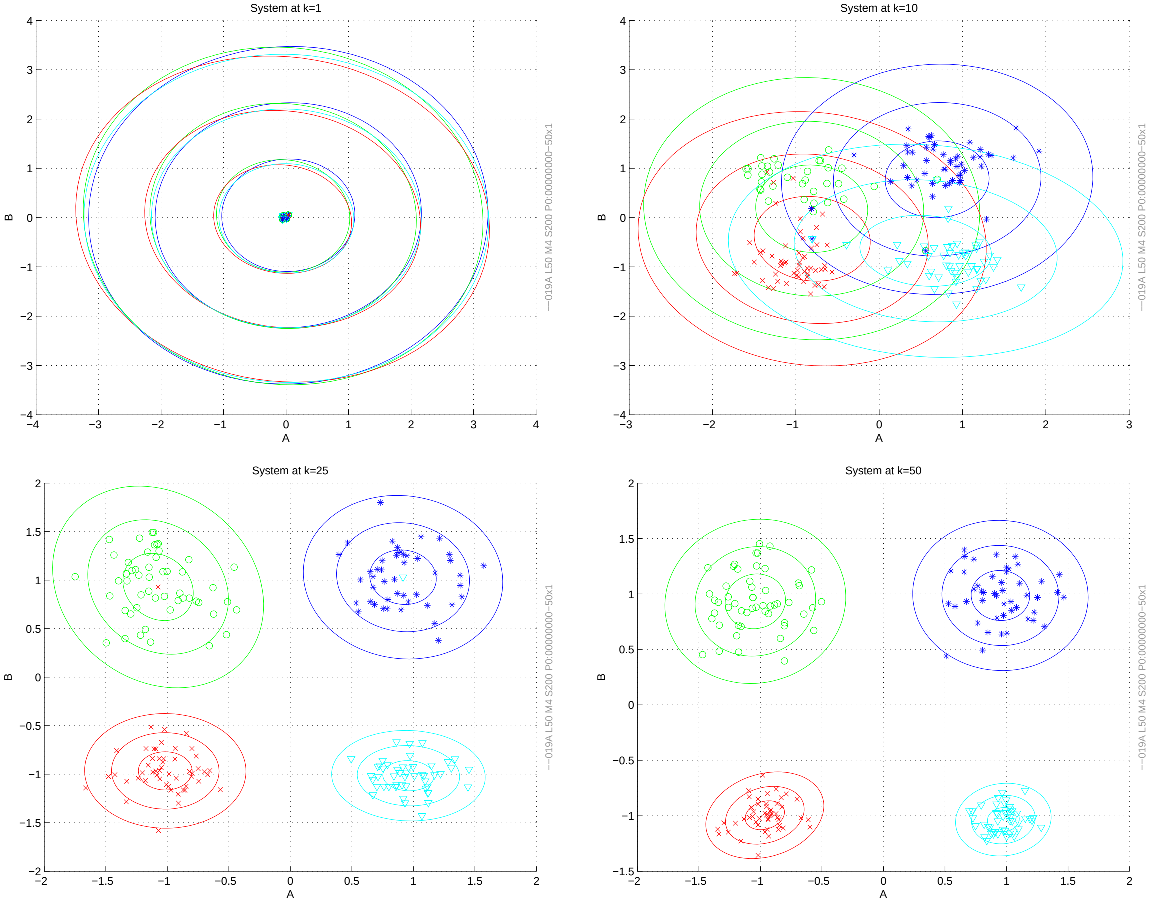


Figure 8.4: Snapshots of unit parameter-distribution at $k = 1, 10, 25, 50$.

And, for simulations,

$$u_k \sim \mathcal{U}\,(0, 100)$$

8.3 Disturbance Model

8.3.1 Simulated Static Disturbance Model

As a first attempt at modelling air handling units, we define a disturbance-based model, where the state, measurement, and input vector are:

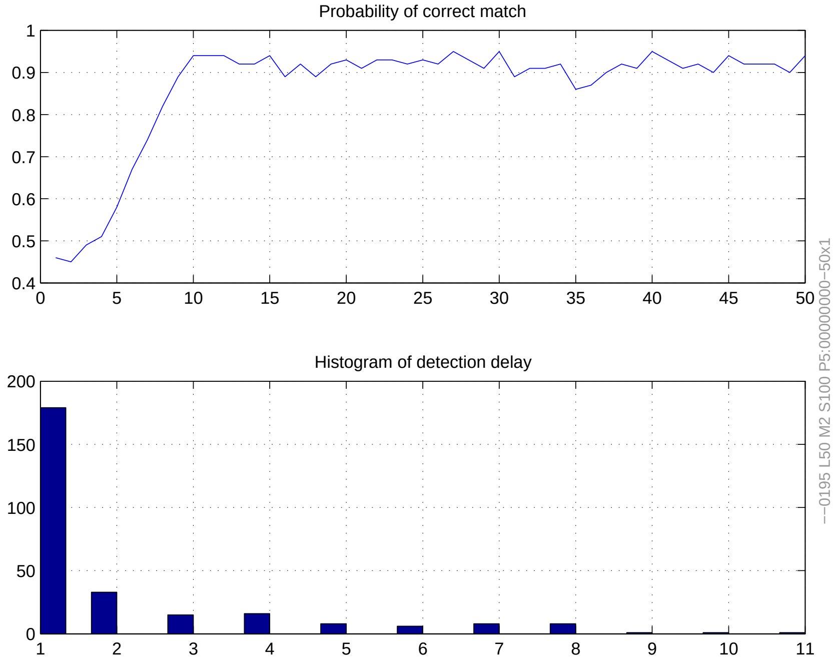


Figure 8.5: Performance of simple static 2-mode parameter estimation (100 units, 5% mode switching).

$$x \triangleq \begin{bmatrix} K \\ D \end{bmatrix} = \begin{bmatrix} \text{Input Gain} \\ \text{Disturbance} \end{bmatrix}$$

$$z \triangleq T = \begin{bmatrix} \text{Supply Temperature} \end{bmatrix}$$

$$u \triangleq \begin{bmatrix} \text{Valve Position} \end{bmatrix}$$

The model is given by:

$$x_k = x_{k-1} + \sqrt{Q} w_k$$

$$z_k = T_{k-1} + K_{k-1} u_k + D_{k-1} + v_k$$

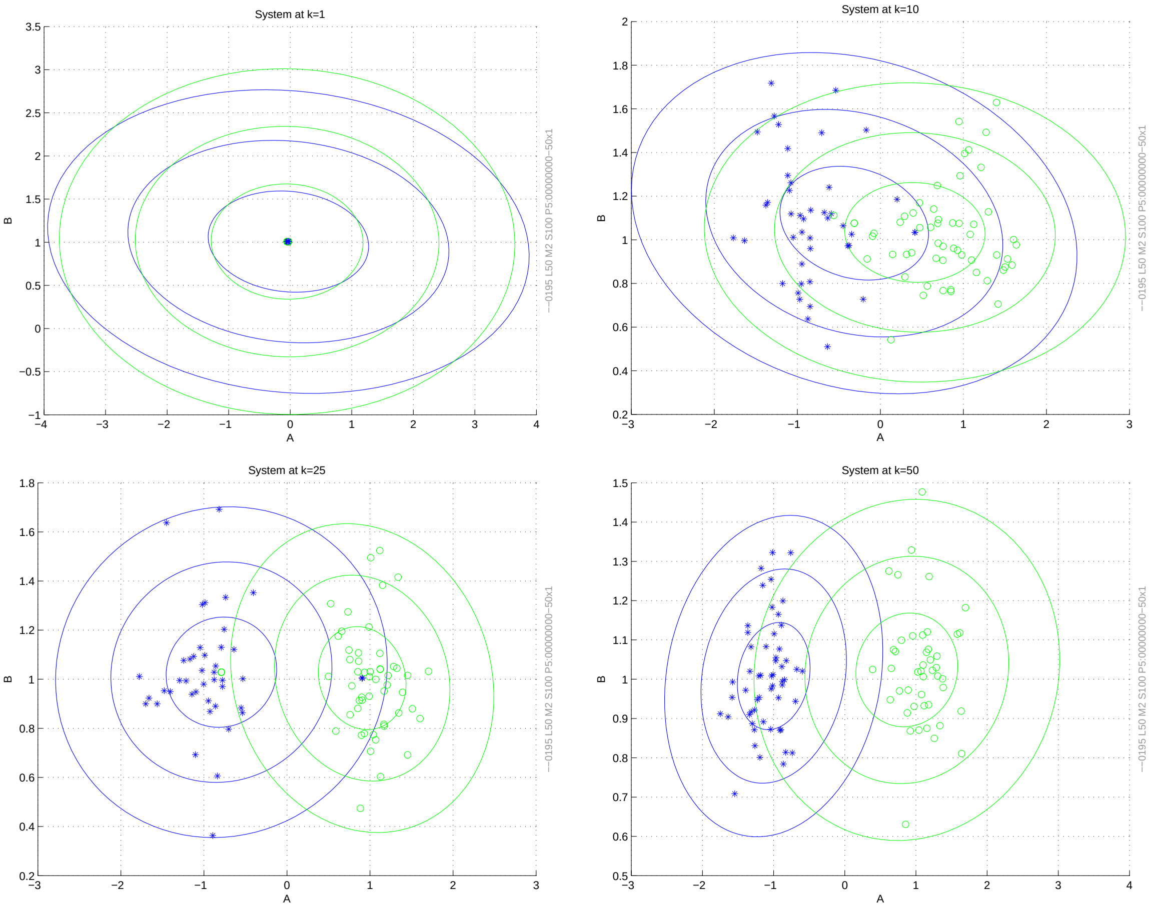


Figure 8.6: Snapshots of unit parameter-distribution at $k = 1, 10, 25, 50$ (5% mode switching).

For the simulation, two modes were defined, as described below:

$$\mathring{\Theta}_0 = \mathcal{N}\left(\begin{bmatrix} 75.0 \\ -0.5 \end{bmatrix}, \begin{bmatrix} 5.0^2 & 0.0 \\ 0.0 & 0.1^2 \end{bmatrix}\right), \quad \mathcal{N}\left(\begin{bmatrix} 40.0 \\ 0.2 \end{bmatrix}, \begin{bmatrix} 5.0^2 & 0.0 \\ 0.0 & 0.1^2 \end{bmatrix}\right)$$

$$\Pi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathring{Q} = \begin{bmatrix} 0.1^2 & 0.0 \\ 0.0 & 0.01^2 \end{bmatrix}, \quad R = 0.1^2$$

Note that the transition probability matrix is the identity matrix, indicating that there is no possibility of a unit changing modes.

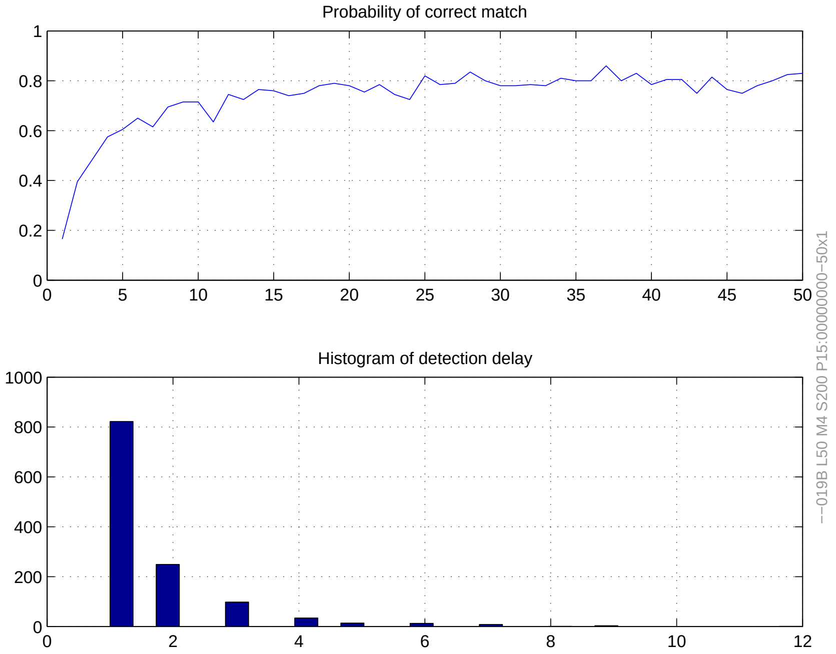


Figure 8.7: Performance of simple dynamic 4-mode parameter estimation (200 units, 5% mode switching).

The performance for a single (typical) run with 120 units is shown in Fig. 8.10. We can see that the mode parameter-clusters are well separated, and that eventually (at $k = 14$), all units are correctly identified.

In the figure, estimated parameters for each unit are colored according to the mode for which they have been identified to belong.

8.3.2 Simulated Dynamic Disturbance Model

Using the same model described above and 120 units, we introduce 1% and 5% transition probabilities, i.e.:

$$\Pi = \begin{bmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{bmatrix}, \quad \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}$$

The performance for a single run is shown in Figs. 8.11 and 8.13. As can be seen from

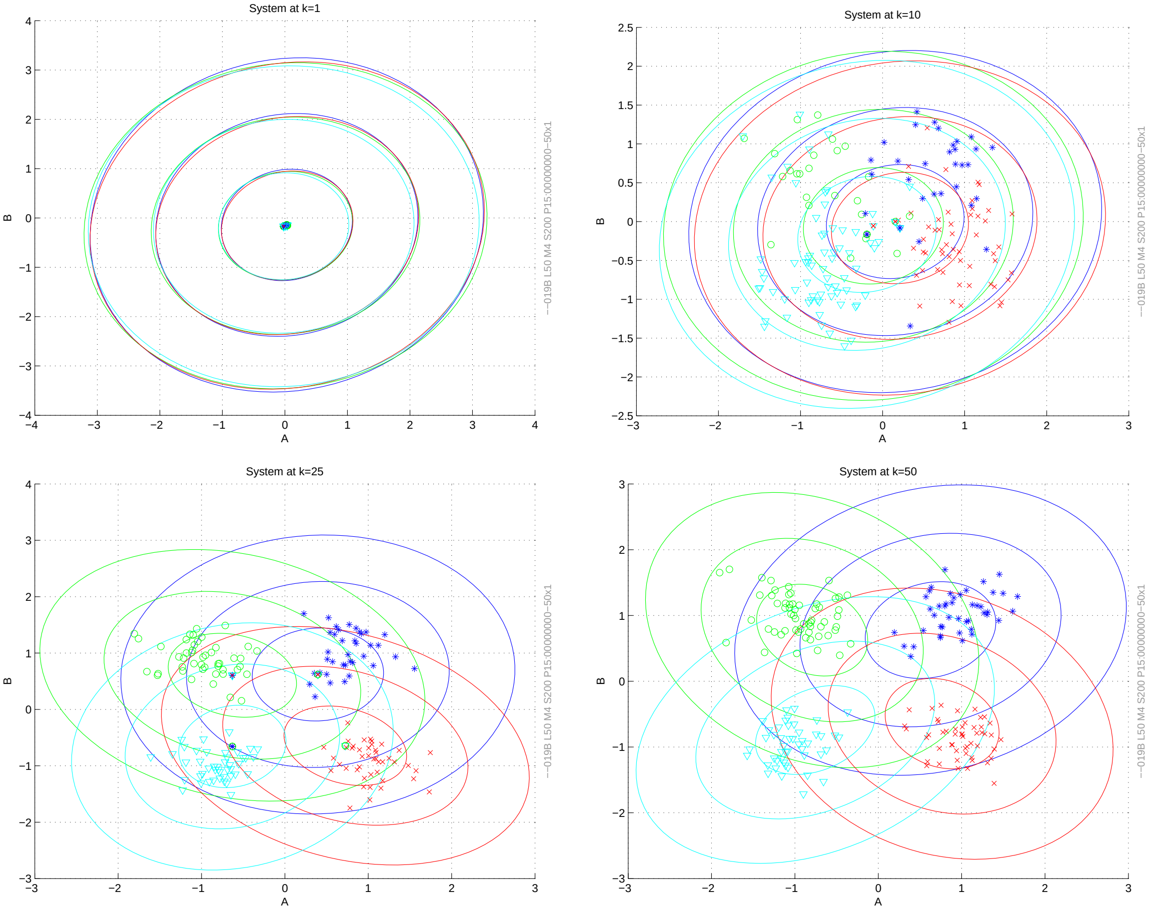


Figure 8.8: Snapshots of unit parameter-distribution at $k = 1, 10, 25, 50$.

the figures, it is likely that one or more units has switched modes at any time sample in particular. The units that have switched modes would eventually be classified after more data arrives.

8.3.3 Real AHU Data Using a Disturbance Model

Data was collected from three different sites with the following characteristics:

SITE1: Large commercial office building in Houston, Texas, with over 70 floors. Supply, Setpoint, and Valve data is available.

SITE2: Large commercial office building in Miami, Florida, with over 70 floors. Supply,

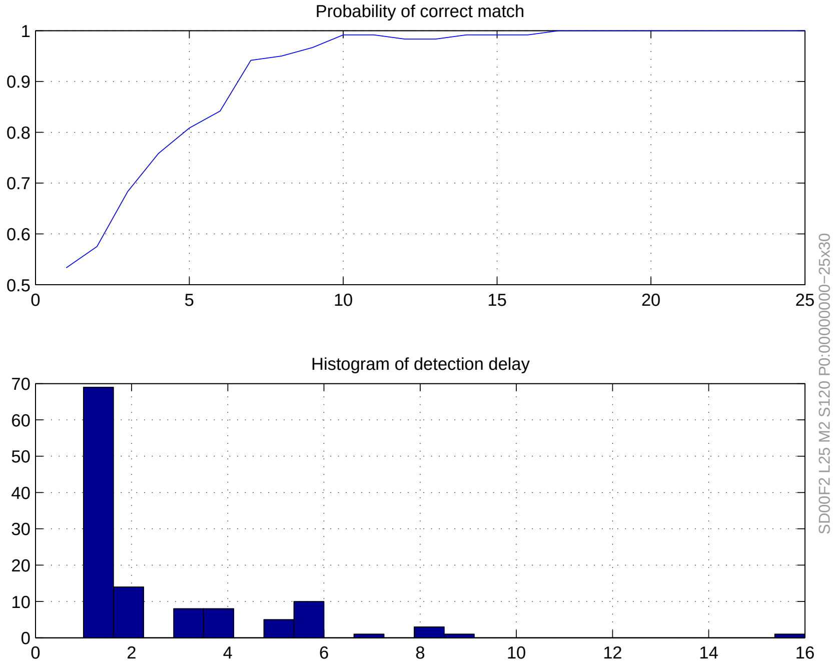


Figure 8.9: Simulated Disturbance Model with no Mode Switching (120 units).

Setpoint, and Valve data is available.

SITE3: Large commercial office building in Houston, Texas, with over 35 floors. Supply, Setpoint, Return, Pressure, VFD Speed, and Valve data is available.

In order to better visualize the amount of data used in producing a single plot, a capture from SITE1 on 2008/07/31 is shown in Fig. 8.15.

Data Collection Issues

Given that the collection sites are commercial properties, their primary concern was not to disrupt operation of the HVAC system in any way. That being the case, most building engineers and management staff are reluctant to introduce experimental code into the automation system. In order to receive permission to test the fault detection algorithm, passive-only data collection was performed, meaning that no external excitation could be introduced into the system.

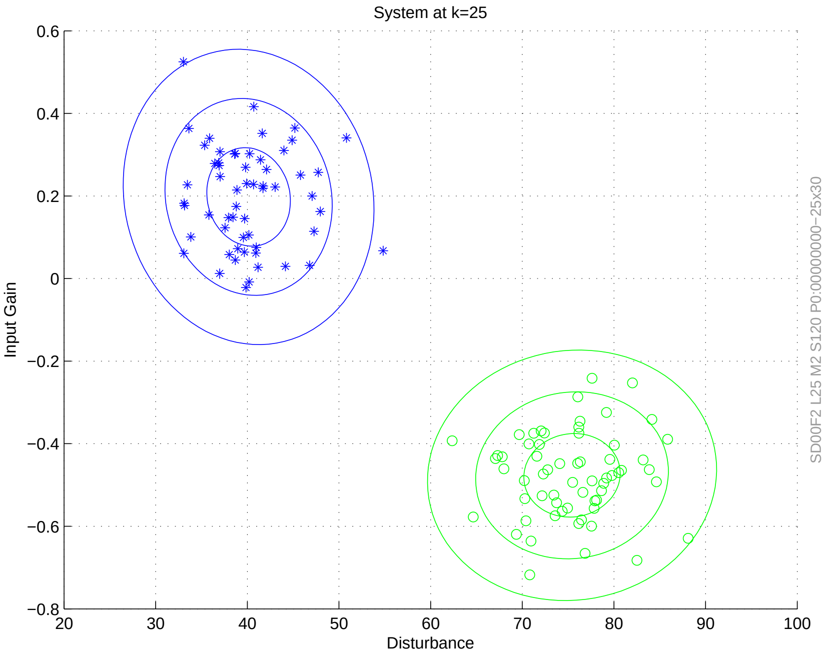


Figure 8.10: Simulated Disturbance Model with no Mode Switching (120 units).

This poses a problem for parameter identification when insufficient excitation is present, as is the case for an HVAC system where the space temperatures have stabilized. For this reason, data collected during early morning start-up proved most useful. After the algorithm has been established, active data collection, where a control signal can be injected into the system should improve the results.

Finally, since no true model is available, fault detection performance cannot be automatically deduced except through anecdotal evidence. For example, some faults may have been due to a technician working on the unit in question at the time, while some faults disappear after routine maintenance.

We present here some preliminary results, without further comment or explanation on the performance in real-world scenarios. Future research should contain more real-world performance metrics such as true detections vs. false alarms.

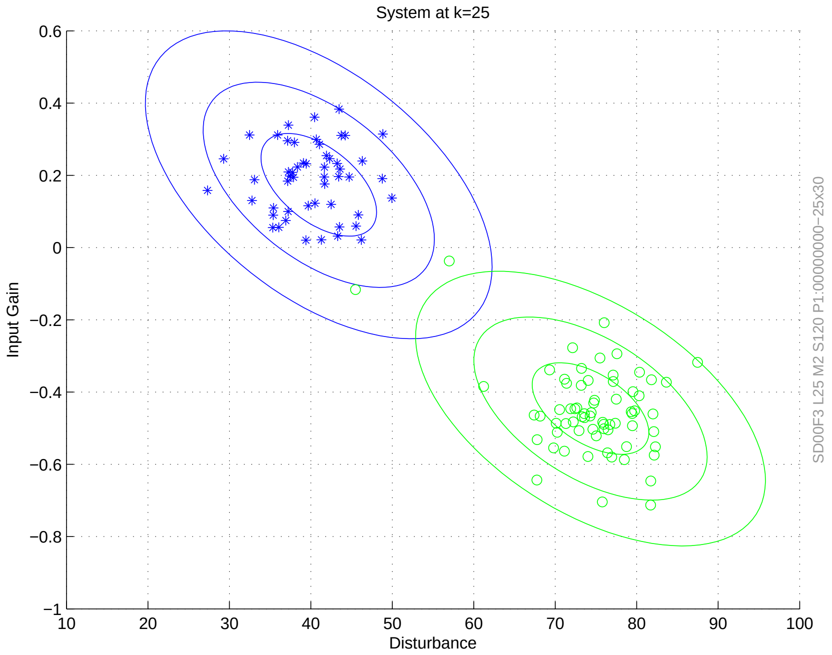


Figure 8.11: Simulated Disturbance Model with 1% Mode Switching (120 units).

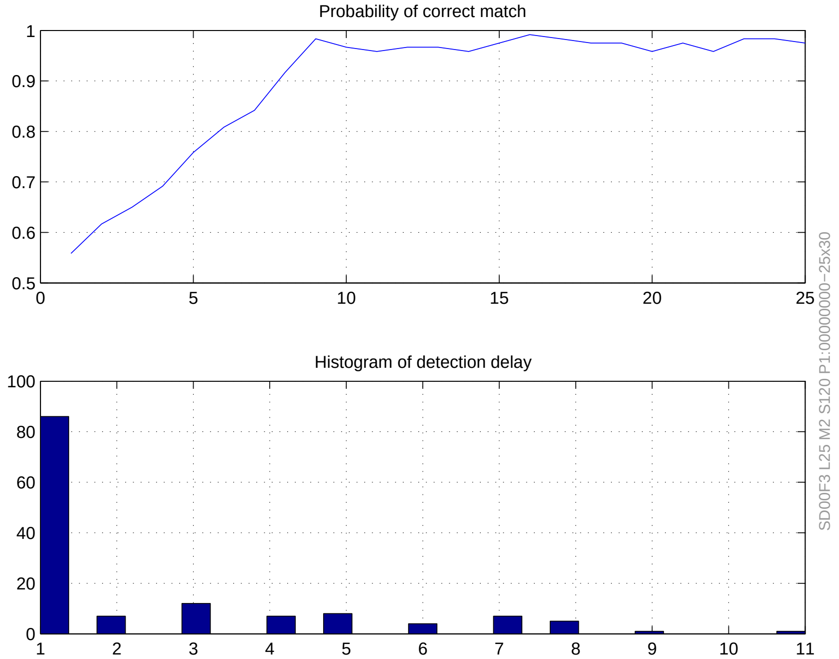
8.4 Real AHU Data Using an Approximate Physical Model

If we have return air information in addition to supply air (as is available at SITE3), we can include this in our model:

$$x \triangleq \begin{bmatrix} K \\ T_r \\ T_s \end{bmatrix} = \begin{bmatrix} \text{Input Gain} \\ \text{Return Temperature} \\ \text{Supply Temperature} \end{bmatrix}$$

$$z \triangleq T = \begin{bmatrix} \text{Return Temperature} \\ \text{Supply Temperature} \end{bmatrix}$$

$$u \triangleq \begin{bmatrix} \text{Valve Position} \end{bmatrix}$$


Probability of correct match

Histogram of detection delay

Figure 8.12: Simulated Disturbance Model with 1% Mode Switching (120 units).

with,

$$\begin{bmatrix} K \\ T_r \\ T_s \end{bmatrix}_k = \begin{bmatrix} K \\ T_r \\ (T_r - T_s)\, K \frac{u}{100} \end{bmatrix}_{k-1} + w_k$$

$$z_k = \begin{bmatrix} T_r \\ T_s \end{bmatrix}_k + v_k$$

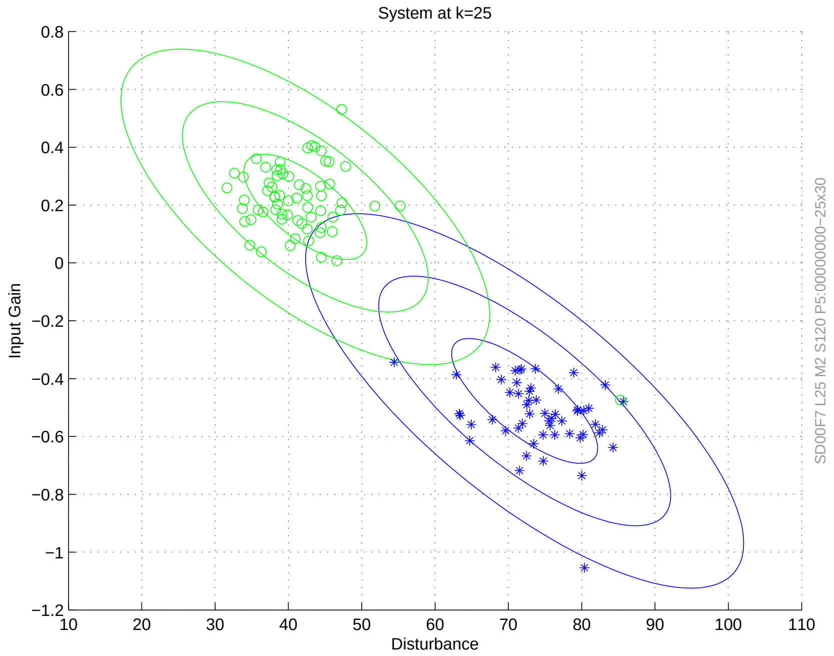


Figure 8.13: Simulated Disturbance Model with 5% Mode Switching (120 units).

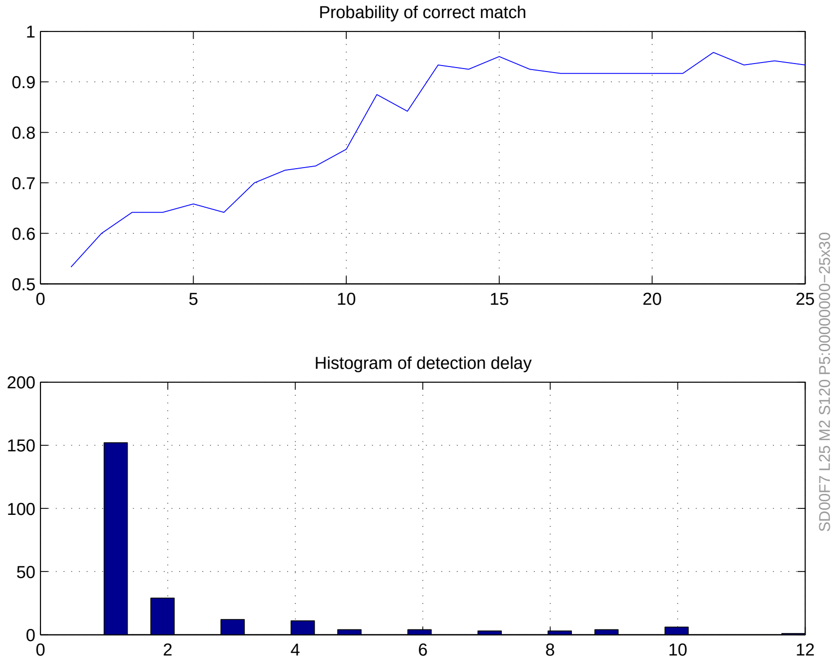


Figure 8.14: Simulated Disturbance Model with 5% Mode Switching (120 units).

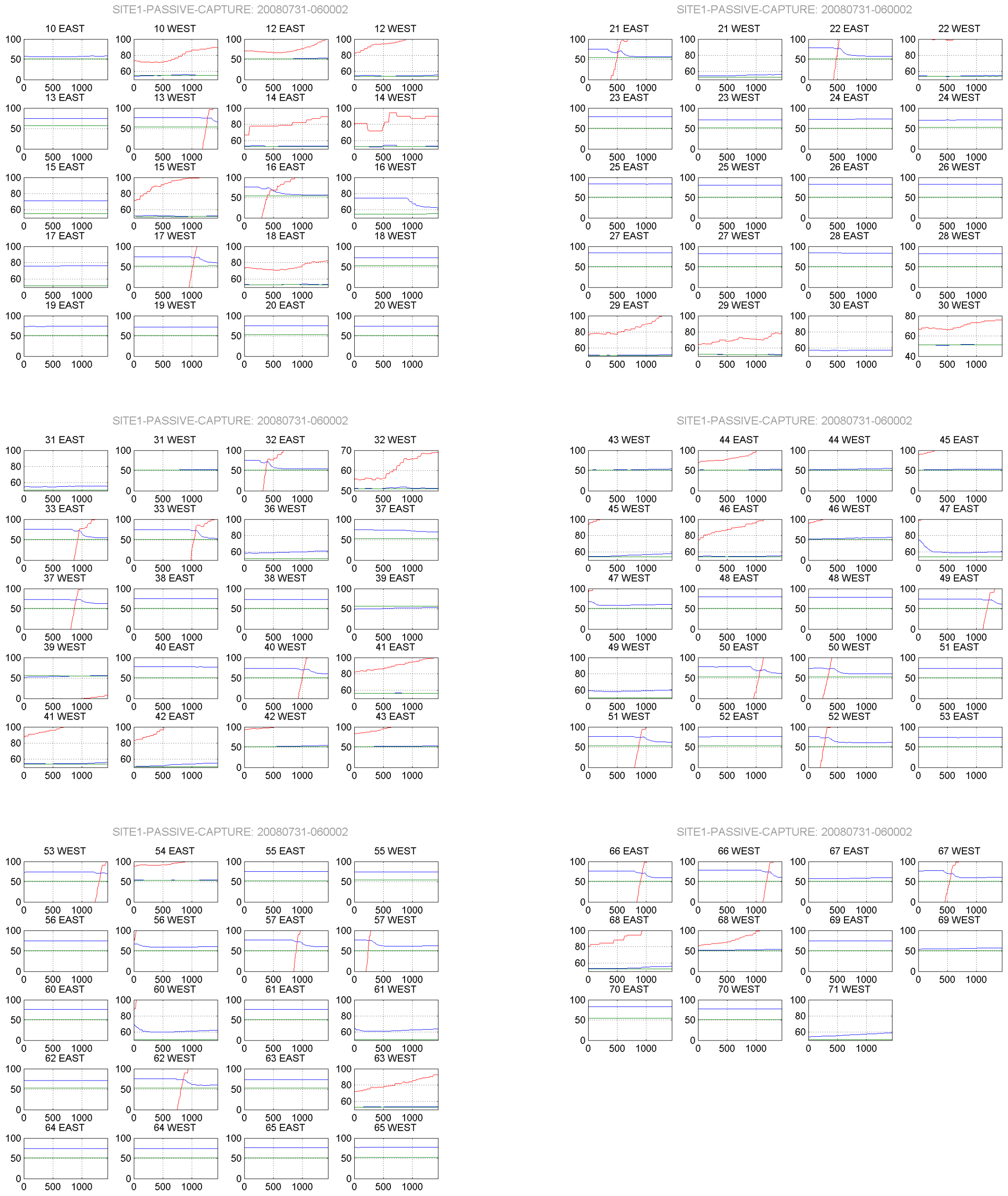


Figure 8.15: Sample data from SITE1.

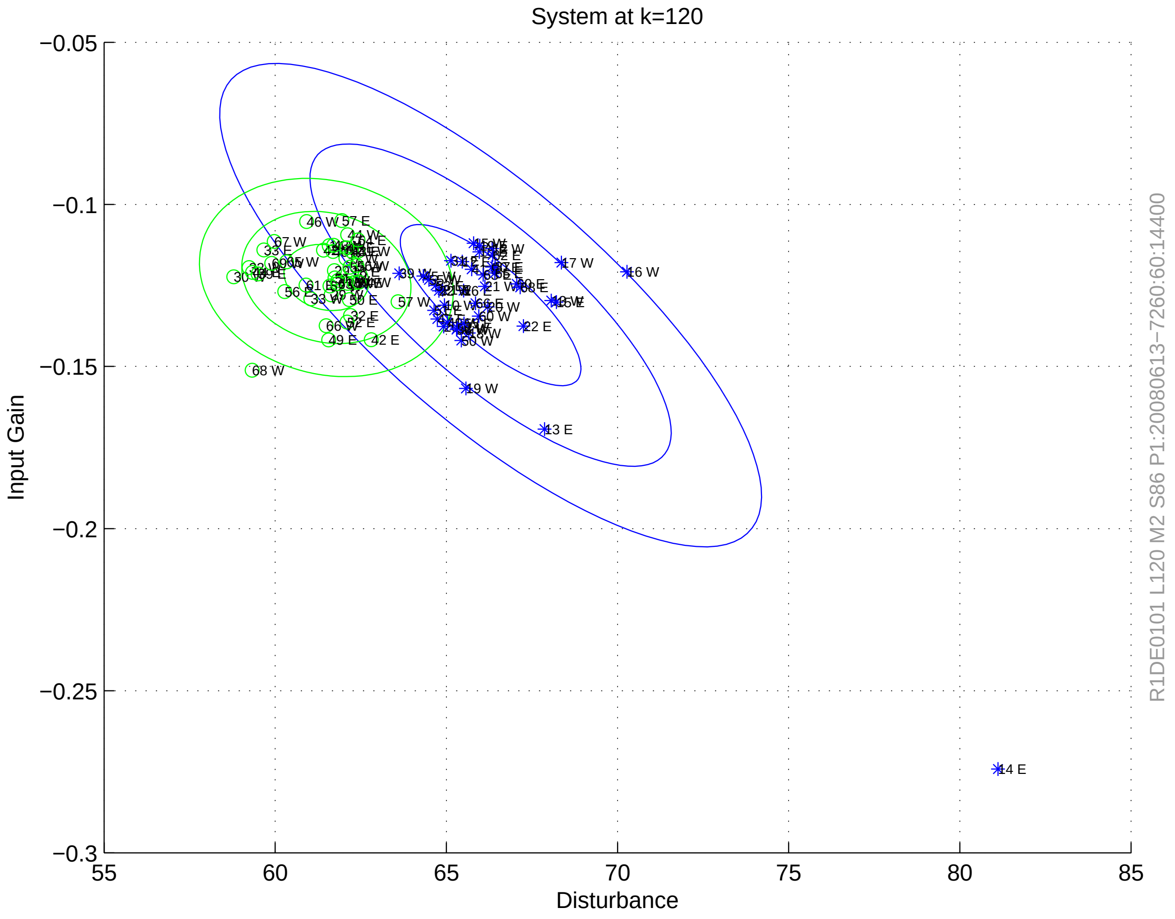


Figure 8.16: Passive identification of SITE2 taken on 2008/06/13 (89 units). We can see that 14-EAST seems to be faulty.

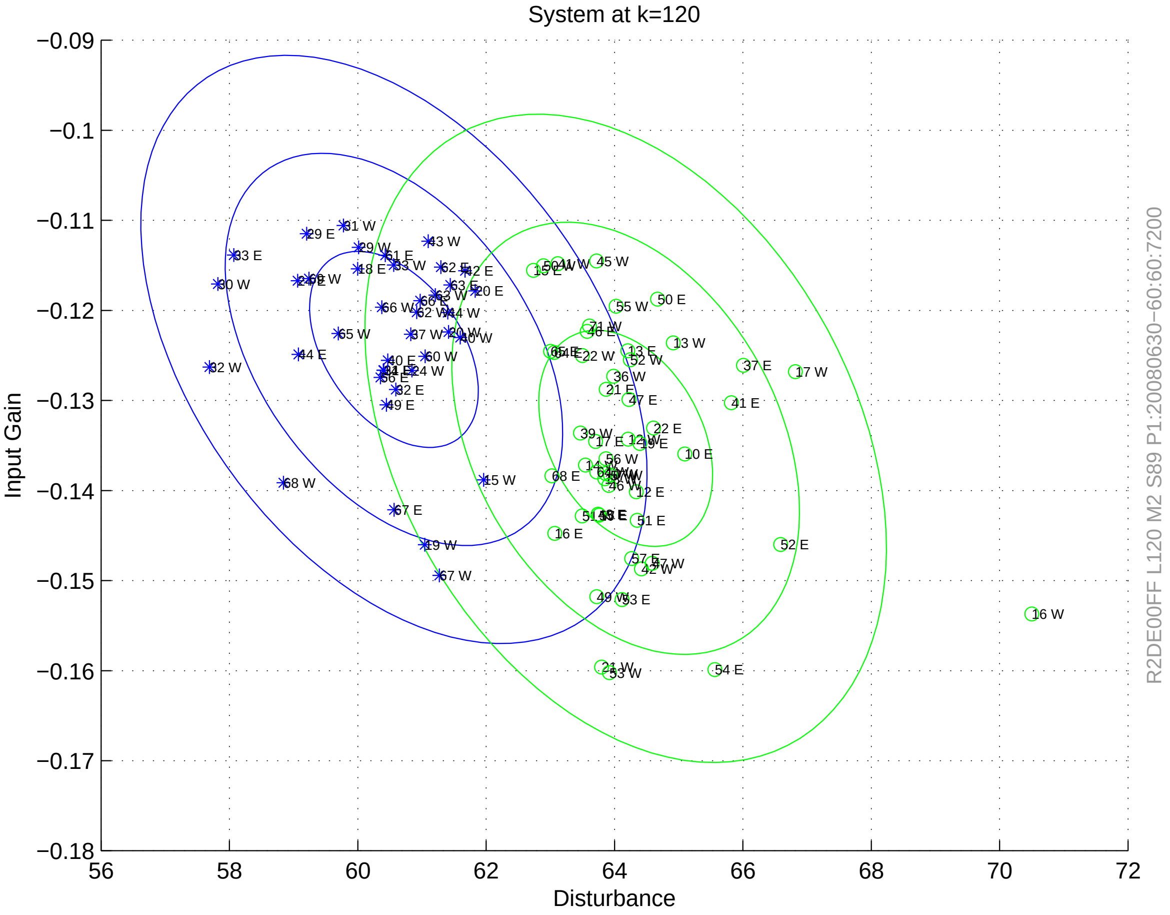


Figure 8.17: Passive identification of SITE2 taken on 2008/06/30 (89 units). We can see that 16-WEST seems to be faulty.

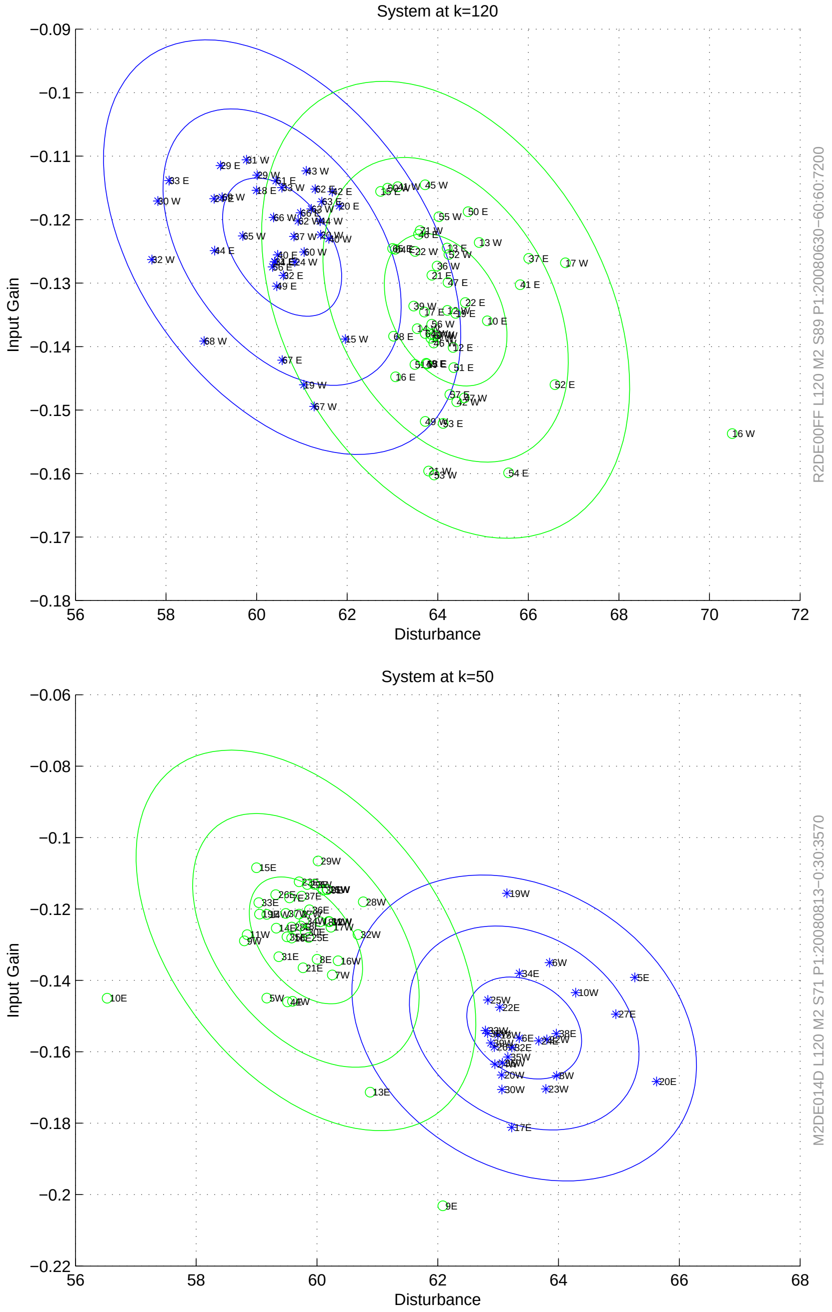


Figure 8.18: Passive identification of two separate systems, SITE1 in Houston (top) and SITE2 in Miami (bottom). Identified parameters and clusters are remarkably similar, however there is no reason that this is necessarily the case for all sites.

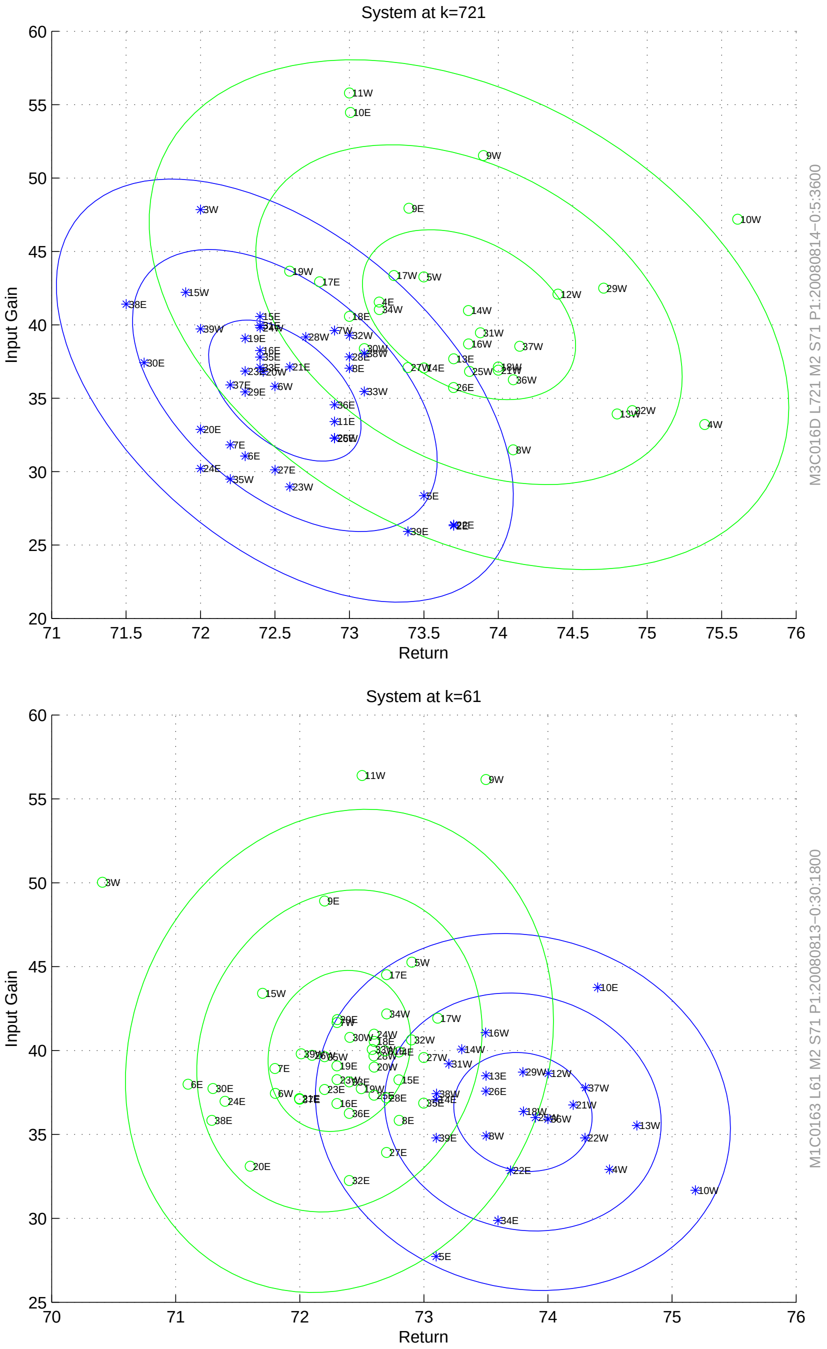


Figure 8.19: SITE3 has return air, speed, and pressure sensors, which allows a more detailed model to be used. Plots show identified parameters on different (successive) days.

9

Performance

A Receiver Operating Characteristic (ROC) curve is a natural way of displaying Type-I (false positive) vs. Type-II (false negative) errors for a fault detection algorithm.

The ROC-curves we present show Type-I errors (false negative) on the x-axis, with one minus Type-II errors (i.e., true positive) on the y-axis. As is customary in detection literature, we refer to the x- and y-axis as Probability of False Alarm (P_{FS}), and Detection Probability (P_D):

$$\begin{aligned} P_{FA} &\triangleq P\left(\text{``}\mathcal{H}_1\text{''}|\mathcal{H}_0\right) \\ P_D &\triangleq P\left(\text{``}\mathcal{H}_1\text{''}|\mathcal{H}_1\right) = 1 - P\left(\text{``}\mathcal{H}_0\text{''}|\mathcal{H}_1\right) \end{aligned}$$

where "$\mathcal{H}_1$" denotes making a decision to choose the Alternate Hypothesis, $\mathcal{H}_1$, based on the detection algorithm.

9.1 Baseline Performance

Before any meaningful comparison can take place, we much choose a baseline which represents the "standard" performance.

In HVAC FDD, the most common approach is non-parametric fault detection based on rules (for example, where a fault is declared if a value exceeds some predefined thresholds).

Clearly, any reasonable sophisticated detection algorithm should be able to perform better than this.

The more sophisticated (theoretically proposed, but not often used in practice) detection algorithms measure a single unit's deviation from some model (fully specified, black-box, neural-network, or otherwise). However such approaches are not well suited to handling drifting parameters, mode-switching, or incipient failures.

Since there are no functionally comparable algorithms available, we will have to manufacture a baseline using the proposed algorithm itself. One possibility is to compare performance of the sub-optimal algorithm to that of the full algorithm. However executing the full algorithm, even for a short run, is prohibitively expensive. Hybrid system, multiple model estimators share the same exponentially increasing mode-switching hypothesis tree, and so they are generally compared with other non-optimal multiple model estimators.

In order to remove the effect of switching from performance comparison, we can construct a scenario with only one mode. This can be seen as the simplest possible scenario, and can also be seen as an upper-bound on algorithm performance.

9.1.1 Uni-Modal Model

The measurements are distributed according to

$$z_k \sim \mathcal{N}\left(\begin{bmatrix} A \\ B \end{bmatrix}, \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}\right)$$

The global parameters, A and B, constitute $\mathring{\theta}$:

$$\mathring{\theta}^{(1)} \sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

9.1.2 Bi-Modal Model

The measurements are distributed according to

$$z_k \sim \mathcal{N}\left(\begin{bmatrix} A \\ B \end{bmatrix}, \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}\right)$$

The global parameters, A and B, constitute $\mathring{\theta}$:

$$\mathring{\theta}^{(1)} \sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$
$$\mathring{\theta}^{(2)} \sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

9.1.3 Quad-Modal Model

The measurements are distributed according to

$$z_k \sim \mathcal{N}\left(\begin{bmatrix} A \\ B \end{bmatrix}, \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}\right)$$

The global parameters, A and B, constitute $\mathring{\theta}$:

$$\mathring{\theta}^{(1)} \sim \mathcal{N}\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

$$\mathring{\theta}^{(2)} \sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

$$\mathring{\theta}^{(3)} \sim \mathcal{N}\left(\begin{bmatrix} -1 \\ -1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

$$\mathring{\theta}^{(4)} \sim \mathcal{N}\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

9.1.4 Failures

Sudden Failure: A fault occurs in mode 1 of unit 1 (designated as $1E$), at time $k_F = 5$,

$$\theta_{k_F}^{(1,1)} = \theta_{k_F-1}^{(1,1)} + \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Incipient Failure: A gradual fault occurs in mode 1 of unit 1 (designated as $1E$), for $k_F \geq 5$, such that

$$\theta_{k_F}^{(1,1)} = \theta_{k_F-1}^{(1,1)} + \begin{bmatrix} 0.2 \\ 0 \end{bmatrix}$$

Pre-Existing Failure: A fault occurs in mode 1 of unit 1 (designated as $1E$), at time

$k_F = 0$ such that

$$\theta^{(1,1)} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.25^2 & 0.0 \\ 0.0 & 0.25^2 \end{bmatrix}\right)$$

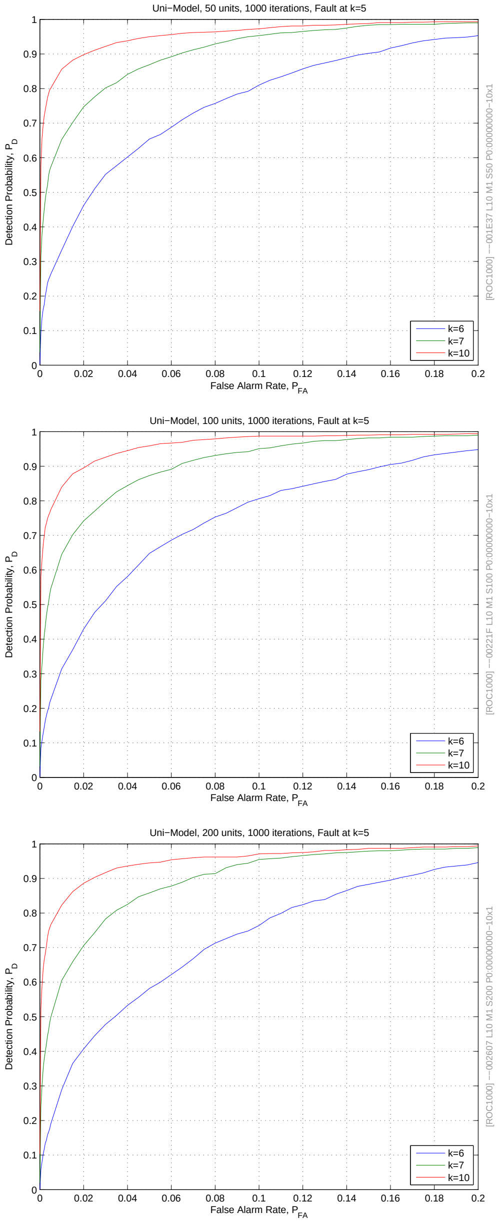


Figure 9.1: ROC for Uni-Mode Models for $k = 6, 7, 10$. It can be seen that there is no significant improvement to having more units - 50 units seems to be able to sufficiently capture the distribution statistics.

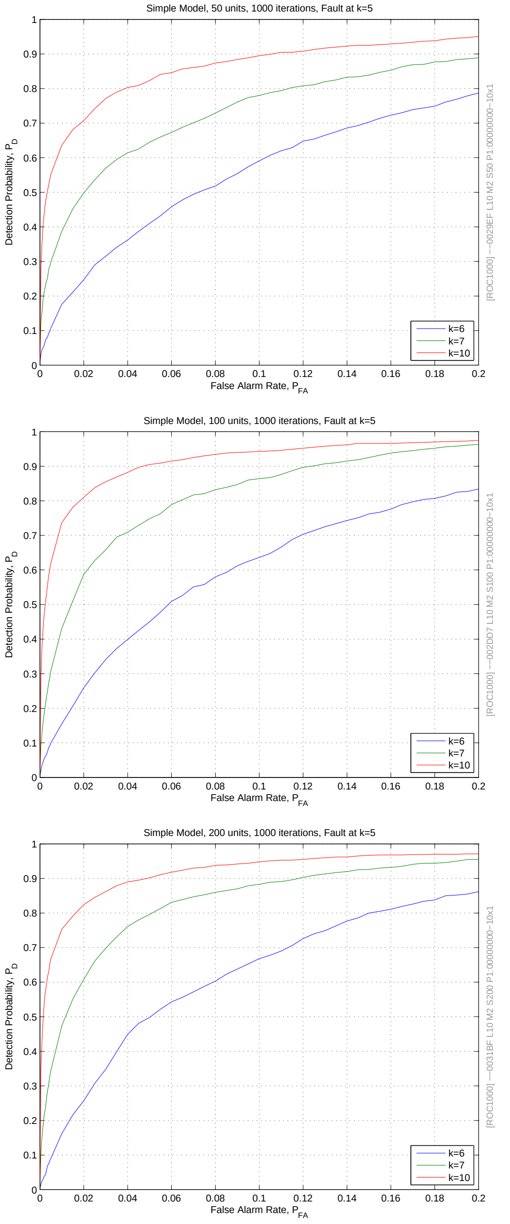


Figure 9.2: ROC for Bi-Mode Models for $k = 6, 7, 10$. It can be seen that there is a slight improvement to having more units; 50 units seems to be too few to capture the distribution statistics adequately.

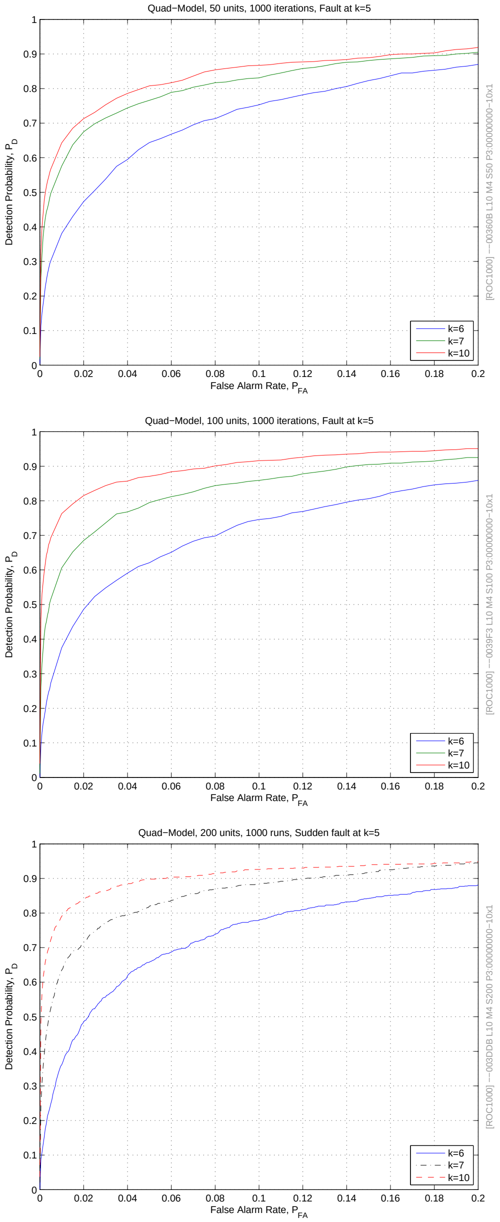


Figure 9.3: ROC for Quad-Mode Models for $k = 6, 7, 10$. It can be seen that there is some improvement when increasing the number of units to 100; 50 units seems to be too few to capture the distribution statistics adequately.

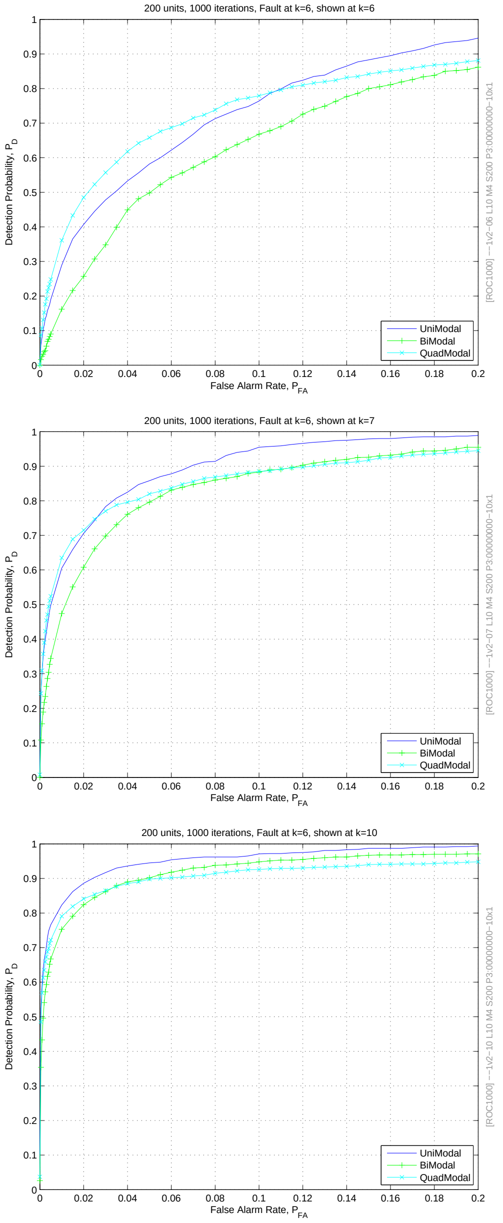


Figure 9.4: ROC comparison for Uni-, Bi- and Quad-Mode Models for $k = 6, 7, 10$. When using a sufficiently large number of units, mode switching does not seem to impact performance significantly in the tested scenario.

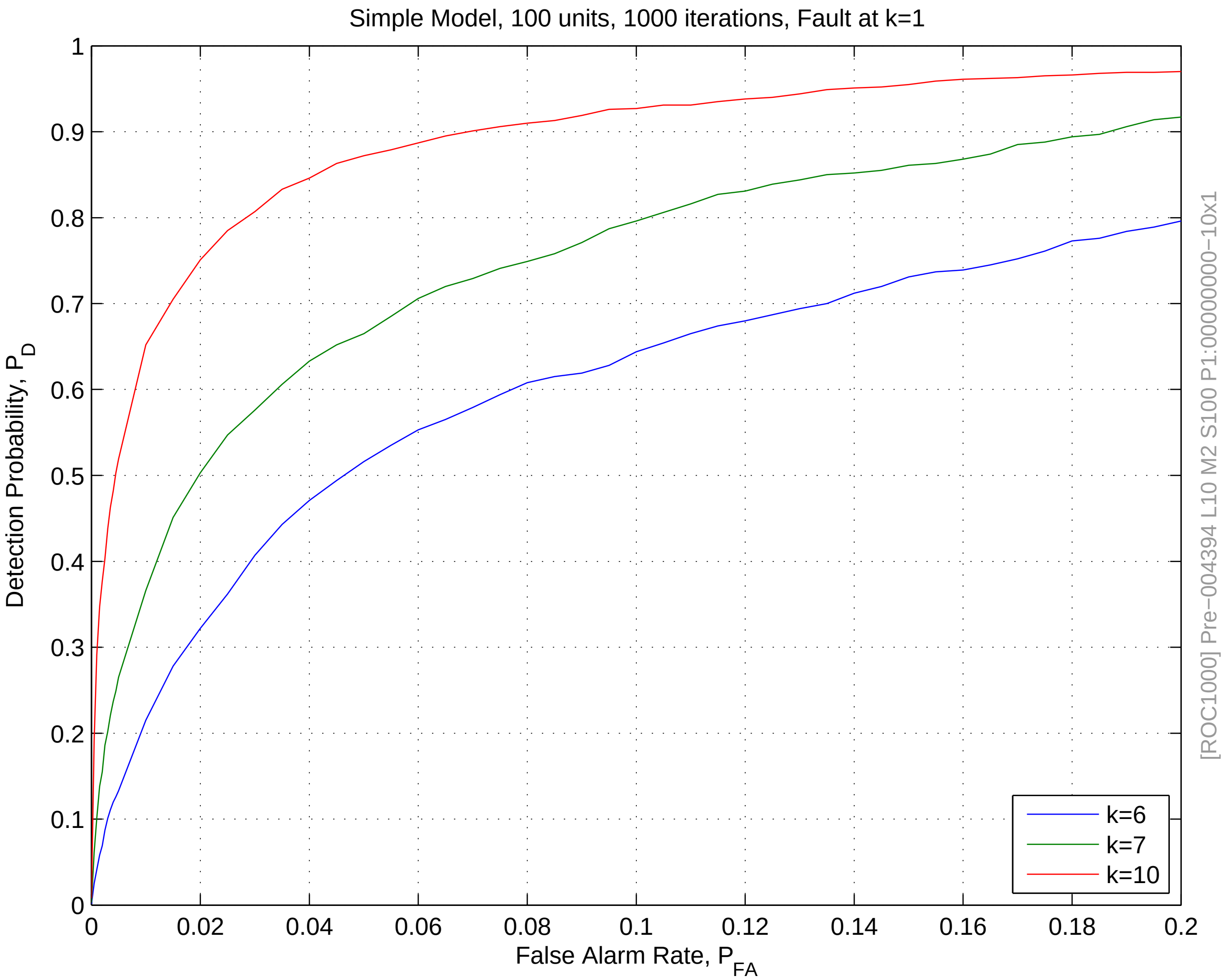


Figure 9.5: Pre-existing fault in Bi-Mode Models. 10^{-4} likelihood exclusion used as discussed in Chapter 7. The plot shows a clear improvement in performance as the global cluster statistics become more accurate (i.e. as $k \to \infty$).

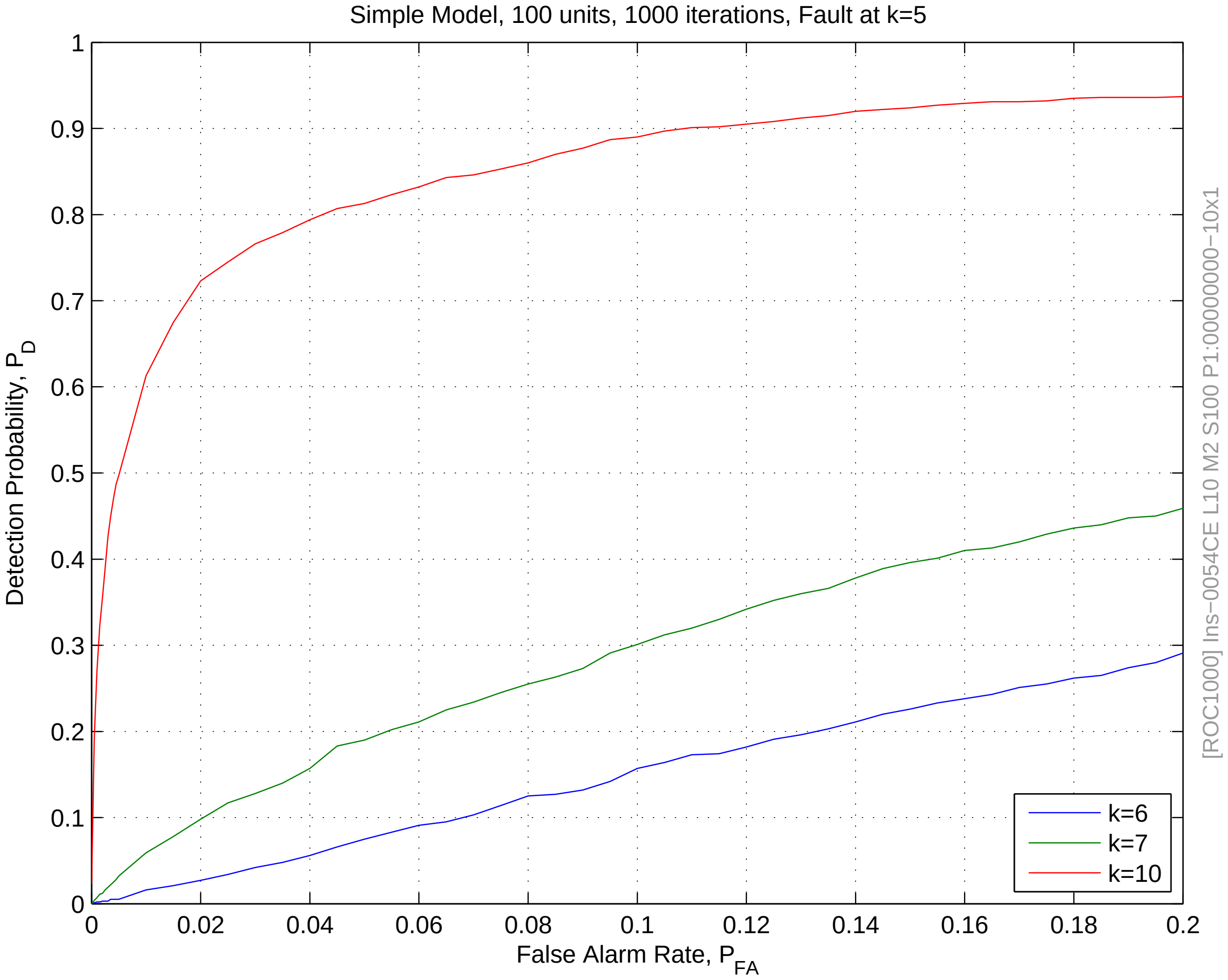


Figure 9.6: Incipient fault in Bi-Mode Models. While the ROC curve looks unusual for $k = 6, 7$, it should be remembered that the error is very small at the beginning, and the random starting location of unit parameters in a Monte Carlo simulation might actually cause some realizations to drift closer to a cluster center during the beginning stages of an incipient failure.

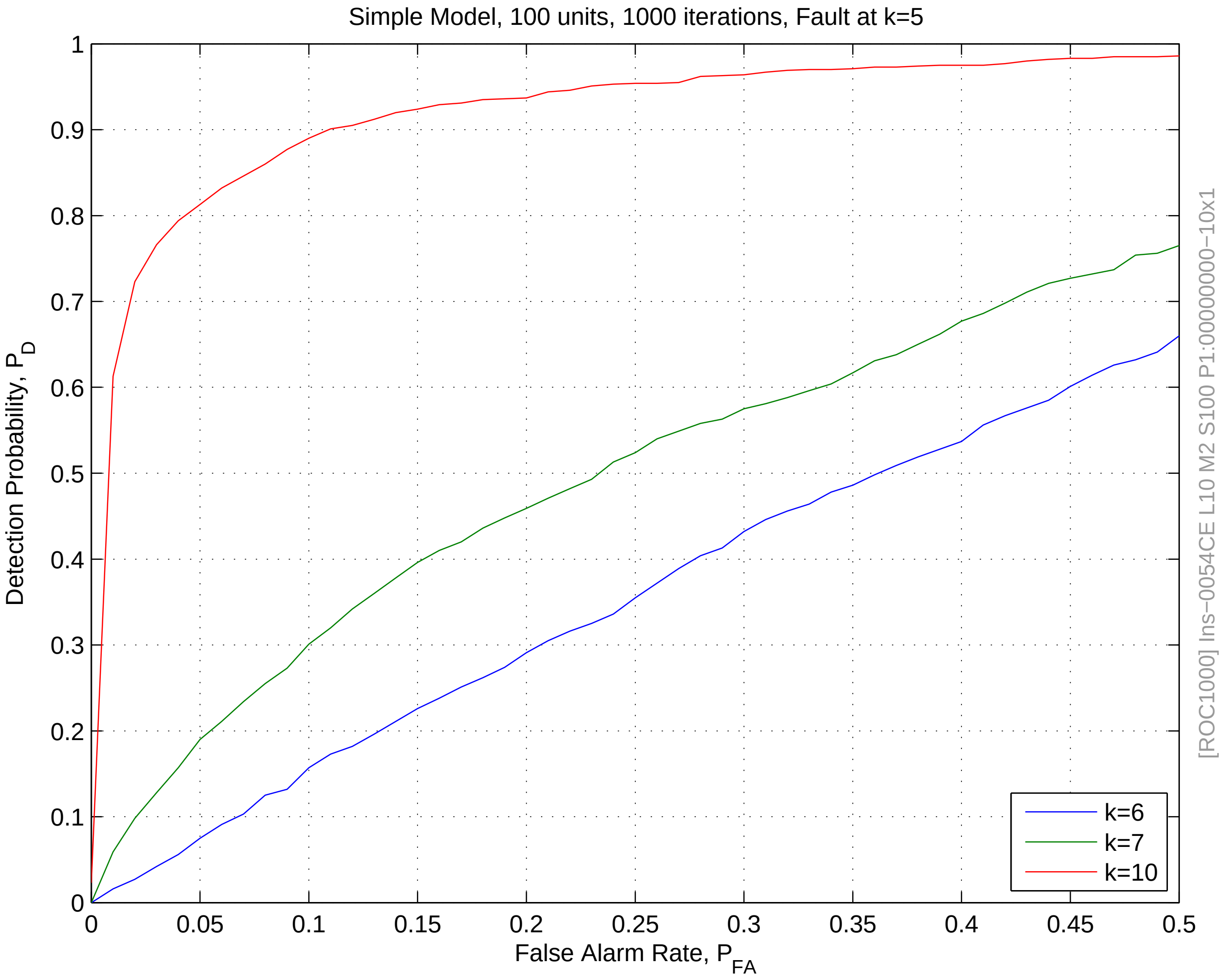


Figure 9.7: Incipient fault in Bi-Mode Models. Shown for $P_{FA} \in (0.0, 0.5)$.

10

Conclusion

Novel techniques for fault detection in multi-unit, multi-mode systems have been developed and tested. The resulting algorithm functions when information about the system modes, system faults, active mode for a unit, and structure and dynamics of the underlying data generation mechanism ("true model") are incomplete or missing.

In Chapter 3, we have examined other works that addressed fault detection in HVAC and approaches that had already been attempted and found that no techniques for dealing with multiple-modes have been suggested, and that most methodologies required a system model.

We have proposed a theoretical framework in Chapter 5, which led to a novel solution which uses information from all units in a system to generate a global parameter distribution. A practical implementation is presented in Chapter 6 which, based on techniques used in multiple-model estimation, allows for fast mode-switching when classifying the units in order to generate the distribution. Also, in order to allow fault detection for incipient and pre-existing faults, a novel application of the Truncated Normal Distribution (as detailed in in Chapter 7) was used.

The algorithm was implemented using MATLAB$^{\text{TM}}$. A simulated scenario was used to compute Receiver Operating Characteristics curves (Chapter 9). It was shown that sudden, incipient, and pre-existing failures were detectable however performance was hard to quantify due to the lack of a baseline algorithm to compare against.

A more complex testing framework was also implemented, in order to simulate an HVAC system with multiple units. The algorithm was tested on the simulated data and seemed to work very well (Chapter 8).

Finally, code was put into place to collect real-world data from three sites based in Houston and Miami (using Python and C++). The algorithm was applied to the live data and appears to show significant promise (Chapter 8).

10.1 Limitations and Capabilities

As mentioned previously, the assumptions underlying the presented algorithm are novel and hence, the algorithm has a number of interesting limitations and capabilities reproduced below for convenience.

10.1.1 Limitations

Ensemble Requirement The algorithm is dependent on the availability of multiple correlated units. It is completely inapplicable to systems with a single unit, and suffers performance degradation for systems with few units (e.g., less than twenty units for a 2-mode system).

Mode Representation Requirement In order to detect jumps from one mode to another, the new mode must have some representation, i.e., one or more units must already be in that mode, otherwise the jump could be considered a fault. Obviously, this is not an issue with single-mode systems however one of the more significant strengths of the algorithm is the ability to effortlessly adapt as units jump from mode to mode.

Slower Detection Than Uni-Mode,No-Drift Algorithms Since the distribution of the global parameters may have a larger covariance than that of a local parameter estimate, algorithms that address only a single mode and assume that parameter drift is either non-existent, or can be ignored over the time in which a fault occurs, will generally perform better (if the stated assumptions are true). Effectively, this limits applicability of single-mode slow-drift parameter algorithms to detection of sudden failures only, which may be sufficient for many problem domains. However, it should be noted that the proposed solution can be used to augment single-mode slow-drift algorithms, thereby allowing fast detection using such algorithms, while providing for detection of incipient or preexisting failures.

Central Processing Requirement As presented, the solution assumes a central location where information must be collected for analysis, which may be problematic for sensor networks. However, the information transfer requirement is minimal, consisting not of actual sample data but rather parameter statistics estimates (such as mean and covariance). Additionally, complete system synchronization is not an absolute requirement, but information from most nodes should be collected at a rate faster than parameter drift is expected to occur.

10.1.2 Capabilities

Multi-Mode Capable The proposed solution excels at addressing multi-mode systems, even when models are initially unavailable for all the modes.

Incipient and Preexisting Fault Parameter estimation explicitly accounts for parameter drift, and since fault detection does not depend on sudden deviation of local parameters, incipient and preexisting faults can be detected. This differs from techniques that account for gradual parameter drift, and assume that a fault is detectable by a sudden parameter change.

Can be used with other FD algorithms The proposed solution can be used to supplement other fault detection methodologies (such as the single-mode slow-drift algorithms mentioned above), thereby allowing for the positive features of both, and providing for detection of incipient or preexisting failures using the new algorithm.

Can Include Tertiary Mode Information If information is available about which mode is active at a particular unit, this information can be incorporated into the algorithm through the transition probability matrix or the initial mode probabilities.

Fast Parameter Initialization After Jumps The probability distribution for global parameters is being constantly estimated using data from every unit at the site. So parameter estimation after the sudden jump of a unit has the advantage of being able to use the global information to set its initial values, even if that unit has never before been seen in that mode. This reduced the "improbability"-spike which would occur otherwise, greatly reducing false positives.

10.2 Future Work

Future work will include algorithm implementation using "production" languages (such as C++ and Python) suitable for integration as an "on-line" system into large commercial real-time building automation systems. One of the primary benefits of an on-line system is that detected faults may be verified by building technicians and engineers. Experience has shown that some transient faults can only be confirmed by on-site building staff (for example, a building engineer replacing a filter could present as a fault)

Additionally, new visualization techniques such as new performance metrics, or video representations, will need to be developed to assist practitioners to better understand system performance over time and across units,.

Additionally, it is the intent of the authors to compile sample data sets in order to facilitate future work. The data collection needs for implementing fault detection using multi-unit algorithms are considerable and likely to be a stumbling block for academic researchers. In practice, data collection may require meetings with building engineers and senior building management staff, training sessions (for technicians and engineers on how to report on the efficacy of the detection algorithm), and money (in the form of implementation costs and participation incentives to offset the risk of testing an experimental solution on a live building).

A

Estimation Primitives

This appendix describes the estimation algorithms which are used as components in the proposed algorithm. Mode switching and fault detection algorithms are also needed and are discussed below.

A.1 State-Space System Representation

A very common assumption in detection, identification, and estimation is that the noise is Gaussian in nature. We will consider two cases:

Linear System: The system model can be represented using a linear state-space representation with additive process and measurement noise.

$$x_k = F_{k-1}x_{k-1} + w_k$$
$$z_k = H_k x_k + v_k$$

In this representations, we omit the known input, u_k, for simplicity. Any known input may be incorporated into a change of mean(s) of the process and/or measurement noises.

Non-linear System: A non-linear representation is required to describe the system. Process noise and measurement noise are still considered additive.

$$x_k = f_{k-1}\left(x_{k-1}, u_k\right) + w_k$$

$$z_k = h_k\left(x_k, u_k\right) + v_k$$

A.2 Linear Systems - Kalman Filter

The Kalman filter (KF) is a recursive, unbiased, minimum variance, consistent estimation technique that uses all the available data to make an optimal estimate. It provides the best possible (unbiased) estimate if the system is linear and known.

The KF is described in Table A.1. We will reference this filter as:

$$\left[\hat{x}_{k|k}^{(m,s)}, P_{k|k}^{(m,s)}, L_k^{(m,s)}\right] = \texttt{KF}\left(\hat{x}_{k-1|k-1}^{(m,s)}, P_{k-1|k-1}^{(m,s)}\right)$$

A.3 Nonlinear Systems - Unscented Transform Filter

The unscented transform (UT) is a method for estimating the statistics of a random variable which undergoes a non-linear transformation. Unlike the Extended Kalman Filter (EKF), which represents the non-linearity with a truncated Taylor series expansion, a filter built on the unscented transform (UTF) propagates specially chosen sample points through the non-linearity such that the resulting statistics capture the first two moments of the transformed distribution. This addresses the two main problems faced by the EKF: complexity of deriving the Jacobians required in the EKF, if they exist at all, and inaccuracy or instability caused by the truncation of the Taylor series expansion.

Table A.1: Kalman filter block

Predict

$$\begin{aligned}
\hat{x}_{k|k-1}^{(m,s)} &= F_{k-1}^{(m,s)}\hat{x}_{k-1|k-1}^{(m,s)} + \bar{w}_{k-1}^{(m,s)} \\
P_{k|k-1}^{(m,s)} &= F_{k-1}^{(m,s)}P_{k-1|k-1}^{(m,s)}\left(F_{k-1}^{(m,s)}\right)^T + Q_{k-1}^{(m,s)} \\
S_k^{(m,s)} &= H_k^{(m,s)}P_{k|k-1}^{(m,s)}(H_k^{(m,s)})^T + R_k \\
\tilde{z}_k^{(m,s)} &\triangleq z_k - H_k^{(m,s)}\hat{\theta}_{k|k-1}^{(m,s)} - \bar{v}_k^{(m,s)} \\
L_k^{(m,s)} &\triangleq p\left(\tilde{z}_k^{(s)}, S_k^{(m,s)}\right)
\end{aligned}$$

Update

$$\begin{aligned}
K_k^{(m,s)} &= P_{k|k-1}^{(m,s)}(H_k^{(m,s)})^T(S_k^{(m,s)})^{-1} \\
\hat{x}_{k|k}^{(m,s)} &= \hat{x}_{k|k-1}^{(m,s)} + K_k^{(m,s)}\tilde{z}_k^{(m,s)} \\
P_{k|k}^{(m,s)} &= P_{k|k-1}^{(m,s)} - K_k^{(m,s)}S_k^{(m,s)}(K_k^{(m,s)})^T
\end{aligned}$$

As discussed in [39], the weights and sample points are chosen such that:

$$\sum_{i=0}^{N}\alpha_i = 1, \quad \bar{x} = \sum_{i=0}^{N}\alpha_i x_i, \quad \mathrm{cov}\,(x) = \sum_{i=0}^{N}\alpha_i\left(x_i - \bar{x}\right)\left(x_i - \bar{x}\right)^T$$

One choice of sample points, $\mathcal{Y} \triangleq \{y_i\}$ and weights, $\mathcal{W} \triangleq \{\alpha_i\}$ is [32]:

$$(y_i, \alpha_i) = \begin{cases} \left(\bar{x}, \frac{\kappa}{N_x+\kappa}\right) & i = 0 \\ \left(\bar{x} + \left[\sqrt{\left(N_x+\kappa\right)C_x}\right]_i, \frac{1}{2(n_x+\kappa)}\right) & i = 1 \ldots N_x \\ \left(\bar{x} - \left[\sqrt{\left(N_x+\kappa\right)C_x}\right]_{i-N_x}, \frac{1}{2(N_x+\kappa)}\right) & i = N_x + 1 \ldots 2N_x \end{cases}$$

where N_x is the length of the state vector, $\kappa \in \Re$, $\left[\sqrt{\left(n_x+\kappa\right)C_x}\right]_i$ is the ith column of the matrix square root of $\sqrt{\left(n_x+\kappa\right)C_x}$.

In order to simplify the discussion, we define:

$$(\bar{y}, C_y) = \mathtt{UT}\left(g\left(x\right), \bar{x}, x_+, C_x, C_+\right)$$

$$\Rightarrow \mathcal{Y} = g\left(\mathcal{X}\left(\bar{x}, C_x\right)\right), \quad \bar{y} = x_+ + \sum_{i=0}^{N} \alpha_i y_i, \quad C_y = C_+ + \sum_{i=0}^{N} \alpha_i \left(y_i - \bar{y}_i\right)\left(y_i - \bar{y}_i\right)^T$$

Utilizing the principles of best linear unbiased estimation [67, 69], the UTF can now be described as shown in Table A.2. We will reference this filter as:

$$\left[\hat{x}_{k|k}^{(m,s)}, P_{k|k}^{(m,s)}, L_k^{(m,s)}\right] = \mathtt{UTF}\left(\hat{x}_{k-1|k-1}^{(m,s)}, P_{k-1|k-1}^{(m,s)}\right)$$

Table A.2: Unscented transform filter block

Predict

$$\begin{aligned}
\left(\hat{x}_{k|k-1}^{(m,s)}, P_{k|k-1}^{(m,s)}\right) &= \mathtt{UT}\left(f_{k-1}^{(m,s)}\left(x, u(s)_k\right), \hat{x}_{k-1|k-1}^{(m,s)}, \bar{w}_{k-1}^{(m,s)}, P_{k-1|k-1}^{(m,s)}, Q_{k-1}\right) \\
\left(\hat{z}_{k|k-1}^{(m,s)}, S_k^{(m,s)}\right) &= \mathtt{UT}\left(h_k^{(m,s)}\left(x, u(s)_k\right), \hat{x}_{k|k-1}^{(m,s)}, \bar{v}_k^{(m,s)}, P_{k|k-1}^{(m,s)}, R_k\right) \\
\tilde{z}_k^{(m,s)} &= z_k^{(s)} - \hat{z}_{k|k-1}^{(m,s)} \\
L_k^{(m,s)} &\triangleq p\left[\tilde{z}_k^{(m,s)}, S_k^{(m,s)}\right]
\end{aligned}$$

Update

$$\begin{aligned}
C_k^{(m,s)} &= \mathrm{cov}_\alpha\left(\tilde{x}_{k|k-1}^{(m,s)}, \tilde{z}_{k|k-1}^{(m,s)}\right) \\
&= \sum_{i=0}^{N} \alpha_i \left(\hat{x}_k^i - \hat{x}_{k|k-1}^{(m,s)}\right)\left(\tilde{z}_{k|k-1}^i - \tilde{z}_{k|k-1}^{(m,s)}\right)^T \\
K_k^{(m,s)} &= C_k^{(m,s)}\left(S_k^{(m,s)}\right)^{-1} \\
\hat{x}_{k|k}^{(m,s)} &= \hat{x}_{k|k-1}^{(m,s)} + K_k^{(m,s)} \tilde{z}_k^{(m,s)} \\
P_{k|k}^{(m,s)} &= P_{k|k-1}^{(m,s)} - K_k^{(m,s)} S_k^{(m,s)} (K_k^{(m,s)})^T
\end{aligned}$$

A.4 Handling Mode Transitions

The interacting multiple model (IMM) estimation algorithm has been shown to be an efficient and cost-effective hybrid estimation technique [40, 52].

The IMM estimator runs M different filters under M different system models. Each of the filters is reinitialized at each time sample with an optimally fused (in the MMSE-sense) set of statistics. Usually, IMM filtering assumes that the estimate is normally distributed, so the estimate mean and covariance are sufficient to describe the estimate to be fused.

It is assumed that a system starts in a random mode, $M_0^{(s)}$, and experiences Markovian mode transitions depending on the active mode at the last time sample,

$$P\{M_0 = m\} = P_i^m$$

$$P\{M_k = m | M_{k-1}^{(s)} = o\} = \Pi_k\left(m, o\right)$$

With the Markovian transitions completely defined, the IMM algorithm is described in Table A.3.

We will reference this algorithm as:

$$\left[\{\hat{x}_{k|k}^{(m,s)}, P_{k|k}^{(m,s)}\}, \{L_k^{(m,s)}\}\right] = \mathtt{IMM}\left(\mathtt{F}_S, \mathtt{F}_L\right)$$

where $\mathtt{F}_S$ denotes the filter used to estimate the states for each mode, and $\mathtt{F}_L$ denotes the filter used to estimate the mode likelihoods and relative probabilities.

A.5 CUSUM

First proposed by Page [50], the CUSUM algorithm is a sequential probability ratio test which is restarted as long as the decision taken is $\mathcal{H}_0$. Page suggested (and it was later proven, [43,59]) that zero is the optimal lower bound if the goal is to reject the null hypothesis.

Using the notation from [8], we denote the log-likelihood ratio of hypotheses as

$$s_k = \log \frac{P\{\mathcal{H}_1 | z_k\}}{P\{\mathcal{H}_0 | z_k\}}$$

The test statistic for the CUSUM algorithm can then be written as:

$$t_k = (t_{k-1} + s_k)^+$$

where $(x)^+ = \sup(0, x)$.

The stopping rule and jump-time is given by,

$$t_d = \min\{k : t_k \geq h\}$$

CUSUM is optimal for simple hypotheses, where the parameter distribution is completely known, before and after the fault. Since this is not the case in most fault detection scenarios, a composite hypothesis must be assumed, in which case likelihood marginalization may be used as in [55].

Table A.3: Interacting Multiple Model block

Fusion: Using the likelihood from $\mathtt{F}_L$, and the state statistics from $\mathtt{F}_S$, estimate the local parameter statistics:

$$\begin{aligned}
\mu_k^{(m,s)} &\triangleq P\left[m_k^{(m,s)}|z^k\right] = \frac{\hat{\mu}_{k|k-1}^{(m,s)} L_k^{(m,s)}}{\sum_n \hat{\mu}_{k|k-1}^{(n,s)} L_k^{(n,s)}} \\
\hat{x}_{k|k}^{(s)} &= E\left[x_k^{(s)}|\{z_k^{(s)}\}\right] = \sum_m \hat{x}_{k|k}^{(m,s)} \mu_k^{(m,s)} \\
\tilde{x}_{k|k}^{(s)} &\triangleq \left(\hat{x}_{k|k} - \hat{x}_{k|k}^{(m,s)}\right) \\
P_{k|k}^{(s)} &= \sum_m \left[P_{k|k}^{(m,s)} + \tilde{x}_{k|k}^{(s)} \left(\tilde{x}_{k|k}^{(s)}\right)^T\right] \mu_k^{(m,s)}
\end{aligned}$$

Mix: Mode probabilities are adjusted for diffusion through the $\Pi_k(\dots)$ function:

$$\begin{aligned}
\hat{\mu}_{k|k-1}^{(m,s)} &\triangleq P\left\{m_k^{(m,s)}|z^{k-1}\right\} = \sum_n \pi_k^{(mn,s)} \mu_{k-1}^{(n,s)} \\
\mu_{k-1}^{j|i} &\triangleq P\left\{m_{k-1}^{(j)}|m_k^{(m,s)}, z^{k-1}\right\} = \frac{\Pi_k(j,i)\, \mu_{k-1}^{(j)}}{\hat{\mu}_{k|k-1}^{(m,s)}}
\end{aligned}$$

Reinitialization: The parameter statistics are mixed according to relative mode probabilities:

$$\begin{aligned}
\bar{x}_{k-1|k-1}^{(m,s)} &\triangleq E\left[x_{k-1}|m_k^{(m,s)}, z^{k-1}\right] = \sum_j \hat{x}_{k-1|k-1}^{(j)} \mu_{k-1}^{j|i} \\
\tilde{x}_{k-1|k-1}^{(mj,s)} &\triangleq \bar{x}_{k-1|k-1}^{(m,s)} - \hat{x}_{k-1|k-1}^{(j,s)} \\
\bar{P}_{k-1|k-1}^{(m,s)} &= \sum_j \mu_{k-1}^{(j|m,s)} P_{k-1|k-1}^{(j,s)} + \mu_{k-1}^{(j|m,s)} \tilde{x}_{k-1|k-1}^{(mj,s)} \left(\tilde{x}_{k-1|k-1}^{(mj,s)}\right)^T
\end{aligned}$$

B

Expectation Maximization for Multi-Unit Scenarios

B.1 Model

Unit index $s = 1, 2, \ldots, S$

Mode index $m = 1, 2, \ldots, M$

For each unit $s = 1, 2, \ldots, S$ the measurements $z_k^{(s)},\ k = 1, 2, \ldots, K$ are obtained from a Gaussian mixture:

$$z_k^{(s)} \sim f^{(s)}(z) = \sum_{m=1}^{M} \pi_m^{(s)} \mathcal{N}\left(z; \mu_m, C_m\right)$$

The possible modes (mixtures components) $\mathcal{N}\left(z; \mu_m, C_m\right),\ m = 1, 2, \ldots, M$ are common for all units but the distributions of these modes $\left\{\pi_m^{(s)}\right\}_{i=1}^{N}$within a unit s are different for different units $s = 1, 2, \ldots, S$.

Data:

$$Z^{(s)} = \left\{z_k^{(s)} : k = 1, 2, \ldots, K\right\},\ Z = \left\{Z^{(s)} : s = 1, 2, \ldots, S\right\}$$

Unknown parameters to be estimated:

$$\theta = \left\{\mu_m, C_m, \pi_m^{(s)} : m = 1, 2, \ldots, M; s = 1, 2, \ldots, S\right\}$$

Objective: Given data Z find MLE of θ using EM.

B.2 Derivation of EM

Define hidden variables (missing data):

$$a^{(s)} = m \iff z_k^{(s)} \sim \mathcal{N}\left(z; \mu_m, C_m\right)$$

Clearly,

$$\pi_m^{(s)} = P\left\{a^{(s)} = m\right\}$$

and, for a D-dimensional measurement, $z_k^{(s)}$,

$$p\left(z_k^{(s)}|a^{(s)} = m, \theta\right) = \mathcal{N}\left(z_k^{(s)}; \mu_m, C_m\right)$$
$$= (2\pi)^{-\frac{D}{2}} \left|C_m\right|^{-1/2} \exp\left(-\frac{1}{2}\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\left(z_k^{(s)} - \mu_m\right)\right)$$

B.2.1 E-step

Let $\theta^{[n]}$ $-$ nth EM iterative estimate of θ.

Then

$$\begin{aligned} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} &= \frac{p\left(a^{(s)} = m, z_k^{(s)}|\theta^{[n]}\right)}{p\left(z_k^{(s)}|\theta^{[n]}\right)} \\ &= \frac{p\left(z_k^{(s)}|a^{(s)} = m, \theta^{[n]}\right) P\left\{a^{(s)} = m|\theta^{[n]}\right\}}{\sum_{l=1}^{M} p\left(z_k^{(s)}|a^{(s)} = l, \theta^{[n]}\right) P\left\{a^{(s)} = l|\theta^{[n]}\right\}} \end{aligned}$$

B.2.2 M-step

Calculate the expected log-likelihood to be maximized[1]

$$\begin{aligned}
Q(\theta) &= E\left[\ln \prod_{s=1}^{S} \prod_{k=1}^{K} p\left(a^{(s)}, z_k^{(s)}|\theta\right)|Z\right] \\
&= E\left[\sum_{s=1}^{S} \sum_{k=1}^{K} \ln p\left(a^{(s)}, z_k^{(s)}|\theta\right)|Z\right] \\
&= \sum_{s=1}^{S} \sum_{k=1}^{K} E\left[\ln p\left(a^{(s)}, z_k^{(s)}|\theta\right)|Z\right] \\
&= \sum_{s=1}^{S} \sum_{k=1}^{K} E\left[\ln p\left(a^{(s)}, z_k^{(s)}|\theta\right)|Z\right] \\
&= \sum_{s=1}^{S} \sum_{k=1}^{K} \sum_{m=1}^{M} P\left\{a^{(s)} = m|z_k^{(s)}, \theta\right\} \ln p\left(a^{(s)} = m, z_k^{(s)}|\theta\right) \\
&= \sum_{s=1}^{S} \sum_{k=1}^{K} \sum_{m=1}^{M} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \ln p\left(a^{(s)} = m, z_k^{(s)}|\theta\right) \\
&= \sum_{s=1}^{S} \sum_{k=1}^{K} \sum_{m=1}^{M} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \\
&\quad \times \ln\left(p\left(z_k^{(s)}|a^{(s)} = m, \theta\right) P\left\{a^{(s)} = m|\theta\right\}\right)
\end{aligned} \tag{B.1}$$

Constraints:

$$\sum_{m=1}^{M} P\left\{a^{(s)} = m|\theta\right\} = 1,\ s = 1, 2, \ldots, S \tag{B.2}$$

[1] Replace $P\left\{a^{(s)} = m|z_k^{(s)}, \theta\right\}$ by $P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}$.

Form the Lagrangian:

$$\begin{aligned}
L(\theta) &= Q(\theta) + \sum_{s=1}^{S} \lambda_s \left(1 - \sum_{m=1}^{M} P\left\{a^{(s)} = m|\theta\right\}\right) \\
&= \sum_{s=1}^{S}\sum_{k=1}^{K}\sum_{m=1}^{M} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \Bigg(\ln (2\pi)^{-\frac{D}{2}} - \frac{1}{2}\ln|C_m| \\
&\quad -\frac{1}{2}\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\left(z_k^{(s)} - \mu_m\right) + \ln P\left\{a^{(s)} = m|\theta\right\}\Bigg) \\
&\quad + \sum_{s=1}^{S} \lambda_s \left(1 - \sum_{m=1}^{M} \ln P\left\{a^{(s)} = m|\theta\right\}\right) \\
&= \sum_{s=1}^{S} \Bigg[\sum_{k=1}^{K}\sum_{m=1}^{M} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \Bigg(\ln (2\pi)^{-\frac{D}{2}} - \frac{1}{2}\ln|C_m| - \frac{1}{2}\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\left(z_k^{(s)} - \mu_m\right) \\
&\quad + \ln P\left\{a^{(s)} = m|\theta\right\}\Bigg) + \lambda_s \left(1 - \sum_{m=1}^{M} \ln P\left\{a^{(s)} = m|\theta\right\}\right)\Bigg]
\end{aligned}$$

where $\lambda_s,\ s = 1, 2, \ldots, S$ - Lagrangian multipliers

Now make $\frac{\partial L(\theta)}{\partial \theta} = 0$ to get $\theta^{[n+1]}$ as follows

For $\mu_m,\ m = 1, 2, \ldots, M$

$$\begin{aligned}
\frac{\partial L(\theta)}{\partial \mu_m} &= \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \left(-\frac{1}{2}\frac{\partial}{\partial \mu_m}\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\left(z_k^{(s)} - \mu_m\right)\right) \\
&= \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} C_m^{-1}\left(z_k^{(s)} - \mu_m\right) \\
&= 0
\end{aligned}$$

gives

$$\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} C_m^{-1}\mu_m = \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} C_m^{-1} z_k^{(s)}$$

which yields the next iteration $\mu_m^{[n+1]}$ for μ_m:

$$\mu_m^{[n+1]} = \frac{\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} z_k^{(s)}}{\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}}$$

For $C_m,\ m = 1, 2, \ldots, M$

$$\begin{aligned}
\frac{\partial L(\theta)}{\partial C_m} &= \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\left(-\frac{1}{2}\frac{\partial}{\partial C_m}\ln|C_m| - \frac{1}{2}\frac{\partial}{\partial C_m}\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\left(z_k^{(s)} - \mu_m\right)\right) \\
&= \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\left(-\frac{1}{2}C_m^{-1} + \frac{1}{2}C_m^{-1}\left(z_k^{(s)} - \mu_m\right)\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\right) \qquad \text{(B.3)} \\
&= 0
\end{aligned}$$

gives

$$\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} C_m^{-1} = \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\left(C_m^{-1}\left(z_k^{(s)} - \mu_m\right)\left(z_k^{(s)} - \mu_m\right)' C_m^{-1}\right)$$

$$\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} C_m = \sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\left(z_k^{(s)} - \mu_m\right)\left(z_k^{(s)} - \mu_m\right)'$$

which yields the next iteration $C_m^{[n+1]}$ for C_m:

$$C_m^{[n+1]} = \frac{\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\left(z_k^{(s)} - \mu_m^{[n+1]}\right)\left(z_k^{(s)} - \mu_m^{[n+1]}\right)'}{\sum_{s=1}^{S}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}}$$

For $\pi_m^{(s)},\ s = 1, 2, \ldots, S,\ m = 1, 2, \ldots, M$

$$\begin{aligned}
\frac{\partial L(\theta)}{\partial \ln P\left\{a^{(s)} = m|\theta\right\}} &= \sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\frac{\partial \ln P\left\{a^{(s)} = m|\theta\right\}}{\partial P\left\{a^{(s)} = m|\theta\right\}} - \lambda_s\frac{\partial P\left\{a^{(s)} = m|\theta\right\}}{\partial P\left\{a^{(s)} = m|\theta\right\}} \\
&= \sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\frac{1}{P\left\{a^{(s)} = m|\theta\right\}} - \lambda_s\frac{\partial P\left\{a^{(s)} = m|\theta\right\}}{\partial P\left\{a^{(s)} = m|\theta\right\}} \\
&= 0
\end{aligned}$$

gives

$$P\left\{a^{(s)} = m|\theta\right\} = \frac{1}{\lambda_s}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \qquad \text{(B.4)}$$

To get λ_s plugging in (B.4) into the constraint (B.2)

$$\begin{aligned}\sum_{m=1}^{M} P\left\{a^{(s)} = m|\theta\right\} &= \frac{1}{\lambda_s}\sum_{m=1}^{M}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\} \\ &= 1\end{aligned}$$

gives

$$\lambda_s = \sum_{m=1}^{M}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}$$

Then

$$\begin{aligned}P\left\{a^{(s)} = m|\theta\right\} &= \frac{\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}}{\sum_{m=1}^{M}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}} \\ &= \frac{1}{M}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}\end{aligned}$$

which is the next iteration $\pi_i^{(s)[n+1]}$ for $\pi_m^{(s)}$:

$$\pi_m^{(s)[n+1]} = \frac{1}{M}\sum_{k=1}^{K} P\left\{a^{(s)} = m|z_k^{(s)}, \theta^{[n]}\right\}$$

Note that $\pi_i^{(s)[n+1]}$ depend directly only on data $Z^{(s)}$ and indirectly on $Z\backslash Z^{(s)}$ through $\theta^{[n]}$. Thus, even though $Z\backslash Z^{(s)}$ is not directly related to the distribution $\pi_m^{(s)}$, it provides information for estimating $\pi_i^{(s)}$ through the estimates of the common (coupling) parameters μ_m, C_m.

C

Fault Construction Approach

C.1 Introduction

This appendix outlines one of the initial approaches taken to address fault detection in large-scale systems with similar units.

The principal idea was to perform online model identification using a technique similar to EMA [37, 41, 42], from which an error model set is generated. This approach requires the user to parametrically define a set of possible errors, or a set of error-generation functions which could be tested against. For example, partial and complete sensor or actuator failures can be easily simulated through very basic manipulation of the observation and input matrices in a state-space system.

After exploring this mechanism, it was decided that the limitation of this approach were unrealistic and unlikely to perform well in real-world scenarios.

C.2 Representation

A Gaussian Mixture, $f(z)$, is a convex combination of Gaussian distributions:

$$\begin{aligned}
\mathcal{N}(z|\mu,\Sigma) &= \frac{1}{|2\pi\Sigma|^{\frac{1}{2}}}\cdot\exp\left[-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\right]\\
f_i(z) &= \mathcal{N}(z|\mu_i,\Sigma_i)\\
f(z) &= \sum_{i=1}^{M} a_i f_i(z)
\end{aligned}$$

where $\sum_{i=1}^{M} a_i = 1$.

C.3 Probability Estimation

The likelihood that a particular measurement, z, came from component i is:

$$L(z|m_i) = \frac{1}{|2\pi\Sigma_i|^{\frac{1}{2}}}\cdot\exp\left[-\frac{1}{2}(z-\mu_i)^T\Sigma_i^{-1}(z-\mu_i)\right] \tag{C.2}$$

Each measurement arrives from one of the M Gaussian components. It is possible to make a hard decision as to which component a measurement originates from by picking the component with the largest likelihood as given in Eq. C.2. Alternately, a soft decision may be made by normalizing the likelihoods across all components:

$$P(m_i|z) = \frac{L(z|m_i)}{\sum_{i=1}^{M} L(z|m_i)}$$

Since each measurement may come from a different component, and we assume that the mixture is fixed, combining the soft-decisions for multiple measurements, $\mathcal{Z} = [z_1 \ldots z_N]$, we get:

$$P(m_i|\mathcal{Z}) = \frac{1}{N}\sum_{k=1}^{N}\frac{L(z_k|m_i)}{\sum_{i=1}^{M} L(z_k|m_i)} \tag{C.4}$$

C.4 Recursive Probability Update

By observation, we can rewrite the sample mean calculation in recursive form:

$$\bar{x}_k = \frac{k-1}{k}\bar{x}_{k-1} + \frac{1}{k}x_k$$

Applying this to the probability estimate, Eq. C.4, yields

$$\begin{aligned} \gamma &= \frac{k-1}{k} \\ P\left(m_i|z_k\right) &= \gamma P\left(m_i|z_{k-1}\right) + \left(1-\gamma\right)\frac{L\left(z_k|m_i\right)}{\sum_{i=1}^{M} L\left(z_k|m_i\right)} \end{aligned}$$

C.5 Multiple Model Estimator

Using the idea we presented above, we can write a multiple model estimator based on the AMM that assumes (in the model probability update phase) a mixture rather than a single model being true. This should give us a much better estimate of the mixture probabilities.

$$L_k^{(i)} \triangleq p\left[\tilde{z}_k^{(i)}|m_k^{(i)}, z^{k-1}\right] \doteq \frac{\exp\left[-\frac{1}{2}(\tilde{z}_k^{(i)})^T(S_k^{(i)})^{-1}(\tilde{z}_k^{(i)})\right]}{|2\pi S_k^{(i)}|^{\frac{1}{2}}} \tag{C.7a}$$

$$\alpha_k^{(i)} = \frac{k-1}{k}\alpha_{k-1}^{(i)} + \frac{1}{k}\frac{L_k^{(i)}}{\sum_{j=1}^{M} L_k^{(j)}} \tag{C.7b}$$

We will refer to a multiple model estimator derived from an AMM estimator with the above probability update equations as the Expected Mixture Multiple Model Estimator (E_{em3}). In reality, this estimator is not expected to perform well for error detection. However it is useful as a component of the two-stage fault detector described next.

C.6 Blended Estimation

We can use the results of the EM3 estimator as the first stage of a "hierarchical" estimator. The second stage uses the resulting relative model probabilities in the EM3 estimator to create a "blended" model, in a fashion similar to that of the EMA estimator [37, 41, 42].

This blended model is then used to generate an error set. The blended and error models form the model set in the second stage of filtering using an interacting multiple model (IMM) algorithm.

We refer to this 2-stage hierarchical estimator as the Blended Estimator (E_b).

C.7 FDD Approach

- The mixture estimation filter is used to identify the model probabilities, which are then used to design an FDD model set for each time sample. This on-the-fly model set is used for fault detection.

- A single mixture estimation filter is used that contains a set of models such that the truth is expected to be contained within (the $\mathcal{H}_0$ set). Additional models represent faults. The fault models are adapted on-line as a result of the changing mixing probabilities of the $\mathcal{H}_0$ set.

- In the spirit of EM Gaussian location and covariance updates, a bank of parameter estimators can be updated based on $P(\tilde{z}_i|\theta_i)$ for estimator i. This allows us to simultaneously estimate parameters for multiple modes based on a soft-assignment for which mode the observation came from.

- SSPRT detection can be performed on $\mathcal{H}_0$ using $\hat{\mathcal{M}}_0$ and a set of fault models,$\mathcal{M}_1 \ldots \mathcal{M}_M$, designed around $\hat{\mathcal{M}}_0$. For a window length, w, mixing probabilities (Eq. C.7) are computed using only the last w samples, i.e.,

$$\alpha_k^{(i,w)} = \frac{1}{w+1} \sum_{n=0}^{w} \frac{L_{k-n}^{(i)}}{\sum_{j=1}^{M} L_{k-n}^{(j)}} \tag{C.8a}$$

This provides a better mixture representation of $\mathcal{H}_1$. A bank of SSPRT are then run against the candidate $\tilde{\mathcal{M}}_1$'s.

C.8 GM Experiments

The truth is given by $\mu_{act}(1)$ for $k = 1 \dots a$, $\mu_{act}(2)$ for $k = a+1 \dots N$. Fault sets are designed by adding pre-specified offsets to the estimated true model.

$$\mu^i_{des} = \mu_{est} + \mu^i_{off}$$

$$\mu_{off} = \begin{bmatrix} 1.0 & 0.0 & 1.0 \\ 0.0 & 1.0 & 1.0 \end{bmatrix}$$

The reference models provided to the IMM estimator have the correct covariance and means given by:

$$\mu_{ref} = \begin{bmatrix} 1.0 & 0.0 & 1.0 & 2.0 & 0.0 & 2.0 \\ 0.0 & 1.0 & 1.0 & 0.0 & 2.0 & 2.0 \end{bmatrix}$$

C.8.1 Favor Correct $\mathcal{H}_0$ Identification

$$a = 20$$

$$\Sigma^2 = \begin{bmatrix} 0.1^2 & 0 \\ 0 & 0.1^2 \end{bmatrix}$$

$$\mu_{act} = \begin{bmatrix} 0.5 & 0.5 \\ 0.6 & 1.4 \end{bmatrix}$$

C.8.2 Fault Caused by Deviation From $\mathcal{H}_0$

$$a = 20$$

$$\Sigma^2 = \begin{bmatrix} 0.1^2 & 0 \\ 0 & 0.1^2 \end{bmatrix}$$

$$\mu_{act} = \begin{bmatrix} 0.6 & 1.0 \\ 0.3 & 1.5 \end{bmatrix}$$

C.8.3 Large Covariance to Blur Models

$$a = 20$$

$$\Sigma^2 = \begin{bmatrix} 0.32^2 & 0 \\ 0 & 0.32^2 \end{bmatrix}$$

$$\mu_{act} = \begin{bmatrix} 0.5 & 0.5 \\ 1.0 & 1.6 \end{bmatrix}$$

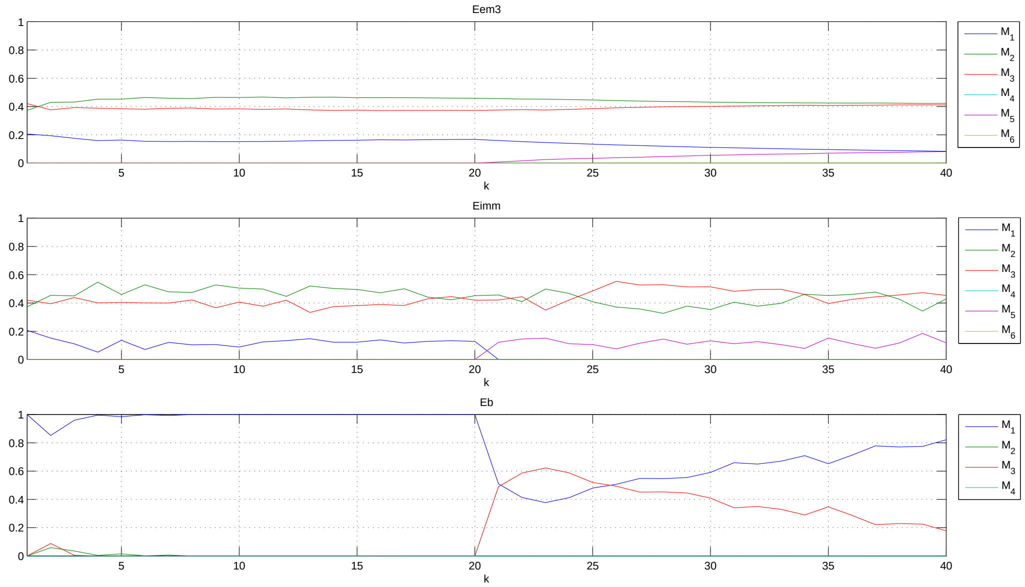


Figure C.1: Favor Correct $\mathcal{H}_0$ Identification

C.9 Results

In Figs. C.1-C.3, models 1-3 are no-fault models for E_{em3} and E_{imm}, with the faults contained in models 4-6. For the E_b estimator, model 1 is only the no-fault model, since it is a weighted combination of models 1-3 in the E_{em3} estimator.

As can be seen from the results, the EM3 estimator, being a non-interacting model estimator, responds very slowly in terms of relative likelihoods.

Fig. C.1 shows the truth to be a combination of the three models in our initial set. This can be seen quite clearly from the E_{em3} and E_{imm} estimators. The E_b estimator uses the combination discovered by E_{em3} to generate the Null Hypothesis model, Model 1. At time $k = 20$, an error occurs which is very quickly recognized to be different from the previously assumed combination - this results in E_b detecting a fault very quickly. While some probability reassignment occurs in E_{imm}, the fault still resembles some of the models in the initial set and so no fault detection occurs.

The next scenario (Fig. C.2) is a situation where the fault is of a larger magnitude. It still

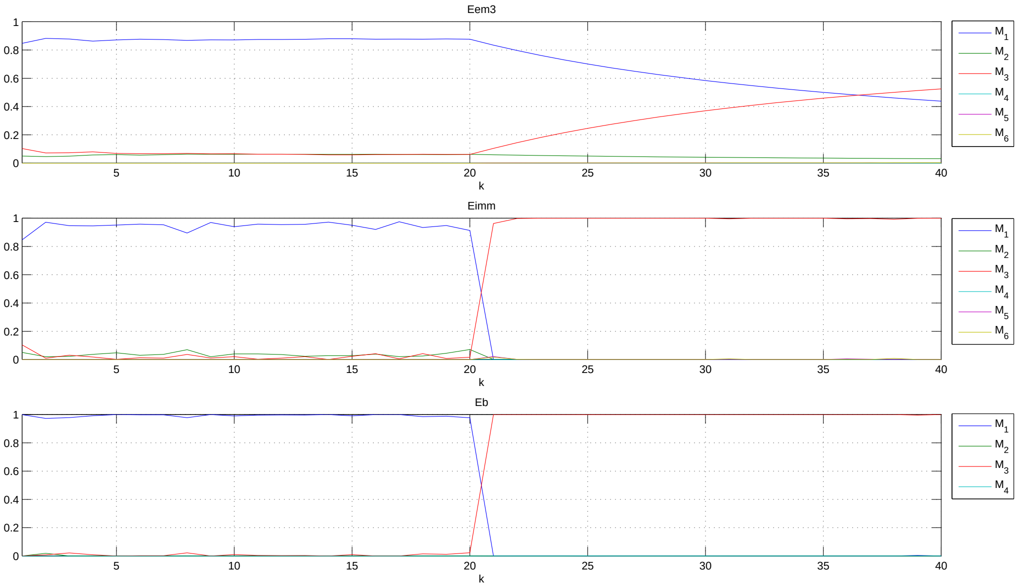


Figure C.2: Fault Caused by Deviation From $\mathcal{H}_0$

closely resembles one of the initial models in the set, but the fault is detected by E_b as sudden deviation from $\mathcal{H}_0$. While E_{imm} detects a change as quickly as E_b, it does not correctly categorize it as a fault.

The final scenario (Fig. C.3) shows a relatively smaller fault and uses a large process noise to "blur" the models. In this case, the E_{imm} estimator performs better, since the new model set has been enlarged, in a way. The fault is close enough (in a probabilistic sense) to one of the covariance-blurred fault models, resulting in a quick identification by the E_{imm} estimator. The E_b estimator also identifies a fault, but more slowly.

C.10 Conclusion

It is clear that two-stage multiple model estimation has promise, and may be an ideal solution for some problem domains. However, we are most concerned with fault detection for systems with similar, correlated units. One of the shortcomings of this approach is that the E_b estimator does not use the additional information which can be gathered from other units in the system.

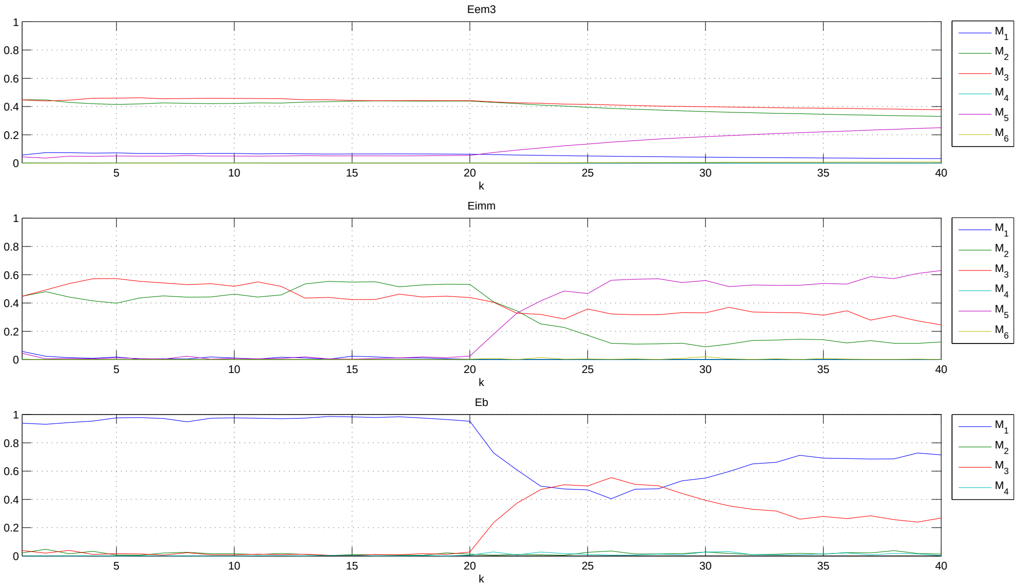


Figure C.3: Large Covariance to Blur Models

Another shortcoming, which is possibly more serious for practitioners, is the need to define valid models, as well as the possible errors, for the system.

D

Gaussian Mixture Identification Approach

D.1 Nomenclature

$\mathcal{H}$	Hypothesis.
$\mathcal{E}$	Estimator.
Δ	Distance.
M_o	Set of Gaussian mixture components for $\mathcal{H}_o$.
μ_m	The mean of Gaussian component m.
σ_m	The variance of Gaussian component m.
$\vec{x}$	Indicates that x is a vector.
p	Dimensionality of the Gaussians in a particular model set.
N	Number of samples in a single detection simulation run.
R	Number of ensembles used in the simulation.
$\vec{z}_r$	Measurement data for run r.
$T_d(\vec{z})$	Test statistic for measurements, $\vec{z}$.
P_{FA}	False alarm rate.
P_D	Detection probability.
T_{FA}^{d}	Threshold for detector d which produces a false alarm rate of P_{FA}.

D.2 Introduction

Finite mixture models have been popularly used to model complex probability distributions. One particularly important mixture model is the Gaussian mixture density, which utilizes the easily represented and versatile Gaussian distribution. A Gaussian mixture model can be used to represent any given probability density with an arbitrary degree of accuracy, if enough mixture components are used [60].

As part of the evaluation of the suitability of Gaussian mixtures in representing the overall parameter distribution in a building, some time was spent in studying the properties of different GM-identification mechanisms. The direct GM-identification approach was ultimately abandoned in favor of the algorithms presented in the main body of this dissertation (for the reasons mentioned in D.8), however this appendix outlines the direction of the research.

D.3 Problem Statement

In the following treatment, we assume the following: There are N samples in a run of data, $[z_1 \dots z_N]$. For a Gaussian white additive noise (GWAN) scenario, a typical set of hypotheses is

$$\mathcal{H}_0 : z_k \sim \mathcal{N}(\mu_0, \sigma_0^2) \tag{D.1a}$$

$$\mathcal{H}_1 : z_k \sim \mathcal{N}(\mu_1, \sigma_1^2) \tag{D.1b}$$

Unless otherwise specified, under $\mathcal{H}_0$, $z_k \sim \mathcal{N}(0, 1^2)$.

For Gaussian Mixture scenarios (GMIX), we formulate the problem as

$$\mathcal{H}_0 : z_k \sim \sum_{m \in M_0} a_m \mathcal{N}(\mu_m, \sigma_m^2) \tag{D.2a}$$

$$\mathcal{H}_1 : z_k \sim \sum_{m \in M_1} a_m \mathcal{N}(\mu_m, \sigma_m^2) \tag{D.2b}$$

where M_0 and M_1 are sets containing the Gaussian mixture components of each hypothesis, respectively. Unless otherwise specified, the null hypothesis, $\mathcal{H}_0$ contains a single component which is distributed as above for the WAN case (i.e. $\mathcal{N}(0, 1^2)$).

The likelihood of a mixture at a point, z_k, is given by:

$$\mathcal{L}(z_k; \vec{\mu}, \vec{\sigma}^2) = \frac{1}{M} \sum_{m=1}^{M} \frac{a_m}{\sqrt{2\pi\sigma_m^2}} \cdot \exp\left[-\frac{1}{2\sigma_m}(x - \mu_m)^2\right] \tag{D.3}$$

So, the likelihood that a single measurement came from a particular hypothesis $\mathcal{H}_o$, or equivalently, a particular model set, M_o, with Gaussian models $m \in M_o$, each of dimension p, is:

$$\mathcal{L}_o\left(z\right) = \sum_{m=1}^{M} a_m \cdot \frac{1}{(2\pi)^{\frac{1}{p}} |\Sigma|^{\frac{1}{2}}} \cdot \exp\left[-\frac{1}{2}(z - \vec{\mu}_m)^T (\Sigma_m)^{-1} (z - \vec{\mu}_m)\right] \tag{D.4}$$

D.3.1 Threshold Generation

- R runs, each with N samples, are generated for the $\mathcal{H}_0$ model.
- Each detector, d, is fed the data and produces a single test statistic number for run r, $T_d(\vec{z}_r)$.
- Problematic values such as `Inf` or `NaN` are removed and replaced with the average threshold for that detector[1].
- $T_d(\vec{z}_r)$ is sorted (for each detector) in ascending order, and T_{FA}^d is calculated for detector d at false alarm rate P_{FA} by selecting the $T_d(\vec{z}_r)$ located at $(1 - P_{FA})R$.

[1] So far, `Inf` and `NaN` are very rare, but can appear once or twice in 10,000 iterations when using the EM algorithm in a GLRT estimator

D.3.2 Detection

- R runs, each with N samples, are generated for the $\mathcal{H}_1$ model.
- Each detector, d, is fed the data and produces a single test statistic number for run r, $T_{r,d}$.
- Problematic values such as `Inf` or `NaN` are removed and replaced with the average test statistic for that detector.
- The portion of samples that exceed T^d_{FA} for a particular detector, d, is calculated for each false alarm rate, P_{FA}. This is the detection probability, P_D.

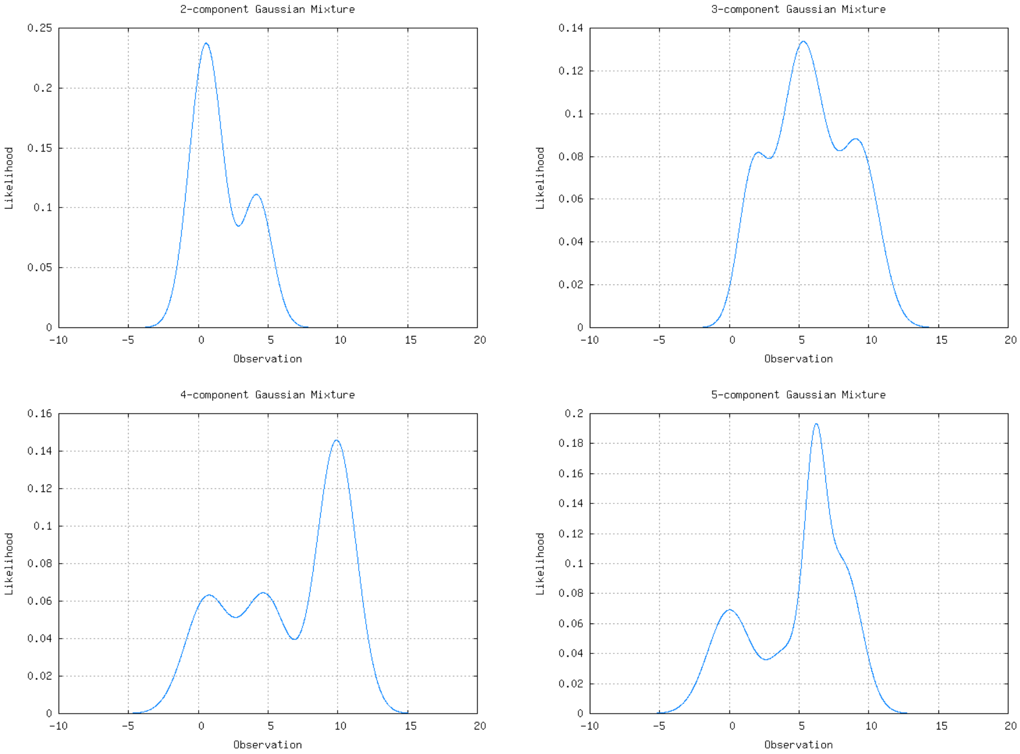


Figure D.1: Examples of Gaussian Mixtures

D.4 Distance Measures

In order to better evaluate detector performance, we would like to be able to establish how difficult a particular detection problem is. While many measures exist for the standard Gaussian white additive noise scenario (Eq. D.1), a Gaussian mixture scenario (Eq. D.2) poses a problem.

D.4.1 Deflection Coefficient

The deflection coefficient for a detector, d, is given by:

$$\Delta_{DC}(\vec{x}) = \frac{(E[T_d(x|H_1)] - E[T_d(x|H_0)])^2}{var[T_d(x|H_1)]} \tag{D.5}$$

In GWAN scenarios, Δ_{DC} completely describes detector performance - the larger the deflection coefficient, the better the performance.

D.4.2 Simple Deflection Experiment

A 2G-Mixture was generated:

$$H_0 : x_k \sim \mathcal{N}(0, 1^2) \tag{D.6a}$$

$$H_1 : x_k \sim 0.5\mathcal{N}(-\mu, 1^2) + 0.5\mathcal{N}(\mu, 1^2) \tag{D.6b}$$

and μ was varied from $0 \rightarrow 2$ (giving $\Delta_{DC} = 0 \rightarrow 4$). See Fig. D.4.2 for results.

D.4.3 Kullback-Leibler Divergence

The Kullback-Leibler (KL) divergence for a distribution is given by:

$$\Delta_{KL}\left(f(z)|g(z)\right) = \int_{-\infty}^{\infty} f(z) \log_2 \frac{f(z)}{g(z)} dz \tag{D.7a}$$

It can be understood, from an information theoretic perspective, to be the expected extra encoding length required to represent a sample distribution ($f(z)$) using a code based on a different distribution ($g(z)$). Δ_{KL} is always positive ($f(z)$ and $g(z)$ represent probability distributions), and never equal to zero unless $f(z) \equiv g(z)$.

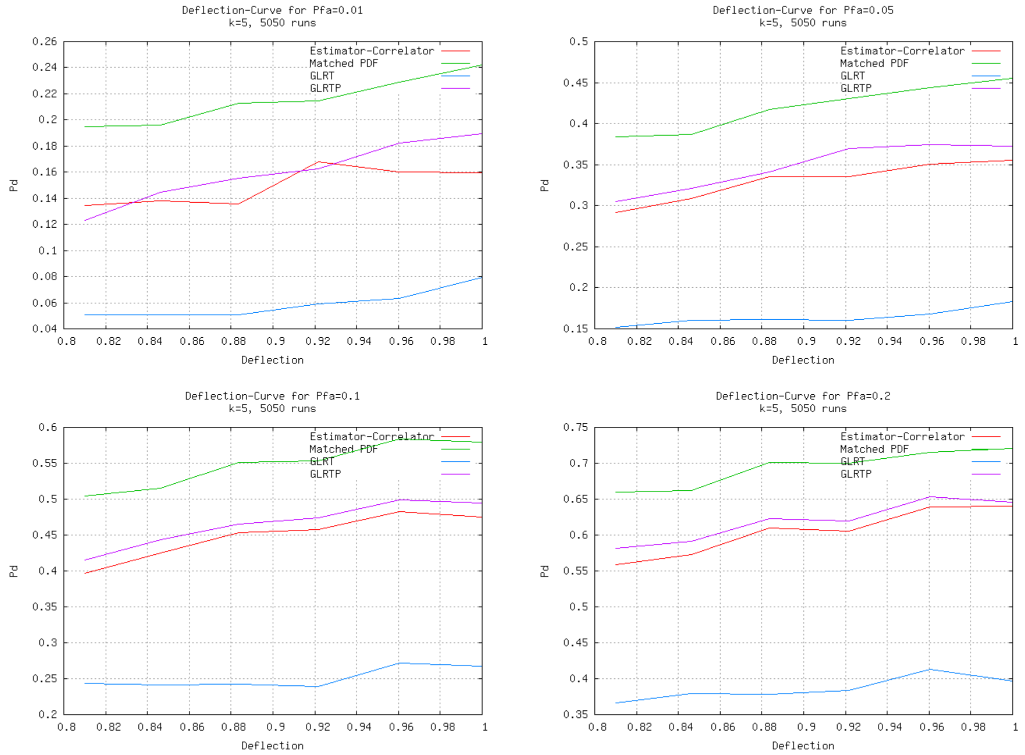


Figure D.2: *HistOverlap* experiment

It can be noted from Eq. D.7 that the KL distance is not symmetric, and may be infinite if there is incomplete coverage of $f(z)$ by $g(z)$ (if $g(z) = 0$ at some z_k, such that $f(z) \neq 0$). For Gaussians in particular, $g(z_k) > 0 \; \forall k \neq \pm \inf$.

This makes sense, since no amount of extra encoding can represent a symbol not in the original alphabet (assuming Huffman-style encoding).

Implementation

Since there is no closed form for Eq. D.7, numerical methods must be applied to solve it:

$$\Delta_{KL}\left(f(z)|g(z)\right) = \sum_{k=\frac{-A}{\Delta t}}^{\frac{B}{\Delta t}} f(k\Delta t)\log_2 \frac{f(k\Delta t)}{g(k\Delta t})\Delta t \tag{D.8a}$$

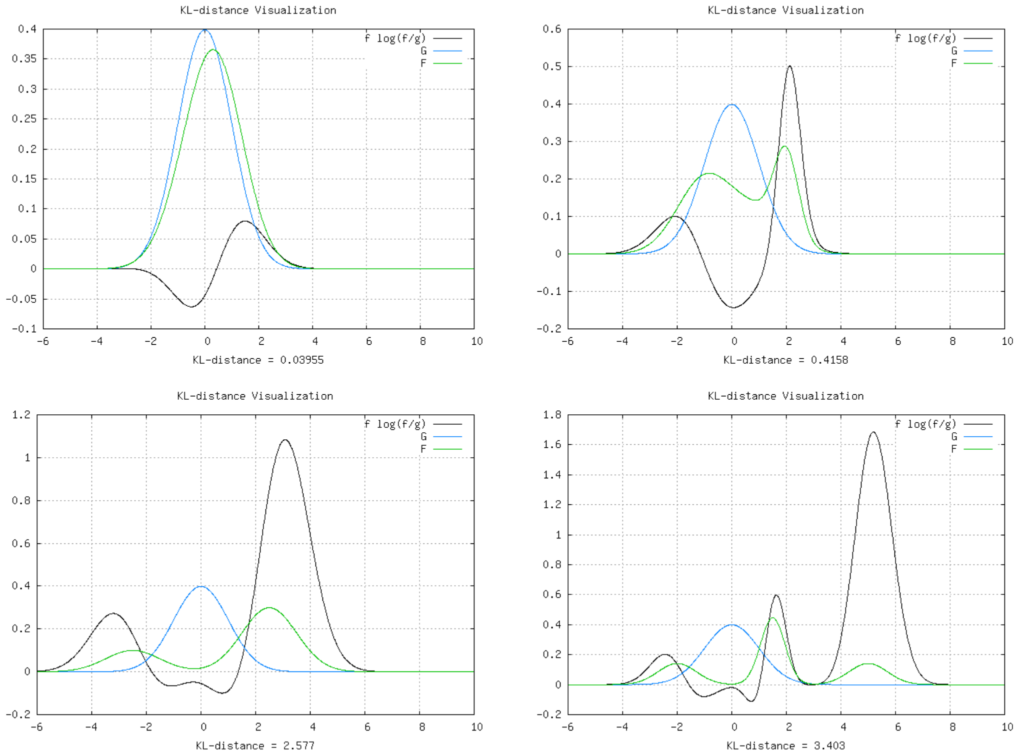


Figure D.3: $f(z)\log_2\frac{f(z)}{g(z)}$ for different mixtures

D.5 Detectors

D.5.1 Random Detector

In order to perform a basic sanity check of the detection framework, we include a random detector:

$$T_{rnd}(\vec{x}) \sim \mathcal{U}(0,1) \tag{D.9}$$

The random detector replaces the chance-line in our plots. It adds an insignificant computational burden to use the random detector (less than 0.1% overhead in most cases).

D.5.2 Matched Filter

The matched filter assumes that the signal is known. In the usual formulation, with the noise being white and uncorrelated, the matched filter detector is given by:

$$T_{mf}(\vec{x}) = \frac{1}{N}\sum_{k=1}^{N} x_k s_k \tag{D.10}$$

It correlates the data with the signal. If the noise is orthogonal to the signal (such as zero-mean uncorrelated noise), then this is an optimal detector for the single-Gaussian scenarios.

D.5.3 Estimator-Correlator

If we assume no information at all, the best we can do for the Estimator-Correlator for a set of non-GM hypotheses is the energy detector (see next section). For a set of GM hypotheses, the Estimator-Correlator is equivalent to the GLRT detector[2] (see below).

D.5.4 Energy-Detector

This detector simply measures the energy present in the signal. It is ideal for detection without any information about the signal for a WAN hypotheses (Eq. D.1).

$$T_{\mathcal{E}}(\vec{x}) = \frac{1}{N}\sum_{k=1}^{N} x_k^2 \tag{D.11}$$

While the energy detector coincides with the Estimator-Correlator for the WAN-Hypothesis, the energy of H_1 does not necessarily have more energy than H_0, and so applying this detector to such a scenario could cause it to perform worse than the chance line (or Random Detector).

We will denote the energy of a signal as $\mathcal{E}(x_k)$.

[2] Or is it?

D.5.5 $\bar{H}_0$-Detector

Another method of detecting the presence of a signal in the absence of any information on H_0 is to detect if H_0 is not true:

$$L(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[-\frac{1}{2\sigma}(x-\mu)^2\right] \tag{D.12a}$$

$$T_{\bar{H}_0}(\vec{x}) = \frac{1}{N}\sum_{k=1}^{N} \log\left[\frac{1}{L(x_k, \mu_0, \sigma_0^2)}\right] \tag{D.12b}$$

Ultimately, for $H_0 \sim \mathcal{N}(0, 1^2)$, this translates into:

$$T_{\bar{H}_0}(\vec{x}) \propto \frac{1}{N}\sum_{k=1}^{N} \log\left[\frac{1}{\exp\left[-\frac{x^2}{2\sigma}\right]}\right] \tag{D.13a}$$

$$T_{\bar{H}_0}(\vec{x}) \propto \frac{1}{N}\sum_{k=1}^{N} x_k^2 \tag{D.13b}$$

Since this statistic is essentially equivalent to that of the Estimator-Correlator, the performance is identical. For this reason, we no longer consider the $\bar{H}_0$-Detector.

D.5.6 Matched PDF Detector

For a Gaussian Mixture, we formulate the Matched PDF Detector as a matched filter that has perfect knowledge of the mixture densities (i.e. the PDF), but not the actual samples drawn. Since the GM scenarios do not have an additive noise component (see Eq. D.2), providing the actual signal is equivalent to providing the correct estimation decision.

The H_1 likelihood for the matched PDF filter is given by:

$$L\,(x) = \sum_{m=1}^{M} a_m \mathcal{N}(\mu_m, \sigma_m) \tag{D.14a}$$

$$\sum_{m=1}^{M} a_m = 1 \tag{D.14b}$$

$$L(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[-\frac{1}{2\sigma}(x-\mu)^2\right] \tag{D.15a}$$

$$T_{pdf}(\vec{x}) = \frac{1}{N}\sum_{k=1}^{N} \log\left[\frac{\sum_{m=1}^{M} a_m L(x_k; \mu_m, \sigma_m)}{L(x_k, \mu_0, \sigma_0^2)}\right] \tag{D.15b}$$

where the mixture densities, $a_m\mathcal{N}(\mu_m, \sigma_m)$ are given.

D.5.7 Generalized Likelihood Ratio Test

The GM-GLRT detector uses the likelihood function of the Matched PDF Detector given in Eq. D.14, but is not given the mixture densities. The Expectation-Maximization algorithm (EM) is used to estimate the probabilities, means and variances. Only the model order is given.

Performance of GLRT Detector

The performance of the GLRT detector is especially bad, in some cases (Sec. D.6.3) approaching that of the random detector. This makes sense, since $H_0 = \mathcal{N}(0, 1^2)$ can be represented under H_1 by appropriate choice of μ, σ^2, and a, for example, for a 2-G mixture:

$$\mu = [0.0, 0.0] \qquad \sigma^2 = [1, 1] \qquad a = [*, *] \tag{D.16a}$$

$$\mu = [0.0, *] \qquad \sigma^2 = [1, *] \qquad a = [1, 0] \tag{D.16b}$$

So, while generating the thresholds, the GLRT detector computes $\frac{L(H_1)}{L(H_0)}$, which will likely be greater than 1, since the estimated Gaussian mixture is usually closer to the sample distribution than the actual generating distribution (for example, if the sample data drawn from $\mathcal{N}(0,1^2)$ is actually distributed as $\mathcal{N}(-0.12, 1.17^2)$).

This situation is exacerbated if the separation between H_0 and H_1 is small as seen in Sec. D.6.3.

We can expect the GLRT detector to perform badly for scenarios where $H_1 \approx H_0$. The maximum likelihood estimate of H_1 is H_0 under H_0, and H_1 under H_1. Essentially, the detector statistic measures the distance of H_1 from H_0.

Furthermore, the fewer samples available, the farther the maximum likelihood estimate for $\frac{\hat{H}_1}{H_0}$ is from the truth. This means that the test statistic generated under H_0 is unnaturally high.

D.5.8 Generalized Likelihood Ratio Test - Probability Estimation Only

The GLRTP detector assumes that the locations and variances of the Gaussians in the mixture composing H_1 are known, and only the mixing probabilities are unknown. This formulation of the GLRT detector is closer to the multiple-model detection problem, since we generally know the feasible model-space for H_0 and H_1.

As can be seen from the scenarios run, this version of GLRT performs better than all the other detection schemes listed except for the clairvoyant detectors (the Matched Filter and Matched PDF Filter).

The probabilities are estimated by applying soft assignments to each sample. The mixing probability for component q, P_q, is given by[3]:

$$P_q = \frac{1}{N} \sum_{k=1}^{N} \frac{L\left(x; \mu_q, \sigma_q^2\right)}{\sum_{m=1}^{M} L\left(x; \mu_m, \sigma_m^2\right)} \tag{D.17a}$$

[3] Is this intuitive? Or does is need some proof or reference? What is this called?

D.6 Results

D.6.1 Scenario: Framework Test

The signal, noise, and observation (s_k, n_k, and x_k, respectively) are given by:

$$s_k = A$$

$$n_k = \mathcal{N}\left(0, 1^2\right)$$

$$H_0 : x_k = n_k$$

$$H_1 : x_k = s_k + n_k$$

where $A = 0.5$

See Fig. D.6.1 for results.

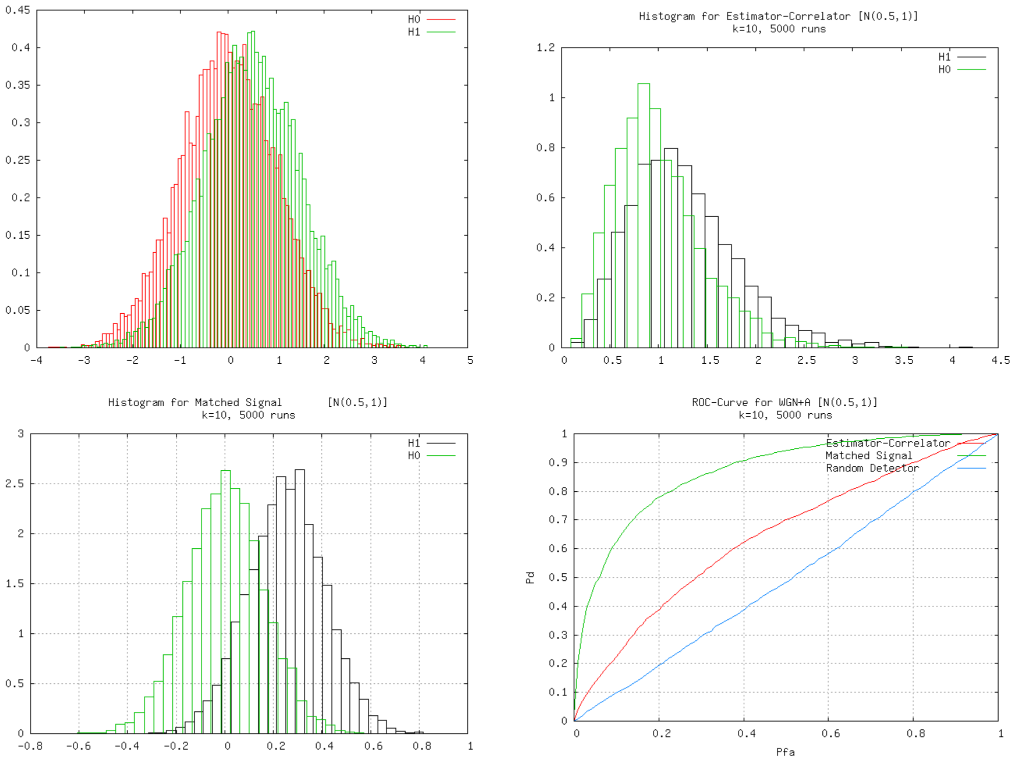


Figure D.4: *FrameTest* experiment

D.6.2 Scenario: GM Baseline

The signal, noise, and observation (s_k, n_k, and x_k, respectively) are given by:

$$s_k \sim \sum_{m=1}^{M} a_m \mathcal{N}(\mu_m, \sigma_m^2)$$

$$n_k = \mathcal{N}\left(0, 1^2\right)$$

$$H_0 : x_k = n_k$$

$$H_1 : x_k = s_k$$

where $\mu = [-0.5, 1.5]$, $\sigma^2 = [0.35, 0.35]$, and $a = [0.5, 0.5]$

See Fig. D.6.2 for results.

D.6.3 Experiment: GM Baseline With Low-Energy Signal

The above experiment is repeated with a signal containing less energy than the noise. This highlights the poor performance of Energy-detector schemes for a test that is not in the form:

$$H_0 : x_k = n_k$$

$$H_1 : x_k = n_k + s_k$$

For this scenario, $\mu = [-0.25, 0.25]$, $\sigma^2 = [0.25, 0.25]$, and $a = [0.5, 0.5]$

See Fig. D.6.3 for results.

D.6.4 Scenario: GM Baseline With Low-Energy Signal - 25 samples

$\mu = [-0.25, 0.25]$, $\sigma^2 = [0.25, 0.25]$, and $a = [0.5, 0.5]$

See Fig. D.6.4 for results.

D.6.5 Scenario: GM3 - 25 samples

$\mu = [-1.5, -0.5, 1.5]$, $\sigma^2 = [0.35, 1, 0.35]$, and $a = [0.2, 0.3, 0.5]$

See Fig. D.6.5 for results.

D.7 GM-EM Estimation

Gaussian Mixture Expected Mode Estimation can be performed in the hybrid estimation framework in batch mode. Initially, we will assume that there is no modal jump during the period in question.

To begin, the states are estimated with the assumption that all models are equiprobable. This means that the states for each model are reinitialized at the beginning of each cycle using the mixture probabilities, but the probabilities are not allowed to change during the course of a single iteration. After the iteration is complete, EM is run on the batch of data, using $\mathcal{N}(\bar{z}^K, S^K)$ for each model as the Gaussian distributions to mix.

The cycle is rerun using the new model probabilities (i.e. the mixture probabilities as identified by EM). The new observation predictions result in a new $\mathcal{N}(\bar{z}^K, S^K)$ for each model.

This process is iterated until the probabilities (and hence, observations) converge.

D.8 Conclusion

Problems:

- Current formulation is batch
- Current formulation assumes no mode jump
- Since the model probabilities cannot change within an iteration, there is an increased chance that some models will go unstable. Any model whose state diverges will impact the performance of all other models (since the state cannot be removed by decreasing it's relative probability). In order to reduce the possibility of divergence, the probabilities can be allowed

to change for the first few samples on the first iteration. An uninformative prior can be used, i.e. $\Pi = \frac{1}{M}\bar{1}$.

Benefits:

- This technique can represent mixtures very well.
- If the initial poor estimate of mixing probabilities does not cause the estimator to go unstable, then better estimation of the mixture probabilities should not do so.

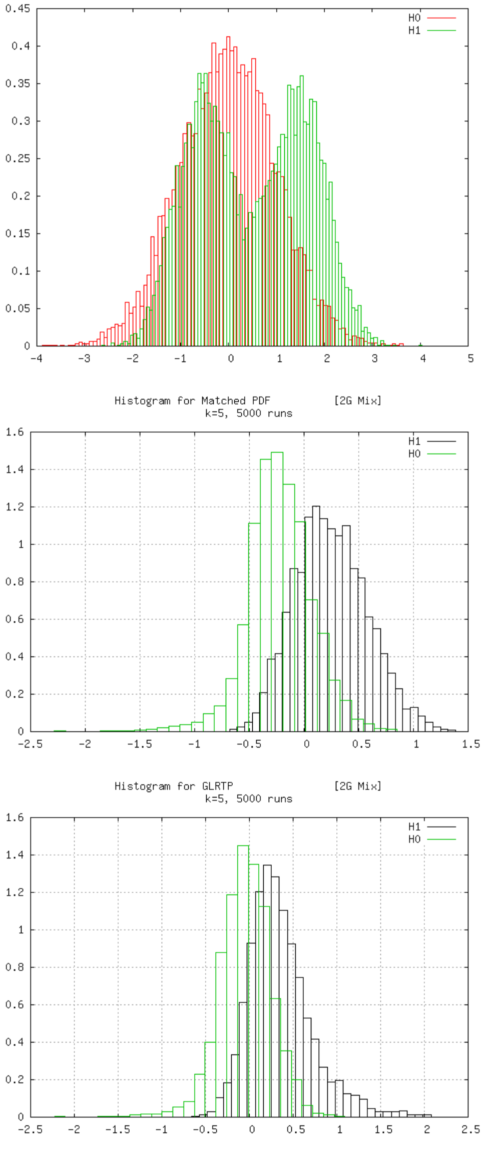


Figure D.5: *GM2* experiment

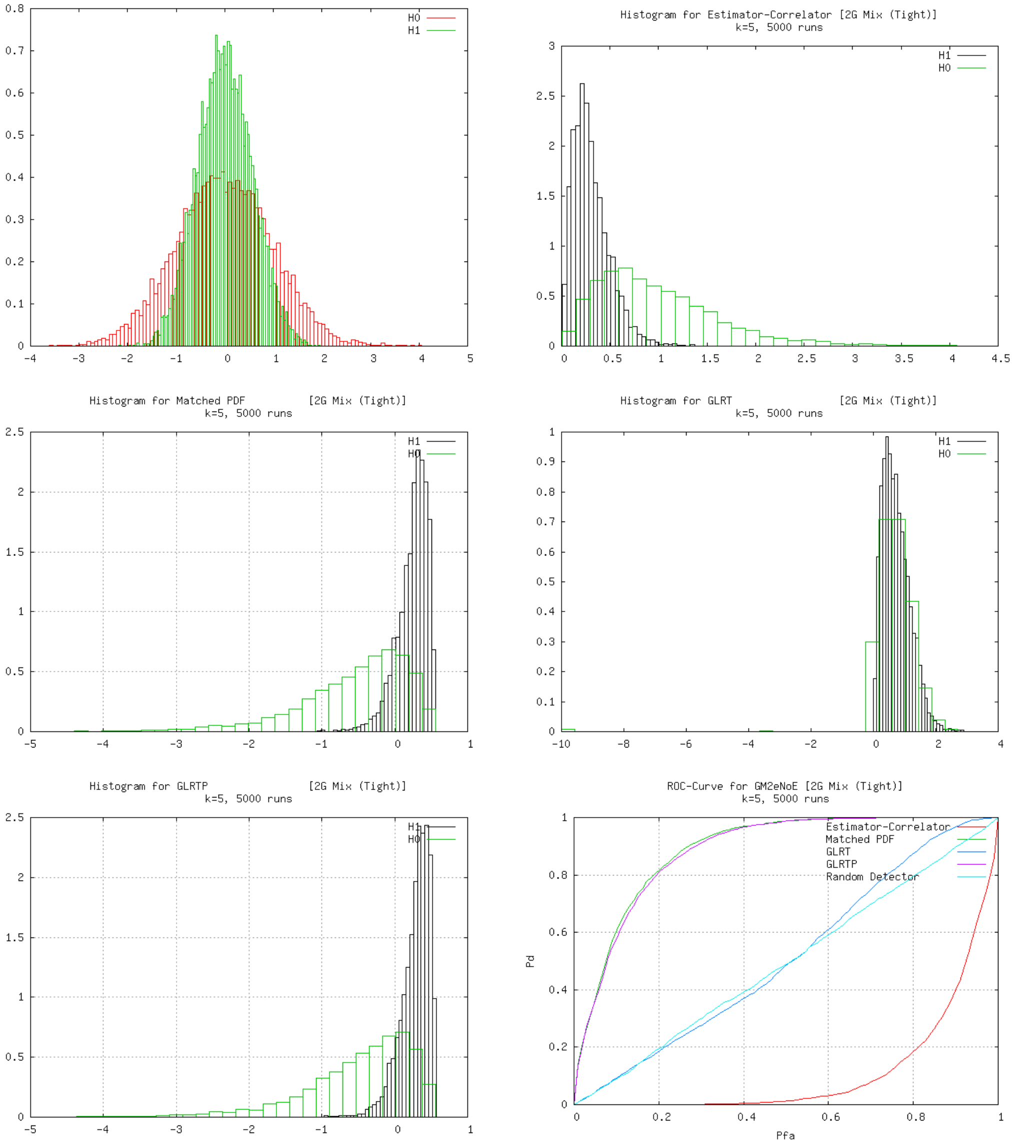


Figure D.6: *GM2-NoE* experiment

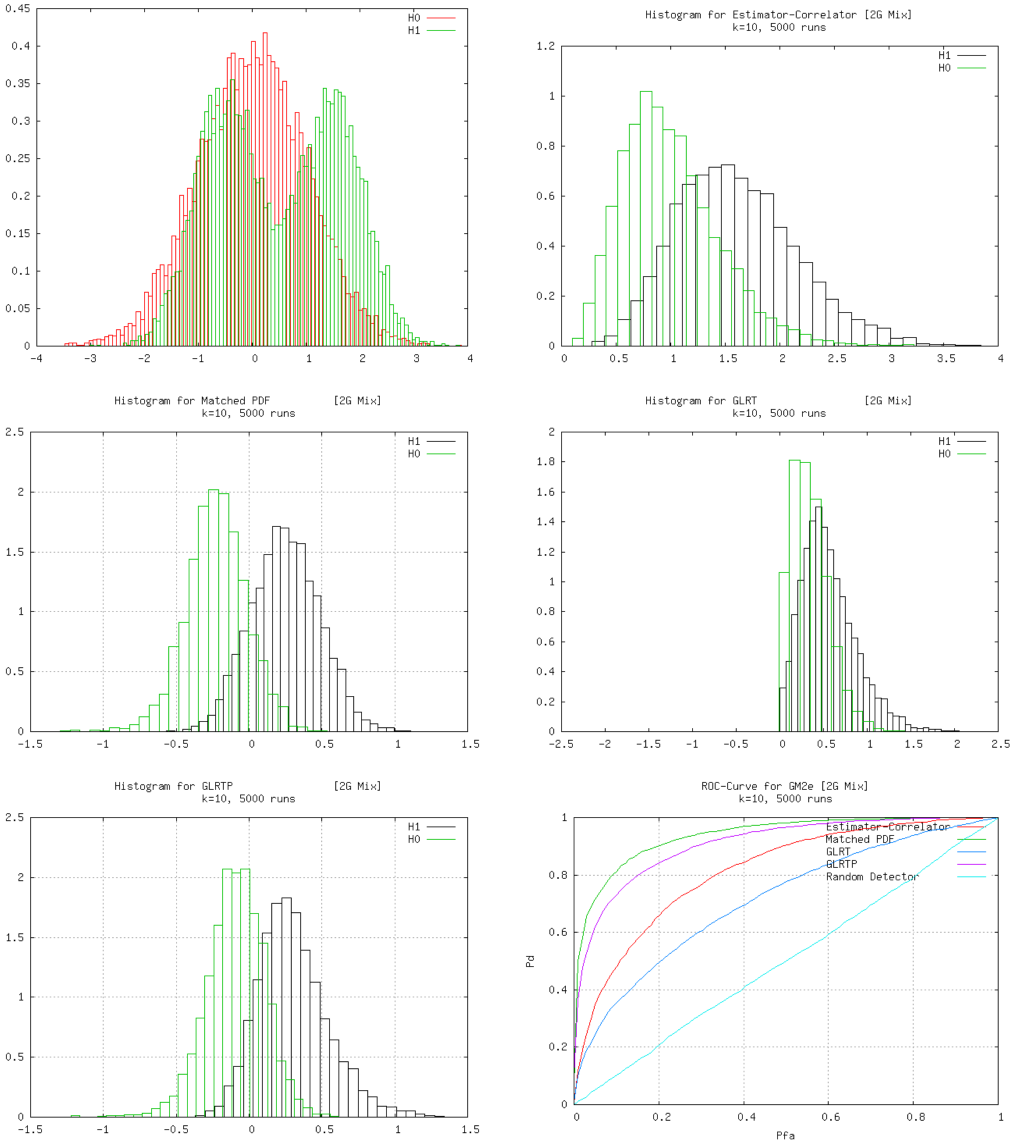


Figure D.7: *GM2-25* experiment

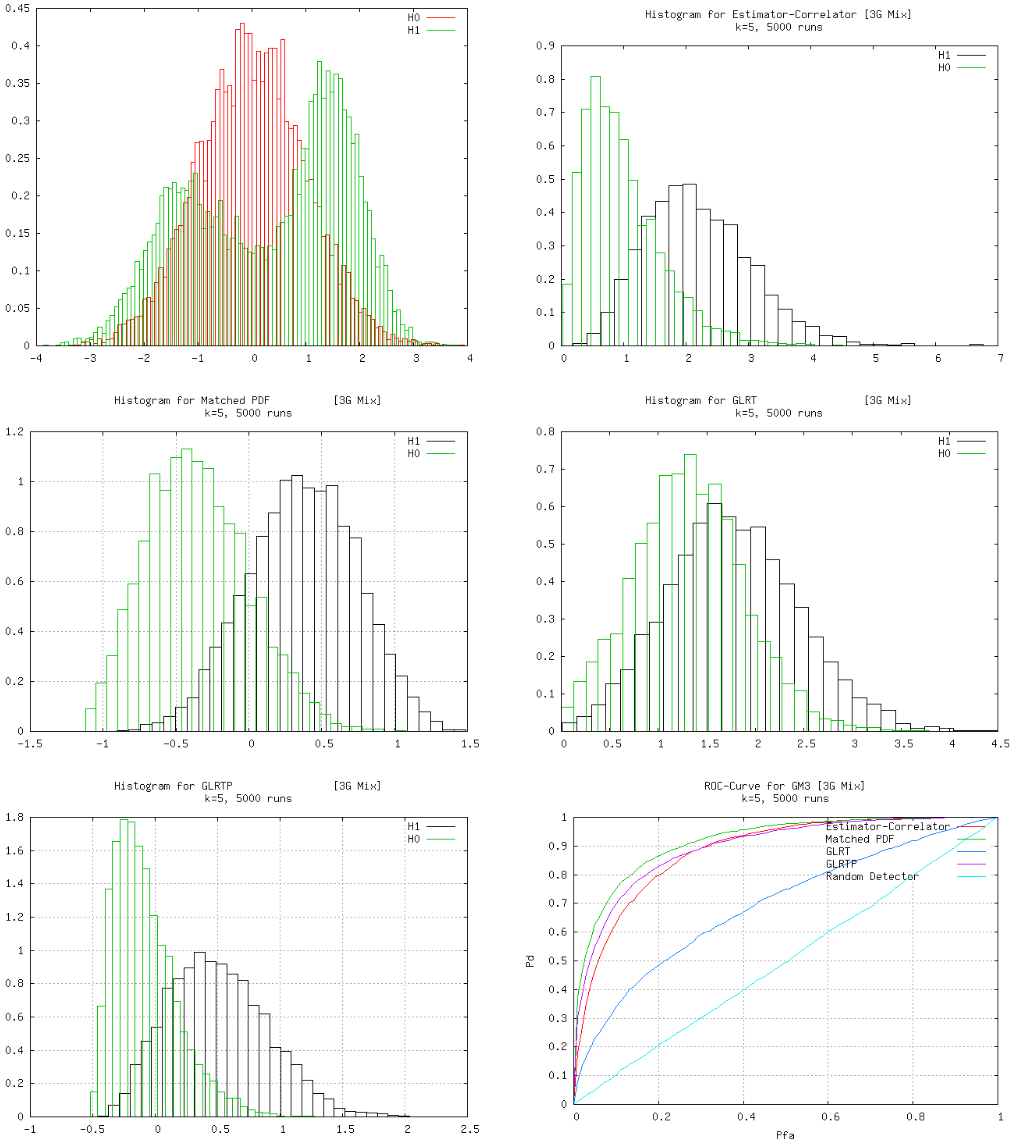


Figure D.8: *GM3-25* experiment

Bibliography

[1] M. S. Al-Homoud. Optimum Thermal Design of Office Buildings. *International Journal of Energy Research*, 21(10):941–957, 1997.

[2] Y. Bar-Shalom. Update with Out-of-Sequence Measurements in Tracking: Exact Solution. *IEEE Transactions on Aerospace and Electronic Systems*, 38:769–777, July 2002.

[3] Y. Bar-Shalom and X.-R. Li. *HVAC Systems and Equipment Handbook.* American Society of Heating Refrigerating and Air Conditioning Engineers, Atlanta, GA, 2004.

[4] A. Bashi, V. P. Jilkov, and X. R. Li. Fault Detection for Systems with Multiple Unknown Modes and Similar Units and Its Application to HVAC. *IEEE Transactions on Control Systems Technology.* Submitted.

[5] A. Bashi, V. P. Jilkov, and X. R. Li. Fault Detection for Systems with Multiple Unknown Modes and Similar Units - Part I. In *Proc. 2009 International Conf. on Information Fusion*, pages 732–739, Seattle, WA, USA, 2009.

[6] A. Bashi, V. P. Jilkov, and X. R. Li. Fault Detection for Systems with Multiple Unknown Modes and Similar Units - Part II: Application to HVAC. In *Proc. 2009 International Conf. on Information Fusion*, pages 740–747, Seattle, WA, USA, 2009.

[7] A. Bashi, V. P. Jilkov, X. R. Li, and H. Chen. Distributed Implementations of Particle Filters. In *Proc. 2003 International Conf. Information Fusion*, pages 1164–1171, Cairns, Australia, July 2003.

[8] M. Basseville and I. Nikiforov. *Detection of Abrupt Changes: Theory And Application.* Prentice Hall, Information and system science series, Englewood Cliffs, NJ, 1993.

[9] A. C. Bebbington. A Method of Bivariate Trimming for Robust Estimation of the Correlation Coefficient. *Applied Statistics*, 27(3):221–226, 1978.

[10] H. A. P. Blom and Y. Bar-Shalom. Time-Reversion of a Hybrid State Stochastic Difference System with a Jump-Linear Smoothing Application. *IEEE Transactions on Information Theory*, IT-36(4):836–847, July 1990.

[11] J. E. Braun and M. S. Breuker. Evaluating the Performance of a Fault Detection and Diagnostic System for Vapor Compression Equipment. *International Journal of Heating, Ventilating, and Air Conditioning and Refrigerating Research*, 4(4):401–425, 1998.

[12] J. E. Seem C. C. Federspiel. Temperature Control in Large Buildings. In W. S. Levine, editor, *The Control Handbook*, pages 1191–1203. CRC Press, Boca Raton, FL, 1996.

[13] B. Chen and J. K. Tugnait. Interacting Multiple Model Fixed-Lag Smoothing Algorithm for Markovian Switching Systems. *IEEE Transactions on Aerospace and. Electronic Systems*, 36(1):243–250, Jan. 2000.

[14] B. Chen and J. K. Tugnait. Tracking of Multiple Maneuvering Targets in Clutter Using IMM/JPDA Filtering and Fixed-Lag Smoothing. *Automatica*, 37(2), Feb. 2001.

[15] S. Dasgupta. Learning Mixtures of Gaussians. In *FOCS '99: Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, page 634, Washington, DC, USA, 1999. IEEE Computer Society.

[16] A. L. Dexter. Fuzzy Model Based Fault Diagnosis. *IEE Proceedings on Control Theory and Applications*, 142(6):545–550, November 1995.

[17] A. L. Dexter and M. Benouarets. A Generic Approach to Identifying Faults in HVAC Plants. *ASHRAE Transactions*, 102:550–556, 1996.

[18] D. D. Dyer. On Moments Estimation of the Parameters of a Truncated Bivariate Normal Distribution. *Applied Statistics*, 22(3):287–291, 1973.

[19] J. Hyvärinen (ed.). *Final Report Vol 2: Technical Papers of IEA Annex 25.* Finland, Technical Research Centre of Finland, Building Technology, Englewood Cliffs, NJ, 1997.

[20] T. Elteto and S. Molnar. On the Distribution of Round-Trip Delays in TCP/IP Networks. *24th Annual IEEE Intl. Conf. on Local Computer Networks*, 00:172, 1999.

[21] P. M. Frank. Fault Diagnosis In Dynamic Systems Using Analytical And Knowledge-Based Redundancy—A Survey And Some New Results. *Automatica*, 26(3):459–474, 1990.

[22] J. J. Gertler. Survey Of Model-Based Failure Detection and Isolation in Complex Plants. *Control Systems Magazine*, 8(6):3–11, Dec 1988.

[23] W. Greene. *Econometric Analysis.* Englewood Cliffs: Prentice Hall, 2003.

[24] H. T. Grimmelius, J. K. Woud, and G. Been. On-line Failure Diagnosis for Compression Refrigeration Plants. *International Journal of Refrigeration*, 18(1):31–41, 1995.

[25] R. Hallouzi, R. Babuska M. Verhaegen, and S. Kanev. Model Weight and State Estimation for Multiple Model Systems applied to Fault Detection and Identification. In *Proc. IFAC Symposium on System Identification (SYSID)*, pages 511–534, Newcastle, Australia, March 2006.

[26] R. Hallouzi, M. Verhaegen, and S. Kanev. Model Weight Estimation for FDI using Convex Fault Models. In *Proc. IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS)*, Beijing, China, August 2006.

[27] M. J. Holmes. The Simulation of Heating and Cooling Coils for Performance Analysis. In *Proc. of System Simulation in Buildings*, Lige, Belgium, 1982.

[28] W. C. Horrace. Some Results on the Multivariate Truncated Normal Distribution. *Journal of Multivariate Analysis*, 94(1):209–221, 2005.

[29] T. Hosomura. Land Cover Classification by Using Screening and Truncated Normal Distribution. *Geoscience and Remote Sensing Symposium*, 01:199–200, 2001.

[30] J. Hyvärinen and S. Karki (eds.). *Final Report Vol 1: Building Optimization and Fault Diagnosis Source Book*. Finland, Technical Research Centre of Finland, Building Technology, Englewood Cliffs, NJ, 1996.

[31] V. P. Jilkov, X. R. Li, and L. Lu. Performance Enhancement of the IMM Estimation by Smoothing. In *Proc. of the Fifth International Conference on Information Fusion*, volume 1, pages 713–720, 2002.

[32] S. J. Julier and J. K. Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems. In *Proceedings of AeroSense: The 11^{th} International Symposium on Aerospace/Defence Sensing, Simulation and Controls, Session: Multi Sensor Fusion and Resource Management II*, Orlando, FL, 1997.

[33] S. Katipamula and M. R. Brambley. Methods for Fault Detection, Diagnostics, and Prognostics for Building Systems - A Review, Part I. *International Journal of HVAC&R Research*, 11(1):3–25, 2005.

[34] S. Katipamula and M. R. Brambley. Methods for Fault Detection, Diagnostics, and Prognostics for Building Systems - A Review, Part II. *International Journal of HVAC&R Research*, 11(2):169–187, 2005.

[35] G. E. Kelly. Description of a Reference Air-Handling System. In *IEA Annex 25 Working Paper*, Lige, Belgium, 1992.

[36] W-Y. Lee, C. Park, and G. E. Kelly. Fault Detection in an Air Handling Unit Using Residual and Recursive Identification Methods. In *ASHRAE Trans*, volume 102, pages 528–539, 1996.

[37] X. R. Li and V. P. Jilkov. Expected-Mode Augmentation for Multiple-Model Estimation. In *Proc. 2001 International Conf. on Information Fusion*, 2001.

[38] X. R. Li and V. P. Jilkov. A Survey of Maneuvering Target Tracking, Part IV: Decision-Based Methods. In *Proc. 2002 SPIE Conf. on Signal and Data Processing of Small Targets*, volume 4728, pages 511–534, Orlando, FL, USA, April 2002.

[39] X. R. Li and V. P. Jilkov. Survey of Maneuvering Target Tracking: Approximation Techniques For Nonlinear Filtering. In O. E. Drummond, editor, *Signal and Data Processing of Small Targets 2004. Edited by Drummond, Oliver E. Proceedings of the SPIE, Volume 5428, pp. 537-550 (2004).*, volume 5428 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 537–550, August 2004.

[40] X. R. Li and V. P. Jilkov. Survey of Maneuvering Target Tracking, Part V: Multiple-Model Methods. In *IEEE Transactions on Aerospace and Electronic Systems*, volume 41, pages 1255–1321, October 2005.

[41] X. R. Li, V. P. Jilkov, and J.-F. Ru. Multiple-Model Estimation with Variable Structure–Part VI: Expected-Mode Augmentation. *IEEE Transactions on Aerospace and Electronic Systems*, 41(3):853–867, July 2005.

[42] X. R. Li, V. P. Jilkov, J. F. Ru, and A. Bashi. Expected-Mode Augmentation Algorithms for Variable-Structure Multiple-Model Estimation. In *Proc. IFAC 15th World Congress*, Barcelona, Spain, 2002.

[43] G. Lordern. Procedures for Reacting to a Change in Distribution. *Annals Mathematical Statistics*, 42:1897–1908, 1971.

[44] D. P. Malladi and J. L. Speyer. A Generalized Shiryaev Sequential Probability Ratio Test for Change Detection and Isolation. *IEEE Trans. Automatic Control*, AC-44(8):1522–1534, 1999.

[45] P. S. Maybeck and R. D. Stevens. Reconfigurable Flight Control Via Multiple Model Adaptive Control Methods. *IEEE Transactions on Aerospace and. Electronic Systems*, AES-27(3):470–480, May 1991.

[46] R. K. Mehra, C. Rago, and S. Seereeram. Failure Detection and Identification using a Nonlinear Interactive Multiple Model (IMM) Filtering Approach with Aerospace Applications. In *11th IFAC Symp. on System Identification*, Fukuoka, Japan, July 1997.

[47] T. E. Menke and P. S. Maybeck. Sensor/Actuator Failure Detection in the Vista F-16 by Multiple Model Adaptive Estimation. *IEEE Transactions on Aerospace and. Electronic Systems*, AES-31(4):1218–1229, Oct. 1995.

[48] G. B. Nath. Estimation in Truncated Bivariate Normal Distribution. *Applied Statistics*, 20:313–319, 1971.

[49] Charles D. Ormsby. *Generalized Residual Multiple Model Adaptive Estimation of Parameters and States*. PhD thesis, Air Force Institute of Technology, Wright Patterson Air Force Base, Ohio, 2003.

[50] E. S. Page. An Improvement to Wald's Approximation for Some Properties of Sequential Tests. *Journal of Royal Statistics Society B*, 16:136–139, 1954.

[51] H. Peitsman and V. Bakker. Application of Black-Box Models to HVAC Systems for Fault Detection. *ASHRAE Transactions*, 102:628–640, 1996.

[52] R. R. Pitre, V. P. Jilkov, and X. R. Li. A Comparative Study of Multiple-Model Algorithms for Maneuvering Target Tracking. In *Proc. SPIE - Signal Processing, Sensor Fusion and Target Recognition XIV*, volume 5913, Orlando, FL, USA, March 2005.

[53] C. Rago, R. Prasanth, R. K. Mehra, and R. Fortenbaugh. Failure Detection and Identification and Fault Tolerant Control Using the IMM-KF with Applications to the Eagle-Eye UAV. In *Proc. of the 37th IEEE Conference on Decision and Control*, volume 4, pages 4208–4213, Tampa, FL, USA, December 1998.

[54] J. Ru, V. P. Jilkov, X. R. Li, and A. Bashi. Sequential Detection of Target Maneuvers. In *Proc. 2005 International Conf. on Information Fusion*, pages 345–351, Philadelphia, PA, USA, July 2005.

[55] J. Ru, V. P. Jilkov, X. R. Li, and A. Bashi. Detection of Target Maneuver Onset. *IEEE Trans. on Aerospace and Electronic Systems*, 45(2):536–554, April 2009.

[56] J. Ru and X. R. Li. Variable-Structure Multiple-Model Approach to Fault Detection, Identification, and Estimation. *IEEE Transactions on Control Systems Technology*, 16(5):1029–1038, September 2008.

[57] J. Ru, X. R. Li, and V. P. Jilkov. Multiple-Model Detection of Target Maneuvers. In *Proceedings of the SPIE*, volume 5913, pages 100–108, September 2005.

[58] J.-F. Ru and X. R. Li. Interacting Multiple-Model Algorithm with Maximum Likelihood Estimation for FDI. In *Proc. 2003 IEEE International Symp. Intelligent Control*, pages 661–666, Houston, TX, Oct. 2003.

[59] A. N. Shiryaev. The Problem of the Most Rapid Detection of a Disturbance in a Stationary Process. *Soviet Math. Dokl*, (2):795–799, 1961.

[60] H. W. Sorenson and D. L. Alspach. Recursive Bayesian Estimation Using Gaussian Sums. *Automatica*, 7:465–479, 1971.

[61] D. M. Titterington. *Statistical Analysis of Finite Mixture Distributions.* John Wiley & Sons, 1986.

[62] S. M. Verbout, J. M. Ooi, J. T. Ludwig, and A. V. Oppenheim. Parameter Estimation for Autoregressive Gaussian-Mixture Processes: The EMAX Algorithm. *IEEE Transaction on Signal Processing*, 46(10):2744–2756, 1998.

[63] D. F. Votaw Jr., J. A. Rafferty, and W. L. Deemer. Estimation Of Parameters in a Truncated Trivariate Normal Distribution. *Psychometrika*, 15(4):339–347, 1950.

[64] M. Wetter and J. Wright. Comparison of a Generalized Pattern Search and a Genetic Algorithm Optimization Method. In *Proc. of the 8th International IBPSA Conference*, volume 3, pages 1401–1408, Eindhoven, Netherlands, August 2003.

[65] M. Yoshimura and N. Ito. Effective Diagnosis Methods for Air-Conditioning Equipment Intelecommunications Buildings. In *Proc. 11th International Telecommunications Energy Conference*, pages 1–21, Florence, Italy, October 1989.

[66] K. Zhang, X. R. Li, and Y. Zhu. Optimal Update With Out-Of-Sequence Measurements. *IEEE Transactions on Signal Processing*, 53(6):1992–2004, June 2005.

[67] K.-S. Zhang and X. R. Li. Optimal Sensor Data Quantization for Best Linear Unbiased Estimation Fusion. *IEEE Conference on Decision and Control*, pages 2656–2661, Dec 2004.

[68] Y. M. Zhang and X. R. Li. Detection and Diagnosis of Sensor and Actuator Failures Using IMM Estimator. *IEEE Transactions on Aerospace and. Electronic Systems*, AES-34(4):1293–1312, Oct. 1998.

[69] Z.-L. Zhao, X. R. Li, and V. P. Jilkov. Best Linear Unbiased Filtering With Nonlinear Measurements for Target Tracking. *Proc. 2003 SPIE Conf Signal and Data Processing of Small Targets*, August 2003.

[70] X. H. Zhuang, Y. Huang, K. Palaniappan, and Y. X. Zhao. Gaussian Mixture Density Modeling, Decomposition, and Applications. *IEEE Transactions on Image Processing*, 5(9):1293–1302, September 1996.

Vita

Anwer Bashi was born in Basra, Iraq, in 1973. In 1991, he came to study electrical engineering at the University of New Orleans, graduating with a BSEE in 1996, and an MSEE in 1997 with a focus on Controls and Estimation. While at college, Anwer participated in student life in a number of different organizations, including IEEE, the Dean's Student Organizational Council, the UNO Technology Committee, student government, and the Society for Advanced Graduate Engineering Studies.

His research for his Ph.D. focused on distributed computing, hybrid systems, controls and estimation, and fault detection and diagnosis.

Anwer has authored or coauthored 15 conference papers, journal articles, or book chapters.