Copyright
by
Katherine Patricia Carpenter
2010

The Thesis committee for Katherine Patricia Carpenter certifies that this is the approved version of the following Thesis:

Fault Detection and Precedent-free Localization in Thermal-fluid Systems

APPROVED BY

SUPERVISING COMMITTEE:

Alexandre K. da Silva, Supervisor

Dragan Djurdjanovic, Supervisor

Fault Detection and Precedent-free Localization in Thermal-fluid Systems

by

Katherine Patricia Carpenter, B.S.

THESIS

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE IN ENGINEERING

THE UNIVERSITY OF TEXAS AT AUSTIN
December 2010

Acknowledgments

I would like to sincerely thank my advisors, Alexandre K. da Silva and Dragan Djurdjanovic, for their support and guidance throughout this project.

Thank you to all my friends who make life more interesting.

I would like to thank my parents, Jon and Denise, and my sisters and brother for the love and support they have always given me.

Fault Detection and Precedent-free Localization in Thermal-fluid Systems

Katherine Patricia Carpenter, M.S.E.
The University of Texas at Austin, 2010

Supervisors: Alexandre K. da Silva
Dragan Djurdjanovic

This thesis presents a method for fault detection and precedent-free isolation for two types of channel flow systems, which were modeled with the finite element method. Unlike previous fault detection methods, this method requires no a priori knowledge or training pertaining to any particular fault. The basis for anomaly detection was the model of normal behavior obtained using the recently introduced Growing Structure Multiple Model System (GSMMS). Anomalous behavior is then detected as statistically significant departures of the current modeling residuals away from the modeling residuals corresponding to the normal system behavior. Distributed anomaly detection facilitated by multiple anomaly detectors monitoring various parts of the thermal-fluid system enabled localization of anomalous partitions of the system without the need to train classifiers to recognize an underlying fault.

Table of Contents

Acknowledgments **iv**

Abstract **v**

List of Tables **viii**

List of Figures **ix**

Chapter 1. Introduction **1**

1.1 Motivation 1

1.2 Objective and Challenges 2

1.3 Organization of Thesis 4

Chapter 2. Background in Current Fault Detection Methods in Thermal-fluid Systems **5**

2.1 Classifications of Fault Detection and Isolation Methods 6

2.2 Validity of Modeling Thermal-fluid Systems with Neural Networks 9

2.3 Current Neural-network-based Fault Detection and Diagnosis Methods in Thermal-fluid Systems 13

2.4 Growing Structure Multiple Model System (GSMMS) Modeling Approach 17

Chapter 3. GSMMS-based Fault Detection and Localization **19**

3.1 Introduction of the Growing Structure Multiple Model System (GSMMS) Modeling Approach 19

3.2 Training the GSMMS 20

3.3 Analysis of the Residuals 24

3.4 Fault Isolation 26

Chapter 4. Numerical Models and Verification **30**
4.1 Channel Flow Numerical Formulation 31
4.2 Verification of Channel Flow 33
4.3 Verification of Channel Flow with an Obstacle 35

Chapter 5. Anomaly Detection and Localization in Channel Flow **39**
5.1 Description of Anomalies 39
5.2 Finding the Orders of the Model 40
5.3 Fault Detection in Simple Channel Flow with an Inlet and Outlet Temperature Sensor 42
5.4 Fault Detection in a Simple Channel Flow with Distributed Sensing 46
5.5 Fault Localization in a Simple Channel Flow using Distributed Anomaly Detectors 50
5.6 Fault Detection and Localization in a More Dynamic System . 54

Chapter 6. Anomaly Detection and Localization in Channel Flow with an Obstacle **61**
6.1 Description of Anomalies and Normal Behavior 61
6.2 Fault Detection and Localization Upstream of the Obstacle . . 62
6.3 Fault Detection and Localization Downstream of the Obstacle 66
6.4 Fault Detection and Localization Above the Obstacle 70

Chapter 7. Conclusions and future work **72**
7.1 Conclusions 72
7.2 Future work 76

Bibliography **77**

Vita **84**

List of Tables

5.1 The $\tilde{x}$ positions of the inputs tested. For all inputs, $\tilde{y} = 0.05$. 47

List of Figures

3.1 The regional confidence value (CV) is the normalized area of the overlap of the PDFs. 26

3.2 Distribution of anomaly detectors in a diesel engine EGR system. Once an anomaly is detected, the overall AD (depicted in (a)) splits into five ADs monitoring pertinent subsystems (depicted in (b)). Taken from [7]. 28

3.3 CVs output by subsystem-level ADs illustrated in Fig. 3.2(b). Plot (a) illustrates CVs output when at $t = 1350s$, faults were simulated into the EGR valve. Plot (b) shows the CVs output when at $t = 1350s$, when faults were simulated into the PI controller. Since the only CV to drop was the AD associated with the EGR valve and PI controller, respectively, the culprit subsystem can be identified. Taken from [7]. 29

4.1 System model of 2-D channel flow. 31

4.2 Channel flow with an obstacle. 36

5.1 Locations of Anomalies 1, 2 and 3. 40

5.2 Example of how the temperature changed with time (both nondimensionalized). 43

5.3 Anomaly 3, input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and output ($\tilde{x} = 1$, $\tilde{y} = 0.05$) positions. 44

5.4 CV plot for Anomaly 3 with 1 input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and 1 output ($\tilde{x} = 1$, $\tilde{y} = 0.05$). 45

5.5 CV plots for Anomalies 1 and 2 with 1 input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and 1 output ($\tilde{x} = 1$, $\tilde{y} = 0.05$). 46

5.6 The positions of inputs. The positions of the three anomalies are also shown (note: for each simulation, only one anomaly was simulated). 48

5.7 Anomalies 1, 2 and 3 with multiple inputs where the output is at the outlet of the channel. Note that the CV scale is different for Anomaly 3. 49

5.8 Anomalies 1, 2 and 3 where different output positions were used for each anomaly. 52

5.9 CV plots when both Anomaly 1 and 2 occur simultaneously. Notice that at Position 5, the overall CV drop is greater than that at Position 4, indicating a second anomaly is occurring. . 54

5.10 Example of changing temperature (non-dimensionalized) and Reynolds number. 56

5.11 Anomalies 1, 2 and 3 for a more dynamic system 58

5.12 CV plots when an input directly upstream was used. Positions 4, 5, 6, 7 correspond to $\tilde{x} = 0.3, 0.4, 0.5, 0.6$ 59

6.1 Locations of Anomalies 1, 2 and 3. 62

6.2 Anomaly 1 is upstream of the obstacle. The position number in the CV plots represents the output position. 63

6.3 Velocity and temperature field upstream of the obstacle. . . . 65

6.4 Anomaly 2 is downstream of the obstacle. The position number indicated in the CV plots represents the output position. . . . 66

6.5 Velocity and temperature field downstream of the obstacle. . . 67

6.6 Example of the temperature gradient during normal behavior. To better show the gradient near the wall, a maximum temperature gradient of 3 was set, which explains the white space. The highest gradients occur along the top of the obstacle. . . . 69

6.7 Anomaly 3 is above the obstacle. The position number indicated in the CV plots represents the output position. 70

Chapter 1

Introduction

1.1 Motivation

Thermal-fluid systems (i.e., heat exchangers, fuel cells, etc.) are host to a variety of potential problems, such as fouling, overheating, leakage and general wear and deterioration. Such faults will not only negatively affect system performance but their effects develop slowly and may go unnoticed until abrupt failure occurs. Besides the difficulty of detecting the presence of a fault, localization of the source of the fault is also challenging because effects of a fault can propagate throughout the system.

In the last two decades, fault detection methods developed for thermal-fluid systems have used data-driven models, such as neural networks, to model dynamic behavior of the monitored systems and use residual comparisons to detect and isolate faults. The use of neural networks in thermal science has been increasing in the recent years due to their robustness in dealing with more complex phenomena compared with traditional, first-principle based methods, where differential equations approximating the system dynamics are solved by numerical techniques [1]. For example, fault detection methods were created for a steam turbine [2], furnace [3] and a solar water heater [4]. In each method, the system was modeled using a neural network, and their modeling residuals were analyzed and matched with residuals of a known fault. However, these fault detection methods were precedent-based methods, which means that a priori knowledge regarding the potential faults and their effects was needed for detection and isolation. Therefore, such methods are limited to whether faulty behavior data is readily available for training the neural network or if a priori knowledge of the fault characteristics exists. Clearly, these constraints limit the applicability of precedent-based methods because as system complexity increases, it becomes infeasible to anticipate all possible faults at all possible locations. Also, many existing fault detection methods are only capable of detecting a single fault occurring. Thus, a more sophisticated fault detection method is needed.

In this thesis, a precedent-free localization method based on distributed anomaly detection is presented and is applied to simple, dynamic thermal-fluid systems. The new approach requires only normal system behavior data to detect and localize the source of abnormal behavior.

1.2 Objective and Challenges

Previously, the Growing Structure Multiple Model System (GSMMS) based fault detection method, which was developed by [5], has been applied to lumped parameter systems, such as an electronically controlled throttle system [6], an exhaust gas recirculation (EGR) system [7] and a diesel power generator [8]. In the aforementioned systems, anomalies were detected and localized within the system using only normal behavior data. The GSMMS-based detection method uses a self-organizing map (SOM) [9] to model the system through a “divide and conquer” approach, where the operating space of the system is decomposed into smaller regions within which analytically tractable, dynamic models can be postulated. Simpler forms of dynamic models enable the use of simple anomaly detection approaches in each region. Anomalous behavior is detected when statistically significant departures away from the normal modeling residual patterns are identified in sufficiently many regions of the piecewise dynamic models.

The main goal of this thesis is to apply the GSMMS-based fault detection method and the ensuing precedent-free fault localization method based on the distributed anomaly detection paradigm to a thermally dynamic, transient channel flow system. Channel flow is a distributed system described by a set of partial differential equations, and the aforementioned anomaly detection and isolation approaches have never been applied to such systems. The objectives of this thesis are to observe and evaluate the capabilities and limitations of the fault detection method applied to a distributed system by investigating

- if the method can detect faults using only normal behavior data.
- if the method can localize the source of the faults using normal behavior data.
- how different system models composed of different inputs and outputs affect both detection and localization.

- how system complexity affects detection and localization sensitivity.

1.3 Organization of Thesis

The remainder of this thesis is as follows. Chapter 2 first gives a brief background on general fault detection methods, emphasizing the use of data-driven methods. Then, the use of neural networks for modeling thermal-fluid systems is described, and finally, background on the use of neural networks in thermal-fluid systems for fault detection and diagnosis is given. Chapter 3 describes the GSMMS-based anomaly detection and precedent-free localization. Chapter 4 describes the two numerical models used in this study and their verification. Chapter 5 presents the results of the fault detection and localization method when applied to simple channel flow, and Chapter 6 presents the results when the method is applied to channel flow with an obstacle. Chapter 7 presents the conclusions and suggests future work.

Chapter 2

Background in Current Fault Detection Methods in Thermal-fluid Systems

A fault is defined generally as a deviation from acceptable behavior of a system, whereas a failure refers to a permanent inability of the system to maintain its operating conditions [10, 11]. The goal of fault detection is to simply decide whether a fault exists or not within a system. Fault isolation follows fault detection whose goal is to determine the kind, location and time of detection of the fault within the system. Once the fault is isolated, fault identification can be accomplished by determining the dynamic model of the behavior in the presence of that fault. Together, fault isolation and identification, is referred to as fault diagnosis [10]. Fault detection and diagnosis (FDD) are vital in engineering systems to maintain normal operating conditions and prevent catastrophic failure.

Potential component faults can generally be categorized as hard faults or soft faults. Hard faults are caused by abrupt malfunctions or component damage, where the system stops functioning entirely. Soft faults encompass any degradation, wear and tear, fouling, leaking, clogging or any other gradually evolving fault that could occur in a system.

In this chapter, the different methodologies and strategies of FDD are first reviewed,with emphasis on the use of neural networks in FDD. Secondly, the validity of modeling of thermal-fluid systems using neural networks is discussed, which leads into the descriptions of the current FDD strategies that detect soft faults in thermal-fluid systems using neural networks. Finally, the Growing Structure Multiple Model System (GSMMS) approach to system modeling is introduced and its advantages over the currently used neural networks are discussed.

2.1 Classifications of Fault Detection and Isolation Methods

Generally, fault diagnosis methods can be classified into the following three categories: quantitative model-based methods, qualitative model-based methods and process history-based methods [11]. All these methods require a priori knowledge or system data but differ in terms of how the methods approach the problem of fault diagnosis. This section will briefly describe these three general approaches.

Quantitative model-based approaches monitor residuals of the measured variables and compare them with those calculated from a mathematical model. Residuals are the differences between the predicted output and the actual output of a system. If an abnormality occurs, system parameters will change, and this change will be reflected in the modeling residuals caused by the inconsistencies that arise between the predicted and actual output. Typically, quantitative models are general input-output or state-space models, as well as first-principles models and frequency response models [11]. There are a plethora of examples of quantitative-based methods in literature that encompass a variety of fields, including the automotive, aerospace and chemical industries; more survey papers can be found in [12–15]. Several examples of applications of quantitative-based methods to heat exchangers can be found in literature. Wakui and Yokoyama [16] developed an on-line, model-based method to monitor a shell-and-tube type heat exchanger where its accuracy depended on calculating correct performance correction factors and heat transfer coefficients. Shah et al. [17] created a dynamic, nonlinear model of the lumped state space form of a cross-flow plate-and-fin heat exchanger to monitor fouling by estimating state-dependent parameters. However, first-principles models cannot generally be used due to the difficulty in modeling complex behavior, such as transient phenomena and two-phase flow. Thermal phenomena often represent nonlinear systems, which severely limits the applicability of quantitative-based approaches to thermal-fluid systems. Also, many simplifications are often assumed in the models, such as constant properties or simplified geometries. Furthermore, potential faults must be known a priori to be included in the model. If the fault is not included in the model, the residuals may not recognize the fault.

Whereas quantitative model-based approaches express the models as mathematical functional relationships, qualitative-based approaches express the relationships in terms of qualitative functions, developed as either qualitative causal models or abstraction hierarchies [18]. Qualitative-based approaches incorporate cause and effect logic, where generally, if-then-else rules are used to create causal models, such as digraphs and signed digraphs, and a priori knowledge is used to predict likely system faults. Clearly, the limitations of these methods are that the designer must know all the causes and effects between the chosen system parameters and must know how the inputs affect the outputs. As a system becomes larger and more complex, this task becomes increasingly difficult and infeasible.

Differently from quantitative and qualitative model-based approaches, which both require specific knowledge about the process or system, history-based methods only require large amounts of historical process data [19]. The data is then transformed, known as feature extraction (methods differ in how feature extraction is performed), to create the diagnostic system. The obvious advantages of history-based methods are they require relatively little effort to implement and little a priori knowledge.

Thus, history-based approaches have largely been used to advance FDD for thermal-fluid systems. Because of the complexities involved in thermal-fluid systems, quantitative and qualitative model-based approaches become inherently inadequate as a system grows. In particular, neural networks are often used in history-based methods, which assumes thermal-fluid systems can be accurately be modeled by them. In the following section, the achievements of modeling thermal-systems using neural networks are overviewed.

2.2 Validity of Modeling Thermal-fluid Systems with Neural Networks

Neural networks are part of the computer-based algorithms known as soft-computing, which try to create simple models of human intelligence and evolution. Other soft-computing models methodologies include metaheuristic optimization algorithms (genetic algorithms), fuzzy-logic control, expert systems, data mining, etc. [1]. Traditional hard-computing methods, based on solving first-principle differential equations through numerical solutions, are not adequate or robust enough to deal with today's increasingly more complex thermal problems, which usually encompass transient systems, two-phase flow, etc., and systems that require optimization and control. Also, the phenomena may not be entirely understood and thus, cannot be modeled based on first principles. For these reasons, neural networks, known as data-driven models, have become popular as they can take advantage of the many process measurements that can be monitored continuously. Furthermore, data-driven approaches can model complex, dynamic phenomena without the common, simplified assumptions that many first principle-based models must use.

The strength of neural networks is its powerful ability to accurately recognize the inherent relationships within an input-output data set of a physical system despite complexities such as nonlinearity, multiple variables and parameters and noisy data [1]. Given sets of examples, neural networks essentially "learn from experience". From a data set consisting of inputs and outputs, the neural network maps the inputs onto the outputs. In thermal-fluid literature, the most popular neural network used is the multi-layer perceptron (MLP) neural network with the back-propagation learning algorithm [2–4, 20, 21]. An MLP neural network consists of an input layer, one or more hidden layers and an output layer. Each layer consists of neurons or nodes with weights associated with each node. The output signal, y, can then be computed as

$$y = F(\sum_{i=0}^{M} w_i x_i) \tag{2.1}$$

where $[x_1, x_2, \ldots, x_M]$ represents the input signals, $[w_1, w_2, \ldots, w_M]$ are the weights, and F is the activation or transfer function, where the hyperbolic tangent and sigmoid functions are typically employed.

By iterating over and over again through the input-output data, the neural network uses a learning algorithm to adjust the weights. The learning algorithm determines how and which weights are adjusted. With the back-propagation learning algorithm, the output is estimated from a given input, and the error between the predicted and the desired output is computed; this error is then back-propagated back through the neural network, and the weights are adjusted so as to decrease the error. Training consists of many iterations through the training data. Since the desired output is known and is used to adjust the weights, the learning is called supervised. After training, the neural network can be given testing data, different from the training data, and will be capable of predicting the correct outputs.

Note that neural networks themselves are a current research area. Various types of neural networks can be characterized based on such characteristics as their architecture, how they are trained and the direction of data flow [22]. However, the above description gives the general idea of how neural networks are programmed. Also, despite the MLP being one of the simplest neural networks, its use is widespread, particularly in the thermal-fluid literature.

The following examples illustrate the power and ease of neural networks when applied to different systems that would otherwise be difficult and time consuming to model with the governing equations.

Modeling transient or unsteady heat conduction requires a large amount of computational power to solve such problems through numerical methods. However, neural networks can quickly and accurately model unsteady systems. Jambunathan et al. [23] used a neural network to model 1D transient heat conduction. Heat transfer coefficients at different points were accurately predicted (average errors of up to 2.7%) in a duct that was being heated by the flow of hot air. Kuroe and Kimura [24] used a neural network to model 2D unsteady heat conduction and then predict temperatures throughout the field. Mittal and Zhang [25] developed a neural network to predict food freezing time with typical relative errors of less than 5% to ensure optimum food quality, which would otherwise require tedious calculations.

Heat exchangers, which are ubiquitous in many thermal-fluid systems, are difficult to model due to such complexities as moisture in the air condensing or freezing on the fins, which modify the flow field, partial refrigerant evaporation, which may cause an inhomogeneous flow distribution, turbulence, complex geometries, existence of hydrodynamic and thermal entrance regions, vortices and temperature-dependent fluid properties. Correlations for heat exchangers are often inadequate to address such issues. However, neural networks are able to model such phenomena without difficulty. For example, using the limited amount of data provided by the heat exchanger manufacturer, Pacheco-Vega et al. [26] used a neural network to model a multi-row, multi-column fin-plate type heat exchanger with staggered tubes where Freon 22 was used as the refrigerant. The neural network predicted the total heat rate with a root-mean-square error (percentage difference between the predictions and the experimental measurements) of less than 1.5%. Dynamic control of heat exchangers is almost impossible using predictions from first principles due to the difficulties stated earlier, and assumptions and simplifications regarding the model are usually made. Diaz et al. [27] used a neural network to model the time-dependent behavior of a heat exchanger to control the air temperature passing over it, where the neural network controller performed better than when using traditional PI and PID controllers.

If the phenomena of the system are not understood, creating a model from first principles is impossible. However, neural networks can model such systems using only the available process data. Liu et al. [28] predicted the boiling heat transfer enhancement due to additives, an enhancement mechanism that is not well understood and thus, precludes the use of a mathematical model. With parameters that describe the molecular characteristics of the additive as the inputs, a model with an accuracy of over 90% was obtained.

The previous examples show that neural networks are able to accurately capture complex phenomena without a mathematical model derived from the governing equations and are much less time-consuming than other methods.

The recent growth of neural networks in thermal-fluid systems opens up many possibilities and potential advancements. Obviously, in relation with FDD, the strengths of neural networks, such as, their ease and flexibility in modeling of complex, nonlinear behavior and ability to rapidly make predictions, make its use in detecting faults in thermal-fluid systems promising. The aforementioned examples justify modeling thermal-fluid systems with neural networks.

In the next section, current fault detection and diagnosis methods for various thermal-fluid systems that use neural networks are discussed, and their strengths and limitations are highlighted.

2.3 Current Neural-network-based Fault Detection and Diagnosis Methods in Thermal-fluid Systems

With condition-based maintenance, the natural degradation of equipment is monitored in real time, whereas with periodic maintenance, equipment is checked at regular intervals regardless of their state. Condition-based maintenance is therefore more cost-effective compared with periodic maintenance because in condition-based maintenance, resources are only used to check the equipment if a fault occurs. Neural networks can quickly predict system behavior, which is required for on-line monitoring in condition-based maintenance and the reason why many FDD systems created for thermal-fluid systems use neural networks. However, different techniques exist. The following examples illustrate these differences.

The objective of many studies is to diagnose faults, i.e., to detect and recognize known faults, by training a neural network from theoretical relationships or an assumed mathematical model. Riveral and Napolitano [29] reduced the impact of fouling in a plate heat exchanger by predicting the deposit thickness, the overall heat transfer coefficient and the critical time (time before a system must be stopped before cleaning) in a pasteurization process using heat flux, pressure and temperature sensors. Their model was based on a set of heat balances and a fouling model. Tian and Sun [30] proposed an FDD method to detect chemical pipeline leakage by monitoring the fluid flow process. A neural network was trained according to a pipeline leakage model that calculated flow rates based on geometric characteristics, pressure, fluid properties and friction factors. A leak was then detected by calculating the difference between the simulated and measured value and comparing the resulting residual with a threshold value. These methods are limited by the accuracy of the model and assumptions. With more complex phenomena, assumptions, such as constant properties, cannot be used and existing correlations for friction factors or heat transfer coefficients will not be available.

In many FDD methods, the neural network is specifically trained to recognize different fault type patterns, which again focuses on diagnosing faults, not detecting abnormal behavior. Karlsson et al. [2] investigated seven types of faults common in steam turbines where the neural network was trained to memorize the fault patterns. Normal and faulty data were generated using a steam cycle simulation. Calisto et al. [3] used computational fluid dynamics (CFD) to model a ”virtual” furnace, and a neural network was trained to detect and identify fouling and steam leaks. Sorsa et al. [20] studied FDD for a heat exchanger-continuous stirred tank reactor system and were able to detect and classify ten different faults using a neural network, which were simulated. Though each study was able to detect and identify their respective faults, any fault unanticipated by the designer would either go unnoticed or be incorrectly identified. With any new fault, the neural network would have to be trained again. Also, Karlsson et al. and Calisto et al. both highlighted that multiple faults occurring simultaneously would not be correctly identified.

Not all neural network-based FDD methods require faulty data. Many methods focus on detecting abnormal behavior by using a combination of residuals to detect and isolate different faults. Kalogirou et al. [4] developed an automatic solar water heater fault diagnosis system by training multiple neural networks to predict different temperatures of a fault-free system. A residual calculator was also developed that compares both the current measurement data and the fault-free predictions, where the residuals are compared against three constant threshold values. The magnitude of the residuals categorized the current state as either being normal, low probability, high probability or failure. Because false alarms are prone to occur due to noisy measurements, a fault was detected when five consecutive failure states are shown. Cui and

Wang [31] created a model-based FDD strategy for a steady-state centrifugal chiller system based on six physical performance indexes, which were selected so that different performance indexes were sensitive to different faults based on basic thermo-physical principles. A neural network was trained using fault-free data, and a set of rules for five different faults and how they impacted six performance indices classified the fault. Again, residual thresholds were used to detect faults. Setting adequate thresholds is key because setting thresholds too low results in false alarms, whereas setting thresholds too high results in missed detections. Also, to isolate faults, the designer must know how different process measurements are affected by faults so that different residuals are insensitive to some faults and not to others.

It is important here to emphasize the subtle difference between fault diagnosis and fault detection approaches. Diagnosis aims to recognize and classify particular faults, typically by matching residual patterns, whereas detection recognizes abnormal behavior in the system, typically be analyzing the residual patterns.

Rather than using only the residual value to detect faults, statistical tests on the residuals can be performed. Chetouani [32] used a nonlinear autoregressive with eXogenous input (NARX) model, obtained through the use of a neural network, and the well-known CUSUM test [1] (cumulative sums test) to detect faults in a reactor-exchanger. According to the CUSUM test, any

---

[1] The CUSUM test is a well-known, powerful tool that detects changes in process dynamics by estimating the mean of the residual signal [33]

jump in the mean indicates a fault; however, the minimum jump magnitude must be determined a priori [32]. Lalot and Palsson [21] detected fouling in a cross-flow heat exchanger also using a neural network and the CUSUM test, where the data was generated using the finite volume method. They found that analysis of just the residuals would not have been as sensitive as their developed technique.

2.4 Growing Structure Multiple Model System (GSMMS) Modeling Approach

Despite the success that many researchers have achieved using MLP neural networks, one drawback is that the structure of the neural network must be determined. Many tests must be performed to find the best combination of the number of layers and the number of nodes in each layer. Alternatively, the Kohonen's Self-Organizing Map (SOM) [9], a vector quantization technique, uses a "divide and conquer" approach to modeling, where the operating space is decomposed into smaller sub-regions, as opposed to the "global" models that are constructed using MLP neural networks. Differently from MLP neural networks, SOMs use an unsupervised learning algorithm, where the goal is to determine how the data are clustered together, i.e., to recognize the underlying organization and structure of the data. The weight vectors of an SOM define a Voronoi Tessellation:

$$V_m = \{x : \|x - \xi_m\| \ \leq \|x - \xi_j\|, \qquad \forall m \neq j\} \tag{2.2}$$

In this way, the operating space is divided into sub-regions, V_m, of "similar" input-output patterns. However, SOMs still require that the number of nodes and the number of topological connections be determined.

To decrease the number of decisions and assumptions that must be determined when using SOMs, addition and deletion mechanisms can be incorporated. These attributes are used in growing SOMs [34], which led to the development of the Growing Structure Multiple Model System (GSMMS) approach to dynamic system modeling [5]. With the GSMMS approach, the operating space can be refined by inserting a new node near a poorly modeled region; thus, the structure is allowed to "grow", which allows the SOM to determine its own appropriate size.

With the GSMMS modeling approach, less assumptions and decisions on the size and structure of the neural network are needed compared with the MLP neural network. Furthermore, with the "divide and conquer" approach, residual analysis and interpretation can be performed on the simpler, regional residuals, which eliminates the need to deal with non-stationary residual analysis, enabling one to determine the anomaly detection thresholds in a tractable, rigorous manner. The next chapter explains the fault detection methodology based on GSMMS modeling and the residual analysis. Together, GSMMS modeling and the residual analysis can detect and localize faults using only normal behavior.

Chapter 3

GSMMS-based Fault Detection and Localization

This chapter will first explain the recently introduced Growing Structure Multiple Model System (GSMMS), followed by a description of how the residuals are analyzed and interpreted so that abnormal behavior can be detected. Later, a description will be given of how distributed GSMMS-based anomaly detectors can be used to localize sources of abnormal behavior precedent-free (without the need to *a priori* observe signatures of the fault that generated the anomaly).

3.1 Introduction of the Growing Structure Multiple Model System (GSMMS) Modeling Approach

The GSMMS uses a multiple model structure, where each local model domain is defined by the self-organizing map (SOM) [9] induced Voronoi tessellation of the state-space of the model [6]. Within each region, local models are assumed to be of the linear form

$$\mathbf{F}_m(s(k)) = \mathbf{a}_i^T \mathbf{s}(k) + \mathbf{b}_i \tag{3.1}$$

where $\mathbf{a}_i$ and $\mathbf{b}_i$ denote the vectors of parameters of the local model i and

$$\mathbf{s}(k) = [\mathbf{y}^T(k), \dots \mathbf{y}^T(k-n_a+1), \mathbf{u}^T(k-n_d), \dots, \mathbf{u}^T(k-n_d-n_b+1)]^T \tag{3.2}$$

is the state vector, where $\mathbf{y}(k) = [y_1(k), \dots, y_p(k)]^T$ is a vector of p outputs and $\mathbf{u}(k) = [u_1(k), \dots, u_m(k)]^T$ is a vector of m inputs of the system, n_d is the time delay between when the input reaches the output, and n_a and n_b are respectively the autoregressive and external input orders of the local model. Following Johansen and Foss [35], the global model is then defined as

$$\hat{y}(k+1) = \sum_{m=1}^{M} \nu(\mathbf{s}(k))\mathbf{F}_m(\mathbf{s}(k)) \tag{3.3}$$

where $\nu(\mathbf{s}(k))$ describes how the local models are interpolated into the global model. Following Liu [5], $\nu(\mathbf{s}(k))$ is a simple gating function

$$\nu_m(\mathbf{s}(k)) = \begin{cases} 1 & \mathbf{s}(k) \in V_m \\ 0 & \mathbf{s}(k) \text{ otherwise} \end{cases} \tag{3.4}$$

that says that each local model $\mathbf{F}_m(\mathbf{s}(k))$ is only valid in region m.

3.2 Training the GSMMS

The training process for the GSMMS yields the following:

1. The structural parameters or weight vectors of the model. The weight vectors partition the operation space into regions of similar input-output

patterns and are obtained through unsupervised clustering of input/output vectors, $\mathbf{s}(k)$ (state-vectors), in the training set.

2. The local model parameters for each sub-region.

Essentially, training consists of successive passes through the training data (normal behavior data), which updates and adjusts the SOM weight vectors (and thus the resulting state-space partition induced by the corresponding Voronoi tessellation, as well as the local model parameters within each Voronoi region). Unsupervised clustering of the SOM weight vectors, $\xi_m, m \in 1, 2, \ldots, M$, is obtained via recursive adjustments.

$$\xi_m(k+1) = \xi_m(k) + \zeta_m(k) h(k, \mathrm{dis}(m, b(k)))[\bar{\mathbf{s}}_b - \xi_m(k)] \tag{3.5}$$

where k is the index of the training item $\mathbf{s}(k)$, and $\bar{\mathbf{s}}_b$ is the sample mean of the training vectors for which b is the Best Matching Unit (BMU), For each training sample, a BMU, $b(k)$, which is the index of the local model or the weight vector who best matches the training sample, is defined as

$$b(k) = \arg\min_m \|\mathbf{s}(k) - \xi_m\| \tag{3.6}$$

The function, $h(k, dis(m, b(k)))$, is the neighborhood function, which describes how each vector is updated using training samples in neighboring regions and defined as

$$h(k, \mathrm{dis}(m,k))) = exp\Big(\frac{-dis(m, b(k))^2}{2\sigma^2(k)}\Big) \tag{3.7}$$

The neighborhood function shrinks with increasing distance away from the BMU and with increasing passes through the data.

The width parameter, $\sigma^2(k)$, defines the effective range of the weighting function and decreases to zero as $k \to \infty$ to achieve convergence and global ordering of the SOM [9]. In other words, at the beginning of the training, each training sample has an effect on the parameters of a wide area of local models, but as $k \to \infty$, the affected area of local models narrows because the further away a given region m is from the BMU, the less significant the effects are of the current observation on the parameter estimates of the model in the region m. Following [7], in this thesis, we adopt $\sigma(k) = 1/k$. Finally, the term, $dis(m, b(k))$, is the shortest distance between the node m and the BMU and is found using the Breadth-first procedure [36].

The penalty term, $\zeta_m(k)$ in Eq. 3.5 helps achieve a more accurate model by balancing the effects of visiting frequencies [1] and modeling errors across different regions. If a region is not frequently visited, the region could be poorly approximated and in need of more local models. Furthermore, if modeling errors in a given region are high, that region may need to be refined and additional SOM models should move toward it. The weight vector updating Eq. 3.5 already ensures that the state space is partitioned according to

---

[1] Visiting frequency for a given region refers to the number of times the training data samples are associated with that particular region.

visitation frequencies (more nodes in more frequently visited regions). Thus, the penalty term is described by

$$\zeta_m(k) = \frac{e_m(k)}{\sum\limits_{m=1}^{M} e_m(k)} \tag{3.8}$$

where $e_m(k)$ is the root mean squared (RMS) modeling errors in the m^{th} region. As a result of the penalty term, weight vectors will tend to move towards regions with higher modeling errors, resulting in a finer partition in those areas.

The local model parameters in region m are determined by minimizing the sum of the weighted squared output errors in each region using

$$J_m(\theta_m) = \frac{1}{k} \sum_{i=1}^{k} w_m(\mathbf{s}(i)) \|y(i) - \hat{y}_m(i)\|^m \tag{3.9}$$

where θ_m denotes the model parameters for the m^{th} region, $y(i)$ is the training output at time i, and $\hat{y}_m(i)$ is the predicted output of model m at time i. The weighting function, $w_m(\mathbf{s}(i))$, determines the effect of sample i on the neighboring regional models estimates, as defined as

$$w_m(\mathbf{s}(i)) = \exp\Big(\frac{-\mathrm{dis}(m, b(k))^2}{2\sigma(k)^2}\Big), \qquad m = 1, 2, \ldots, M \tag{3.10}$$

In this way, each training sample affects all the local models, with this effect reducing as the distance from the model corresponding to the BMU grows.

The self-organizing map is allowed to “grow” after a predetermined number of passes through the training data by inserting a new node in a poorly modeled region. This allows the underlying model structure to grow and adapt to the data. Training ends when one of the two stopping criteria was met, 1) the total RMS error was below a pre-determined tolerance or 2) the number of nodes exceeded a pre-determined number.

3.3 Analysis of the Residuals

Once the training is finished, the statistical characteristics of the modeling residuals during normal behavior are known, where the residuals are defined as the differences between the actual system output and the GSMMS predicted output. If any anomaly enters the system, i.e., if the system dynamics changes in any way, the modeling residuals behavior changes as well. Thus, to detect an anomaly, one can compare the characteristics of the training residuals with that of the current residuals, and abnormal behavior can be indicated when differences are detected. However, interpretation of the residuals is not a trivial task.

The operating regions within the GSMMS all have different levels of approximation accuracy. If the system inputs change and drives the system to different a operation region, the modeling residuals will consequently change. Thus, the modeling residuals can change, not only because of an anomaly. The “divide and conquer” framework of the GSMMS models is able to work around these potential false alarms because residual interpretation can be done based on the simpler, regional residuals.

Following [5], system deviation is quantified within each region m by using the concept of regional confidence values (CVs), defined as

$$\mathrm{CV}(m,k) = \frac{|f_m(e) \cdot g_m(e,k)|}{\|f_m(e)\| \|g_m(e,k)\|} \tag{3.11}$$

where $f_m(e)$ is the probability density function (PDF) of the modeling residuals displayed during normal behavior and $g_m(e,k)$ is the PDF of the residuals corresponding to the current behavior at time k, $|\cdot|$ denotes the inner product,

$$|f_m(e) \cdot g_m(e,k)| = \int_{-\infty}^{\infty} f_m(e) g_m(e,k) de \tag{3.12}$$

and $\|\cdot\|$ is the L_2 norm,

$$\|f_m(e)\| = \sqrt{\int_{-\infty}^{\infty} [f_m(e)]^2 de} \tag{3.13}$$

The regional confidence value, $CV(m,k)$, describes the normalized area of the overlap of the PDFs in that region, as shown in Fig. 3.1. If $CV(m,k) = 1$, the current residual PDF matches with the residual PDF obtained during training, which indicates normal behavior. The PDF $f_m(e)$ was approximated using Gaussian Mixture Models due to their universal approximation capability [37], and $g_m(e,k)$ was calculated by updating $f_m(e)$ recursively during operation [38]. A single, global CV is created as the geometric mean of the regional CVs,

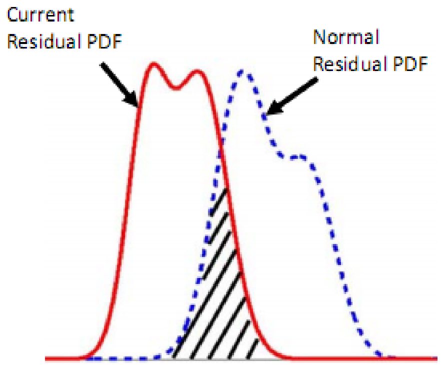


Figure 3.1: The regional confidence value (CV) is the normalized area of the overlap of the PDFs.

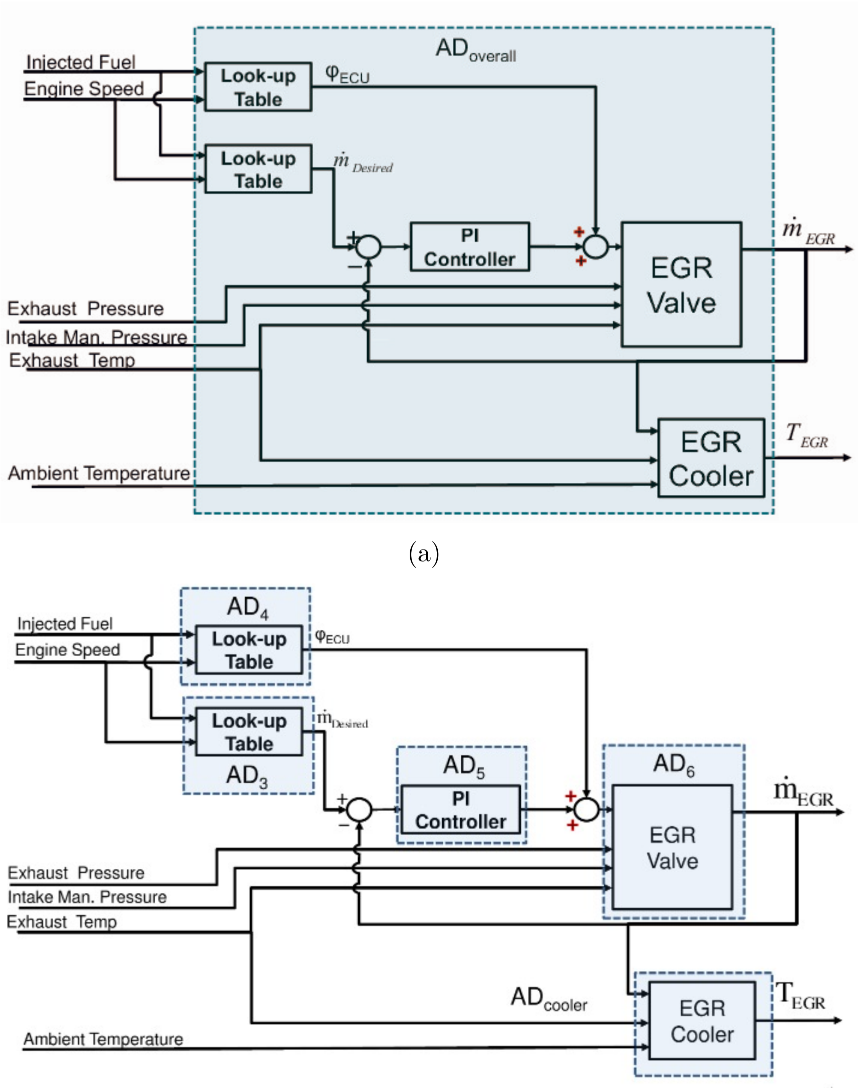
$$\mathrm{CV}_{\mathrm{global}}(k) = \Big(\prod_{m=1}^{M} \mathrm{CV}(m,k)\Big)^{1/M} \tag{3.14}$$

in order to emphasize departures of individual regional CVs away from one [7].

3.4 Fault Isolation

Using only normal behavior data, fault isolation can be achieved using the paradigm of distributed anomaly detection, where a set of anomaly detectors (ADs) are used to monitor pertinent sub-systems. The following example (taken from [7]) shows how precedent-free localization is achieved. Initially, as shown in Fig. 3.2(a), an overall anomaly detector monitors an entire EGR system based on the GSMMS model of its normal behavior. If an anomaly is detected, five anomaly detectors are distributed, as shown in Fig. 3.2(b), where each use the GSMMS model of normal behavior specific to its subsystem. Figure 3.3(a) shows the CV output from the five ADs when a fault was occurring in the EGR valve, and Fig. 3.3(b) shows the CV output from the five ADs when a fault was occurring in the PI controller. Clearly, since only one AD for each fault dropped, the culprit subsystem can be identified. The other four sub-systems each show that normal behavior was occurring since the CVs were essentially one, i.e., the GSMMS modeling residuals observed during normal behavior and the most recently observed modeling residuals matched. Thus, precedent-free localization can be achieved through distributed anomaly detection.

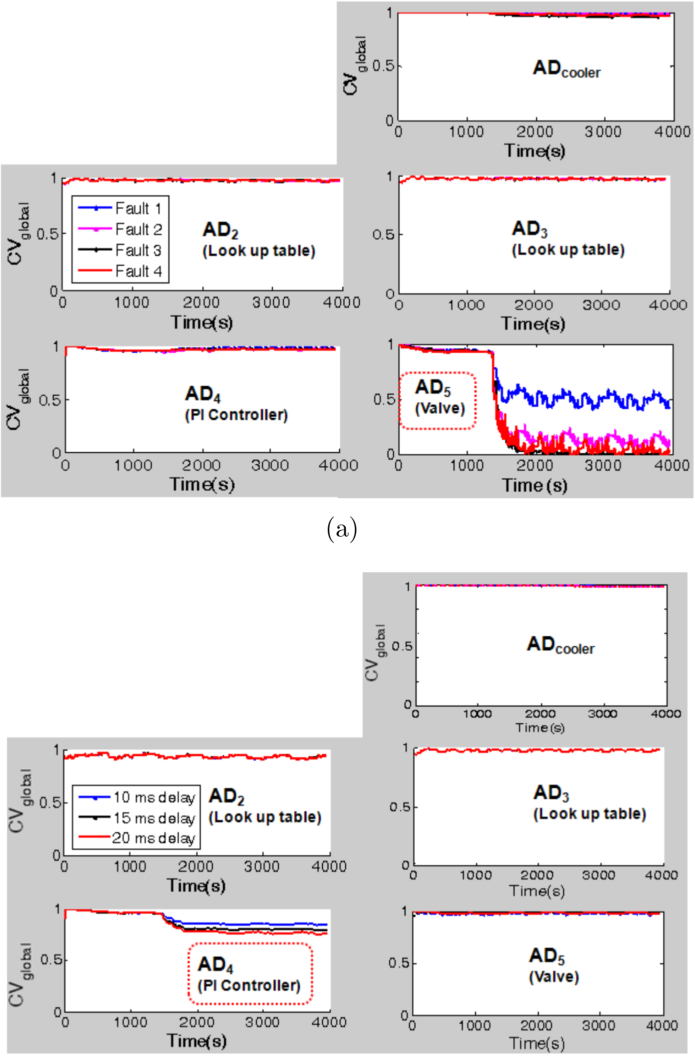
In this thesis, we will use a simpler distributed anomaly detection concept to localize sources of abnormal behavior to different positions of a thermal-fluid system.



(a)

(b)

Figure 3.2: Distribution of anomaly detectors in a diesel engine EGR system. Once an anomaly is detected, the overall AD (depicted in (a)) splits into five ADs monitoring pertinent subsystems (depicted in (b)). Taken from [7].



(a)

(b)

Figure 3.3: CVs output by subsystem-level ADs illustrated in Fig. 3.2(b). Plot (a) illustrates CVs output when at $t = 1350s$, faults were simulated into the EGR valve. Plot (b) shows the CVs output when at $t = 1350s$, when faults were simulated into the PI controller. Since the only CV to drop was the AD associated with the EGR valve and PI controller, respectively, the culprit subsystem can be identified. Taken from [7].

Chapter 4

Numerical Models and Verification

All training, testing and anomalous data for GSMMS-based anomaly detectors were created by the finite element method using the commercial software package, COMSOL Multiphysics [39]. Rather than using data obtained from an experimental apparatus, data from simulations could be obtained relatively quickly; thus, the focus could be on the performance and sensitivity of the fault detection and localization method. As can be seen from the previous chapter, the data-driven GSMMS approach to modeling and anomaly detection requires only input-output data to model the system, whose residuals are then used in the fault detection and localization.

Two systems were modeled. For each model, the geometry and non-dimenionalized variables were based from two separate studies [40, 41]. First, a simple channel flow was chosen as the inaugural system to verify that the GSMMS could model a thermal-fluid system, to confirm that faults could be detected and localized in a simple setting and to observe how the positions of the inputs and outputs affect the sensitivity and performance of the method.

Secondly, a channel flow with an obstacle was considered. Such a system has a more complex hydrodynamic behavior than a simple channel flow.

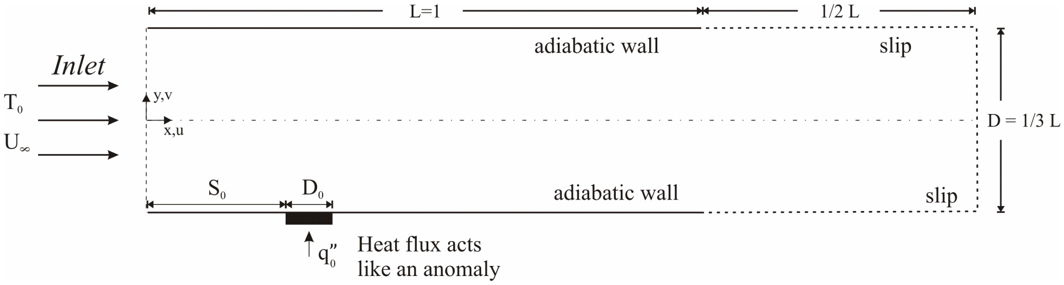


Figure 4.1: System model of 2-D channel flow.

The purpose of tests with this model was to observe if the same detection and localization approach used in the simple channel flow could be applied to a more complex system.

This chapter will present the numerical models and their verification.

4.1 Channel Flow Numerical Formulation

The first system was a transient, 2-D channel where laminar flow and constant properties were assumed. Normal behavior was channel flow through adiabatic walls with an inlet temperature that changed at random intervals. Anomalous behavior was created by introducing a linearly increasing heat flux through a small portion of the wall. By continuously changing the inlet temperature, detecting a heat flux (the anomaly) through the wall becomes more challenging than if the inlet temperature did not change, which would have been trivial. Figure 4.1 illustrates the system model.

The geometry and variables were taken from [40]. The channel was of length L and width D, with fluid entering from the left with a free-stream velocity of U_∞. The width of the heater was D_o at a length of S_o from the entrance. An extension was placed downstream to simulate more realistic outlet conditions. There was no slip at the walls and free slip at the fluid-fluid interface.

The simplified mass, momentum and energy equations used are the following:

$$\nabla \cdot \mathbf{v} = 0 \tag{4.1}$$

$$\rho \frac{D\mathbf{v}}{Dt} = \text{-}\nabla \mathbf{P} + \mu \nabla^2 \mathbf{v} \tag{4.2}$$

$$\rho c_p \frac{DT}{Dt} = k \nabla^2 T \tag{4.3}$$

For convenience, the equations were non-dimensionlized with the following dimensionless variables:

$$(\tilde{x}, \tilde{y}, \tilde{D_o}, \tilde{S_0}) = \frac{x, y, D_o, S_o}{L}, \qquad (\tilde{u}, \tilde{v}) = \frac{(u, v)}{U_\infty} \tag{4.4}$$

$$\tilde{T} = \frac{T - T_\infty}{q_o'' L/k}, \qquad \tilde{P} = \frac{P}{\rho U_\infty^2}, \qquad \tilde{t} = \frac{U_\infty}{L} t \tag{4.5}$$

Equations 4.1, 4.2 and 4.3 then become

$$\nabla \cdot \tilde{\mathbf{v}} = 0 \tag{4.6}$$

$$\frac{D\tilde{\mathbf{v}}}{D\tilde{t}} = -\nabla\tilde{\mathbf{P}} + \frac{1}{Re}\nabla^2\tilde{\mathbf{v}} \tag{4.7}$$

$$RePr\frac{D\tilde{T}}{D\tilde{t}} = \nabla^2\tilde{T} \tag{4.8}$$

where

$$Re = \frac{U_\infty D}{\nu}, \qquad Pr = \frac{\nu}{\alpha} \tag{4.9}$$

The outlet conditions are

$$\frac{\partial\tilde{u}}{\partial\tilde{x}} = \frac{\partial\tilde{v}}{\partial\tilde{x}} = \frac{\partial\tilde{T}}{\partial\tilde{x}} = 0 \tag{4.10}$$

and the boundary conditions at the fluid-fluid interface are

$$\frac{\partial\tilde{u}}{\partial\tilde{x}} = \frac{\partial\tilde{T}}{\partial\tilde{x}} = 0 \tag{4.11}$$

where $D()/Dt = \partial()/\partial t + \mathbf{v} \cdot \nabla()$, $\nabla = \partial/\partial x + \partial/\partial y$.

4.2 Verification of Channel Flow

The simulations were performed in COMSOL [39], where the Fluid-Thermal Interaction mode with the Direct (PARDISO) linear solver was used.

The time dependent simulations used the Backwards Differentiation formula with a time step of 0.01. For all cases, the relative error was set as $1 \cdot 10^{-5}$ and the absolute error was set as $1 \cdot 10^{-6}$.

Mesh verification was performed in three phases. First, to verify the accuracy of the mesh, global conductance values were calculated, where C is defined as

$$C = \frac{Q'}{k(T_{max} - T_{\infty})} \tag{4.12}$$

where C is the global conductance, Q' is the total heat flow through the heat source, and T_{max} is the maximum temperature that may occur at any point on the wall. The conditions were $Re = 10^3$, $Pr = 0.7$, $\tilde{D} = 0.3$, $\tilde{D}_o = 0.1$, and conductance values were calculated with heat fluxes positioned at $S_o = 0.1, 0.3, 0.5, 0.7, 0.9$ at the steady-state condition. A mesh of 11,006 elements was simulated, and the conductance values were evaluated. Then, the mesh was refined further to 25,253 elements, and the results changed less than 1%. To ensure the accuracy of the mesh, results were compared with [40], which studied heaters in channel flow. The only exception between the two geometries is that in [40], a front extension was also used. The mesh of 11,006 elements agreed within 8% to the comparison values.

Second, the transient model was verified by first letting the model run until it reached “steady-state” and comparing the same conductance values for the mesh and the refined mesh. The results changed less than 1% between mesh refinements and agreed within 8% to the comparison values. To ensure that the mesh could accurately model a changing inlet temperature, a sinusoidal temperature oscillating with a small amplitude was used. Again, the results changed less than 1% between mesh refinements and agreed within 8% of the comparison values.

Lastly, temperature measurements at various locations within the system (five points spaced evenly along the centerline at $\tilde{y} = 0.5$ and five points spaced evenly at $\tilde{y} = 0.05$) between refinements when the model was simulated for two time units with an inlet temperature of $\tilde{T} = 2$ and the system temperature was initially set at $\tilde{T} = 1$. There was less than 1% difference with the same mesh refinements as before.

Thus, a model with 11,006 elements was chosen for the channel flow study.

4.3 Verification of Channel Flow with an Obstacle

The geometry and variables of the channel flow with an obstacle was based on the study by [41]. Figure 4.2 illustrates the model. Again, laminar, incompressible fluid flow is assumed through the 2-D channel with adiabatic walls. In this study, $H = 1$, $L_e = 2$, $L_o = 8$ and $w = h = 0.25$.

The governing equations (Eqns. 4.1, 4.2 and 4.3) were non-dimensionlized with the following dimensionless variables:

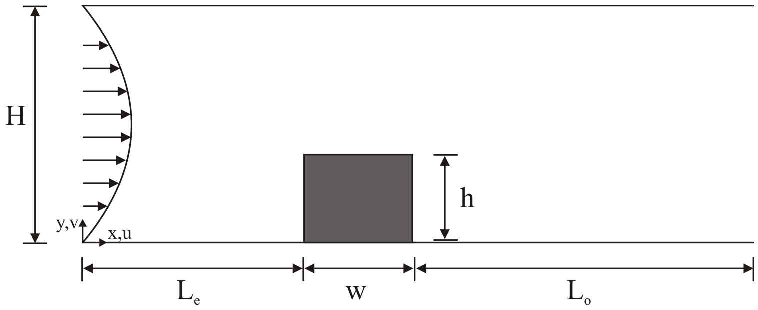


Figure 4.2: Channel flow with an obstacle.

$$(\tilde{x}, \tilde{y}, \tilde{h}, \tilde{w}, \tilde{S}_o) = \frac{x, y, h, w, S_o}{H}, \qquad (\tilde{u}, \tilde{v}) = \frac{(u, v)}{U_m} \tag{4.13}$$

$$\tilde{T} = \frac{T - T_\infty}{q''_o H/k_f}, \qquad \tilde{P} = \frac{PH}{\mu_f U_m}, \qquad \tilde{t} = \frac{U_m}{H} t \tag{4.14}$$

The non-dimensionlized governing equations then become

$$\nabla \cdot \tilde{\mathbf{v}} = 0 \tag{4.15}$$

$$\frac{D\tilde{\mathbf{v}}}{D\tilde{t}} = \text{-}\nabla\tilde{\mathbf{P}} + \frac{1}{Re}\nabla^2\tilde{\mathbf{v}} \tag{4.16}$$

$$RePr\frac{D\tilde{T}}{D\tilde{t}} = \nabla^2\tilde{T} \tag{4.17}$$

where

$$Re = \frac{U_m H}{\nu_f}, \qquad Pr = \frac{\nu_f}{\alpha_f} \tag{4.18}$$

The subscript 'f' refers to the fluid properties.

The mean velocity is given by $u_m = \frac{1}{H}\int_0^H u(y)dy$. In this study, fluid enters the channel at a specified inlet temperature with a fully, developed parabolic profile given by

$$u = 6y(1-y), \qquad v = 0 \tag{4.19}$$

The outlet conditions were

$$\frac{\partial \tilde{u}}{\partial \tilde{x}} = \frac{\partial \tilde{v}}{\partial \tilde{x}} = \frac{\partial \tilde{T}}{\partial \tilde{x}} = 0 \tag{4.20}$$

The no-slip condition and adiabatic condition ($u = 0$, $v = 0, \frac{\partial T}{\partial y} = 0$ were used at the upper and lower channel walls. At the solid obstacle and fluid interface, the following conditions were used:

$$\tilde{u} = 0, \quad \tilde{v} = 0, \quad \tilde{T}_f = \tilde{T}_s, \quad k_f \frac{\partial \tilde{T}_f}{\partial \mathbf{n}} = k_s \frac{\partial \tilde{T}_s}{\partial \mathbf{n}} \tag{4.21}$$

where the subscript 'f' represents the fluid, 's' represents the solid and 'n' represents the direction normal to the surface.

To validate the mesh, the Nusselt number, defined as $Nu_x = \frac{hH}{k_f} =$ -$\frac{1}{\tilde{T}_w}\frac{\partial \tilde{T}_f}{\partial \mathbf{n}}$, was calculated along the obstacle wall and compared with the results from [41] for $Re = 200$ and $Pr = 0.72$ when $k_s/k_f = 10$. To compare the results, a heat flux, $q" = 1$, was prescribed into the base of the solid, and the temperature boundary condition at the entrance, $\tilde{T} = 0$. First, a steady-state comparison was conducted and a mesh of 227,475 elements was used. Then, to test the grid independence, a finer mesh (471,742 elements) was tested with a difference of less than 2% of the original values. Then the Nusselt numbers were compared with [41] and the results agreed within 10% of the values, except at locations extremely close to the two top corners (at distances of 0.002 from each corner) of the obstacle, which agreed within 36%. To verify the transient model, a sinusoidal inlet temperature was used, and the results agreed within 2% of the original values. Thus, a mesh of 227,475 elements was chosen.

Chapter 5

Anomaly Detection and Localization in Channel Flow

This chapter will begin by describing the anomaly, which was a heat flux through a portion of the wall, that was simulated for channel flow. Then, the results of detecting and localizing anomalies in different parts of the system and various combinations of inputs and outputs needed to accomplish that will be discussed. It will be shown that both detection and localization capabilities depend largely on the sensor configuration within the system. Detection and precedent-free localization was achieved for three anomalies that were positioned at different locations in the system. Lastly, a more dynamic system was created, and the results show that detection sensitivity decreased.

5.1 Description of Anomalies

A heat flux through a portion of the channel wall was simulated at three different positions along the bottom wall, as shown in Fig. 5.1. The fault was a heat flux that linearly increased with time ($0 \leq \tilde{q}" \leq 1$ from $50 \leq \tilde{t} \leq 300$) through a small portion of the channel wall.

Anomalies 1, 2 and 3 each had a width of $\tilde{D}_o = 0.1$ and were located

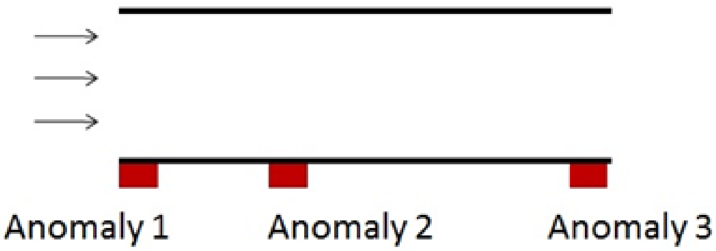


Figure 5.1: Locations of Anomalies 1, 2 and 3.

at $\tilde{S}_o = 0.0$, 0.3 and 0.9, respectively (see Fig. 4.1). When referring to anomaly simulations, for example, Anomaly 1 will refer to a simulation where normal behavior was first simulated, followed by anomalous behavior being simulated (heat flux through the wall) at the Anomaly 1 position. The same convention will be used from Anomaly 2 and Anomaly 3 at their respective positions. The following sections explore the capabilities and limitations of the GSMMS-based distributed anomaly detection method with different sensor configurations providing inputs and outputs to the system.

5.2 Finding the Orders of the Model

Before fault detection can proceed, the orders of the model must be chosen. To reiterate from Chapter 3, the GSMMS partitions the operating space into regions, and in each region, local models of linear form

$$\mathbf{F}_m(s(k)) = \mathbf{a}_i^T \mathbf{s}(k) + \mathbf{b}_i \tag{5.1}$$

where $\mathbf{a}_i$ and $\mathbf{b}_i$ denote the vectors of parameters of the local model i and

$$\mathbf{s}(k) = [\mathbf{y}^T(k), \ldots \mathbf{y}^T(k - n_a + 1), \mathbf{u}^T(k - n_d), \ldots, \mathbf{u}^T(k - n_d - n_b + 1)]^T \quad (5.2)$$

are assumed, where $\mathbf{y}(k) = [y_1(k), \ldots, y_p(k)]^T$ and $\mathbf{u}(k) = [u_1(k), \ldots, u_m(k)]^T$ are respectively, vectors of outputs and inputs, n_d is the time delay between when the input reaches the output, and n_a and n_b are the orders of the model.

The orders and the time delay were chosen based from the combination that resulted in the lowest root mean square (RMS) error on the testing data. Just like the training set, the testing data corresponded to the normal system behavior, except that the inlet conditions were different from the training data. The inlet conditions will be discussed in the next section.

For the rest of the study, all the input(s)/output orders and the time delay were chosen in this manner, i.e., many combinations of the orders and time delay were tried, and the combination with the lowest RMS was chosen for each system model. A recent paper [42] showed that the model order and delay parameters can be selected more systematically, but it was out of the scope of this study. For all the input(s)/output data in this chapter, temperature measurements were recorded in time increments of $\tilde{t} = 0.2$. The data was polluted with 0.5% noise to simulate sensor noise.

5.3 Fault Detection in Simple Channel Flow with an Inlet and Outlet Temperature Sensor

Intuitively, in the presence of an anomaly within a thermal-fluid system where there are fluid flow and heat transfer phenomena, anomalous effects will propagate throughout the system, thus complicating the localization of the sources of the anomalies.

The "eyes" of the fault detection method are the inputs and the output used by the anomaly detector. If the inputs and the output are not significantly affected by the anomaly, the anomaly will go unnoticed. It was expected that the anomaly, modeled as a heat flux through the wall, would affect the temperature distribution in the channel, which would thereby result in discrepancies between the GSMMS-predicted temperature outputs and the actual temperature outputs.

In this section, it was assumed that the system behaves normally when it is characterized by a constant Reynolds number ($Re_D = 100$) flow with the inlet temperature changing at random times for random time intervals. An example of the inlet temperature in a short time interval is shown in Fig. 5.2.

The inlet temperature changed at random time intervals for a period anywhere from $\tilde{t} = 4-7$ and ranged between $\tilde{T} = 0-1$ (as described in Chapter 4, both temperature and time were non-dimensionalized). Thermal inertia was accounted for by smoothing the temperature transitions using COMSOL's general function type. Random intervals and temperatures were created in MATLAB, which were then used to create the inlet conditions for each COMSOL

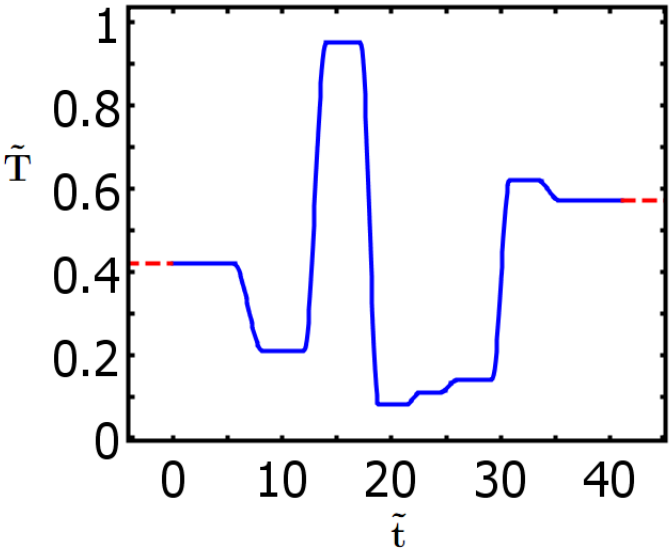


Figure 5.2: Example of how the temperature changed with time (both non-dimensionalized).

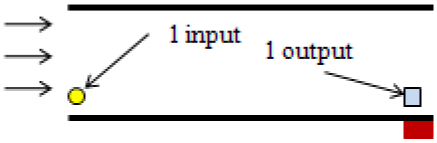


Figure 5.3: Anomaly 3, input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and output ($\tilde{x} = 1$, $\tilde{y} = 0.05$) positions.

simulation by using the ’constant’ extrapolation method where a ’continuous first derivative’ smoothing of 0.2 was used.

First, to verify that the method could detect anomalous behavior, the simplest configuration, shown in Fig. 5.3, was tested. In this configuration, only one input placed at the beginning of the channel ($\tilde{x} = 0$, $\tilde{y} = 0.05$) was used, while the output was obtained from a sensor placed at the very end of the channel ($\tilde{x} = 1$, $\tilde{y} = 0.05$). Thus, the GSMMS model uses the input from the beginning of the channel to predict the temperature at the end of the channel. An anomaly was placed at the end of the channel, essentially directly below the output temperature sensor. Therefore, one expected to achieve detection relatively easily since the output was relatively close to the anomaly.

Figure 5.4 shows the confidence value (CV, defined by Eqn. 3.14) vs. time plot. Between $\tilde{t} = 0-50$, the system behaved normally, which is confirmed by the high CVs over that period of time. At $\tilde{t} = 50$, the heater was turned on, beginning at $\tilde{q}" = 0$ and linearly increasing to $\tilde{q}" = 1$ by the end of the simulation ($\tilde{t} = 300$). Clearly, the CV dropped during that time, which clearly indicates that abnormal behavior was occurring.

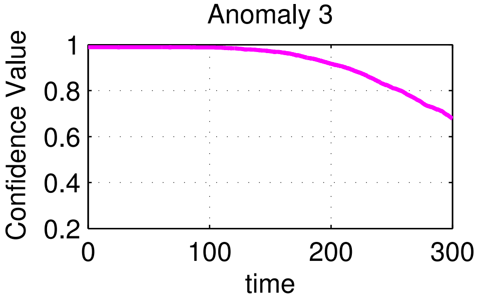


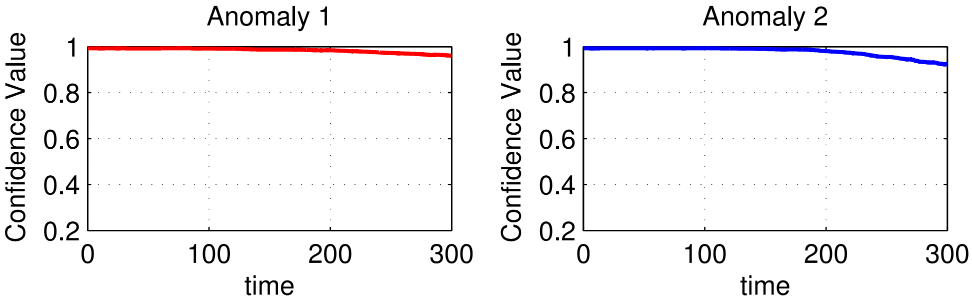
Figure 5.4: CV plot for Anomaly 3 with 1 input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and 1 output ($\tilde{x} = 1$, $\tilde{y} = 0.05$).

However, let us now observe Figs. 5.5(a) and 5.5(b), which show the CVs corresponding to the same input/output pair but with Anomalies 1 and 2 inserted into the system, respectively. There is obviously a detection lag because the anomaly was moved further upstream from the output. [1]

Furthermore, there is an obvious difference in the sensitivity of the anomaly detection in the cases of Anomalies 2 and 3 since the CV drop is less pronounced than in Fig. 5.4. For Anomaly 3, the overall CV drop was approximately 0.7, whereas the overall CV drop for Anomalies 1 and 2 were 0.95 (hardly dropped) and 0.9, respectively. The reason for the differences is that in the case of Anomaly 1 and 2, the anomaly's effects have dissipated by the time they have reached the temperature sensor at the end of the channel (output sensor). Conversely, in the case of Anomaly 3, the anomaly's effects have not dissipated much because the anomaly is located close to the output.

---

[1] Note that the same inlet temperature conditions and the same linearly increasing heat flux were used as before; only the location of the anomaly was simulated differently.



(a) Anomaly 1

(b) Anomaly 2

Figure 5.5: CV plots for Anomalies 1 and 2 with 1 input ($\tilde{x} = 0$, $\tilde{y} = 0.05$) and 1 output ($\tilde{x} = 1$, $\tilde{y} = 0.05$).

The above results show that since anomalous effects dissipate, the anomaly detection sensitivity will be different depending on where the anomaly is located relative to the position of the output. Thus, a more refined approach to sensor configuration placement is needed that can better detect a potential fault located anywhere in the system.

5.4 Fault Detection in a Simple Channel Flow with Distributed Sensing

In this section, the effects of distributed sensing and the inclusion of more inputs in the anomaly detection model is explored. In this way, we can observe if adding more inputs to the underlying GSMMS will increase detection sensitivity at a particular output location. Table 5.1 gives the $\tilde{x}$ positions for the inputs that were tested. Figure 5.6 shows the input sensor configurations that were tested, and Fig. 5.7 shows the effect of adding more inputs for the

Table 5.1: The $\tilde{x}$ positions of the inputs tested. For all inputs, $\tilde{y} = 0.05$.

	$\tilde{x}$
1 input	0.0
3 inputs	0.0, 0.3, 0.7
5 inputs	0.0, 0.2, 0.4, 0.6, 0.8
10 inputs	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9

three anomalies.

As seen in Figs. 5.7(a) and 5.7(b) the detection sensitivity was not affected as the number of inputs increased for Anomalies 1 and 2, i.e., adding more inputs did not increase the overall CV drop. In fact, for both Anomaly 1 and 2, when the anomaly was placed at the beginning and middle of the channel, respectively, the CV dropped more with fewer inputs. The reason for less detection sensitivity with more inputs is likely because the GSMMS model included more inputs that were affected by the anomaly and thus, ends up in the "unusual" (not well-trained) SOM regions where the local models are not very reliable (and hence, not very sensitive either). Alternatively, with less inputs affected by the anomaly, the GSMMS remains in the "usual" (well-trained) SOM regions, where local models are reliable, accurate, and therefore more sensitive to anomalies.

Differently, the results for Anomaly 3 indicate that generally, the greater the number of inputs, the more the CV will drop; however, a limit does seem to exist, i.e., as more inputs were added, there was less and less of an overall drop in CV. The reason why we see this pattern with Anomaly 3 and not with

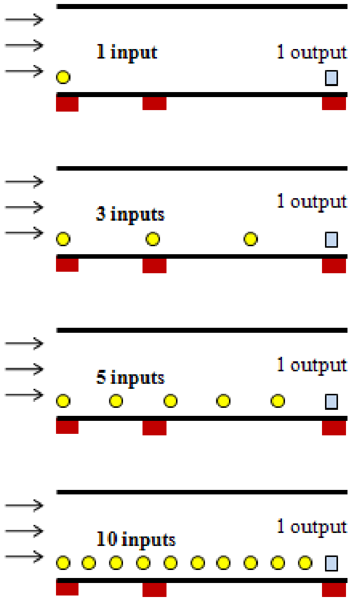
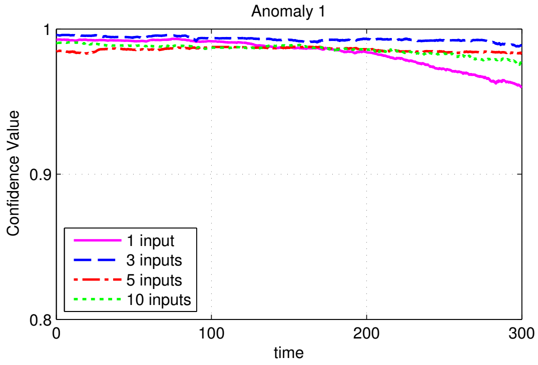
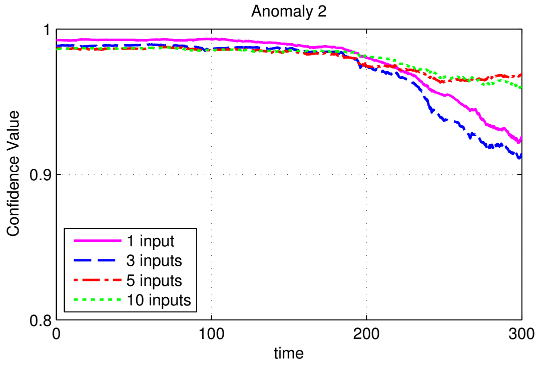


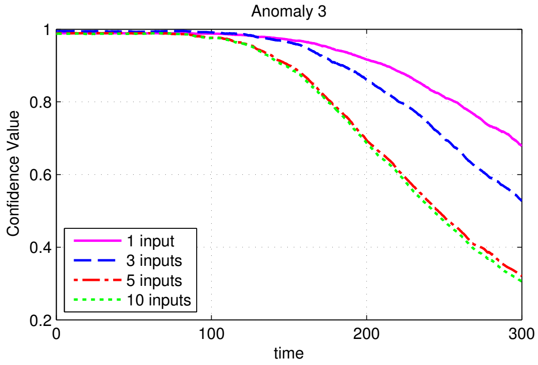
Figure 5.6: The positions of inputs. The positions of the three anomalies are also shown (note: for each simulation, only one anomaly was simulated).



(a) Anomaly 1



(b) Anomaly 2



(c) Anomaly 3

Figure 5.7: Anomalies 1, 2 and 3 with multiple inputs where the output is at the outlet of the channel. Note that the CV scale is different for Anomaly 3.

Anomalies 1 and 2 is that unlike the cases of Anomalies 1 and 2, the bulk of the inputs are unaffected by Anomaly 3 and thus, the GSMMS is able to gain additional information from the additional inputs without jumping into SOM regions where local GSMMS models are unreliable.

From the aforementioned results, it is clear that the relative distance between the output and the anomaly is important. Only if the output is close to the anomaly will more inputs benefit fault detection sensitivity. However, if an anomaly's effects have dissipated, the anomaly can go undetected and the use of more inputs will not significantly enhance the sensitivity.

Once an anomaly is detected, the next logical step is to localize its source (or sources). Let us use distributed GSMMS-based anomaly detectors to localize the source of anomalous behavior.

5.5 Fault Localization in a Simple Channel Flow using Distributed Anomaly Detectors

In the previous section, it was shown that a single sensor configuration will be prone to "blind spots" in which anomalies could not be detected and/or localized. In this section, rather than changing the number of inputs, only the position of the output will be changed. By inspecting the pattern of CV plots, both detection and localization will be attempted.

Figure 5.8 shows the pattern of CVs output by distributed GSMMS anomaly detectors for Anomalies 1, 2 and 3. Anomaly detectors were formed based on the GSMMS using two inputs from the beginning of the channel

($\tilde{x} = 0$, $\tilde{y} = 0.05$ and 0.95) and an output located downstream of the input [2]. The individual anomaly detectors (corresponding to individual CV plots) differ according to the position of the output. The output position was moved downstream from left to right. [3]. There were ten output positions spaced evenly from inlet to outlet, where the larger position number indicates a position further downstream. The output positions used for each Fig. 5.8 subplot is indicated in the channel drawing below the CV plots.

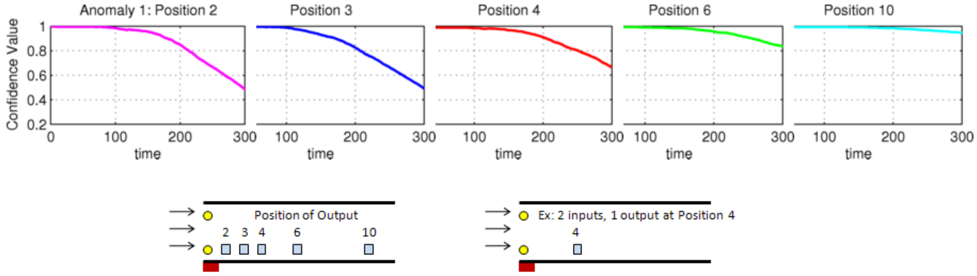
Figure 5.8(a) shows the results for Anomaly 1, the anomaly placed at the beginning of the channel. Clearly, anomalous behavior is indicated by the large overall CV drop for Positions 2 and 3. As the output was moved further downstream, the overall CV dropped less and less. This result agrees with the previous results: as the distance between the output and the anomaly becomes larger, the anomalous effects dissipate, and the corresponding CV drops less. Nevertheless, unlike what we had in the previous section where we only had a single anomaly detector (a single CV profile), the pattern of CVs output by the series of anomaly detectors we considered in Fig. 5.8(a) clearly indicates that the anomaly's point of origin is between the inlet and Position 2 or 3, which is where the anomaly is indeed located.

Using the same approach, Anomalies 2 and 3 can be detected and localized from using CV patterns shown in Figs. 5.8(b) and 5.8(c), respectively. For

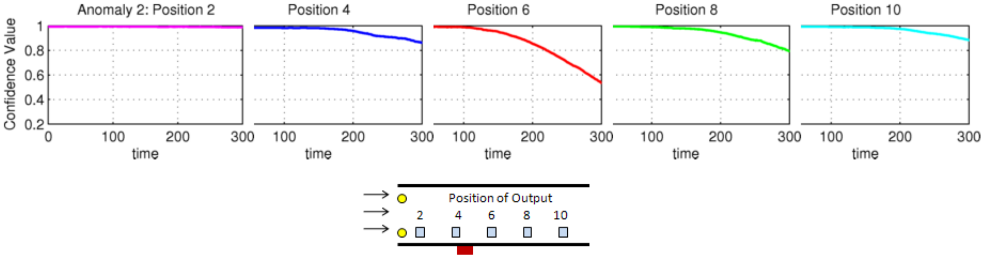
---

[2] all at a height of $\tilde{y} = 0.05$

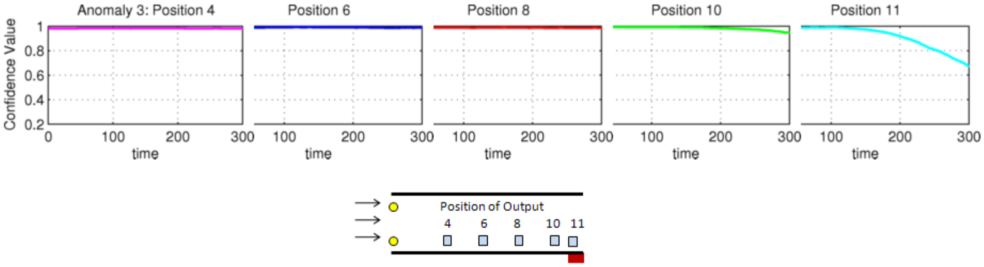
[3] Note that each anomaly detector and corresponding CV plot is a result of a different GSMMS model



(a) Anomaly 1: Positions 2, 3, 4, 6, 10 correspond to $\tilde{x} = 0.1, 0.2, 0.3, 0.5, 0.9$, respectively.



(b) Anomaly 2: Positions 2, 4, 6, 8, 10 correspond to $\tilde{x} = 0.1, 0.3, 0.5, 0.7, 0.9$, respectively.



(c) Anomaly 3: Positions 4, 6, 8, 10, 11 correspond to $\tilde{x} = 0.3, 0.5, 0.7, 0.9, 1.0$, respectively.

Figure 5.8: Anomalies 1, 2 and 3 where different output positions were used for each anomaly.

Anomaly 2, as shown in Fig. 5.8(b), normal behavior is indicated at Position 2. At Position 4, the CV drops slightly, indicating anomalous behavior, which is expected because Position 4 and the anomaly's location overlap slightly. At Position 6, the CV experienced the largest drop, and as the output is positioned further downstream, the overall CV drop decreases. Thus, an anomaly is detected and from the pattern of CV plots, it can be localized between Positions 4 and 6. Similar analysis of CV patterns in Fig. 5.8(c), the CV plots pinpoints Anomaly 3. Thus, the same sensor configuration approach was used to detect and localize an anomaly occurring at three different locations in the system.

In addition, the same approach can be used to detect and localize, multiple anomalies occurring simultaneously in the system, which has been a limitation of past fault detection methods. Figure 5.9 shows the CV plots when both Anomaly 1 and 2 occur simultaneously. As expected, the CV drops at Positions 2 and 3, which is the same as Fig. 5.8(a). However, at Position 5, the CV drops more than the previous position's CV. Downstream of Position 5, the overall CV drop decreases with position. Knowing that effects dissipate in a thermal-fluid system, this result indicates that two anomalies are occurring, with the first one being at the beginning of the channel, and the second anomaly being between Positions 4 and 6.

In this section, it was shown that distributed anomaly detectors can be used to detect and localize anomalies in a thermal-fluid system. Conceivably, only the number of sensors and their spacing limit the granularity of localiza-

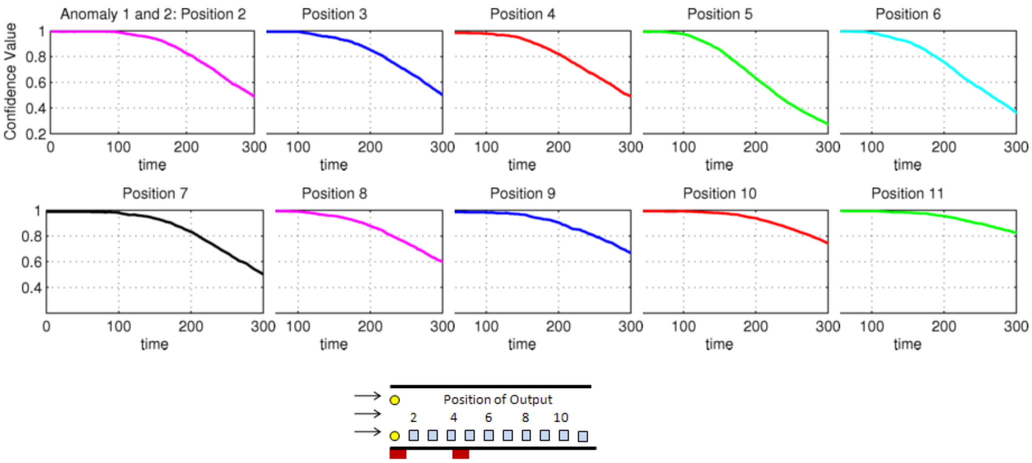


Figure 5.9: CV plots when both Anomaly 1 and 2 occur simultaneously. Notice that at Position 5, the overall CV drop is greater than that at Position 4, indicating a second anomaly is occurring.

tion, i.e., with more anomaly detectors, a higher resolution of localization can be achieved.

5.6 Fault Detection and Localization in a More Dynamic System

In the previous sections, the system was characterized by $Re_D = 100$, which is a relatively "slow" system. The anomalous effects had time to propagate before they reached the outlet, and fault detection and localization are more likely to be achieved if phenomena occur slowly since effects will dissipate at a slower rate. In this section, the results from a more dynamic system are described and discussed.

The previous results were from a system where only the inlet temperature changed at random intervals, and the Reynolds number remained constant. To create a more dynamic system, simulations were created where *both* the inlet temperature and the Reynolds number changed simultaneously. The range of the Reynolds number was between 100 and 1000, and it changed at increments of 100 at random intervals between $\tilde{t} = 4 - 8$. Figure 5.10 shows an example of how the temperature and Reynolds number could change with time. Note how the two parameters changed at different times for different time intervals.

As was done previously, Anomalies 1, 2 and 3 were simulated. Normal behavior was simulated from $\tilde{t} = 0 - 50$, and the heat flux began at $\tilde{t} = 50$, linearly increasing from $\tilde{q}" = 0 - 1$. Figure 5.11 shows the results for the three anomalies.

Figures 5.8 and Fig. 5.11 can be directly compared because the same anomalies were simulated with the same inputs and output. The only difference is both temperature and Reynolds number were changing in the Fig. 5.11 simulations, whereas only temperature was changing in the Fig. 5.8 simulations.

The most obvious difference between the two simulations is that with the more dynamic system, the overall CV drop was much less (note that the CV scale is different in Fig. 5.11). This is expected because with a “faster” system, the anomalous effects will be harder to discern from normal behavior. For example, for Anomaly 1, when $\tilde{q}" = 1$ at the end the simulation, the overall CV drop from Fig. 5.8(a) was approximately 0.5, whereas in Fig. 5.11(a), the

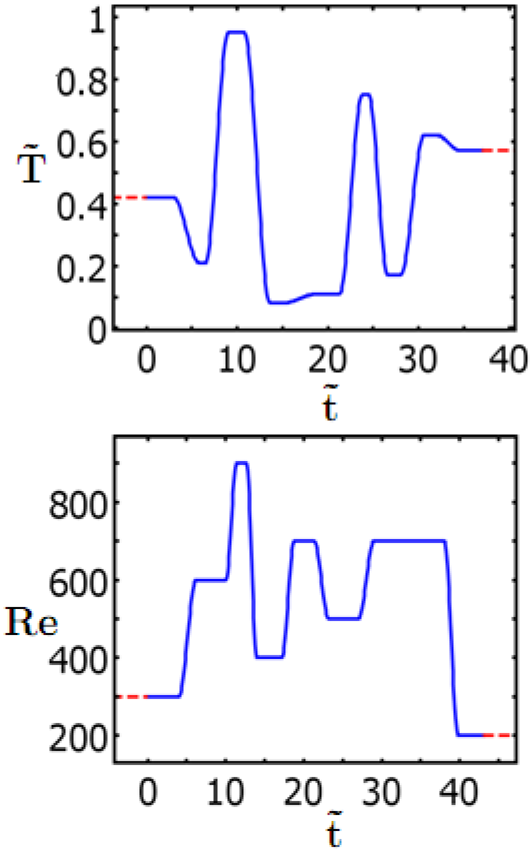


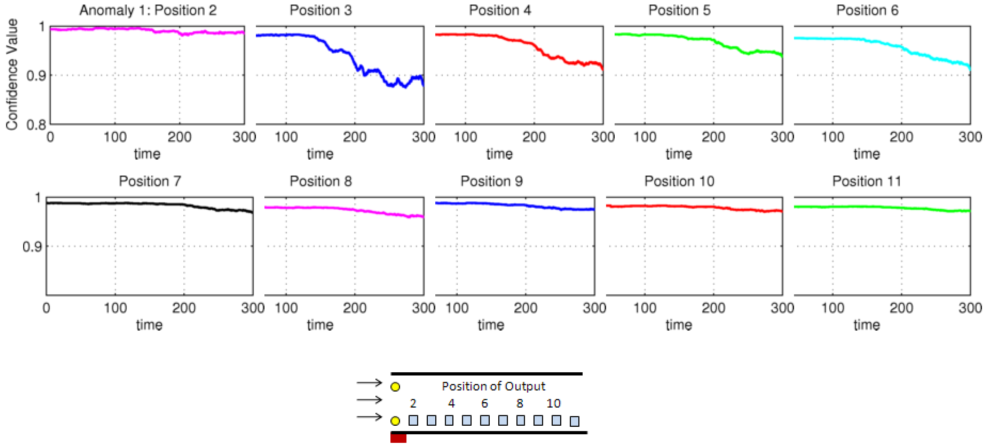
Figure 5.10: Example of changing temperature (non-dimensionalized) and Reynolds number.

overall CV drop was approximately 0.9.

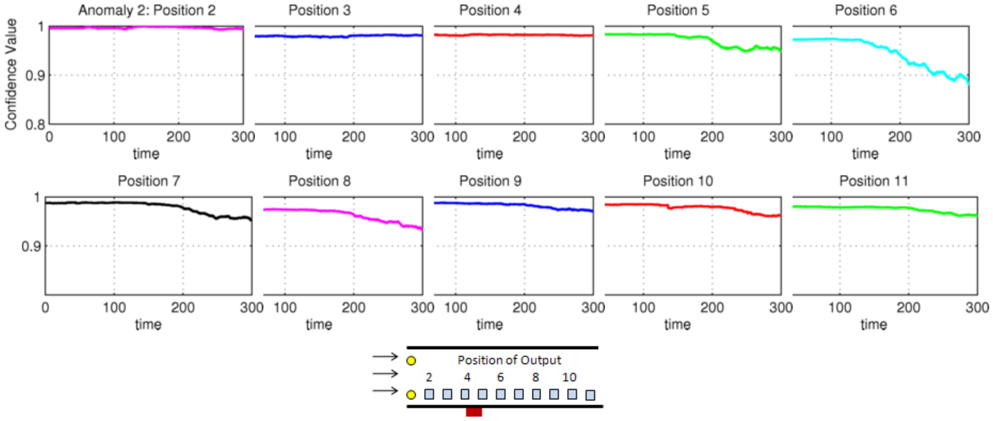
Also, localization becomes less clear with a more dynamic system. For each anomaly in Fig. 5.11, the position that corresponded to the CV that dropped first was one position further downstream compared with those in Fig. 5.8. For example, in Fig. 5.8(a) (Anomaly 1), the CV at Position 2 was the first CV to drop, while for the same anomaly, in Fig. 5.11(a), the CV at Position 3 dropped first, not Position 2. Also noticeably, for Anomaly 3, the CV at Position 11 hardly dropped, as seen in Fig. 5.11(c), whereas the CV at Position 11 dropped to approximately 0.7, as seen in Fig. 5.8(c).

The loss of detection sensitivity may not be entirely due to an inherently more dynamic and "faster" system, where anomalous effects have less time to reside in the system before exiting. The foundation of the fault detection method is system modeling using the GSMMS. If the system is not accurately modeled, fault detection will not be as sensitive to deviations away from normal behavior. The results from Fig. 5.11 used two inputs positioned at the very beginning of the channel. Intuitively, inputs located closer to the output would more likely predict a more accurate output.

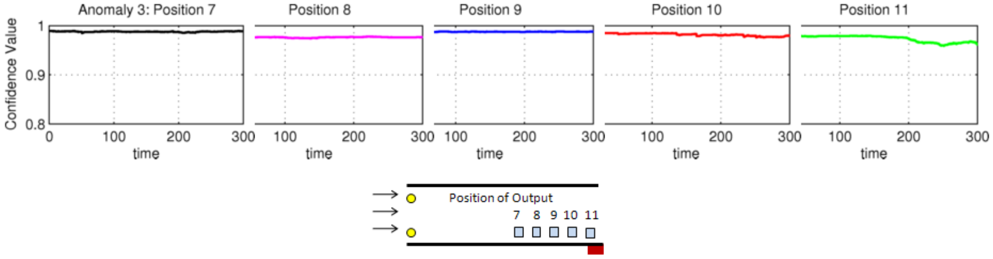
For the following results, the inputs were positioned close to the output, normal behavior occurred between $\tilde{t} = 0 - 50$, and then the heater was turned at $\tilde{t} = 50$ and increased linearly from $\tilde{q}" = 0 - 1$ until the end of the simulation ($\tilde{t} = 500$). Figure 5.12 shows the results for a single input that is one position upstream (a distance of $\tilde{x} = 0.1$) of the output for Anomaly 2. Clearly, the CV of Position 5 dropped, though the overall drop is approximately the same



(a) Anomaly 1

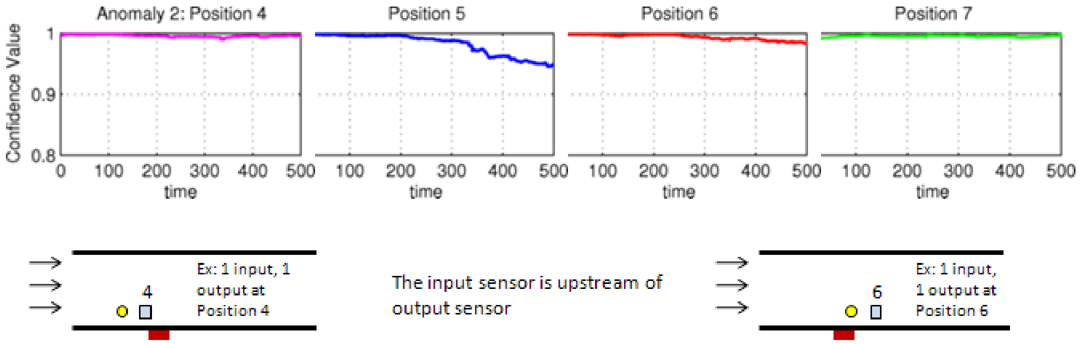


(b) Anomaly 2



(c) Anomaly 3

Figure 5.11: Anomalies 1, 2 and 3 for a more dynamic system



Ex: 1 input, 1 output at Position 4

The input sensor is upstream of output sensor

Ex: 1 input, 1 output at Position 6

Figure 5.12: CV plots when an input directly upstream was used. Positions 4, 5, 6, 7 correspond to $\tilde{x} = 0.3, 0.4, 0.5, 0.6$

as its counterpart in Fig. 5.11(b). However, differently from Fig. 5.11(b), the downstream CVs indicate normal behavior. This is most likely because the anomalous effects have already dissipated and thus, the downstream outputs are not as affected by the anomaly. In this way, localization becomes more definite as clearly seen in Fig. 5.12, the anomaly's point of origin is between Positions 4 and 5.

In this chapter, it was shown that fault detection and precedent-free localization can be achieved when multiple anomaly detectors are used. A single sensor configuration could be "blind" to different faults because the anomalous effects could dissipate entirely by the time they reach the output. Using multiple anomaly detectors ensures that more of the system can be monitored. However, fault detection and localization greatly depend on the dynamics of the system. The "faster" the anomalous effects travels through and exits the system, the less sensitive detection becomes. Using inputs that are close to the output can make localization clearer, though the CV plots need to be interpreted carefully.

In the next chapter, a more complex system is simulated to observe if the distributed anomaly detection approach is effective.

Chapter 6

Anomaly Detection and Localization in Channel Flow with an Obstacle

In the previous chapter, fault detection and precedent-free localization was achieved in simple channel flow through distributed anomaly detection. However, channel flow is characterized by relatively simple flow dynamics. In this chapter, the distributed anomaly detection approach is applied to a more dynamically complex system: a channel flow with a sharp obstacle.

6.1 Description of Anomalies and Normal Behavior

The system, shown in Fig. 4.1, is channel flow and contains an obstacle that obstructs the flow. As was done in the previous chapter, a heat flux that linearly increased with time ($0 \leq \tilde{q}" \leq 1$ from$100 \leq \tilde{t} \leq 200$) through a portion of the wall acted as the anomaly. For each anomaly, the heat flux began at $\tilde{t} = 100$. Three anomalies were simulated, as shown in Fig. 6.1. Anomaly 1 (positioned at $\tilde{S}_o = 1.7$) was located upstream of the obstacle. Anomaly 2 was located downstream of the obstacle (positioned at $\tilde{S}_o = 2.35$), and Anomaly 3 was located on the channel wall above the obstacle (positioned at $\tilde{S}_o = 2.1$). Also, as was done previously in Chapter 5, the width of each heat flux was

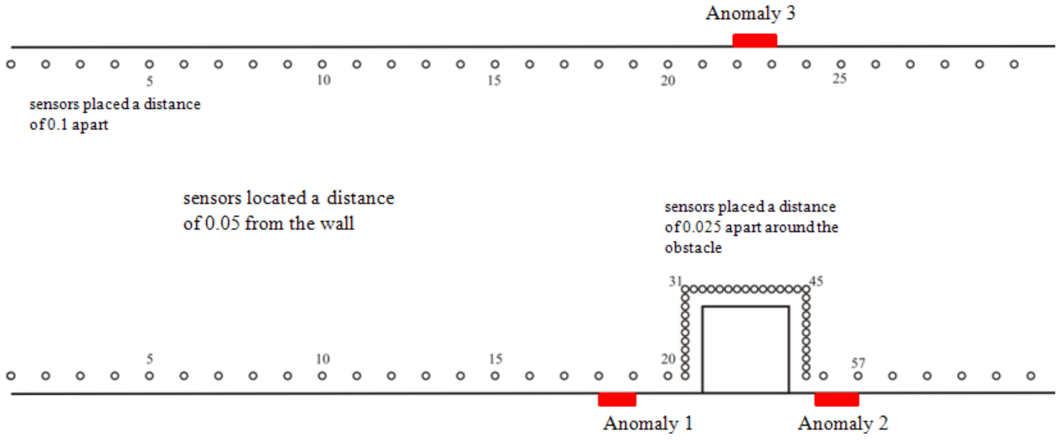

Anomaly 3
5
10
15
20
25
sensors placed a distance of 0.1 apart
sensors located a distance of 0.05 from the wall
sensors placed a distance of 0.025 apart around the obstacle
31
45
5
10
15
20
57
Anomaly 1
Anomaly 2

Figure 6.1: Locations of Anomalies 1, 2 and 3.

$\tilde{D}_o = 0.1$.

In the following simulations, normal system behavior was characterized by a constant Reynolds number of 200 and a Prandtl number of 0.72. The inlet temperature ranged from $\tilde{T} = 0 - 1$ and changed at random times for random intervals (between $\tilde{t} = 4 - 6$). Temperature measurements were sampled every $\tilde{t} = 0.1$, and 0.5% noise was added to each measurement.

6.2 Fault Detection and Localization Upstream of the Obstacle

Similarly to what we did in Chapter 5, we will use distributed GSMMS-based anomaly detection and interpretation of the CVs output and then localize an anomaly's point of origin. The sensors were located a distance of $\tilde{y} = 0.05$ away from the walls, spaced a distance of $\tilde{x} = 0.1$ apart along the

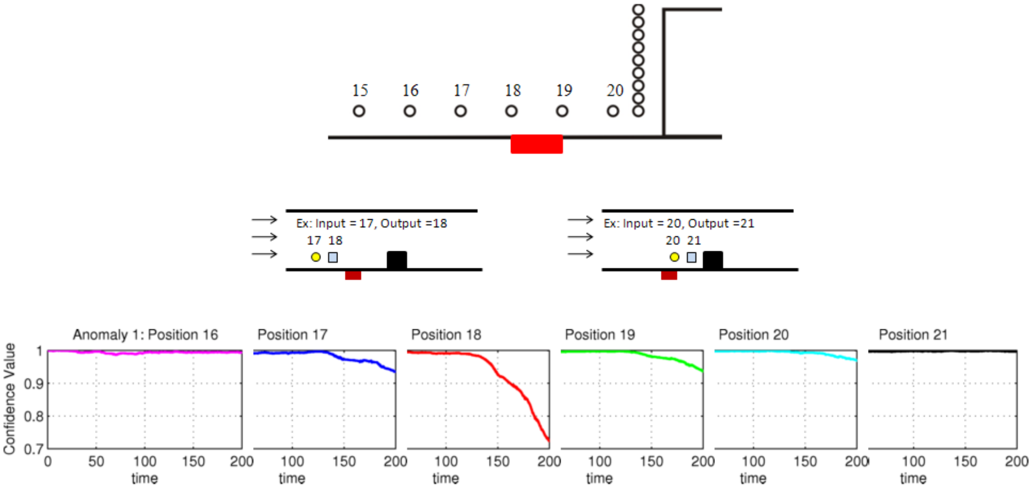


Figure 6.2: Anomaly 1 is upstream of the obstacle. The position number in the CV plots represents the output position.

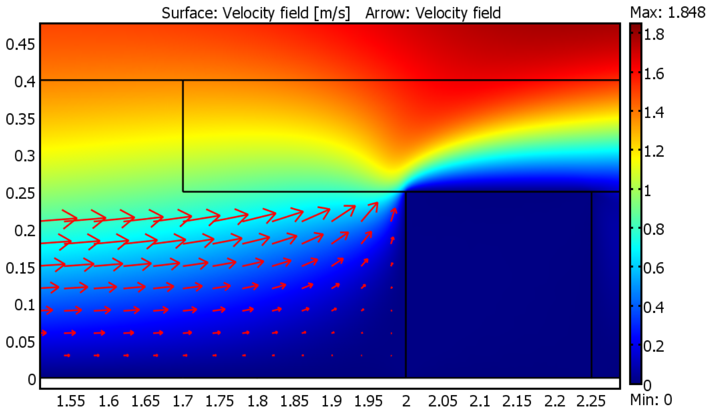
channel walls and spaced a distance of $\tilde{x} = 0.025$ along the obstacle wall. The sensor configuration is shown in Fig. 6.1. For each sub-system, one input was used, which was the sensor directly upstream of the output. Also, as was done in Chapter 5, the orders of each model were chosen based on the lowest RMS.

Figure 6.2 shows the result for Anomaly 1, the anomaly upstream of the obstacle. For clarity, only a portion of the wall is shown. The position number shown in the title of each CV plot refers to the output position. The corresponding input is always the previous Position number.

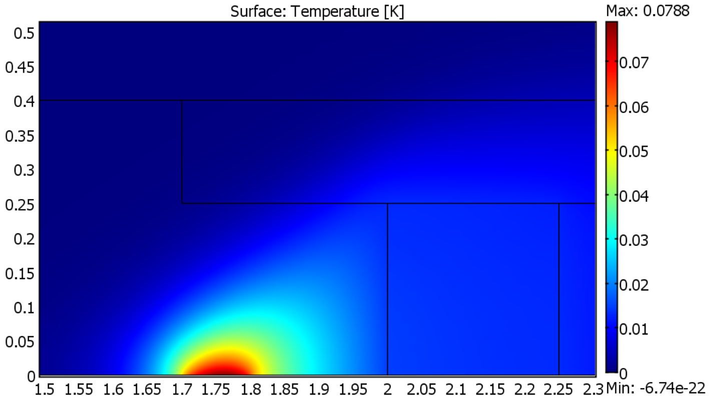
From Fig. 6.2, abnormal behavior is clearly indicated at Position 18, where the overall CV dropped to approximately 0.73. Also, the CV at Positions 17, 19 and 20 dropped approximately 0.05, 0.05 and 0.02, respectively, by the end of the simulation.

However, localization is less clear. Anomalous behavior can be isolated between Positions 16 and 21 since both the CVs at those locations indicate normal behavior. Anomaly 1 was located between Positions 18 and 19, but the first CV to drop was located at Position 17, which is slightly upstream of the anomaly. However, the largest drop occurred at Position 18, which is above the anomaly. This CV pattern differs from the results in Chapter 5, where the first CV to drop was either the output above the upstream corner (the CV at Position 18 in Fig. 6.2) or one of the two outputs downstream of the anomaly (Positions 19 or 20).

The reason for this behavior can be explained by Fig. 6.3 (note that the black lines above the obstacle are meshing boundaries since a finer mesh was required above the obstacle due to the high gradients). Figure 6.3(a) shows that no recirculation exists at Position 17 ($\tilde{x} = 1.6$, $\tilde{y} = 0.05$), but the velocity is very small. Therefore, it is reasonable to assume that the heat flux is propagating out in all directions and not being dissipated downstream. Figure 6.3(b) shows a steady-state example when the heat flux at the Anomaly 1 position is $\tilde{q} = 1$, and the inlet temperature is $\tilde{T} = 0$. The results show that the anomalous effects can affect the upstream sensor at Position 17, which is why its CV dropped. Thus, it is difficult to localize the anomaly upstream of the obstacle without knowing how the particular anomaly propagates outwards into the system.



(a) Velocity field upstream of the obstacle. The arrows show that there is no recirculation.



(b) Temperature field upstream of the obstacle, which shows a steady-state example when $\tilde{q}'' = 1$. The figure shows that at the highest heat flux, the temperature at ($\tilde{x} = 1.6$, $\tilde{y} = 0.05$) is affected by the anomaly. Note that the heat flux from Anomaly 1 is situated between $1.7 \leq \tilde{x} \leq 1.8$, and the output's CV to drop first, as seen in Fig. 6.2, is located at $\tilde{x} = 1.6$.

Figure 6.3: Velocity and temperature field upstream of the obstacle.

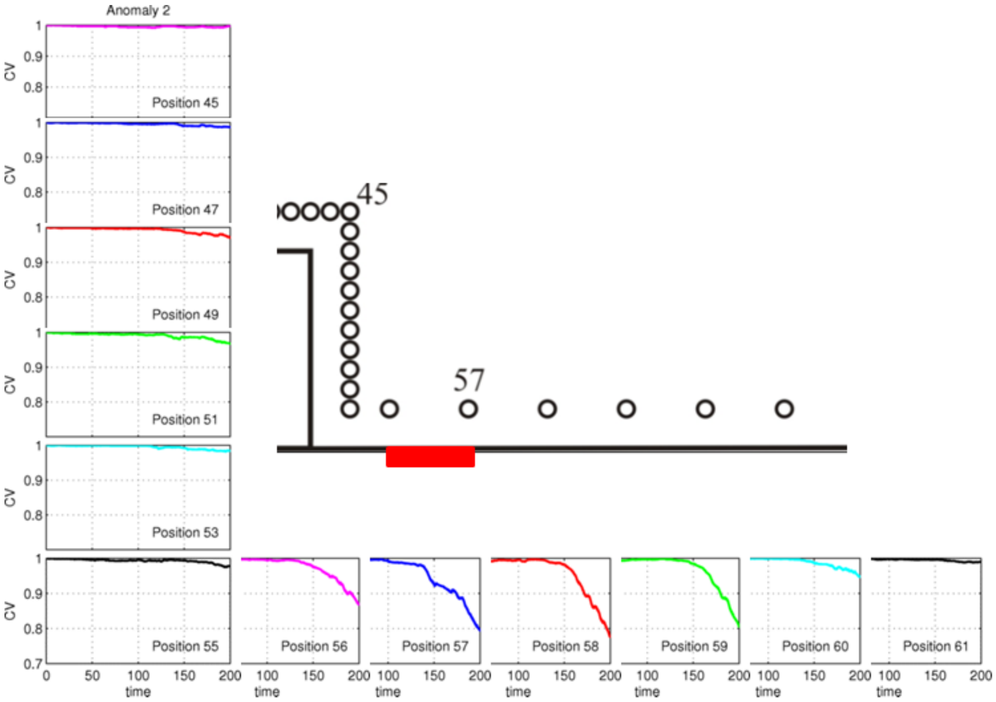
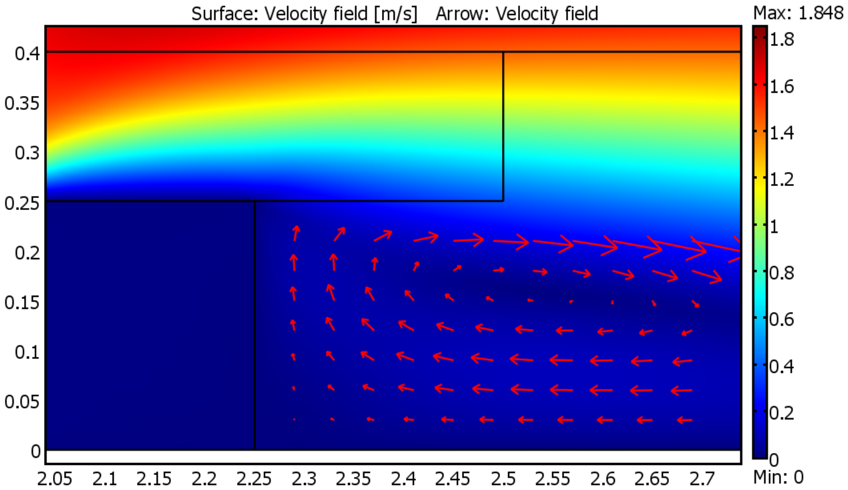


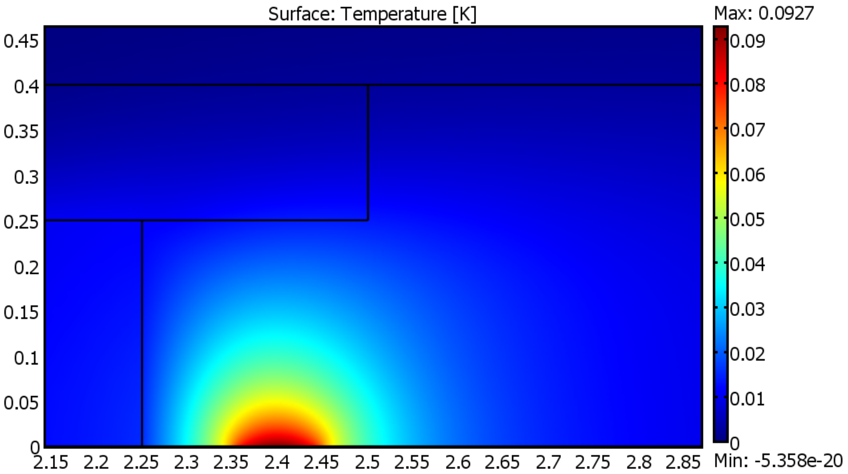
Figure 6.4: Anomaly 2 is downstream of the obstacle. The position number indicated in the CV plots represents the output position.

6.3 Fault Detection and Localization Downstream of the Obstacle

Figure 6.4 shows the CV plots for Anomaly 2, which is downstream of the obstacle. The CVs along the obstacle wall drop slightly (Positions 47-55). The first CV to substantially drop is at Position 56, while the largest overall CV drop was at Position 58. However, the CVs at Positions 57, 58 and 59 all approximately dropped the same amount, which is different from previous results. Typically, only a single, large CV drop was observed.



(a) Velocity field downstream of the obstacle. The arrows show that there is recirculation.



(b) Temperature field downstream of the obstacle, which shows a steady-state example when $\tilde{q}" = 1$.

Figure 6.5: Velocity and temperature field downstream of the obstacle.

The relatively large CV drop at the three adjacent positions indicate an anomaly whose size is larger than what has been seen, i.e., a heat flux with a larger $\tilde{D}_o$ (larger width along the channel wall). To explain this result, Fig. 6.5(a) shows the velocity field downstream of the obstacle, where a recirculation pattern exists; the fluid is slowly rotating in the clockwise direction, from upstream to downstream. Figure 6.5(b) shows the temperature field downstream of the obstacle. As seen previously in Fig. 6.3, since the velocity is so low, the heat flux essentially propagates in all directions outwards.

However, the low velocities and the existence of recirculation does not explain why the three adjacent CVs all dropped to approximately the same level. Figure 6.6, which shows an example of the temperature gradient during normal behavior, offers an explanation. Figure 6.6 shows that a higher temperature gradient appears upstream of the obstacle compared with the temperature gradient downstream of the obstacle. It is obvious from Figs. 6.3(b) and 6.5(b) that a heat flux results in relatively high temperature gradients around the vicinity of its origin. Thus, upstream of the obstacle, the outputs are not as "surprised" by the anomalous effects as the outputs downstream of the obstacle, which explains the single large CV drop in Fig. 6.2, and several large CV drops in Fig. 6.4. Upstream of the obstacle, the dynamics of a heat flux are similar to the dynamics during normal behavior but not similar to the dynamics downstream of the obstacle. Thus, upstream of the obstacle, the GSMMS predictions for each output will be closer to the actual output compared with the predictions downstream of the obstacle.

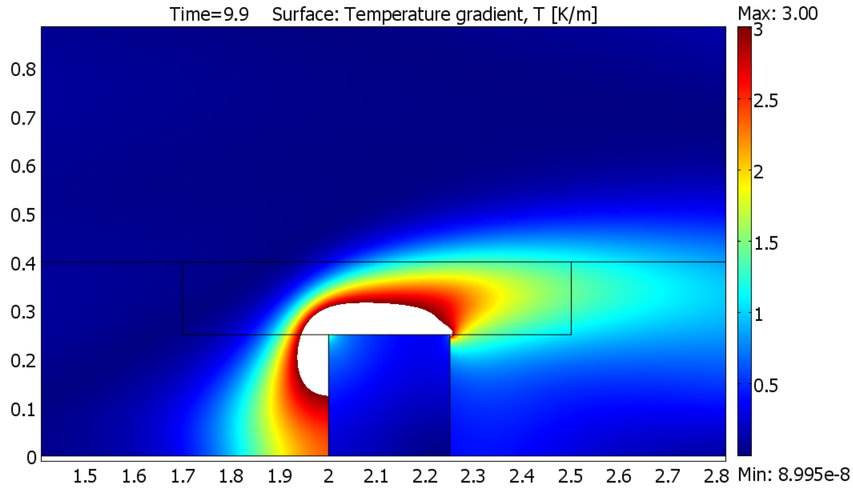


Figure 6.6: Example of the temperature gradient during normal behavior. To better show the gradient near the wall, a maximum temperature gradient of 3 was set, which explains the white space. The highest gradients occur along the top of the obstacle.

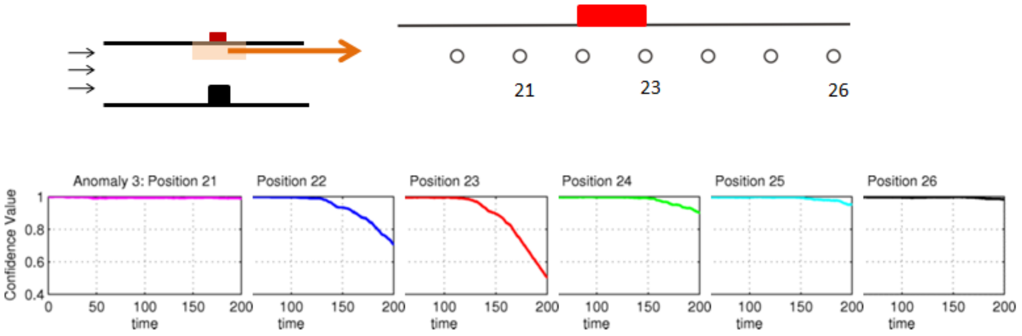


Figure 6.7: Anomaly 3 is above the obstacle. The position number indicated in the CV plots represents the output position.

6.4 Fault Detection and Localization Above the Obstacle

Figure 6.7 shows the CV plots for Anomaly 3, the anomaly above the obstacle. Since the dynamics above the obstacle are similar to simple channel flow, the CV plot pattern is similar to the results from channel flow without the obstacle.

The CV plot at Position 22 dropped first, and the CV plot at Position 23 exhibited the largest drop. Both Positions 24 and 25 only dropped slightly. Thus, anomalous behavior was detected, and its origin can be determined to be between Positions 21 and 23.

From the results of this chapter, it was found that anomalous behavior can be detected in a more hydrodynamically complex system by dividing the system into many sub-systems and monitoring the entire system with simple one-input one-output models, where inputs are directly upstream of the output. With this approach, localization of the anomaly can also be achieved through the multiple anomaly detectors. However, localization becomes less clear because the anomalous effects can propagate upstream. Also, the same type of anomaly can result in a different CV plot pattern, depending on the dynamics in the system. Thus, as systems become more complex, knowledge of the system dynamics may be increasingly needed to properly interpret the CV plot patterns.

Chapter 7

Conclusions and future work

7.1 Conclusions

Past fault detection and localization methods require extensive a priori knowledge of the system and knowledge of the potential fault characteristics. Therefore, such precedent-based methods are limited to whether this knowledge exists. In this thesis, a precedent-free fault detection and localization method was applied to selected channel flow systems. Using only normal-behavior data, faults located at different positions were detected and then localized.

The previously developed Growing Structure Multiple Model System (GSMMS) based fault detection and localization method was used to accomplish this goal. Data-driven models, such as GSMMS models, are advantageous compared with models based on first principles because they usually require fewer assumptions, are less time consuming to create and enable one to utilize process measurements that are readily available. Specifically, the GSMMS-based method is a "divide and conquer" approach to system modeling that uses a self-organizing map (SOM) to partition the operating space of the system (state space of the system) into regions of similar input-output patterns, and a linear model is fit within each region. Thus, a highly complex, nonlinear system can be accurately modeled using models that are locally tractable.

Just like in most other anomaly detection schemes, an abnormality is detected when modeling residuals differences between system outputs and those predicted by the model display abnormal dynamic patterns. Residuals are the differences between the model-predicted output and the monitored output. In the past anomaly detection methods, faults were detected when residuals become greater than a pre-defined threshold. If the threshold is set too low, numerous false alarms will result, and if the threshold is set too high, faults can go undetected. In this thesis, a more refined residual analysis was performed that eliminated the need to set pre-defined thresholds. Instead, residual analysis was performed locally, within each GSMMS region, which simplified the analysis. For each region within the system model, a confidence value (CV) was calculated as the overlap of the probability density function (PDF) of the modeling residuals corresponding to the normal system behavior data and the currently observed GSMMS residuals. Thus, if no fault exists in the system, the CV will be close to one, indicating that the current residual behavior matches the normal residual behavior well and will be reduced when the current system is deviating from normal behavior. For simplicity, a global CV was created as the geometric mean of the regional CVs, indicating the general deviation away from normal behavior that drops lower as soon as one (or a few) regional CVs drop. Obviously, only normal behavior knowledge is needed for a fault to be detected.

Numerical verification of the GSMMS anomaly detection and fault localization based on distributed anomaly detectors was conducted on two simulated thermal-fluid systems. Data measurements for both systems were obtained by simulating channel flow using the finite element method. The first system was a single channel flow in which normal behavior was first defined by a changing inlet temperature with a constant Reynolds number. The GSMMS models used temperature measurements at various points in the channel as the inputs and outputs. A heat flux through a portion of the channel wall, which linearly increased from zero to one, was simulated as the anomaly, and CV plots were created.

Three anomaly positions were separately simulated to compare the degree of detection and localization: at the beginning, near the middle and the end of the channel. It was found that the relative distance between the anomaly and the output sensor was important. If the distance was too great, the anomalous effects will have dissipated upon reaching the output and thus, a single input/output configuration becomes "blind" to certain faults, depending on the anomaly's point of origin. Therefore, multiple anomaly detectors that monitor different sub-systems (sections of the system) are needed to prevent "blind spots". Using distributed anomaly detection, the three anomalies were detected by interpreting the CV plots pattern. Since anomalous effects dissipate in channel flow, the sub-system with the largest overall CV drop indicates the approximate point of origin of the anomaly. Thus, not only were the three anomalies detected using the same input/output configuration, but they were also localized within the system using only normal system behavior. Furthermore, the same approach was able to detect and localize multiple anomalies occurring simultaneously, which has been a major limitation in previous fault detection methods.

Since channel flow is a relatively simple system, more complex systems were tested. First, the channel flow system was made more dynamic by simulating normal behavior where both the inlet temperature and the Reynolds number changed at different times for different time intervals. Using the same anomalies and the same input/output configuration, the anomalies were detected and localized but with less sensitivity than before, i.e., each sub-system's overall CV drop was less than when only the inlet temperature changed.

Secondly, a different channel flow system with a sharp obstacle was created. Using the same paradigm of distributed anomaly detection, three anomalies that were positioned upstream, downstream and above the obstacle were detected. It was noted that anomaly localization was more challenging to achieve than in the case of a smooth channel flow because of the interaction of the heat flux (the anomaly) with the system. Upstream of the obstacle, the velocities are very low, which allows anomalous effects to propagate upstream, thereby affecting the output and consequently, the CV plot upstream of the anomaly. Downstream of the obstacle, the velocities are also very low and because the anomaly has a larger effect on the dynamics in that area downstream of the obstacle, the overall CV drop is greater than the same anomaly occurring upstream. Thus, localization is not as clear in a more complex system.

7.2 Future work

The main objective of this work was to verify that the GSMMS-based, precedent-free fault detection and localization method could successfully be applied, for the first time, to a distributed parameter system. Both detection and localization was achieved in two channel flow systems of different complexity using only normal behavior data. To further advance this work, the following are recommendations for future work:

- In this study, only a single type of fault was simulated: a heat flux through a portion of the wall. Other faults need to be simulated, such as fouling or leaking, which are typical examples of faults in a thermal-fluid system.
- In this study, the highest Reynolds number occurring in the system was 1000, which is still a laminar flow, whereas numerous practical systems are characterized by turbulent flows. Therefore, a GSMMS-based model of a turbulent flow system also needs to be obtained and its accuracy verified. It is expected that with a higher Reynolds number, both fault detection and localization will be more challenging because of the increase in system complexity.
- Since anomalous effects tend to dissipate as they travel downstream, the sensor configuration is of vital importance. A systematic way to optimize the sensor configuration is needed to achieve good fault localization (coverage of "blind spots") with the lowest possible sensing costs.

Bibliography

[1] K. Yang. Artificial neural networks (ANNs): A new paradigm for thermal science and engineering. *J. Heat Transfer*, 130:1–19, 2008.

[2] C. Karlsson, J. Arriagada, and M. Genrup. Detection and interactive isolation of faults in steam turbines to support maintenance decisions. *Simulation Modelling Practice and Theory*, 16:1689–1703, 2008.

[3] H. Calisto, N. Martins, and N. Afgan. Diagnostic system for boilers and furnaces using CFD and neural networks. *Expert Systems with Applications*, 35:1780–1787, 2008.

[4] S. Kalogirou, S. Lalot, G. Florides, and B. Desmet. Development of neural network-based fault diagnostic system for solar thermal applications. *Solar Energy*, 82:164–172, 2008.

[5] Jianbo Liu. *Autonomous Anomaly Detection and Fault Diagnosis*. PhD thesis, University of Michigan, 2007.

[6] J. Liu, D. Djurdjanovic, K. Marko, and J. Ni. Growing structure multiple model system for anomaly detection and fault diagnosis. *Transactions of ASME, J. Dynamic Systems, Measurement and Control*, 131(5):051006–1–051006–13, 2008.

[7] M. Cholette and D. Djurdjanovic. Precedent-free fault isolation in a diesel engine EGR valve system. submitted to *Trans. Of IMechE, Journal of Automotive Systems*, paper no. JAUTO 1706, 2010.

[8] D. Djurdjanovic, C. Hearn, and Y. Liu. Immune systems inspired approach to anomaly detection, fault localization and diagnosis in a generator. In *Proc. of the 2010 Conference of Grand Challenges in Modeling and Simulation (GCMS)*, Ottawa, ON, 2010. paper no. 71.

[9] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin; New York, 1995.

[10] S. Simani, C. Fantuzzi, and R. Patton. *Model-based fault diagnosis in dynamic systems using identification techniques*. Springer-Verlag, London, 2003.

[11] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri. A review of process fault detection and diagnosis. part i: quantitative model-based methods. *Computers and Chemical Energy*, 27:293–311, 2003.

[12] R. Isermann. Process fault detection based on modeling and estimation methods-a survey. *Automatica*, 20(4):387–404, 1984.

[13] M. Basseville. Detecting changes in signals and systems-a survey. *Automatica*, 24(3):309–326, 1988.

[14] P.M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy-a survey and some new results. *Automatica*, 26:459–474, 1990.

[15] P.M. Frank, S.X. Ding, and T. Marcu. Model-based fault diagnosis in technical processes. *Transactions of the Institution of Measurement and Control*, 22(1):57–101, 2000.

[16] T. Wakui and R. Yokoyama. On-line model-based performance monitoring of a shell-and-tube type heat exchanger using steam and water. *Energy Conversion and Management*, 49:2669–2677, 2008.

[17] S. Shah, G. Liu, and D. Greatrix. On-line fouling detection of aircraft environmental control system cross flow heat exchanger. In *Proceedings of the 2009 IEEE International Conference on Mechatronics and Automation*, Changchun, China, August 2009.

[18] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri. A review of process fault detection and diagnosis. part ii: qualitative models and search strategies. *Computers and Chemical Energy*, 27:313–346, 2003.

[19] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri. A review of process fault detection and diagnosis. part iii: process history based methods. *Computers and Chemical Energy*, 27:327–346, 2003.

[20] T. Sorsa, H. Koivo, and H. Koivisto. Neural networks in process fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics*, 21:815–825, 1991.

[21] S. Lalot and H. Palsson. Detection of fouling in a cross-flow heat exchanger using a neural network based technique. *Int. J. Thermal Sciences*, 49:675–679, 2010.

[22] S. Haykin. *Neural networks, a comprehensive foundation.* Macmillian, New York, 1994.

[23] K. Jambunathan, S.L. Hartle, S. Ashforth-Frost, and V.N. Fontama. Evaluating convective heat transfer coefficients using neural networks. *Int. J. Heat Mass Transfer*, 39:2329–2332, 1996.

[24] Y. Kuroe and I. Kimura. Modeling of unsteady heat conduction field by using composite recurrent neural networks. In *Proceedings of the IEEE International Conference of Neural Networks*, volume 1, pages 323–328, 1995.

[25] G.S. Mittal and J. Zhang. Prediction of freezing time for food products using a neural network. *Food Research International*, 33:557–562, 2000.

[26] A. Pacheco-Vega, M. Sen, K.T. Yang, and R. McClain. Neural network analysis of fin-tube refrigerating heat exchanger with limited experimental data. *Int. J. Heat Mass Transfer*, 44:763–770, 2001.

[27] G. Diaz, M. Sen, K.T. Yang, and R. McClain. Dynamic prediction and control of heat exchangers using artificial neural networks. *Int. J. Heat Mass Transfer*, 44:1671–1679, 2001.

[28] T. Liu, X. Sun, and H. Wang. Neural network analysis of boiling heat transfer enhancement using additives. *Int. J. Heat Mass Transfer*, 45:5083–5089, 2002.

[29] C. Riverol and V. Napolitano. Estimation of fouling in a plate heat exchanger through the application of neural networks. *Journal of Chemical Technology and Biotechnology*, 80:594–600, 2005.

[30] W. Tian and S. Sun. Parameter correction based fault diagnosis for chemical pipeline leakage. In *Proceedings of the 7th World Congress on Intelligent Control and Automation*, pages 3648–3653, 2008.

[31] J. Cui and S. Wang. A model-based online fault detection and diagnosis strategy for centrifugal chiller systems. *Int. J. Thermal Sciences*, 44:986–999, 2005.

[32] Y. Chetouani. A neural network approach for the real-time detection of faults. *Stochastic Environmental Research and Risk Assessment*, 22:339–349, 2008.

[33] D.V. Hinkley. Inference about the change-point from cumulative sum tests. *Biometrika*, 58:509–523, 1971.

[34] D. Alahokoon, S.K. Halgamuge, and B. Srinivasan. Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE Transaction on Neural Networks*, 11:601–614, 2000(3).

[35] T.A. Johansen and B.A. Foss. Identification of non-linear system structure and parameters using regime decomposition. *Automatica*, 31(2):321–326, 1995.

[36] R. Sedgewick. *Algorithms in C++. Part 5, Graphical Algorithms.* Addison Wesley, London, 1995.

[37] G. McLachlan and D. Peel. *Finite Mixture Models.* John Wiley & Sons, Inc., 2000.

[38] Z. Zivkovic and F. van der Heijden. Recursive unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):651–656, 2004.

[39] COMSOL, Inc. *comsol Multiphysics, version 3.5a, User's Guide.*

[40] A.K. da Silva, S. Lorente, and A. Bejan. Optimal distribution of discrete heat sources on a wall with laminar forced convection. *Int. J. Heat Mass Transfer*, 47:2139–2148, 2004.

[41] T. Young and K. Vafai. Convective cooling of a heat obstacle in a channel. *Int. J. Heat mass Transfer*, 41:3131–3148, 1998.

[42] L. Jiang, E. Latronico, and J. Ni. A novel method for input selection for the modeling of nonlinear dynamic systems. In *Proceedings of 2008 ASME Dynamic Systems and Control Conference*, Ann Arbor, Michigan, 2008.

Vita

Katherine Carpenter grew up in Springfield, MO before moving to Dallas, TX in 1994. She attended The University of Texas at Austin and received a Bachelor of Science in Mechanical Engineering in 2008. In the Fall of 2008, she enrolled in graduate school at The University of Texas at Austin.

Permanent address: 3460 Lark Meadow Way
Dallas, Texas 75287

This thesis was typeset with LaTeX[†] by the author.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.