Fault Detection and Diagnosis in Air Conditioners and Refrigerators

A. M. Noonan, N. R. Miller, and C. W. Bullard

ACRC TR-155 August 1999

For additional information:

Air Conditioning and Refrigeration Center
University of Illinois
Mechanical & Industrial Engineering Dept.
1206 West Green Street
Urbana, IL 61801

(217) 333-3115

Prepared as part of ACRC Project 87
Fault Detection and Diagnosis in Air Conditioners and Refrigerators
C. W. Bullard and N. R. Miller, Principal Investigators

The Air Conditioning and Refrigeration Center was founded in 1988 with a grant from the estate of Richard W. Kritzer, the founder of Peerless of America Inc. A State of Illinois Technology Challenge Grant helped build the laboratory facilities. The ACRC receives continuing support from the Richard W. Kritzer Endowment and the National Science Foundation. The following organizations have also become sponsors of the Center.

Amana Refrigeration, Inc.
Brazeway, Inc.
Carrier Corporation
Caterpillar, Inc.
Chrysler Corporation
Copeland Corporation
Delphi Harrison Thermal Systems
Frigidaire Company
General Electric Company
Hill PHOENIX
Honeywell, Inc.
Hussmann Corporation
Hydro Aluminum Adrian, Inc.
Indiana Tube Corporation
Lennox International, Inc.
Modine Manufacturing Co.
Peerless of America, Inc.
The Trane Company
Thermo King Corporation
Visteon Automotive Systems
Whirlpool Corporation
York International, Inc.

For additional information:

Air Conditioning & Refrigeration Center
Mechanical & Industrial Engineering Dept.
University of Illinois
1206 West Green Street
Urbana IL 61801

217 333 3115

Abstract

A fault detection and diagnosis (FDD) method was used to detect and diagnose faults on both a refrigerator and an air conditioner during normal cycling operation. The objective of the method is to identify a set of sensors that can detect faults reliably before they severely hinder system performance. Unlike other methods, this one depends on the accuracy of a number of small, on-line linear models, each of which is valid over a limited range of operating conditions.

To detect N faults, N sensors are needed. Using M>N sensors can further reduce the risk of false positives. For both the refrigerator and air conditioner systems, about 1000 combinations of candidate sensor locations were examined. Through inspection of matrix condition numbers and each sensor's contribution to fault detection calculation, the highest quality sets of sensors were identified. The issue of detecting simultaneous multiple faults was also addressed, with varying success.

Fault detection was verified using both model simulations and experimental data. The results were similar, although in practice only one of the two would probably be used. Both load-type faults (such as door gasket leaks) and system faults were simulated on the refrigerator. It was found that system faults were generally more easily detectable than load faults.

Refrigerator experiments were performed on a typical household refrigerator because it was readily available in a laboratory, but the results of this project may be more immediately useful on larger commercial, industrial or transport refrigeration systems. Air conditioner experiments were performed on a 3-ton split system. Again, the economic benefits of this type of fault detection scheme may also be more feasible for larger field-assembled systems.

Table of Contents

List of Tables

List of Figures

Nomenclature

a	generic matrix element
COP	coefficient of performance
E	energy
f_z	refrigerator damper position (split-air fraction)
h	convection heat transfer coefficient
k	system parameter value
M	total number of sensors
N	total number of faults
P	pressure
RT	compressor run time
T	temperature
TA	air temperature
TXV	thermostatic expansion valve
u	uncertainty
UA	product of heat transer coefficient and area
W	power
x	system variable value
$\Delta\mathbf{x}$	residual vector
$\Delta\mathbf{k}$	vector of parameter values
$\mathbf{J}$	Jacobian matrix
$\mathbf{J}^{-1}$	inverse or pseudo-inverse Jacobian matrix
β	normally distributed uncertainty
δ	normally distributed uncertainty
Δ	change in a value
ϕ	constraint
κ	sensor error
ρ	density
σ	standard deviation
ψ	quantity to be minimized

Subscripts

air	air-side
base	base case value
calc	calculated value
Comp	compressor
CompIn	compressor inlet
cond	condenser
CondOut	condenser outlet
crit	critical value
Dis	discharge (compressor outlet)
EvapIn	evaporator inlet
EvapOut	evaporator outlet
exp	experimental value
fault	fault run value
ff	refrigerator fresh food compartment
frez	refrigerator freezer compartment
LiqLineOut	liquid line outlet
m	intermediate sensor number
model	simulation value
n	intermediate fault number
Shell	compressor shell
Suct	suction (compressor inlet)

Chapter 1

Introduction

1.1 Introduction

This study was motivated by the desire for an inexpensive, reliable method for detecting and diagnosing faults in refrigeration and air conditioning systems during normal cycling operation. The two systems are similar in that they are both vapor-compression cycles whose purpose is to lower the air temperature of a given enclosure. A major benefit of a fault diagnosis system would be the fact that many faults could be detected and repaired before any equipment damage would occur. This project will exploit the fact that these systems exhibit a highly repeatable quasi-steady condition near the end of their operating cycles.

The method presented here makes the assumption that whatever systems it is applied to are well-instrumented and make use of microprocessors. Some parameters require precise control and measurement. It is expected that a sophisticated FDD method would be most useful on high-quality components, in applications where benefits of early detection are greatest (e.g. in large chiller plants or in commercial, industrial, or transport refrigeration systems where faults may cause loss of valuable product). Although the method would be most cost-effective for these types of systems, experiments were performed on (well-instrumented) common household equipment because it was readily available. The method is general enough to be applied to other HVAC and/or refrigeration systems.

1.2 Objectives

The specific objectives of this project are as follows:

1) apply a model based fault detection and diagnosis method to extract as much information as possible from a small set of inexpensive sensors;
2) demonstrate through numerical simulation that this diagnostic method can accurately identify "simulated faults" (alone and in combination) and also minimize the chance of false positive diagnoses;
3) modify existing well-instrumented refrigerators and air conditioners to simulate these same faults; and

4) demonstrate the viability of this diagnostic method over a wide range of test conditions.

These objectives will be addressed in detail throughout the remainder of this report. Chapter 2 will introduce the fault detection and diagnosis (FDD) technique that this study proposes and will describe faults to be detected and sensors that may be used. Chapter 3 will investigate methods of choosing the best sensors for fault detection, in the interest of minimizing the number of required sensors. Finally, Chapter 4 will assess the quality of this FDD technique through both numerical and experimental results.

1.3 Literature review

A recent application of diagnostic techniques to stationary vapor compression air conditioners is reported in Rossi and Braun (1997) and was evaluated by Breuker and Braun (October 1998). Rossi and Braun proposed a method to detect and distinguish among five different faults in an air conditioner: refrigerant leakage, liquid line restriction, leaky compressor valves, fouled condenser coil, and dirty evaporator filter. Their method requires at most 10 measurements; nine temperatures and one humidity measurement. Two of the temperature measurements, the condenser and evaporator inlet (outdoor and indoor, respectively) air temperatures, along with evaporator inlet humidity define the driving states of the system. They define the system's operating condition. Three sensors are therefore needed to define normal operation. This leaves 7 sensors to detect 5 faults. The driving states are measured and used by a steady-state model to predict the other seven system temperatures. Fault detection is then based on differences (residuals) between those predicted temperatures and their measured values. The seven residuals define a "detection vector" in 7-D space. Rossi and Braun use a large nonlinear physical model to predict system performance, but Rossi (1995) determined that the size and iterative nature of the model would be "too numerically burdensome" for field applications. Therefore they used that model's output to develop a smaller empirical "black box" type model to predict the system variables of interest based on the measured driving states.

Diagnosis is performed by a rule-based diagnostic classifier that identifies a unique signature associated with each fault, meaning that each different fault is

represented by a unique detection vector direction in 7-D space. If the magnitude of the detection vector is significantly larger than measurement uncertainties, and if it points in one of the five predetermined fault vector directions, then that particular fault is statistically detectable and classifiable. Rossi and Braun do not address the possibility of multiple faults existing simultaneously. In a later publication, Breuker and Braun used this method proposed by Rossi and Braun to detect the same five faults, but they reduced the number of sensors needed to two inputs (specifying the operating condition) and five outputs (to detect five faults). They concluded that reasonable accuracy is maintained as long as humidity is retained as one of the driving state sensors.

Other researchers have addressed this problem also. Stylianou and Nikanpour (1996) diagnosed faults on a reciprocating chiller. They divided the chiller's operation into three distinct modes: Off-cycle, start-up, and steady-state operation. They used pattern-recognition techniques to diagnose faults that were observable during at least one of the three modes of operation. For example, sensor drift was detected during the off-cycle, inherently transient refrigerant flow control faults (ex. compressor floodback) were detected during start-up, and evaporator and condenser fouling, refrigerant leak, and flow restrictions were detected during steady-state operation. At steady-state, performance quality was calculated using expected COP as a function of condenser and evaporator temperatures. If COP was below its predicted value, performance was defined as faulty and six temperatures and two pressures were estimated using a linear regression model (based on training data), then residuals were calculated and matched using a rule-based method similar to that used by Rossi and Braun.

Grimmelius, et al. (1995) diagnosed faults affecting a compression refrigeration plant. They considered six faults and monitored 20 variables, although 9 of those showed no changes upon fault induction and could probably be discarded. They used a regression model based on previously measured data to predict system behavior and the differences between measured and expected sensor readings to compute residuals. Those residuals were then compared to a rule-based matrix similar to that used by both Rossi and Braun and Stylianou and Nikanpour. On-line FDD is accomplished by comparing sensor measurements to the rule-based matrix and assigning a score between 0 (no fault) and 1 (probable fault) to each possible failure mode based on how closely the sensor

measurements match each fault signature. In a later publication, Grimmelius, et al. (1999) state that a detailed refrigeration plant model is being developed (from first principles) for the express purpose of simulating faults. They state that most existing models cannot be used for FDD because faults cause off-design behavior, which is difficult to validate with manufacturer's data.

Wagner and Shoureshi (1992) review both a limit/trend checking and an innovation-based detection scheme to detect five faults. The limit and trend checking scheme is a model-free approach based, as the name implies, on monitoring system outputs and verifying that they remain within acceptable ranges. The ranges are typically chosen experimentally, and must be narrow enough to avoid major component damage but wide enough to avoid false alarms. The innovation-based (or residual-based) detection scheme predicts sensor readings using a simplified model based on thermodynamic/heat transfer principles and empirical data. The model operates on-line, and failures are indicated when the sum of normalized square innovations is larger than a predetermined threshold.

Researchers have recently tested FDD techniques on air-handling units as well. Ngo and Dexter (1999) compare measured data (from the cooling-coil subsystem of the unit) to 6 different generic fuzzy reference models obtained from simulation data. One model represents fault-free operation, the other five represent faults such as valve leaking, valve sticking, and fouling. They claim that due to sensor bias only large faults can be successfully detected in practice. Kärki and Karjalainen (1999) stipulate that the amount of instrumentation on the air-handling unit must not increase (no extra hardware cost). They do not report specific results, but review FDD methods such as monitoring heat recovery through the monitoring of exhaust air temperature, monitoring powers, process characteristic curves, and fault-symptom trees. Katipamula, et al. (1999) employ diagnostics based on rules derived from engineering models of proper and improper air-handler performance. The rules are implemented as a decision tree structure. They exhibit promising results from a prototype system in order to demonstrate its potential.

The proposed approach is most similar to that of Rossi and Braun, but theoretically it is more powerful. Both approaches depend on known (measured) driving states and both compare predicted states to sensor measurements. Their steady-state

model is constructed with empirical data, whereas our refrigerator and air conditioner models are large computer simulation models. Our underlying FDD tool is a very small linear model that requires only as many state measurements as faults to be detected. Unlike other methods reviewed above, our method contributes an algorithm for selecting the best sensor locations based solely on a mathematical analysis of their contributions to detection accuracy.

In terms similar to those used by Rossi and Braun, our algorithm ensures that the sensors chosen are those which ensure that the unique directions of the detection vectors are as close to orthogonal as possible. This tendency is valuable because if two fault vectors were nearly collinear, distinguishing between those two faults would be difficult. Our method also allows for the addition of extra sensors for extra accuracy. Those extra sensors may also be chosen analytically using the same sensor choice algorithm. Our method also proposes a normalization routine that may be used to determine the optimal time to alert a user of a particular fault (when system performance is sufficiently degraded). That is important because in some cases a fault may be statistically detectable and classifiable, but not severe enough to warrant the repair cost. At the heart of this method is a dependence on unique fault signatures, as in other methods, but sensors and alarm levels are chosen more rigorously. The possibility of diagnosing multiple faults occurring at the same time is also considered here. In theory, the mathematical method presented here will detect and diagnose multiple faults as effectively as single faults.

Chapter 2

Fault Detection and Diagnosis

2.1 FDD method

Data from experiments or simulation analyses can be used to formulate simple linearized vector equation system models of the following form:

$$\Delta \mathbf{x} = \mathbf{J}\, \Delta \mathbf{k} \qquad [2.1]$$

where

- $\Delta \mathbf{x}$ is the vector of M residuals under a particular set of operating conditions (e.g. deviation between measured and expected temperatures), where M = number of sensors used,
- $\Delta \mathbf{k}$ is the vector of changes in N process parameters (e.g. difference between design and actual air flow rate), where N = number of faults accounted for, and
- $\mathbf{J}$ is a matrix of partial derivatives (of $\mathbf{x}_m$ with respect to $\mathbf{k}_n$, called a Jacobian matrix) which models the linearized relationship between the residuals and the parameter changes at a particular set of operating conditions.

If the Jacobian matrix $\mathbf{J}$ is available for a given set of operating conditions, the changes in process parameters can be computed from:

$$\Delta \mathbf{k} = \mathbf{J}^{-1}\, \Delta \mathbf{x} \qquad [2.2]$$

That is, degradations of physical system characteristics $\mathbf{k}$ can be calculated and detected directly from measurements of monitored operating variables $\mathbf{x}$, provided one knows the inverse of the Jacobian matrix. That matrix could be obtained from a system and programmed into a system's controller. Of course, such a linearized model only produces accurate results over a limited range of operating conditions. Results have shown, though, that over the relatively small ranges of parameter degradation considered here, a small linear model provides reasonable accuracy. See Appendix B for further discussion of this linearity assumption and other details of Jacobian matrix construction.

The remainder of this study is dedicated to testing the viability of this proposed FDD method. It could be tested in numerous ways, including a purely empirical test or a test consisting only of model simulation data. The test method chosen here consists of

both simulation and experimental results. Model data is used to choose appropriate sensor sets and to deduce uncertainty ranges for fault detection using those sets, as well as to investigate Jacobian robustness, linearity, etc. Experimental data was then used mainly as a supplement, to verify whether model results are realistic and whether an empirical test would give similar results.

2.2 Simulated faults and candidate sensor locations

This section describes the faults that were simulated on a refrigeration system and an air conditioning system. The most obvious difference is the fact that the refrigerator has three reservoirs (two compartments and an outdoor room) while the air conditioner has only two (indoor and outdoor). This means that the air conditioning system is simpler to analyze, mainly because it doesn't utilize a damper to divide the air flowing over the evaporator. Refrigerator experiments were performed on a typical household refrigerator because it was readily available in a laboratory, but the results of this project may be more immediately useful on larger commercial, industrial or transport refrigeration systems. Air conditioner experiments were performed on a 3-ton split system. Again, the economic benefits of this type of fault detection scheme may also be more feasible for larger field-assembled systems.

In order to simulate the refrigerator and air conditioner systems, large nonlinear computer simulation models were used first to identify candidate sensor locations. Model runs were performed so that all values of potentially measurable variables are known for each system's base case (no faults present) and for all of their fault cases. By separately comparing each variable value seen in each fault run to the value of that same variable in the base case run, all possible changes in sensor readings (Δx_m) are now known. Woodall and Bullard (1996) list all the variables used by the refrigerator simulation model. Bridges, et al. (1995) list all the variables used by the a/c simulation model.

Each model makes use of more than 100 variables, but only a smaller subset of them may be realistically measured in a production unit with reasonable ease and cost. From the list of easily measurable variables, a list of candidate sensor locations was compiled by eliminating all of the x_m values that did not change significantly upon fault

induction. A "significant change" is defined as a change in a variable that is greater than that sensor's measurement uncertainty, to distinguish signal from noise. Measurement uncertainty (2σ) for sensors (regardless of whether they were used on the refrigerator or air conditioner) was assumed to be 0.5% for compressor RunTime fraction, 4.0 psia for pressure, 1.0°F for temperature, and 4.0 W for power.

2.2.1 Refrigerator fault simulation

The fault detection and diagnosis method relies on readings from a number of sensors at various locations throughout the refrigerator system refrigerant and air loops. Figure 2.1 below is a schematic diagram of a typical refrigerator loop.

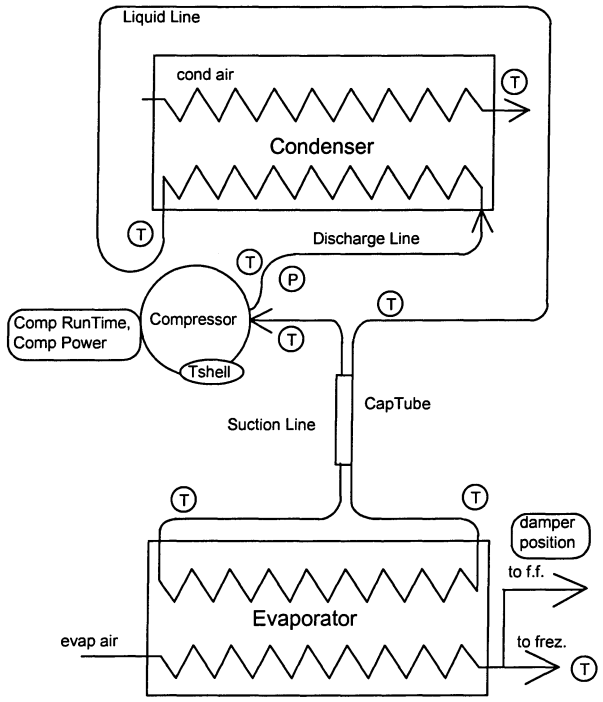

Figure 2.1 Schematic diagram of refrigerator model setup

A total of 13 locations were considered as candidates for sensor locations, based on inspection of simulation model results, and are shown in Figure 2.1. Table 2.1 below summarizes Figure 2.1 and lists the candidate sensor locations. The experimental test

unit used for this project was instrumented with all of the sensors listed in Table 2.1, with the exception of RunTime and evaporator air damper position.

Table 2.1 Candidate refrigerator sensor locations

Temperatures		Others
compressor suction	evaporator fan outlet	discharge pressure
compressor discharge	condenser outlet	compressor run time
compressor shell	evaporator inlet	system power
condenser fan outlet	evaporator outlet	evaporator air damper position
	liquid line outlet	

Using compartment air heaters, the experimental unit was forced to run in the "on" cycle 100% of the time with a fixed damper position. Compartment temperatures were held constant, and heater outputs were used to make an "offline" calculation of effective RunTime and damper position.

Eight refrigerator faults were simulated using the model. Each separate fault was assumed to be caused by a change in some fault-specific operation parameter k_n. The steady-state computer model simulations were performed as follows: all eight faults were simulated as in a real refrigerator for which cabinet loads were known. The resulting values of RunTime (<100%) and damper position required to match capacity to load were computed as variables. The total yearly energy use was then set to increase by an amount ΔE_{crit} (arbitrarily chosen as 5%) over the base case while a single parameter k_n was allowed to vary. With these results it can be seen exactly how much a parameter must be degraded (while all other parameters remain unchanged) before it causes the system to use 5% more energy. By performing this type of simulation for all eight faults, eight different Δk_n values are then known and are "equivalent," in that they all degrade the system's performance equally. Table 2.2 shows the eight simulated faults and their "critical" parameter changes.

The experimental facility differs from an actual operational system as described above, however. Therefore after the first eight model runs were completed, eight more runs were done holding RT equal to 100% and the evaporator damper position equal to 0.94 (the experimental baseline value found by Kelman and Bullard (1998)) while

compartment heater powers were allowed to vary. This second set of eight runs simulates the steady-state experiments conducted in the test facility. All eight faults were simulated using the magnitudes of parameter degradation (Δk_n) shown in Table 2.2. Most model analysis was done at the following conditions: ambient temperature = 75°F, freezer temperature = 5°F, fresh food compartment temperature = 45°F. This procedure

Table 2.2 Refrigerator faults simulated

Fault Simulated	Parameter Varied	% Change
clogged capillary tube	capillary tube exit area	-16%
worn compressor	compressor capacity scaling factor	-8%
low motor efficiency	compressor power map scaling factor	+6%
frosted evaporator	volumetric air flow rate	-20%
fouled condenser	air-side heat transfer coefficient	-21%
reduced condenser airflow	volumetric air flow rate	-50%
freezer gasket leak	freezer compartment UA coefficient	+7%
fresh food gasket leak	fresh food compartment UA coefficient	+20%

allows the results of these model simulations to be compared meaningfully to experimental sensor readings. Our choice of critical parameter values differs only in detail from Rossi and Braun, who measured fault magnitudes based on their level of detectability instead of selecting equivalent magnitudes. Our approach was designed to be generalized to the case of multiple faults as well as to avoid selecting potentially costly sensor locations that would not be particularly useful.

The parameters listed in Table 2.2 may be changed easily within the simulation model, but experimentally the faults are not as easy to simulate. In the laboratory it is often difficult to introduce a fault having exactly the target magnitude. This and other experimental issues are addressed in Appendix A.

2.2.2 Air conditioner fault simulation

A similar list of candidate sensor locations was compiled from air conditioner simulation runs. Figure 2.2 is a schematic diagram of a typical air conditioner loop. A total of 12 locations were considered as candidates for sensor locations, based on inspection of simulation model results, and are shown in Figure 2.2. The experimental

test unit used for this project was instrumented with all of the indicated sensors, with the exception of liquid line outlet temperature (directly upstream of the TXV).

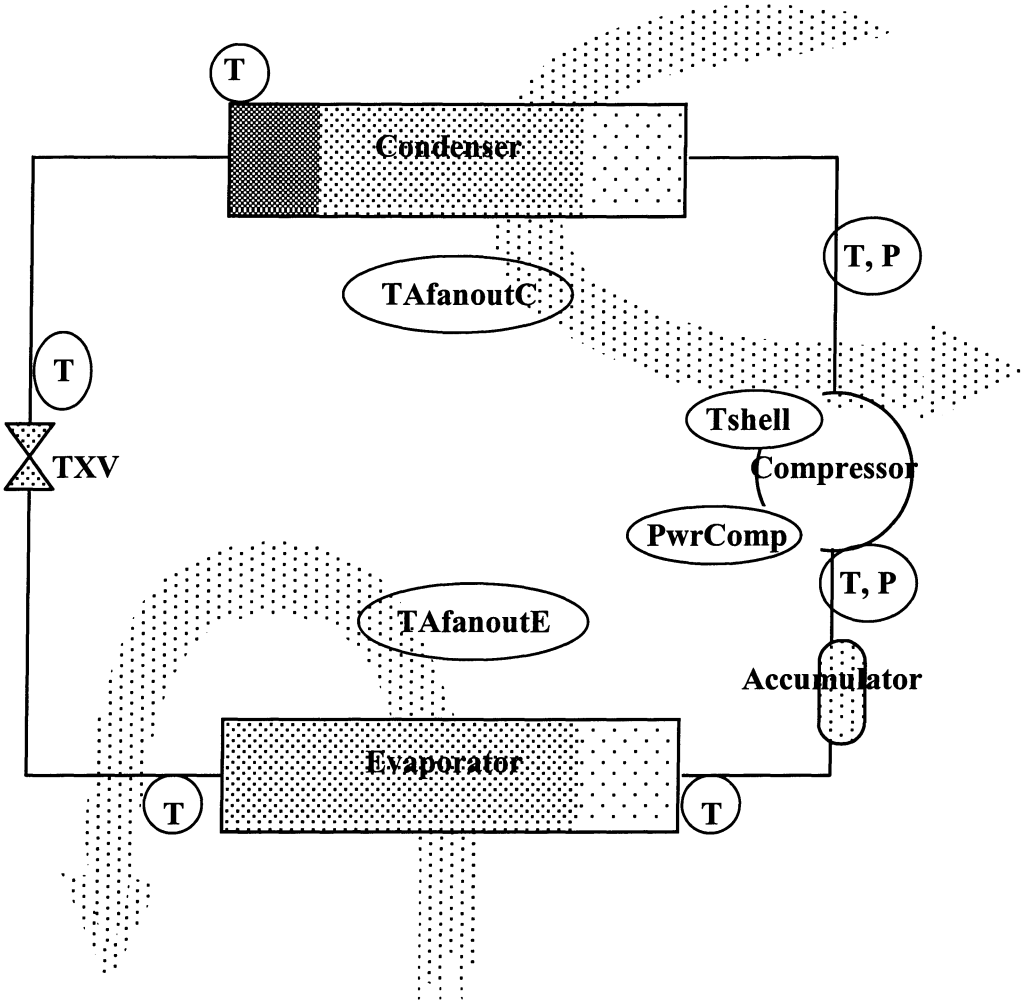

Figure 2.2 Schematic diagram of air conditioner model setup

Table 2.3 summarizes Figure 2.2 and lists the candidate sensor locations.

Table 2.3 Candidate a/c sensor locations

Temperatures		Others
compressor suction	evaporator fan outlet	discharge pressure
compressor discharge	condenser outlet	compressor suction pressure
compressor shell	evaporator inlet	compressor power
condenser fan outlet	evaporator outlet	
	liquid line outlet	

Six air conditioner faults were simulated. Model runs were performed in a manner similar to the refrigerator simulations described in Section 2.2.1. Most simulation analysis was done at the following conditions: outdoor temperature = 95°F, indoor temperature = 80°F, indoor relative humidity = 50%. Unlike the case of a refrigerator compartment where cooling loads are known, they are typically unknown and highly variable for a building. Therefore both the simulation and experiments were conducted at steady state, so RunTime is not available as a candidate sensor location. Indoor and outdoor temperatures and indoor humidity were held constant through the use of heaters and steam addition. Instead of specifying total yearly energy use as the critical performance variable, COP was set to decrease by an amount ΔCOP_{crit} (again, arbitrarily chosen as 5%) over the base case while a single parameter k_n was allowed to vary. By performing this type of simulation for all six faults, six different Δk_n values are then known and are "equivalent," in that they all degrade the system's performance equally. Table 2.4 below shows the six simulated faults and their "critical" parameter changes.

Table 2.4 Air conditioner faults simulated

Fault Simulated	Parameter Varied	% Change
reduced evaporator airflow	volumetric air flow rate	-62%
reduced condenser airflow	volumetric air flow rate	-21%
fouled condenser	air-side heat transfer coefficient	-52%
compressor leak	% discharge gas flowing as normal	-7%
low motor efficiency	compressor power scale factor	+7%
system undercharged	total refrigerant mass	-9%

Two of the simulated faults, compressor leak and low motor efficiency, could not be simulated using the original version of the model, so a few additional equations were

added as documented in Appendix F. As mentioned in Section 2.2.1, parameters may be changed easily within the simulation model, but experimentally the faults are not as easy to simulate. This and other experimental issues are addressed in Appendix A.

Note that the critical value for evaporator air flow rate (-62%) is quite large. In fact, intuitively it seems much larger than should be necessary to decrease COP by 5%. Figure 2.3 shows a plot of simulated COP vs. evaporator air flow. Apparently the base case evaporator air flow rate was greater than optimal, so the initial reduction actually increased COP. This demonstrates a possible shortcoming of any FDD algorithm that expects monotonic or linear effects of faults on system performance.

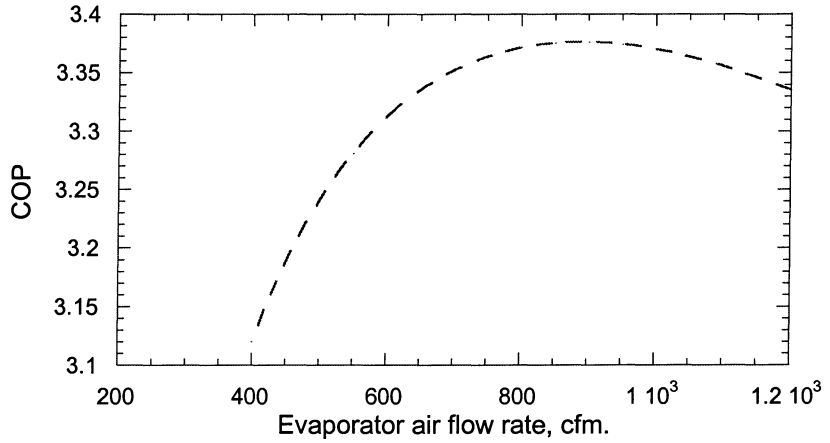

Figure 2.3 Effect of decreased evaporator air flow on simulated COP

The plot shows that a decrease in air flow causes the COP to increase to a point, due to the fact that initially the system power requirement (specifically, evaporator fan power) drops more quickly than evaporator capacity.

Also as evaporator air flow rate is reduced the evaporating temperature falls as the exit air temperature approaches the fin surface temperature. Beyond this point any further reduction in the evaporator air flow rate reduces system capacity, driving down evaporating temperature which in turn reduces compressor mass flow rate. However compressor power drops too, almost proportionally, so the reduction in COP is rather small. At extremely low air flow rates, the simulation model may not be accurate.

Chapter 3

Choosing the Best Sensor Locations

3.1 Introduction

Chapter 2 introduced the fact that there are more possible sensor locations than included faults for both the refrigerator and air conditioning systems. The first objective of this project, introduced in Section 1.2, states the need for a *small* set of sensors. In the interest of cost minimization it would be beneficial to use as few sensors as possible in diagnosing a given number of faults. A set of sensors will be chosen so as to preserve the ability to detect parameter changes and suppress the effect of sensor measurement errors. Note that if the number of sensors (M) does not equal the number of faults (N) then a true matrix inverse, introduced in equation [2.2], cannot be calculated. If the number of sensors is greater than the number of faults included in determining the Jacobian matrix (M>N) then the system is over-specified, but a pseudo-inverse matrix may be calculated so that equation [2.2] is still exact. Strang (1993) gives an explanation of the mathematics involved in calculating a pseudo-inverse. If the number of sensors is less than the number of faults (M<N) a pseudo-inverse may still be calculated, but unfortunately numerical results have shown that in this case the result of equation [2.2] is inconclusive for at least one fault. Hence it appears that if this particular fault diagnosis method is to be used, the number of faults (N) is the lower limit for the number of sensors (M). For the moment it will be assumed that the absolute minimum number of sensors are desired, so the following sections will describe two methods of choosing sensor sets such that M=N.

3.2 Method 1: Condition number

An exhaustive search was performed for both the refrigerator and air conditioner in pursuit of the set of M(=N) sensors that would most reliably diagnose N faults. "Exhaustive search" means that a number (>N) of sensors were considered, and every possible combination of M=N was analyzed. In an attempt to quantify and rank the relative "quality" of each set of sensors considered, a singular value decomposition technique was used. When the decomposition of each square Jacobian matrix was

performed, N "singular values" were computed. These singular values are always greater than or equal to zero. The ratio of the largest singular value to the smallest defines the "condition number" of the matrix. If the smallest singular value is equal to zero, then the condition number is infinite and that matrix is singular, meaning that it has no solution vector.

The condition number of a Jacobian gives a measure of the independence of the included sensors. In the case where M=N (square matrix), as two rows become closer to being multiples of each other, the condition number approaches infinity and the matrix becomes singular (no solution). The condition number of a Jacobian containing unrelated sensors is smaller than that of a Jacobian containing two closely related sensors (compressor discharge and condenser inlet temperatures, for example). Note that the condition number also gives a general indication of how measurement errors in the $\Delta\mathbf{x}$ vector will affect the calculation of the $\Delta\mathbf{k}$ vector in equation [2.2]. Dongarra, et al. (1979) give a thorough and useful description of the condition number and singular value decomposition of a matrix.

3.2.1 Refrigerator results

As described in Section 2.2.1, the refrigerator has 13 possible sensor locations and 8 faults to be detected. Therefore model results (at the following conditions: ambient temp. = 75°F, freezer temp. = 5°F, fresh food compartment temp. = 45°F) were used to select the "best" sets of 8 sensors out of 13 candidates based on the condition numbers of the resulting Jacobian matrices. Eight faults were included in the analysis, therefore eight sensors were considered, resulting in a square (8 × 8) Jacobian. There were more than 1200 possible sensor sets, each with its own Jacobian. Figure 3.1 below shows a distribution of 215 sensor sets having condition numbers less than 1000. The figure shows that in the range of 0-50 (which is the lowest, therefore possibly the best) there are 20 different sensor combinations, which means there is some flexibility in choosing sensor locations.

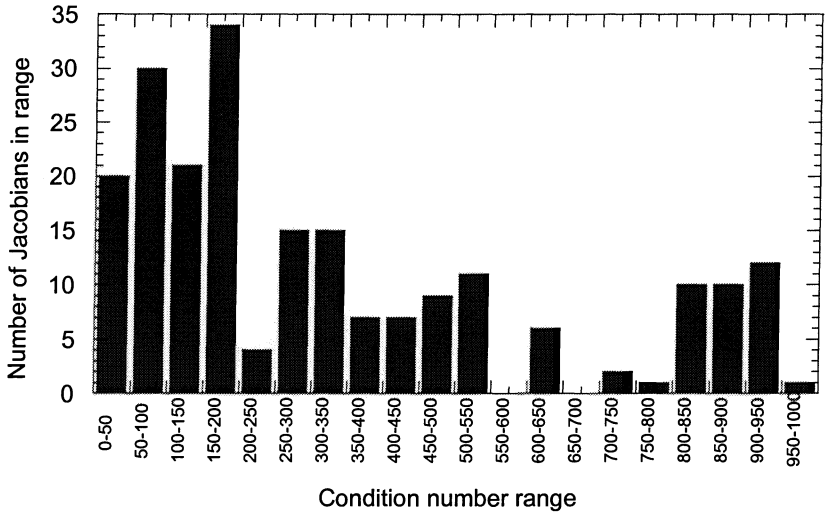

Figure 3.1 The 215 best conditioned sensor sets (refrig)

The lowest condition number available is approximately 36. Table 3.1 below includes the 20 best sensor sets, based solely on Jacobian condition number.

Table 3.1 Simulation sensor sets, ordered by condition number (refrig)

Cond. #	W_{Comp}	P_{Dis}	T_{Shell}	T_{Dis}	$T_{CondOut}$	$T_{LiqlineOut}$	T_{EvapIn}	$T_{EvapOut}$	T_{CompIn}	$TA_{CondOut}$	$TA_{EvapOut}$	RT	f_z
35.8			X		X			X	X	X	X	X	X
35.8				X	X			X	X	X	X	X	X
36.0			X			X		X	X	X	X	X	X
36.0				X		X		X	X	X	X	X	X
36.9	X				X			X	X	X	X	X	X
37.1	X					X		X	X	X	X	X	X
37.2			X		X		X	X	X	X		X	X
37.2				X	X		X	X	X	X		X	X
37.4			X			X	X	X	X	X		X	X
37.4				X		X	X	X	X	X		X	X
38.3	X				X		X	X	X	X		X	X
38.5	X					X	X	X	X	X		X	X
42.8	X			X	X			X	X		X	X	X
43.1	X		X		X			X	X		X	X	X
43.1	X			X		X		X	X		X	X	X
43.3	X		X			X		X	X		X	X	X
44.4	X			X	X		X	X	X			X	X
44.6	X			X		X	X	X	X			X	X
44.6	X		X		X		X	X	X			X	X
44.8	X		X			X	X	X	X			X	X

The results of this analysis indicate that there are numerous sensor sets that may give diagnoses of similar quality. Diagnosis quality is specifically addressed later in Chapter 3. Note that two sensors, RunTime and f_z (damper position), are included in every sensor set. The reason that they appear in every set can be explained by the fact that two of the faults included in the set of eight affect only 2 of the 8 sensors. Two of the faults that were simulated with the model, fresh food gasket leak and freezer gasket leak, are "load faults," which have no effect on the system operating conditions (i.e. cabinet and refrigerant temperatures) and therefore no effect on most sensors. The only two sensors that are affected by load faults are the compressor RunTime and damper position, therefore those two sensors must be present in the final set if load faults are to be detected. Requiring that those two sensors be present eliminated more than 700 possible sensor sets (they had infinite condition numbers), leaving approximately 460 candidate sets remaining. Appendix D illustrates mathematically this distinction between the two types of faults.

Model results indicate that the damper position does not change more than 1%, even for load faults, but even this small change can make a difference in fresh food compartment cooling. It is assumed here that the damper would be electronically controlled in an actual unit, thus making it possible to detect these small changes.

There are some other trends that can be observed in Table 3.1. Note that the evaporator outlet temperature and compressor inlet temperature are included in each of the top 20 sets. They are not mathematically necessary, like RT and f_z are, but just happen to be present in all of these well-conditioned Jacobians. An explanation is that the suction line heat exchanger is located between the evaporator outlet and the compressor inlet, so the difference in those two temperatures gives an indication of how efficiently the heat exchanger is working, which is related to the mass flow rate of refrigerant through it. That information is useful in detecting faults such as a clogged capillary tube and a worn compressor (both reduce mass flow).

It is also apparent that either the evaporator air or refrigerant outlet temperature must be included. Both are indicators of evaporator capacity and are useful in detecting faults such as frost on the evaporator or a worn compressor. Condenser faults are indicated by either the condenser or liquid line outlet temperature, one of which is present

in every set shown in Table 3.1. Those two sensor locations are separated only by the section of the liquid line used to warm the door flange areas, so the difference between them is nearly constant.

There is an interesting relationship among the following four sensors: compressor power, compressor shell temperature, discharge temperature, and condenser air outlet temperature. Two of the four are always present. The shell and discharge temperatures are never in the same set because the shell temperature is related almost linearly to the discharge temperature for many refrigerator and air conditioner compressors. This has been demonstrated by Cavallaro and Bullard (1995) and Mullen et al. (1998). All four of the sensors detect inefficient operation, either directly as compressor power input or indirectly via the increased amount of waste heat rejected. Every sensor set in the top half of Table 3.1 includes the condenser air outlet temperature, so it is apparently a good sensor to use, and a potentially inexpensive substitute for compressor power measurement.

3.2.2 Air conditioner results

As described in Section 2.2.2, the air conditioner has 12 possible sensor locations and 6 faults to be detected. Therefore model results (at the standard industry rating condition: indoor temp. = 80°F, outdoor temp. = 95°F, indoor RH = 50%) were used to select the "best" sets of 6 sensors out of 12 candidates based on the condition numbers of the resulting Jacobian matrices. Six faults were included in the analysis, therefore six sensors were considered, resulting in a square (6 × 6) Jacobian. There were more than 900 possible sensor sets, each with its own Jacobian. Figure 3.2 below shows 267 sensor sets having condition numbers less than 1000. The figure shows that the majority of the sensor sets fall toward the lower end of the chart, which again means there is flexibility in choosing sensor locations.

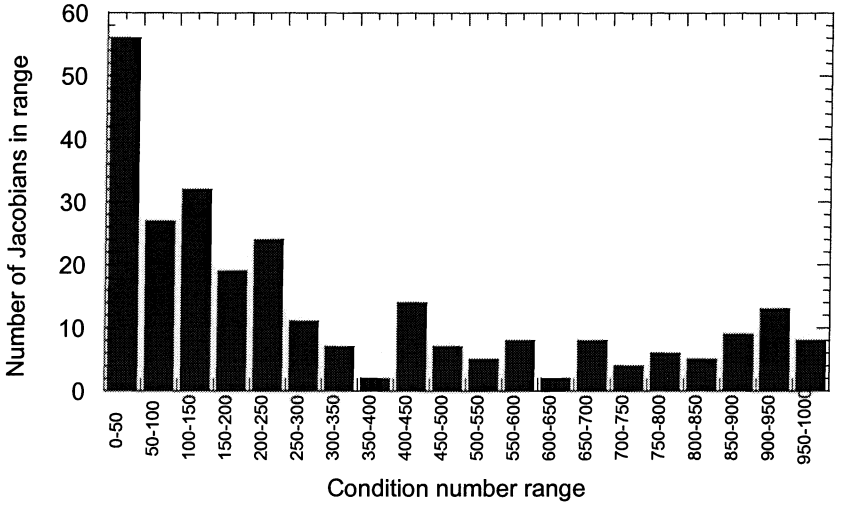

Figure 3.2 The 267 best conditioned sensor sets (a/c)

The lowest condition number available is approximately 9. Table 3.2 below includes the 20 best sensor sets, based solely on Jacobian condition number.

Table 3.2 Simulation sensor sets, ordered by condition number (a/c)

Cond. #	W_{Comp}	P_{Dis}	P_{Suct}	T_{Dis}	$T_{CondOut}$	$T_{LiqlineOut}$	T_{EvapIn}	$T_{EvapOut}$	T_{CompIn}	$TA_{CondOut}$	$TA_{EvapOut}$	T_{Shell}
9.3		X				X			X	X	X	X
12.1		X				X	X		X	X		X
12.3		X				X		X	X	X		X
12.8		X			X				X	X	X	X
14.3		X		X		X			X	X	X	
15.9		X		X	X				X	X	X	
17.2		X					X		X	X	X	X
17.4		X			X		X		X	X		X
17.5		X			X			X	X	X		X
17.7		X	X			X				X	X	X
18.8		X		X		X	X		X	X		
19.2		X		X		X		X	X	X		
19.9		X	X			X			X	X		X
19.9		X				X		X		X	X	X
21.4		X				X	X			X	X	X
21.4		X		X	X		X		X	X		
21.6		X		X	X			X	X	X		
22.3		X						X	X	X	X	X
22.5		X	X		X					X	X	X
24.1		X			X			X		X	X	X

Unlike the refrigerator, no load faults were simulated in the air conditioner case, but there are still trends that can be observed in Table 3.2. Note that in the a/c case it is not necessary to have both the evaporator outlet and compressor inlet temperatures in the same sensor set, as there is no suction line heat exchanger as in a refrigerator. The compressor discharge pressure and condenser air outlet temperature, however, are included in each of the top 20 sets, while the compressor power appears in none of the top 20.

As in the refrigerator case, the shell and discharge temperatures do not appear in the same set because the shell temperature is related nearly linearly to the discharge temperature for many refrigerator and air conditioner compressors, as mentioned in Section 3.2.1.

At least one of three temperatures (evaporator inlet, outlet, or air outlet) appear in each set with the exception of one with condition number = 19.9. This is one of the few sets in which compressor suction pressure is included, which is nearly the same as evaporating pressure and an indicator of evaporator performance. Both condenser outlet and liquid line outlet temperature never appear together in any of the sensor sets, but surprisingly there are a couple sets in which neither appear (condition number = 17.2 and 22.3). In these sets two of the three aforementioned evaporator sensors appear, so apparently the inclusion of more information about the evaporator indirectly assists in diagnosing condenser faults.

3.2.3 Detection accuracy example

The following example illustrates how error propagation is related to the Jacobian's condition number. Figure 3.3 is an uncertainty distribution on the calculation of a single Δk element (in this case, a clogged capillary tube in the refrigerator where the captube exit area was reduced by 16%) using equation [2.2]. Both the Jacobian matrix and the $\Delta \mathbf{x}$ vector were assumed to have random errors associated with them. See Appendix C for a complete discussion of these errors and of uncertainty in the diagnostic method. Two different sensor sets, resulting in Jacobian matrices with two different condition numbers, were used in the following example. The sets are listed in Table 3.3.

Table 3.3 Two sensor sets used in Figure 3.3

Set 1/460: Cond. # = 35	Set 46/460: Cond. # = 77
T_{Shell}	W_{Comp}
$T_{CondOut}$	P_{Dis}
$T_{EvapOut}$	T_{Shell}
T_{CompIn}	T_{EvapIn}
$TA_{CondOut}$	$TA_{CondOut}$
$TA_{EvapOut}$	$TA_{EvapOut}$
RunTime	RunTime
damper position	damper position

Figure 3.3 below compares the **Δk** calculation (reduction of captube exit area, in this case) using sets #1 and #46. Note that the distribution of set #1, with its lower condition number of 36, is not as wide as that of set #46. There is less uncertainty in **Δk** when sensor set #1 is used.

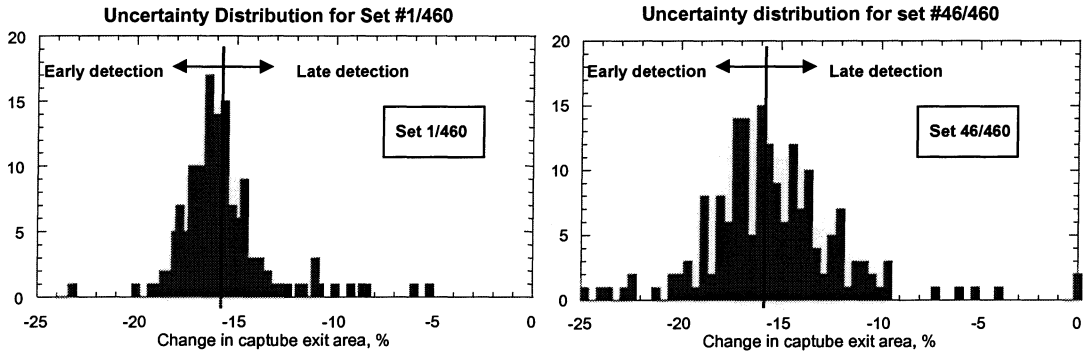

Figure 3.3 Δk uncertainty distributions using sets #1 and #46

Both sensor sets result in distributions that are centered about the correct Δk value of -16% (this value was introduced in Table 2.2), but the uncertainty ranges are different. The widening distribution shown in the figure illustrates the need for a Jacobian with a lower condition number. If any particular calculated Δk_n uncertainty distribution were to include a value of 0% on the x-axis (indicating no parameter change), then a fault causing a 5% loss of energy efficiency could possibly go completely undetected. The area to the right of the correct Δk value is denoted as "late detection," indicating that calculated Δk values lying closer to zero than their actual value (the threshold value of -16% in the case

of Figure 3.3) mask the reality that the actual Δk value may exceed the threshold value before the threshold value is actually calculated, hence detection is late. Conversely, the width of the distribution lying left of the true fault magnitude indicates the potential for "false positives." This argument makes it clear that a narrow, tall distribution is much more desirable than a wide one.

3.3 Method 2: Sensor contributions

It has been established that a Jacobian having the lowest condition number will choose sensors that are most independent of one another, which is also an attempt to suppress the effects of measurement uncertainty. Another way to minimize the propagation of uncertainty due to any one sensor is to minimize the importance of that sensor's signal contribution. Another criterion for sensor selection was devised with this purpose in mind. Consider equation [3.1] below. It is an expanded form of equation [2.2]. When a Δk_n value is calculated, it consists of the sum of M terms:

$$\Delta k_1 = \frac{\partial k_1}{\partial x_1}\Delta x_1 + \frac{\partial k_1}{\partial x_2}\Delta x_2 + \cdots + \frac{\partial k_1}{\partial x_M}\Delta x_M \qquad [3.1]$$

Equation [3.1] suggests that a certain sensor may contribute more information to the detection of one fault than another. It is apparent, then, that the strongest sensor location for a given fault is that whose product $(\partial k_n/\partial x_m)\,\Delta x_m$ contributes most significantly to the Δk_n sum for that fault. For example, the condenser outlet temperature is expected to be a better indicator of a fouled condenser than would some other sensor located on a different component, such as the compressor inlet temperature. If this is the case, then the product in equation [3.1] involving Δ(condenser outlet temp.) would indeed be greater than the product involving Δ(compressor inlet temp.). This fact allows us to consider possibilities concerning a fault's independence from some sensors and its dependence on others. Consider these extreme cases:

a) A fault's detection is completely dependent on one sensor, meaning that only one of the products in equation [3.1] is nonzero.

b) A fault's detection depends equally on every included sensor, meaning that every product in equation [3.1] is equal.

Case (a) may at first glance appear to be the better choice. It is simpler to understand – one sensor goes with each fault. However, from a more conservative perspective, case (b) is actually more desirable because the detection process is not dominated by any one sensor. Additionally, if one sensor were to malfunction in case (b) and contribute an inaccurate term to the calculation it would not affect the final $\Delta\mathbf{k}$ estimate as much as if it were the only sensor that mattered. For example, if a sensor were to fail in case (a), the system would either indicate a false positive or it would never indicate a fault even when one was present, depending on the nature of the failure. In case (b) the diagnosis method would be weakened, but perhaps not disabled because only (100/M)% of the Δk calculation would be inaccurate. Consider a simple example:

Suppose there is a 2×2 inverse Jacobian used with two sensors to detect two faults:

$$\begin{bmatrix} J^{-1}_{1,1} & J^{-1}_{1,2} \\ J^{-1}_{2,1} & J^{-1}_{2,2} \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta x \end{bmatrix} = \begin{bmatrix} \Delta k_1 \\ \Delta k_2 \end{bmatrix} \qquad [3.2]$$

Suppose there are three different sensors being considered for the $\Delta\mathbf{x}$ vector. The first fault k_1 causes the first two sensors x_A and x_B to rise by 5°F. The third sensor x_C does not change. First consider the use of sensors x_A and x_C in the $\Delta\mathbf{x}$ vector. The equation used to calculate the first fault is

$$\Delta k_1 = J^{-1}_{1,1}\,\Delta x_A + J^{-1}_{1,2}\,\Delta x_C = 1.0 \qquad [3.3]$$

But $J^{-1}_{1,2}\,\Delta x_C = 0$, so $J^{-1}_{1,1} = 1/5$. Now suppose sensor one were to fail in such a way that the variable Δx_A was being read as $(\Delta x_A + \kappa)$. Since Δk_1 is completely dependent on Δx_1 (meaning that $J^{-1}_{1,2}\,\Delta x_C = 0$) then Δk_1 is calculated as

$$\Delta k_1 = J^{-1}_{1,1}\,(\Delta x_1 + \kappa) \qquad [3.4]$$

and the error in Δk_1 is $(J^{-1}_{1,1}\,\kappa)$, or $(\kappa/5)$. Now suppose sensors x_A and x_B are used instead, so by equation [3.3], $J^{-1}_{1,1}\,\Delta x_A = J^{-1}_{1,2}\,\Delta x_B$ and $J^{-1}_{1,1} = J^{-1}_{1,2} = 1/10$. If the same sensor failure occurs, then Δk_1 is calculated as

$$\Delta k_1 = J^{-1}_{1,1}\,(\Delta x_A + \kappa) + J^{-1}_{1,2}\,\Delta x_B \qquad [3.5]$$

and the error in Δk_1 is $(J^{-1}_{1,1}\,\kappa)$, or $(\kappa/10)$. The error is now half as large as in the first case. By similar logic, as more sensors are used and M becomes larger, this type of error in the Δk calculations becomes smaller. A sensor failure directly affects the $\Delta\mathbf{x}$ vector, but the extent to which it affects the Δk calculation depends on the element it is multiplied by.

It is apparent, then, that matrices with equal sensor contributions are desired over matrices with independent sensor contributions. The following steps outline a method of quantifying this concept.

Each product that appears in equation [3.1] represents an individual sensor's contribution to the total Δk calculation. Therefore the contribution of sensor n to fault m can be quantified as:

$$\frac{\left(\frac{\partial k_m}{\partial x_n}\Delta x_n\right)}{\Delta k_m}\times 100\% \qquad [3.6]$$

As discussed earlier, the ideal scenario is where each sensor has the same percent contribution to each fault as every other sensor. However in such a complex system that is an unlikely case. Therefore it is desirable to know how "close" a sensor set is to this optimal condition. The following RMS "sum-of-squares" method is proposed:

1) First calculate each sensor's contribution to each fault using equation [3.6]. As an example, this was done for a set of sensors in Table 3.4 below. The Table shows the percent contribution of each sensor to each refrigerator fault. Each row in the table represents a single fault, and each column shows a single sensor's contribution to each of the 8 faults. The sensor set used here is the first one listed in Table 3.1 earlier (the refrigerator Jacobian with the best condition number).

Table 3.4 Sensor contributions (%) for Jacobian condition number=35.8

fault	T_{Shell}	$T_{CondOut}$	$T_{EvapOut}$	T_{CompIn}	$TA_{CondOut}$	$TA_{EvapOut}$	RT	f_z
captube clog	0	4	87	21	0	-11	0	0
worn comp	0	-16	25	52	0	38	0	0
low motor eff.	104	-1	0	1	-4	0	0	0
frosted evap	0	-14	-20	61	0	74	0	0
fouled cond.	1	121	7	-29	1	-1	0	0
low cond air flow	-5	7	2	-5	102	0	0	0
ff gasket leak	0	0	0	0	0	0	27	73
fz gasket leak	0	0	0	0	0	0	73	27

Note that each row sum is equal to 100 (except for rounding error) because the sensor contributions to fault detection must sum to 100%. Figure 3.4 graphically

illustrates Table 3.4 in the case of a clogged capillary tube. Ideally each sensor would contribute equally, but realistically this is not the case.

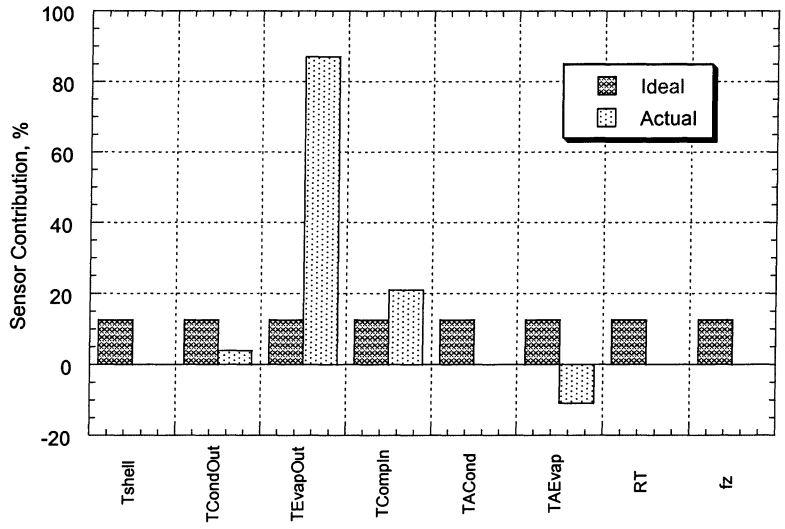

Figure 3.4 Sensor contributions to clogged captube detection

2) Using the resulting matrix from part 1), calculate the square root of the sum of the squares of all 64 elements:

$$RMS = \sqrt{\sum_{m=1}^{M}\sum_{n=1}^{N}(element_{n,m})^2} \qquad [3.7]$$

The minimum value that this function can have is 1, in the case where M=N and each element represents a sensor's % contribution to a parameter calculation. This will only occur when all of the elements are equal, meaning that each sensor is contributing equally to each fault. Appendix E shows a proof of this statement.

3.3.1 Refrigerator results

A calculation of the RMS value of Table 3.4 results in a value of 2.68. This is somewhat greater than 1, so obviously the sensors are not contributing equally. Another exhaustive search was performed on all of the possible Jacobians, this time in search of the lowest RMS values. Table 3.5 below shows the best 20 sensor sets in terms of lowest RMS values.

Table 3.5 Simulation sensor sets, ordered by RMS value (refrig)

RMS	Cond	W_{Comp}	P_{Dis}	T_{Shell}	T_{Dis}	$T_{CondOut}$	$T_{LiqlineOut}$	T_{EvapIn}	$T_{EvapOut}$	T_{CompIn}	$TA_{CondOut}$	$TA_{EvapOut}$	RT	f_z
2.50	53.1	X		X		X				X	X	X	X	X
2.50	51.9	X			X	X				X	X	X	X	X
2.50	53.6	X		X			X			X	X	X	X	X
2.50	52.3	X			X		X			X	X	X	X	X
2.57	54	X	X		X					X	X	X	X	X
2.57	55.3	X	X	X						X	X	X	X	X
2.59	38.3	X				X		X	X	X	X		X	X
2.59	38.5	X					X	X	X	X	X		X	X
2.62	37.2			X		X		X	X	X	X		X	X
2.62	37.2				X	X		X	X	X	X		X	X
2.62	37.4			X			X	X	X	X	X		X	X
2.62	37.4				X		X	X	X	X	X		X	X
2.64	37.1	X					X		X	X	X	X	X	X
2.65	36.9	X				X			X	X	X	X	X	X
2.67	57	X	X					X	X	X	X		X	X
2.67	36			X			X		X	X	X	X	X	X
2.67	36				X		X		X	X	X	X	X	X
2.68	35.8			X		X			X	X	X	X	X	X
2.68	35.8				X	X			X	X	X	X	X	X
2.69	55.6		X	X				X	X	X	X		X	X

All of the sensor sets in Table 3.5 have fairly low condition numbers, so there seems to be good agreement between the two sensor selection methods.

3.3.2 Air conditioner results

For the air conditioner case also, another exhaustive search was performed on all of the possible Jacobians in search of the lowest RMS values. Table 3.6 shows 22 sensor sets whose Jacobians have RMS values less than 2.5. The results for the air conditioner case don't show the same degree of agreement between condition number and RMS value as did the refrigerator case. There seems to be a need for closer inspection of these two methods.

Table 3.6 Simulation sensor sets, ordered by RMS value (a/c)

RMS	Cond	W_{Comp}	P_{Dis}	P_{Suct}	T_{Dis}	$T_{CondOut}$	$T_{LiqlineOut}$	T_{EvapIn}	$T_{EvapOut}$	T_{CompIn}	$TA_{CondOut}$	$TA_{EvapOut}$	T_{Shell}
2.08	106.0	X	X		X		X			X	X		
2.08	106.2	X	X				X			X	X		X
2.10	105.9	X	X	X	X		X				X		
2.10	106.2	X	X	X			X				X		X
2.10	108.6	X	X		X	X				X	X		
2.10	108.9	X	X			X				X	X		X
2.11	105.9	X	X		X		X		X		X		
2.11	106.2	X	X				X		X		X		X
2.12	108.4	X	X	X	X	X					X		
2.12	105.9	X	X		X		X	X			X		
2.12	108.8	X	X	X		X					X		X
2.12	106.2	X	X				X	X			X		X
2.12	105.8	X	X		X		X				X	X	
2.13	106.2	X	X				X				X	X	X
2.14	108.2	X	X		X	X			X		X		
2.14	108.7	X	X			X			X		X		X
2.14	108.3	X	X		X	X		X			X		
2.14	108.7	X	X			X		X			X		X
2.17	108.1	X	X		X	X					X	X	
2.17	108.6	X	X			X					X	X	X
2.48	120.1	X	X				X			X	X	X	
2.49	122.9	X	X			X				X	X	X	

3.4 Condition number vs. RMS comparison

Two methods of choosing sensors have been presented in Sections 3.2 and 3.3. They are *not* interchangeable, as we have seen (the methods will not conclude that the same sets are best). The condition number applies only to the Jacobian matrix itself, and chooses sensors that are most independent of each other. The RMS value of a matrix measures the products of the Jacobian and the $\Delta\mathbf{x}_m$ residual vectors. It strives to minimize the effect of any sensor error by minimizing the importance of any single sensor. Since one of the methods depends only on **J** and the other depends on both **J** and $\Delta\mathbf{x}_m$, a particular sensor set could possibly have a small condition number and a large RMS value, or vice versa.

3.4.1 Refrigerator

An early priority is to investigate the agreement between these two methods. Figure 3.5a below is a plot of the 215 "best" refrigerator sensor sets (first introduced in Figure 3.1) that shows how the RMS value varies with condition number. The plot shows that the relationship between the two is not linear, but there is a definite trend of the best conditioned matrices having the best RMS values.

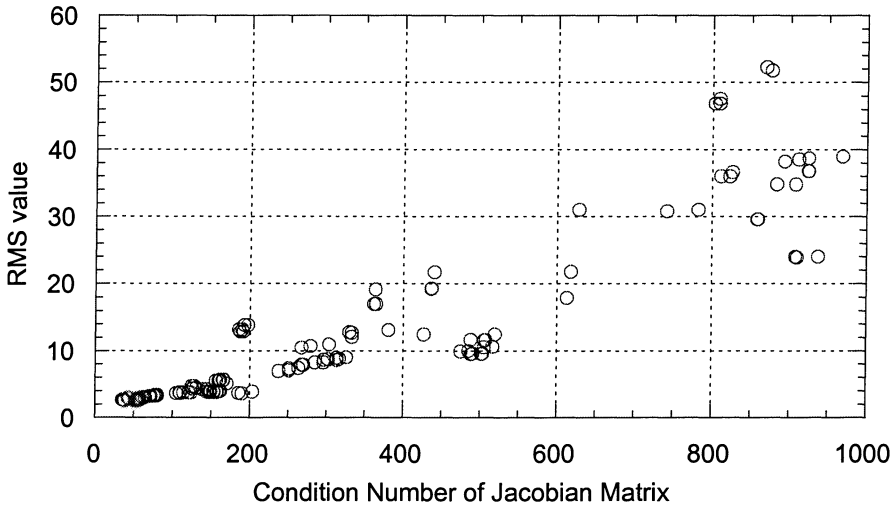

Figure 3.5a RMS value vs. condition number comparison (refrig, I)

Figure 3.5b below is simply an enlarged view of the lower left portion of Figure 3.5a, showing only the matrices with condition numbers less than 180. The same trend is still apparent, although it is obvious that matrices with equivalent condition numbers may have different RMS values, and vice versa.

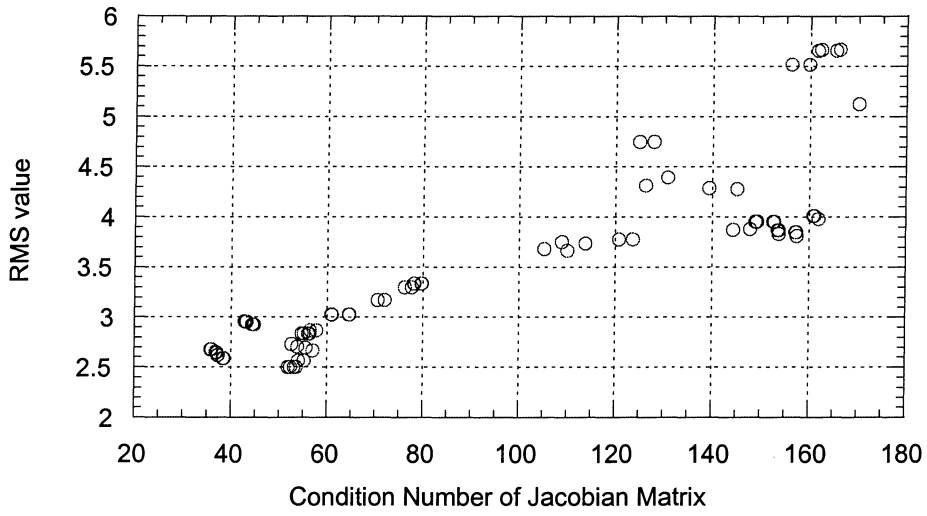

Figure 3.5b RMS value vs. condition number comparison (refrig, II)

These two figures would lead one to believe that there is not a significant trade-off between these two methods of sensor choice. However, the reader will see that the air conditioner results shown in Section 3.4.2 are not as well-behaved.

3.4.2 Air conditioner

Proceeding in a manner similar to that of Section 3.4.1, Figure 3.6a below is a plot of the 267 "best" sensor sets (first introduced in Figure 3.2) that shows how RMS value varies with condition number. The trend seen in Figures 3.5 is not at all apparent here. This is evidence that a matrix with a large condition number can have a small RMS value.

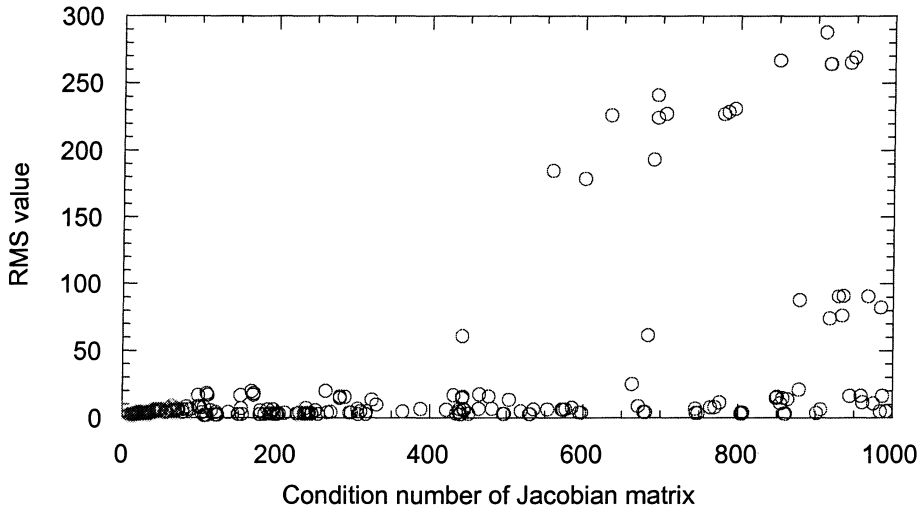

Figure 3.6a RMS value vs. condition number comparison (a/c, I)

Figure 3.6b below is simply an enlarged view of the lower left portion of Figure 3.6a, showing only the matrices with condition numbers less than 200. It appears that some matrices with small condition numbers (<50) do in fact show good RMS values (app. 2.5), but some show RMS values 3× greater than matrices with larger condition numbers (>100).

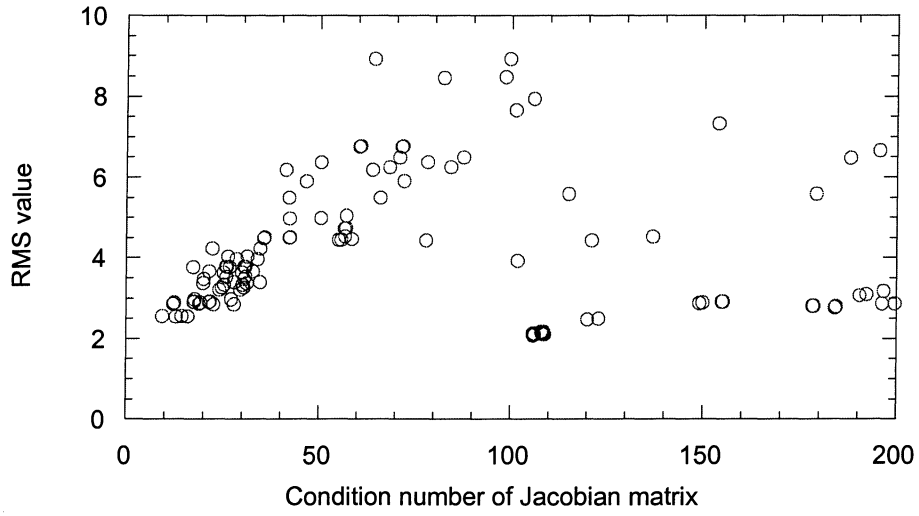

Figure 3.6b RMS value vs. condition number comparison (a/c, II)

3.4.3 Average calculation error

With this evidence, one should now wonder which of these two measures is more important. A logical way to pick out the best sensor set is by considering how measurement errors will propagate through the calculations, in a manner similar to the example discussed in Section 3.2.3. The best sensor set will result in, on average, the narrowest $\Delta k_{n,calc}$ distributions. The only way to measure the average width of every Δk_n distribution for a given Jacobian matrix is to actually plot a distribution similar to Figure 3.3 for every $\Delta k_{n,calc}$ and determine the width of a confidence interval associated with each distribution. As an example, equation [3.8] is an expanded version of equation [2.2].

$$\begin{bmatrix} J^{-1}_{1,1} & J^{-1}_{1,2} & \cdot & \cdot & \cdot & J^{-1}_{1,M} \\ J^{-1}_{2,1} & \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot & \cdot \\ J^{-1}_{N,1} & \cdot & & \cdot & \cdot & J^{-1}_{N,M} \end{bmatrix} \cdot \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta x_M \end{bmatrix} = \begin{bmatrix} \Delta k_1 \\ \Delta k_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta k_N \end{bmatrix} \qquad [3.8]$$

By plugging in N separate, known, unique $\Delta\mathbf{x}$ vectors (one for each fault), N $\Delta\mathbf{k}$ vectors result, each with a different nonzero element. Both the Jacobian matrix and the $\Delta\mathbf{x}$ vector are assumed to have random errors associated with them (see Appendix C for a detailed discussion of uncertainty), so N^2 different Δk_{calc} distributions result, N of which have nonzero nominal values. This result may be thought of as a N×N matrix of normal distributions, where the nominal value of every element is zero except the diagonals, which are $\Delta k_{n,crit}$ values. A 90% confidence interval was chosen to represent the width of a distribution. Figure 3.7 below shows a graphical representation of distribution width, in this case for reduced air conditioner evaporator air flow rate. The best-conditioned Jacobian (shown in Table 3.2) was used to detect the fault. The actual Δk_n value simulated was -62%, and Figure 3.7 shows that the fault was detected to within approximately ±8% of total flow rate with 90% confidence. Of course, a different Jacobian would give a different confidence interval width.

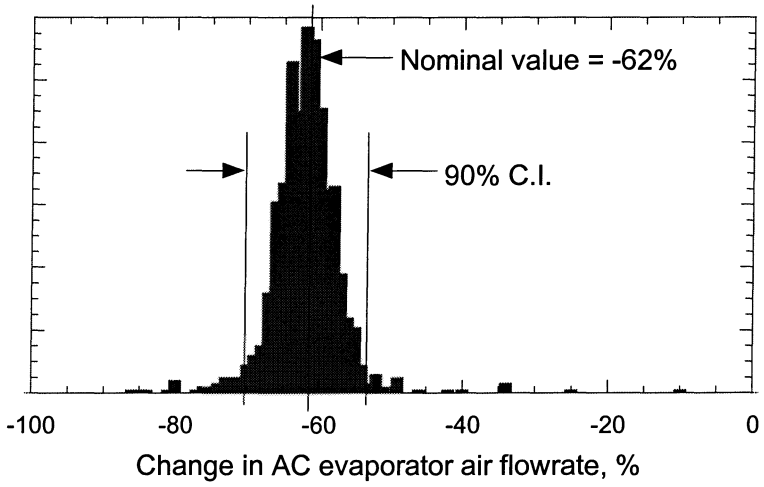

Figure 3.7 90% Confidence interval width

3.4.3.1 Refrigerator

Each Jacobian's "average calculation error" was computed by averaging the distribution width of all N^2 resulting distributions for that matrix (this measure of detection accuracy gives equal weight to both fault distributions and those centered about zero). By computing an average calculation error for every Jacobian, it can be explicitly shown how well each will perform. In an effort to discover whether a Jacobian's condition number or RMS value is ultimately more important, Figures 3.8 and 3.9 plot average calculation error vs. condition number and RMS value, respectively, for the refrigerator case.

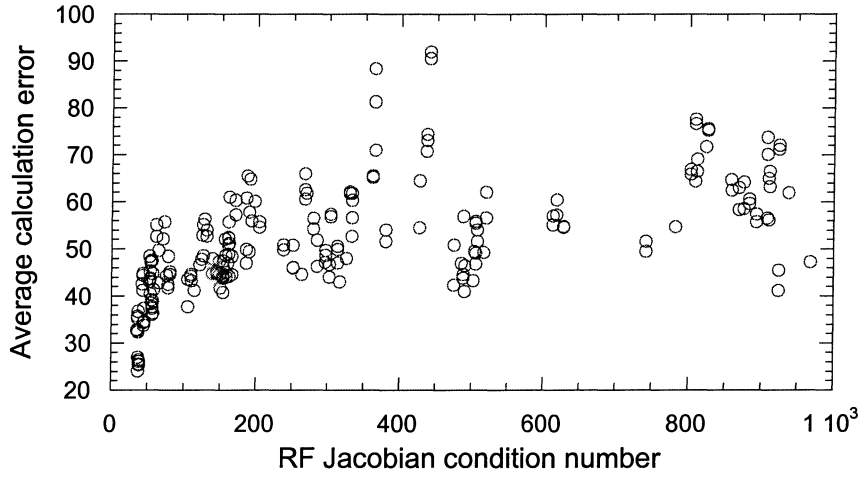

Figure 3.8 Average error vs. condition number (refrig)

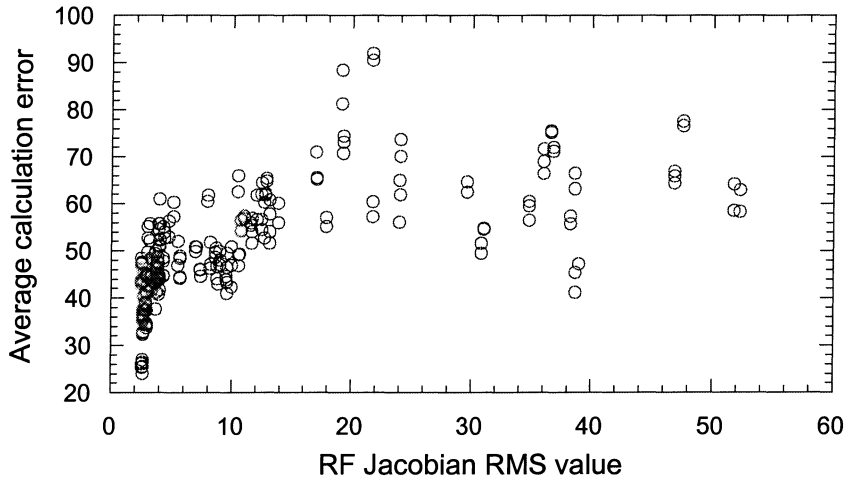

Figure 3.9 Average error vs. RMS value (refrig)

The two graphs show a general trend that a lower condition number and RMS value is better, but neither is a black-and-white indicator of final diagnosis quality.

3.4.3.2 Air conditioner

Following the example of the previous section, Figures 3.10 and 3.11 plot average calculation error vs. condition number and RMS value, respectively, for the air conditioner case.

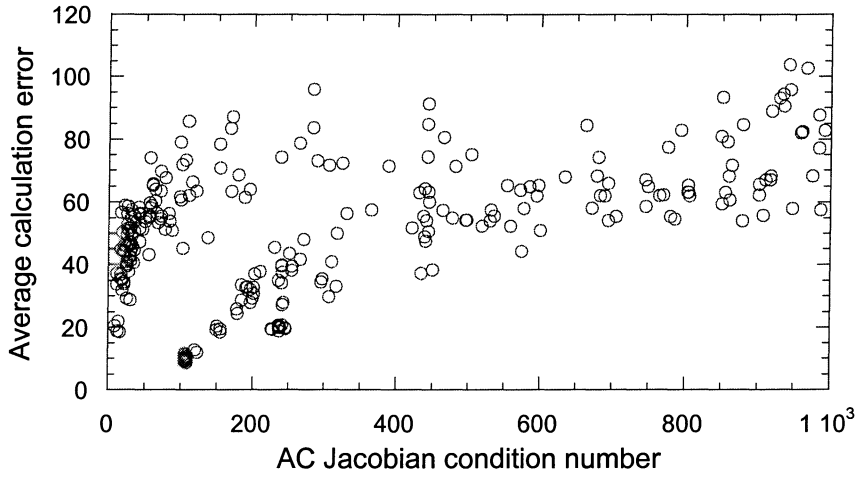

Figure 3.10 Average error vs. condition number (a/c)

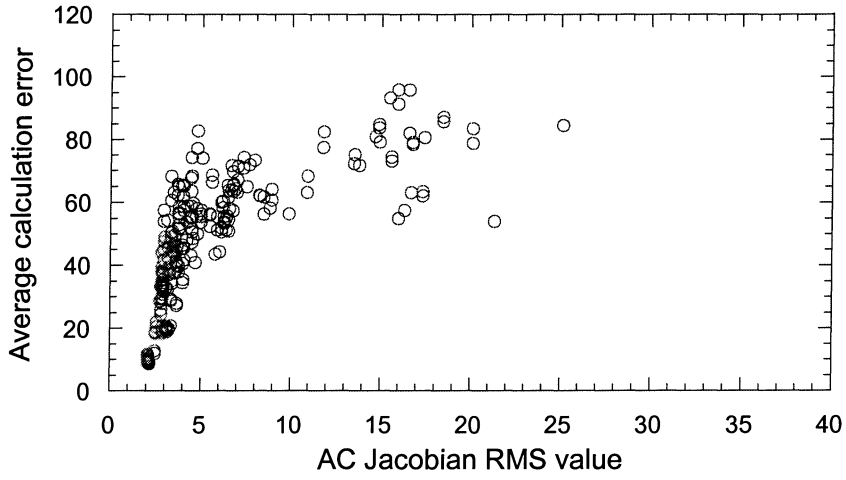

Figure 3.11 Average error vs. RMS value (a/c)

The same general trend seen in the refrigerator case is also apparent here, but Figure 3.10 shows a bit more independence of condition number than does Figure 3.8, specifically the fact that the matrices with the smallest average error have condition numbers slightly greater than 100. These graphs may lead one to believe that RMS value is a bit more meaningful in terms of indicating which Jacobians best quell the propagation of random sensor errors.

3.5 Adding redundant sensors

Earlier it was assumed that M=N was the minimum number of sensors required to detect N faults (see Section 3.1). It is known that M cannot be less than N, but equation [2.2] is still an exact relationship when M>N. An issue that should be addressed is whether extra sensors bring extra accuracy to the $\Delta\mathbf{k}$ calculation, and whether that extra accuracy is worth the extra cost. Simulation results show that extra sensors do in fact improve $\Delta\mathbf{k}$ calculation accuracy. Section 4.1 will provide a quantitative demonstration of calculation accuracy using extra sensors. In addition to this evidence, there is another benefit to the inclusion of extra sensors besides improved accuracy. As stated earlier, there are different types of errors possible in the calculation of $\Delta\mathbf{k}$, including errors due to the natural propagation of uncertainty present in each sensor and due to an unexpected sensor failure of some sort. In the case of a faulty sensor, the greater the number of sensors included in the detection process, the less effect one faulty sensor reading will have on the results (this is similar reasoning to that which led to a desire for a low RMS value). If, for example, each sensor's product $(\partial k_n/\partial x_m)\,\Delta x_m$ (see equation [3.1]) is of the same approximate magnitude, each sensor would contribute approximately (100/M)% to that particular Δk_n sum. Obviously if more sensors are included in the analysis, each individual sensor's contribution is decreased. This would provide insurance against sensor failure and false positives. If a sensor were to relay a faulty reading, it would have less impact on the final Δk_n calculation than if M=N (it also helps to think of a sensor failure simply as a larger-than-normal random uncertainty). There is a trade-off between reliability (more sensors = less chance of a false positive) and cost (fewer sensors = cheaper implementation). There is only a slight negative side to using an extra sensor besides its initial cost: the chance of that sensor failing.

Another advantage that arises with the addition of extra sensors is related to the sensor contribution issue. Consider an example: Suppose there is a fault that is detected mainly by a single sensor. Figure 3.12 below shows sensor contributions using refrigerator sensor set #1 (from Table 3.3) to detect a fouled condenser.

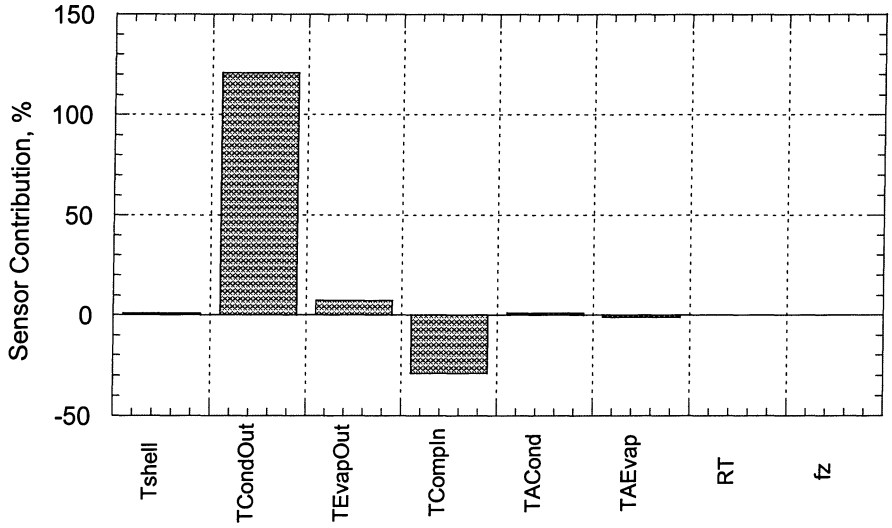

Figure 3.12 Sensor contributions to fouled condenser detection (8 sensors)

It appears that the condenser outlet temperature is the sensor that contributes the most to the detection of this particular fault. Knowledge of the refrigeration system, however, indicates that the condenser outlet and liquid line outlet temperatures are practically identical (the two sensors are redundant). Perhaps by adding an extra redundant sensor the contributions may be distributed more evenly. Figure 3.13 below shows the sensor contributions using 9 sensors (sensor set #1 plus liquid line outlet temperature) to detect the same fault.

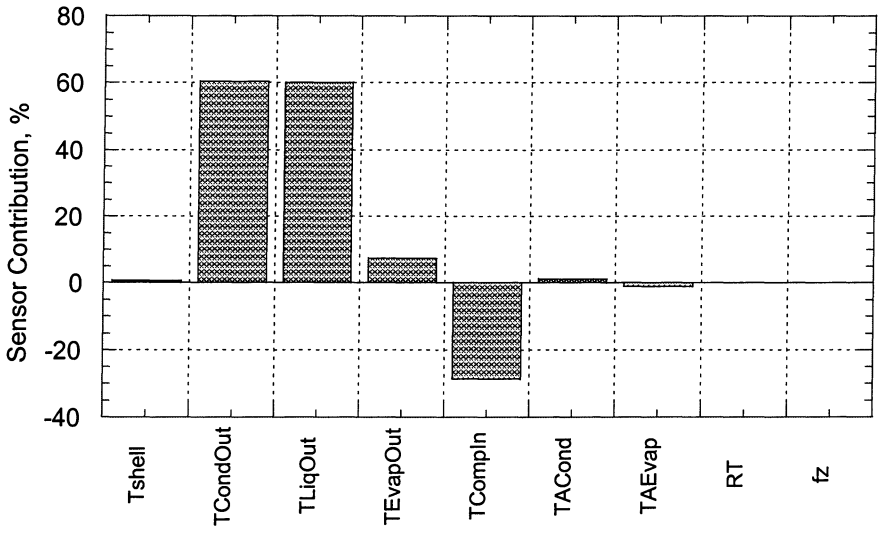

Figure 3.13 Sensor contributions to fouled condenser detection (9 sensors)

It appears that the previous assumption is correct. A redundant sensor will in fact ease the load on any single sensor, thereby reducing the probability of a false positive due to any error in that sensor's reading. In the previous example the liquid line outlet temperature was added because it was known to be dependent on the condenser outlet temperature. This sort of information may not always be known in advance. The RMS value of a matrix can help a user decide *which* extra sensor(s) to add to a square Jacobian. Extra sensors will reduce the RMS value of a Jacobian, but certain sensors will lower it more than others.

Chapter 4

Results

Chapters 2 and 3 outlined a FDD method and a logical algorithm for choosing the best sensor locations. This chapter will review the quality of the fault detection that is obtainable using this FDD method. Issues of interest include fault detectability thresholds and detection accuracy using the minimum number of sensors vs. using extra sensors. Note that this chapter tests the FDD method in two ways. The first uses a detailed simulation model, off-line, to calculate the Jacobian which relates faults to sensor responses. The second obtains the Jacobians experimentally, by inducing faults and observing sensor response. The resulting Jacobians form the heart of the on-line model that can be programmed into a microprocessor to detect faults in a refrigeration or air conditioner system. Sections 4.1 - 4.3 present only model results and analysis, while Section 4.4 deals only with experiments.

4.1 Detection Accuracy

Two important questions that must be answered relate to the issue of fault detectability thresholds, namely: how severe does a fault have to be before the proposed FDD method will detect it? and: how much is detection accuracy improved by increasing the number of sensors? Chapter 3 described two different criteria with which to choose the "best" set of sensors. One criterion involved the condition number of the Jacobian, while the other involved a search for "equality of contribution" among sensors. However, based on results shown in Chapter 3, neither of these is 100% dependable for picking out the best set of sensors based on average calculation error (also introduced in Chapter 3). This section will focus on determining detectability thresholds for the best sensor sets ($M \geq N$).

4.1.1 Air conditioner

Figures 3.10 and 3.11 show that the air conditioner Jacobian showing the lowest average calculation error (about 9%) has a condition number of approx. 108 and an RMS value of 2.2. Table 4.1 below shows the sensor set used to construct this Jacobian.

Table 4.1 Air conditioner set of 6 sensors with lowest calculation error

Cond. # = 108 RMS = 2.2
W_{Comp}
P_{Dis}
T_{Dis}
$T_{CondOut}$
$TA_{CondOut}$
$TA_{EvapOut}$

It was stated that the average calculation error (described in Section 3.4.3) for this sensor set is approximately 9%. Table 4.2 shows the numbers from which that figure (9%) was calculated. It lists the width of each 90% confidence interval (see Figure 3.7 for graphical representation of confidence interval) in the form: Δ(parameter) = [nominal value ± $^1/_2$ width of C.I.] %. Note that all of the nominal values in the table are expected to be exact (the same critical fault magnitudes as listed in Table 2.4). The reason is that the inverse Jacobian $\mathbf{J}^{-1}$ was multiplied by the same set of $\Delta\mathbf{x}$ vectors that was used to create the Jacobian **J** in the first place. The 90% confidence intervals are simply an indication of how much uncertainty is involved even when the nominal calculated value is perfect.

Table 4.2 Set of 90% confidence intervals, best set of 6 sensors

simulated fault:	low evap air flow		low cond air flow		fouled cond		compressor leak		low motor efficiency		low charge	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-
evap air flow	-62	5	0	3	0	3	0	3	0	3	0	3
cond air flow	0	4	-21	5	0	5	0	4	0	4	0	5
cond h_{air}	0	14	0	14	-52	14	0	13	0	13	0	13
% flow thru system	0	3	0	2	0	2	-7	2	0	2	0	2
power map	0	1	0	1	0	1	0	2	7	2	0	1
total charge	0	2	0	2	0	2	0	2	0	2	-9	2

Each column of Table 4.2 (low evap air flow, low cond air flow, etc.) represents the result of the inverse Jacobian multiplied by a single $\Delta\mathbf{x}$ vector, generated by the simulation of that particular fault. The left part of each column shows the nominal calculated value of each parameter, the right part shows the [±] value, or ½ the width of the 90% confidence interval. The shaded boxes highlight the parameter calculations that are expected to be nonzero, based on the fault present.

As an example, consider the second column, where reduced condenser air flow was simulated: all nominal values are just as expected. The only nonzero value is for "cond air flow" since that is the key parameter for this fault. The calculated value (-21% of the base case value) is exactly the amount by which the parameter was degraded in order to simulate the fault. The [±] value for the parameter is 5% of the base case value, meaning that we can be 90% sure that the actual change in the condenser air flow was between -26% and -16% of the base case value. The fact that magnitude of the nominal value is significantly larger than that of the [±] value indicates that a false positive is not likely. The widest confidence interval (highest [±] value) in the second column is associated with the condenser h_{air} parameter calculation. It is 14% of the base case value, but a false positive is still unlikely since its Δk_{crit} value (shown in Table 2.4) is -52%. Note that the [±] values for each individual calculated parameter are fairly consistent no matter which particular fault has been simulated.

This Jacobian's average calculation error (9%) was calculated by averaging the width of every confidence interval (= 2 × [± value]). These air conditioner results show that all of the listed faults are detectable (with at least 90% confidence) at a level that causes the system COP to drop 5%. This can be verified by noting that all nonzero nominal values in Table 4.2 are larger in magnitude than their accompanying uncertainties.

The next issue of interest is the effect that extra sensor addition will have upon fault detectability. With this goal in mind, an exhaustive search was performed of all possible sets of 7 sensors in search of the one with the lowest average calculation error. Figures 4.1 and 4.2 below show average calculation error vs. condition number and RMS value, respectively, for sets of 7 sensors.

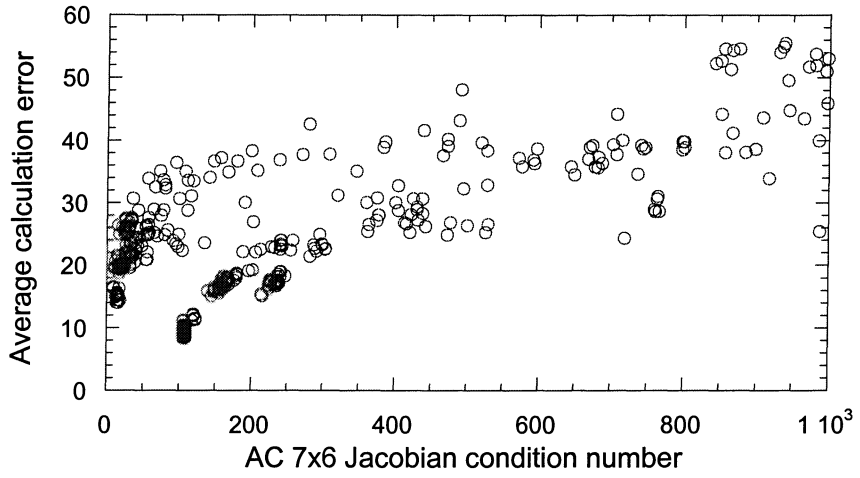

Figure 4.1 Average calculation error vs. condition number (a/c, 7 sensors)

The same trends are apparent here as were in Figure 3.10, but more sets seem to be clustered toward the lower left corner of the plot (note that the y-axis is from 0-60, where in Figure 3.10 it was 0-120).

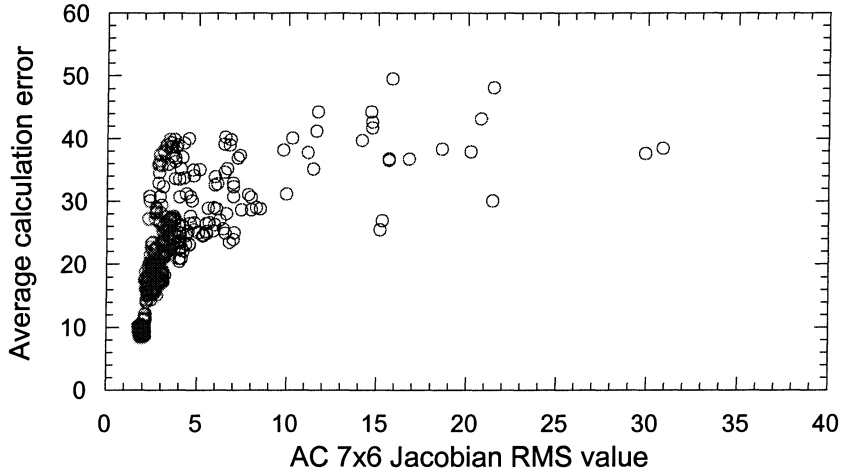

Figure 4.2 Average calculation error vs. RMS value (a/c, 7 sensors)

Again, this plot's shape is similar to that of Figure 3.11. The sensor set that shows the lowest calculation error (approximately 8%) is shown in Table 4.3 below.

Table 4.3 Air conditioner set of 7 sensors with lowest calculation error

Cond. # = 106 RMS = 1.9
W_{Comp}
$P_{Discharge}$
$T_{Discharge}$
$T_{CondOut}$
$T_{LiqLineOut}$
$T_{CondAirOut}$
$T_{EvapAirOut}$

The average error went down about 1% by adding one sensor. Note that the set is the same as that shown in Table 4.1, but with the liquid line outlet temperature added. The reason that detection accuracy improved is because the liquid line outlet and condenser outlet temperatures are closely related (they are separated only by the nearly adiabatic liquid line). The addition of this redundant sensor took half of the burden off the condenser outlet sensor, allowing more equal contributions, as discussed in Section 3.5). The addition of one sensor did not help detection accuracy much, and one wonders how well the FDD method could possibly perform. To answer this question, all 12 of the candidate sensor locations (listed in Table 2.3) were used to detect faults. When all 12 sensors were used, an average calculation error of 7.8% resulted. In the air conditioner case, it seems that the cost of extra sensors probably exceeds any performance benefit, at least for this particular unit.

4.1.2 Refrigerator

As mentioned in Section 2.2, a refrigerator is more complex than an air conditioner due to the fact that evaporator air flow is divided and flows to more than one compartment. The refrigerator results are not as illustrative as results from the previous section, perhaps because of this added complexity, or perhaps due to the presence of load faults (see Appendix D for a complete discussion). The linkage (or lack thereof) between system and load faults may weaken the FDD method. Therefore the results obtained were not as successful as with the air conditioner. Another possible reason is that the refrigerator model was not verified as rigorously as the air conditioner model. Simulation results do not show as much agreement with experimental results, which implies that the

predicted sensor responses do not reflect actual responses. This may indirectly affect Δk_{calc} results. For these reasons, refrigerator results are presented separately in Appendix G.

4.2 Gradual fault development

All results that have been reported thus far have been for faults detected at their critical ΔEnergy or ΔCOP values. When these faults occur in an actual system, they will develop slowly over time. One of the stated objectives of this project is to minimize the chance of false positive diagnoses. This includes distinguishing whether faults have reached a critical level or not. This issue is really a direct test of the linearity assumption that has been made (see Section B.3 for a discussion of this assumption). That is, we want to know if, when a fault is somewhere between its zero and critical levels, the FDD method is able to accurately predict the actual fault level.

In order to test this issue, a Jacobian constructed from the air conditioner sensor set listed in Table 4.1 was used to detect a fault as it developed gradually. The inverse Jacobian was multiplied by $\Delta \mathbf{x}$ vectors that resulted as the condenser airflow was reduced from its base case value to a value 36% less. Figure 4.3 below shows calculated vs. actual air flow reduction.

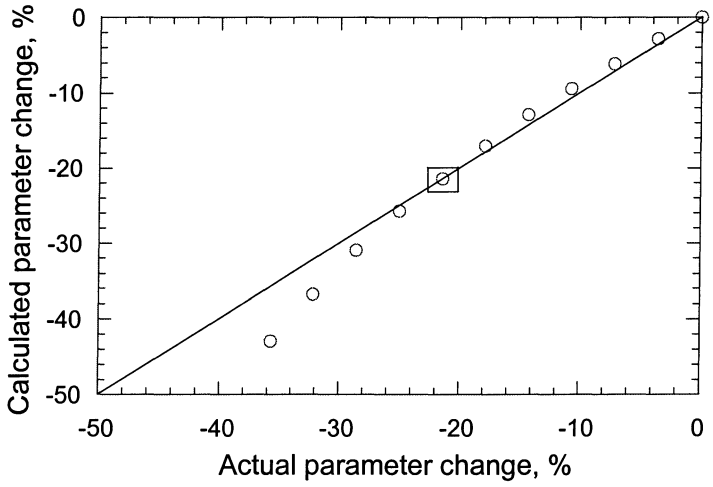

Figure 4.3 Calculated vs. actual parameter change for a/c condenser air flow

The boxed point at approx. -22% is the critical Δk value at which the Jacobian was constructed. This plot shows that from the no-fault condition to the critical level, the predicted parameter value is fairly accurate (the linearity assumption is acceptable). If the fault is left to develop past the critical stage, however, the quality of the prediction deteriorates rapidly, and the fault is significantly overestimated. Note that any linearization between two points gets more accurate as the distance between those point approaches zero. Hence, it is apparent that smaller magnitudes of Δk_{crit} values (such as a/c low motor efficiency, +7%) will give better agreement between calculated and actual than will larger values (such as a/c evaporator air flow reduction, -62%).

Another issue is the Δk_n calculations that should be zero throughout fault development. For example, in the previous example, while the condenser airflow is reduced, other Δk_n calculations should ideally be equal to zero throughout the process. Figure 4.4 below shows the predicted values of $h_{air,cond}$ (which would be degraded in the case of a fouled condenser, but not in this case of reduced airflow) as the condenser airflow is reduced.

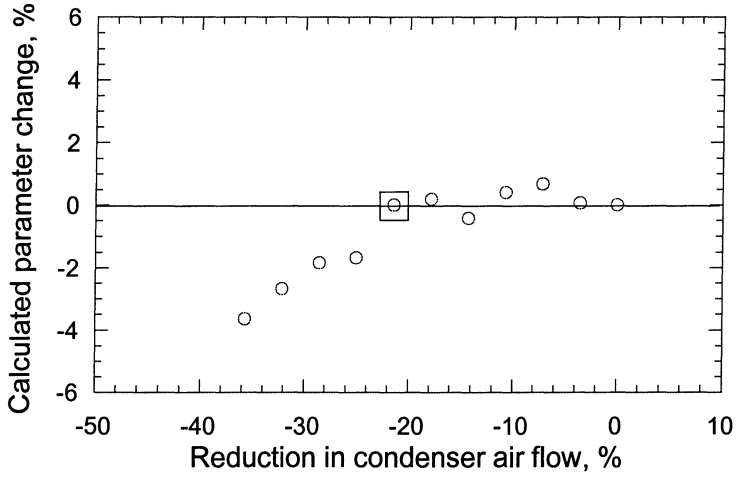

Figure 4.4 Calculated $h_{air,cond}$ vs. actual a/c condenser air flow reduction

Again, the prediction is acceptably close to zero between the fault's zero and critical points, showing that the linearity assumption is good, but degrades quickly outside that range. This issue of a parameter's zero-calculation is not always as well-

behaved as shown here. It is known to be a problem in certain cases, notably the case where refrigerator ΔUA_{gasket} values are actually zero but are calculated considerably far from zero, even within the critical range.

4.3 Jacobian robustness

An important question that model results may be able to answer is whether faults may be effectively diagnosed when the actual operating conditions are different from those at which the Jacobian matrix was generated. If not, then extra sensors (beyond those used as variables in the FDD method) would be needed (as well as a "bank" of Jacobians) in order to identify the system's current operating condition.

4.3.1 Refrigerator

In order to analyze this question, a Jacobian was generated using refrigerator model results at the standard conditions: ambient temperature = 75°F, freezer temp. = 5°F, fresh food temp. = 45°F. The Jacobian was constructed using all 13 of the candidate sensors listed in Table 2.1, to provide a best-case scenario. As discussed in Section 4.1.2, refrigerator results were not as good as air conditioner results, but they are sufficient for these purposes.

Figure 4.5 shows the results when the pseudo-inverse Jacobian was multiplied by a number of $\Delta\mathbf{x}$ (residual) vectors taken from model results in the ambient temperature range 60 - 90°F. The frosted evaporator fault was chosen to illustrate the results; it was fairly typical of most faults. The resulting $\Delta\mathbf{k}$ calculations show how accurately a frosted evaporator can be diagnosed at different ambient temperatures. The actual change (target value for the calculation) in evaporator air flow was -20% (as noted in Table 2.2), as the horizontal line on the plot shows.

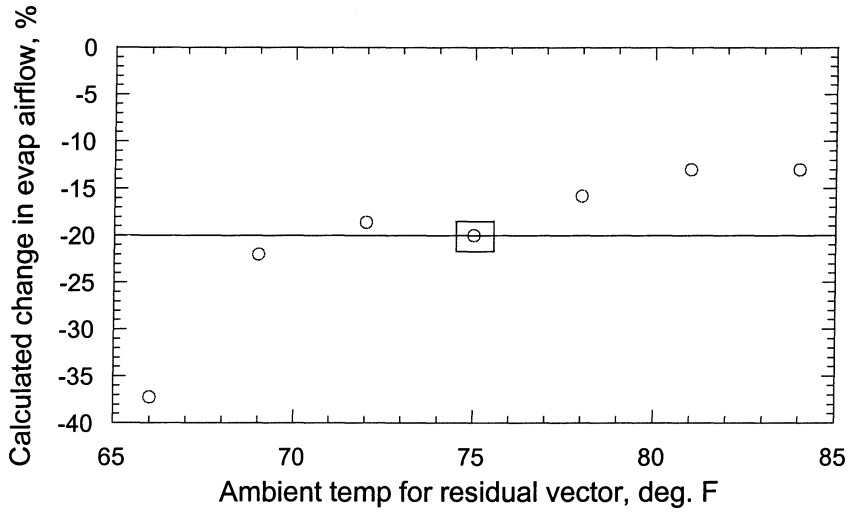

Figure 4.5 Jacobian robustness, ambient temperature variation

The plot shows that the diagnosis is perfect at T_{amb} = 75°F (as expected), but there is significant departure from -20% even when T_{amb} is within 10° of 75°F. When calculation uncertainty is taken into account, the errors shown above could become quite detrimental. This plot implies that separate Jacobians are probably necessary for different ambient temperature ranges.

Another parameter that may affect refrigerator diagnosis is the freezer temperature (more so than fresh food compartment temperature, as approximately 90% of the air flowing over the evaporator is directed to the freezer). A similar plot is shown in Figure 4.6 below, but the same Jacobian was multiplied this time by sensor readings ($\Delta\mathbf{x}$ vectors) observed when faults occurred in a refrigerator operating at various freezer temperature settings.

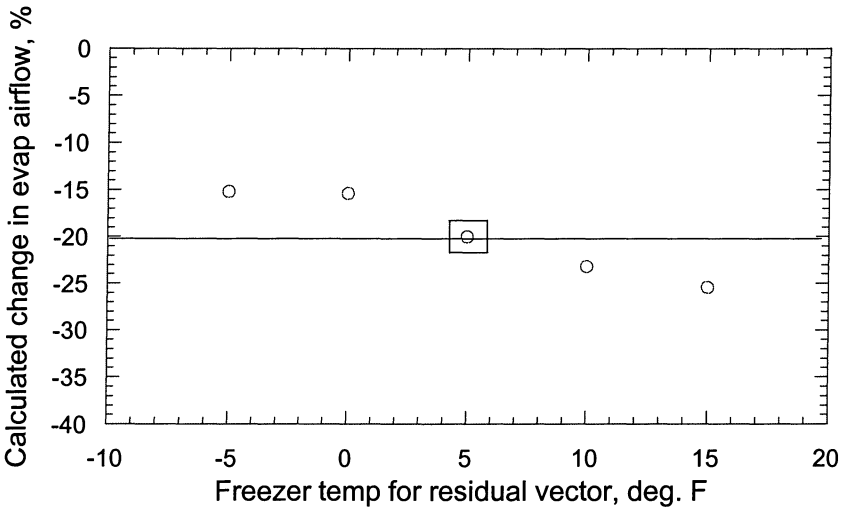

Figure 4.6 Jacobian robustness, freezer temperature variation

The plot shows that the change in flow rate is calculated anywhere between -15% and -25% over a freezer temp. range of ±10°F. The last two plots seem to show that if a piece of equipment is expected to be operated over a wide range of conditions, then several Jacobians are needed. However if operating conditions are known to be very stable and consistent, perhaps only one matrix will be necessary.

4.3.2 Air conditioner

As in the refrigerator case, operating conditions are sure to affect the air conditioner's performance. Since one of the operational parameters is uncontrollable (outdoor temperature) and highly variable both geographically and temporally, numerous Jacobians will surely be necessary. Simulation results confirm this assumption.

Another important question is whether indoor humidity will significantly affect diagnosis results. Humidity sensors are typically more expensive than temperature sensors, so it would be beneficial if it did not have to be measured. As before, simulation results were used to answer this question. A Jacobian was generated using simulation results at the standard conditions: outdoor temperature = 95°F, indoor temp. = 80°F, indoor R.H. = 50%. The Jacobian was constructed using all 12 of the candidate sensors listed in Table 2.3, to provide a best-case scenario. Figure 4.7 shows the evaporator

airflow calculation results when the pseudo-inverse Jacobian was multiplied by a number of $\Delta \mathbf{x}$ (residual) vectors taken from model results in the relative humidity range of 45-70%. The actual change in evaporator air flow was -62% (as noted in Table 2.4), as the horizontal line on the plot shows.

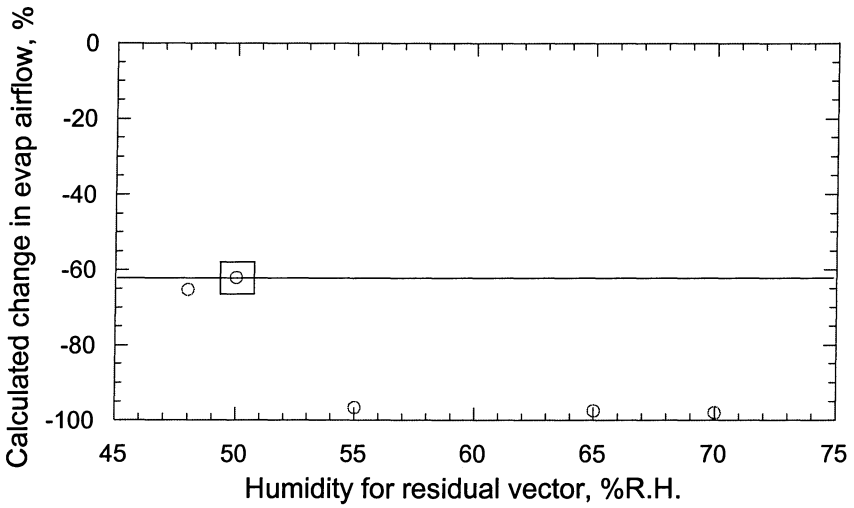

Figure 4.7 Jacobian robustness, a/c evaporator airflow

The reduced evaporator airflow fault was chosen to illustrate the results because it displayed the worst results of all the a/c faults. This is to be expected, as indoor humidity variation will affect indoor components more than outdoor. In fact, humidity has only a minimal effect on the diagnosis of outdoor component faults. Figure 4.8 below is a similar plot, showing the calculation results for reduced condenser airflow as indoor humidity varied. The actual change in airflow was -21%.

These plots show that humidity may or may not be an important factor, depending on which faults a user is interested in detecting.

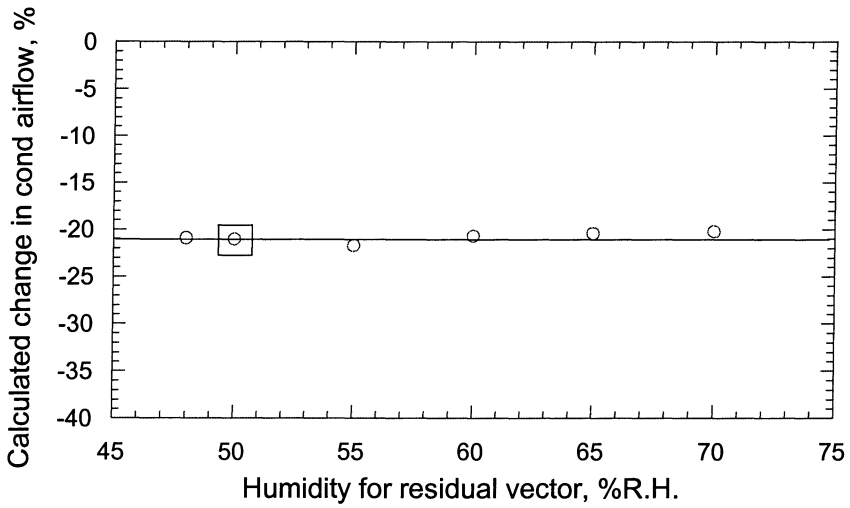

Figure 4.8 Jacobian robustness, a/c condenser airflow

4.4 Experiments

All of the analysis done thus far has been based on simulation model results. These models are helpful in showing general trends that are important to fulfilling project objectives, but one of our goals is to verify model results with experimental data. A number of the faults that were simulated with models were also induced experimentally in the laboratory. Table 4.4 below lists these faults for each system.

Table 4.4 Experimentally induced faults

Refrigerator	Air conditioner
frosted evaporator	reduced evaporator air flow
reduced condenser air flow	reduced condenser air flow
fouled condenser	compressor high-to-low side leak
fresh food gasket leak	low charge
freezer gasket leak	

Appendix A describes each fault experiment that was performed and lists the calculated parameter values for each test (although refrigerator results are discussed in Appendix G for reasons given in Section 4.4.2). Table 4.5 below lists the "critical" fault levels (parameter changes) observed in experiments. These parameter values, along with

variable values, were used to construct experimental Jacobian matrices in the same fashion as was done with model results.

Table 4.5 Experimental critical fault parameter changes

Refrigerator		Air conditioner	
parameter	Δk_{crit}	parameter	Δk_{crit}
evaporator air flow	-20%	evaporator air flow	-15%
condenser air flow	-32%	condenser air flow	-21%
$h_{air,cond}$	-16%	% flow through system	-9%
fresh food UA	19%	total system charge	-19%
freezer UA	21%		

The following sections discuss FDD results and detection accuracy, when experimentally-determined Jacobians were used. As in the model simulation cases, a Monte Carlo technique was used to determine the effects of uncertainty and 90% confidence intervals for parameter value calculations.

4.4.1 Air conditioner

Experimental data for air conditioner runs was taken at the standard capacity rating condition: indoor temperature = 80°F, outdoor temperature = 95°F, indoor relative humidity = 50%. Note that for the experimental case, unlike the simulations, the liquid line outlet temperature was not available as a candidate sensor location. It was not installed on the test unit, but the other 11 locations listed in Table 2.3 were available.

4.4.1.1 Single faults

In keeping with the example set by the simulation cases, an exhaustive search was performed on all possible sets of 4 sensors for those with the lowest condition number, RMS value, and average calculation error. The following plots are similar to those seen in Chapter 3. Figure 4.9 shows RMS value as a function of condition number.

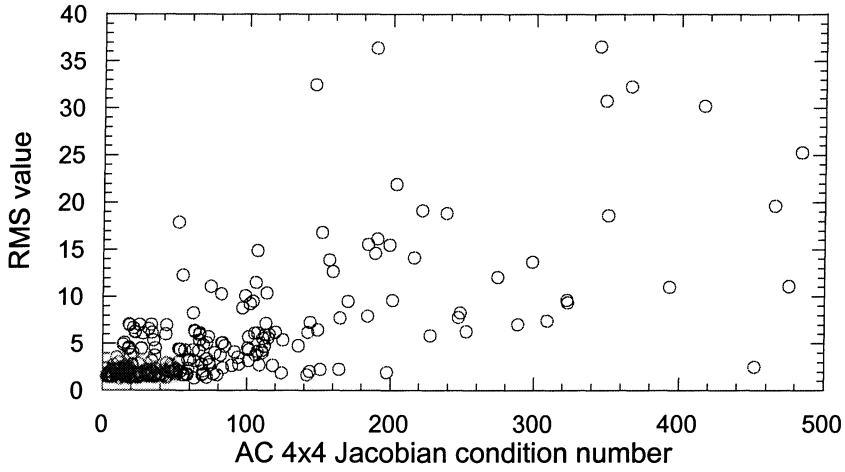

Figure 4.9 RMS value vs. condition number (a/c experiments)

It appears that the same type of trend that was seen in simulation results is seen here as well. There is a general trend of the best conditioned matrices also having the lower RMS values, but there are also many sets with large condition numbers that have low RMS values. Figure 4.10 shows average calculation error (described in Section 3.4) as a function of Jacobian condition number. Figure 4.11 shows average calculation error as a function of RMS value.

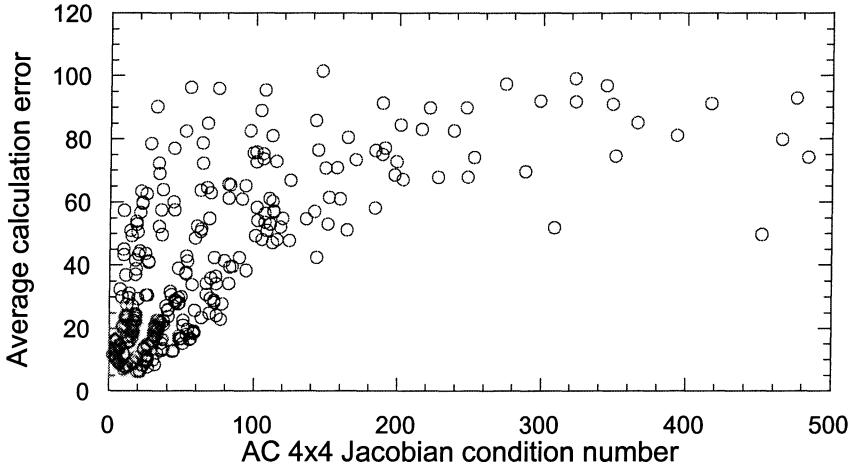

Figure 4.10 Average calculation error vs. condition number (a/c experiments)

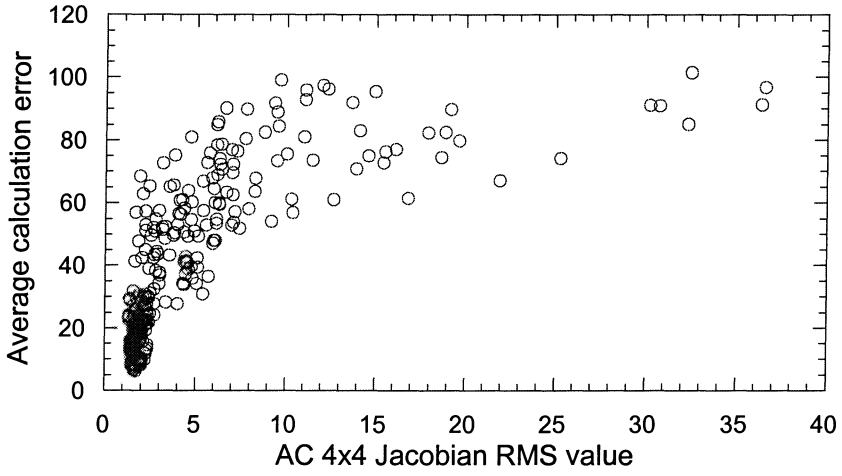

Figure 4.11 Average calculation error vs. RMS value (a/c experiments)

Both Figures 4.10 and 4.11 show results similar to those seen in simulations. Namely, that the matrices with the lowest values of condition number and RMS tend to give more accurate FDD results. These results were used to construct a 4×4 Jacobian for detecting the four a/c faults listed in Table 4.4. The four sensors used to construct the Jacobian were chosen such that the Jacobian gives the minimum average calculation error. Table 4.6 lists the sensor set used.

Table 4.6 Experimental air conditioner sensor set

4 air conditioner sensors
Cond # = 19.7, RMS = 1.71
W_{Comp}
$T_{Discharge}$
$T_{CondOut}$
T_{EvapIn}

The detection accuracy of this Jacobian was tested as in Section 4.1, with 90% confidence intervals. Table 4.7 below shows how accurately each fault (and non-fault) was detected.

Table 4.7 Set of 90% confidence intervals, air conditioner experiments

fault:	low evap airflow		low cond airflow		compressor leak		low charge	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-	nom	+/-
evap airflow	-15	6	0	6	0	6	0	6
cond airflow	0	3	-21	3	0	3	0	3
% flow thru system	0	1	0	1	-10	1	0	1
total charge	0	4	0	4	0	4	-19	4

The results shown here look quite good. As before, the appropriate fault values are highlighted in Table 4.7 for easier reference. The width of the 90% confidence intervals are comparable to those shown for model results in Table 4.2. Note that, as explained in Section 4.1.1, the $\Delta\mathbf{x}$ (sensor) vectors are the same that were used to construct the Jacobian in the first place. Therefore the nominal $\Delta k_{n,calc}$ values are exact (the same values as shown in Appendix A).

In the course of performing these experiments, extra tests were done that caused COP to decrease by less than 5%. Table 4.8 shows FDD results obtained from these $\Delta\mathbf{x}$ vectors, which are independent from the Jacobian. Since the fault levels are less than those that caused a 5% reduction in COP, non-linearity effects are also tested. The three cases listed in Table 4.8 refer to the following parameter conditions:

Case 1: Condenser air flow (-9%)

Case 2: Compressor leak (-4%)

Case 3: Base case, no faults present

Table 4.8 Set of 90% confidence intervals, a/c minor fault experiments

Case #:	1		2		3	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-
evap airflow	1	5	0	5	2	4
cond airflow	-8	2	-7	3	0	2
% flow thru system	-1	1	-8	1	0	1
total charge	2	3	1	3	-2	3

The appropriate fault values are highlighted in Table 4.8 for easier reference. These results are encouraging with the exception of case 2. The severity of the compressor leak was overestimated by nearly a factor of 2, and a -7% change in condenser air flow was falsely calculated. In the case of condenser air flow, a -7% change is significantly less than the critical (5% COP degradation) level, but if it were larger a false positive may be indicated. Note that in Tables 4.7 and 4.8 the confidence interval widths in each row are approximately equal regardless of whether a critical, minor, or no fault is present. This is an indication that the confidence interval width for each parameter calculation may be treated as nearly constant.

4.4.1.2 Multiple faults

One of the objectives of this FDD method is to detect simulated faults alone and in combination. This implies that the method must be able to detect and diagnose multiple faults that occur simultaneously. The method is based on theory, discussed in Chapter 2, that leads one to believe that it should work equally well for single or multiple faults. Multiple fault tests were performed in the laboratory, and will provide a good test for the experimental Jacobian. Three a/c multiple fault runs were performed, and the three cases listed in Table 4.9 refer to the following parameter changes, all of which are at their critical values:

Case 1: Evaporator air flow (-15%) and condenser air flow (-21%)

Case 2: Low charge (-19%) and condenser air flow (-20%)

Case 3: Low charge (-19%) and evaporator air flow (-15%)

Table 4.9 Set of 90% confidence intervals, a/c multiple fault experiments

Case #:		1		2		3	
calculated Δk_n, %	Δk_{crit}	nom	+/-	nom	+/-	nom	+/-
evap airflow	-15	-8	6	-12	9	-21	9
cond airflow	-21	-18	3	-28	4	-3	4
% flow thru system	-10	0	2	1	2	0	2
total charge	-19	1	3	-26	6	-21	5

Once again the appropriate fault values are highlighted. Detection and diagnosis for these multiple fault runs is not quite as good as in the previous table. The confidence intervals for each detected fault have widened a bit due to the presence of other faults, with the exception of "% flow through system," the parameter involved in detecting a compressor leak. See Appendix A for a complete explanation of this and any other parameters. A compressor leak was not induced in conjunction with any other fault, and in fact the FDD method calculated no false positives.

The calculated values of evaporator air flow are incorrect by at least 6% (of base case value) for all three cases. In case 1, Δ(evaporator airflow) is significantly underestimated, which probably means the fault will be detected late. In case 2, a 12% decrease was calculated, which may lead to a false positive indication. The calculated value of condenser air flow is off by 7% (of base case value) in case 2, but is within 3% of its actual value in the other cases. It appears here that the presence of more than one fault, along with possible nonlinearity effects, present added difficulties in detection.

4.4.2 Refrigerator

As discussed in Section 4.1.2, a refrigerator is more complex than an air conditioner due to the fact that evaporator air flow is divided and flows to more than one compartment. Experimental refrigerator results suffer from the presence of load faults just as simulation results did. Another problem with the experiments was with inaccurate instrumentation. Recall that in the steady-state experiments, actual compressor RunTime was 100% and f_z was constant. Calculations were made off-line using readings from compartment heaters to estimate values of RT_{calc} and $f_{z,calc}$ (see Chapter 2 for a description of the heaters and their purpose). Unfortunately, as discussed by Kelman and Bullard (1999), the compartment heaters used in the refrigerator experiments gave consistently suspect readings. Once again, the refrigerator results are not as illustrative as air conditioner results, therefore experimental refrigerator results are presented separately in Appendix G.

Chapter 5

Conclusions and Recommendations

This report documented a FDD method whose purpose is to detect faults before they severely hinder the performance of a refrigerator or air conditioner. Since the method depends on quasi-steady operation, data must be taken when a system is known to have completed a number of undisturbed cycles. The method is general and allows a variable number of sensor locations, in some cases with no significant drop in reliability.

Two methods were proposed as ways of choosing the best set of sensors for detecting a given set of faults. One method, based on the condition number of the Jacobian matrix, strives to select sensors whose readings are independent of each other. The other, the RMS summation method, aims to minimize the effect of any sensor error by minimizing the importance of any single sensor. Both methods appear to be good indicators of which sensor sets might be the best, but neither proved to have the ability to actually choose the best set based solely on minimizing the uncertainty of calculated parameter values. A third measure of sensor set quality, average calculation error, was also investigated. It chooses sets based on calculation results rather than mathematical issues, but is much more computationally intensive and may not always be feasible. It is recommended that future researchers examine differences between these methods and investigate exactly why they suggest given sensor sets. Table 5.1 below shows how many (M=N) sensor sets were considered for each system.

Table 5.1 Summary of simulation sensor set results

	Number of candidate sensor locations	Number of faults	Number of possible sensor sets
Refrigerator	13	8	>1200
Air conditioner	12	6	>900

Table 5.2 shows the best sensor set for each system, based only on the condition number of the Jacobian matrix they produce. These sets are presented as an example of "good quality" sets.

Table 5.2 Best-conditioned simulation sensor sets

Refrigerator	Air conditioner
T_{Shell}	P_{Dis}
$T_{CondOut}$	$T_{LiqLineOut}$
$T_{EvapOut}$	T_{CompIn}
T_{CompIn}	$TA_{CondOut}$
$TA_{CondOut}$	$TA_{EvapOut}$
$TA_{EvapOut}$	T_{Shell}
RunTime	
damper position	

Simulation results show that many faults may be detected using sets of the bare minimum number of sensors (i.e. N sensors for N faults, such as the ones shown above). However, results also show that including more sensors in the detection scheme increases accuracy while guarding against sensor failures. The only drawback to the inclusion of extra sensors is their initial cost. In fact, the reliability of this FDD method will probably depend directly on how much a user is willing to spend on its implementation. The most significant factors will probably be sensor cost and the cost of obtaining quality data relating sensor response to fault magnitude, either through experiments or simulations.

Some limitations have become apparent through this report, and may be addressed in future research. For example, this method ultimately relies on a simple linear system model. However, actual systems have some highly nonlinear responses to certain faults, specifically in the case of an air conditioner that is low on charge, where the condenser exit undergoes a change in flow regime from subcooled to 2-phase.

A problem that was not addressed in this report concerns the fact that all of the detectable faults are those that are originally accounted for in the Jacobian matrix. If an "unanticipated" fault occurs, it will most likely indicate simultaneous false positives for more than one fault (i.e. incorrect faults are indicated). Another possibility, although not likely, is that no fault is indicated and it would go completely undetected. However this may be only a minor issue, as the behavior of these types of systems are well-known by industry, and rarely is a failure a complete surprise to equipment designers.

References

Beaver, A.C., J.M. Yin, C.W. Bullard, and P.S. Hrnjak, "An Experimental Investigation of Transcritical Carbon Dioxide Systems for Residential Air Conditioning," ACRC CR-18, University of Illinois at Urbana-Champaign, 1999.

Breuker, M.S., and J.E. Braun, "Common Faults and Their Impacts for Rooftop Air Conditioners," HVAC&R Research, Vol. 4, No. 3, July 1998.

Breuker, M.S., and J.E. Braun, "Evaluating the Performance of a Fault Detection and Diagnostic System for Vapor Compression Equipment," HVAC&R Research, Vol. 4, No. 4, October 1998.

Bridges, B.D., C.E. Mullen, and C.W. Bullard, "Simulation of Room Air Conditioner Performance," ACRC TR-79, University of Illinois at Urbana-Champaign, 1995.

Cavallaro, A.R., and C.W. Bullard, "Effect of Variable-Speed Fans on Refrigerator Component Heat Transfer," ASHRAE Transactions, 101:2, 7 pp., 1995.

Dongarra, J.J., et al., LINPACK User's Guide, Society for Industrial and Applied Mathematics, Philadelphia, 1979.

Figliola, R.S., and D.E. Beasley, Theory and Design for Mechanical Measurements, John Wiley & Sons, New York, 1991.

Grimmelius, H.T., J.K. Woud, and G. Been, "On-line failure diagnosis for compression refrigeration plants," International Journal of Refrigeration, Vol. 18, No. 1, 1995.

Grimmelius, H.T., et al., "Three State-of-the-Art Methods for Condition Monitoring," IEEE Transactions on Industrial Electronics, Vol. 46, No. 2, 1999.

Kärki, S.H., and S.J. Karjalainen, "Performance Factors as a Basis of Practical Fault Detection and Diagnostic Methods for Air-Handling Units," ASHRAE Transactions, 105:1, 1999.

Katipamula, S., et al., "Automated Fault Detection and Diagnostics for Outdoor-Air Ventilation Systems and Economizers: Methodology and Results from Field Testing," ASHRAE Transactions, 105:1, 1999.

Kelman, S., and C.W. Bullard, "Dual Temperature Evaporator Refrigerator Design and Optimization," ACRC TR-148, University of Illinois at Urbana-Champaign, 1999.

Mullen, et al., "Development and Validation of a Room Air Conditioning Simulation Model," ASHRAE Transactions, 104:2, 1998.

Ngo, D. and A.L. Dexter, "A Robust Model-Based Approach to Diagnosing Faults in Air-Handling Units," ASHRAE Transactions, 105:1, 1999.

Rossi, T.M., "Detection, Diagnosis, and Evaluation of Faults in Vapor Compression Equipment," Ph.D. Thesis, School of Mechanical Engineering, Purdue University, 1995.

Rossi, T.M., and J.E. Braun, "A Statistical, Rule-Based Fault Detection and Diagnostic Method for Vapor Compression Air Conditioners," HVAC&R Research, Vol. 3, No. 1, January 1997.

Srichai, P.R., and C.W. Bullard, "Two-Speed Compressor Operation in a Refrigerator/Freezer," ACRC TR-121, University of Illinois at Urbana-Champaign, 1997.

Strang, G., Introduction to Linear Algebra, Wellesley-Cambridge Press, Wellesley, 1993.

Stylianou, M., and D. Nikanpour, "Performance Monitoring, Fault Detection, and Diagnosis of Reciprocating Chillers," ASHRAE Transactions, 102:1, 1996.

Wagner, J., and R. Shoureshi, "Failure Detection Diagnostics for Thermofluid Systems," Journal of Dynamic Systems, Measurement, and Control, Vol. 114, December 1992.

Woodall, R.J., and C.W. Bullard, "Development, Validation, and Application of a Refrigerator Simulation Model," ACRC TR-99, University of Illinois at Urbana-Champaign, 1996.

Woodall, R.J., and C.W. Bullard, "Simulating Effects of Multispeed Compressors on Refrigerator/Freezer Performance," ASHRAE Transactions, 103:2, 1997.

Appendix A

Experiment Descriptions

A.1 Introduction

When there is a fault in a refrigerator or air conditioner, whether or not it is of interest depends on how much the system's performance is hurt by that fault. Performance degradation in the form of low coefficient of performance (COP) and/or increased compressor run time (RT) leads to increased energy use and higher operation cost. The majority of FDD analysis in this report was done using model results, but an objective of the project is to verify those results with experimental data.

Experiments were performed such that they could be compared meaningfully to simulation model runs. A number of different operating conditions were tested for each system, and data was taken at base case (no faults present) conditions and at different fault conditions. A perfect set of experiments would show identical ΔEnergy or ΔCOP values (as seen in model run results) for all fault runs, so all tests could be compared without adjustment. However, since actual experiments will inevitably show different changes in those variables, the objective is to certify that each experiment decreased the variably of interest by a comparable amount. For reasons presented in Chapter 2, the variable of interest in the refrigerator case is ΔEnergy, and in the air conditioner case ΔCOP.

A.2 Equipment

A.2.1 Refrigerator

The experimental refrigerator used for this project is a 25 cubic foot, side-by-side Amana charged with R-134a. The only major modification made to the unit was the replacement of the original Tecumseh compressor with a two-speed Americold prototype, model RV800. Instrumentation of the unit is described in detail by Srichai and Bullard (1997).

A.2.2 Air conditioner

The experimental air conditioner used is a well-instrumented, 3 ton Carrier residential split system charged with R410A. The experimental facility and system components are described by Beaver (1999).

A.3 Experimental procedure

As described in Chapter 2, the simulation models were used to simulate faults that caused a 5% increase in total energy use (refrigerator) or 5% decrease in COP (a/c). The parameter values that caused these changes were chosen as critical "threshold" values. Model results show that in and around these ranges, it is usually reasonable to assume a near-linear dependence of any variable on any parameter. This same characteristic should also be true of experimental data. It is difficult to induce an exact level of performance degradation under experimental conditions, and to know exactly how much an operational parameter has been changed. The following sections describe steady-state experiments performed on the refrigerator and air conditioner units described in Section A.2. In these experiments, data reduction programs were used to estimate values of the parameters of interest, as well as COP values, and to verify that they were reasonably close to target values.

A.3.1 Refrigerator

All results shown in this section were obtained at the following conditions: Ambient temperature = 75°F, freezer temperature = 5°F, fresh food compartment temperature = 45°F. When refrigerator model runs were performed, the main variable of interest was system energy use (instead of COP). However experimentally, for reasons explained in Section 2.2, the actual compressor run time was always 100% and system power did not always increase upon fault induction. Therefore experiments were performed, then the data was used to estimate parameter values, then $\Delta k_{n,exp}$ values were compared to $\Delta k_{n,model}$ values. An experimentally-induced fault was deemed to have acceptable magnitude if the experimentally-measured parameter change roughly matched the model-predicted change. An experimental Jacobian matrix was then constructed using changes in parameters and changes in sensor readings just as in the model

simulation cases. The data reduction programs used to estimate parameter values were written by Srichai and Bullard (1997).

A.3.1.1 Base case experiments

Base case data was taken simply by letting the unit run with no artificial modifications. These test points are meant to represent fault-free operation. Table A.1 shows base case results.

Table A.1 Refrigerator base case

parameters	base case
evaporator air flow, cfm	60.2
condenser air flow, cfm	131.0
h_{air}, condenser, Btu/hr-°F	4.30
UA_{ff}, W/°F	0.91
UA_{fz}, W/°F	0.97

These parameter values are what other fault experiment values are to be compared to.

A.3.1.2 Frosted evaporator

For this fault experiment, frost was allowed to accumulate as it would in real operating mode while both compartment temperatures were held constant. The system was run for approximately 48 hours to degrade performance sufficiently. Due to the continuous nature of frost growth, there was no way to have a steady-state amount of frost on the coil.

In order to promote frost growth on the evaporator surface, wet sponges and towels were placed in the fresh food compartment while the system operated. These moistened the air circulating over the evaporator and allowed frost to build up faster than would be expected normally. After the test the refrigerator was allowed to defrost, and the frost that had built up on the evaporator was collected as liquid, thus allowing the mass of water to be measured. Table A.2 shows parameter values for this fault.

Table A.2 Refrigerator frosted evaporator

parameters	base case value	frosted evaporator	delta values, % of base value
evaporator air flow, cfm	60.2	48.0	-20.3 %
condenser air flow, cfm	131.0	131.0	0 %
h_{air}, condenser, Btu/hr-°F	4.30	4.30	0 %
UA_{ff}, W/°F	0.91	0.91	0 %
UA_{fz}, W/°F	0.97	0.97	0 %

The observed decrease in air flow indicates that frost on the evaporator significantly decreased the available flow area. More than 1.6 lbm. of water was collected from the evaporator coil upon defrost, which indicates that it can take on a significant amount of frost before performance is hurt.

A.3.1.3 Blocked condenser air flow

The condenser air flow was blocked by placing duct tape over a portion of the condenser fan's discharge vent. This experiment simulates some type of fan blockage, such as the refrigerator being pushed too close to a wall or excessive debris built up on the vent. Table A.3 shows parameter values for this test.

Table A.3 Refrigerator blocked condenser air flow

parameters	base case value	reduced condenser air	delta values, % of base value
evaporator air flow, cfm	60.2	60.2	0 %
condenser air flow, cfm	131.0	89.0	-32.1 %
h_{air}, condenser, Btu/hr-°F	4.30	4.30	0 %
UA_{ff}, W/°F	0.91	0.91	0 %
UA_{fz}, W/°F	0.97	0.97	0 %

An adverse effect that may have occurred during this experiment is condenser air recirculation, but the fan speed was not able to be varied, so this was the most accurate simulation possible. The refrigerator model did not account for recirculation, but in our experiment the effect was probably present.

A.3.1.4 Fouled condenser

The condenser on the test unit is a wire-on-tube type located underneath the unit, near the floor. In an actual unit, condenser fouling may occur due to dust and debris that builds up on the coil. It was fouled in the laboratory by placing sheets of cardboard and towels over the top portion of its heat transfer surface area. Figure A.1 illustrates the condenser orientation and cardboard placement.

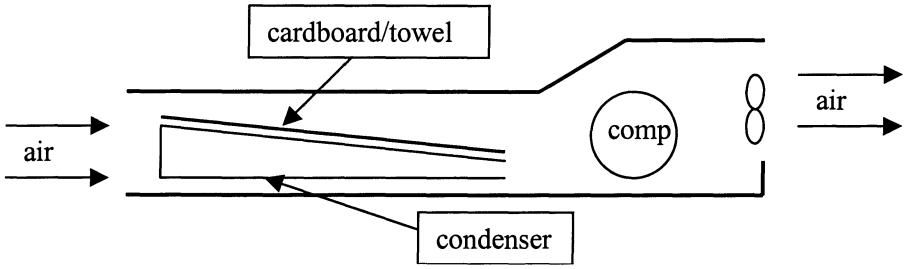

Figure A.1 Refrigerator condenser housing, side view

Table A.4 shows parameter values for this test.

Table A.4 Refrigerator fouled condenser

parameters	base case value	fouled condenser	delta values, % of base value
evaporator air flow, cfm	60.2	60.2	0 %
condenser air flow, cfm	131.0	131.0	0 %
h_{air}, condenser, Btu/hr-°F	4.30	3.63	-15.6 %
UA_{ff}, W/°F	0.91	0.91	0 %
UA_{fz}, W/°F	0.97	0.97	0 %

Air flow over the condenser may have been inadvertently (but minimally) affected by this experiment, due to the fact that cardboard sheets and towels are not perfectly flat and were not placed exactly parallel to the flow direction.

A.3.1.5 Door gaskets leaks

Both freezer and fresh food compartment gasket leaks were induced on the test unit. When a compartment gasket is leaking, the heat transfer between that compartment and the ambient air is expected to increase due to warm ambient air being allowed into the compartment. This might occur when gaskets are old and worn. The faults were simulated in the lab simply by placing a number of paper clips between the refrigerator body and the appropriate door gasket. The door was then closed on the paper clips, causing small openings that allowed warm ambient air into the cold compartments. Small or large leaks could be induced by varying the number of clips used. Table A.5 shows parameter values for each of the gasket leak tests.

Table A.5 Refrigerator gasket leaks

parameters	base case value	fresh food gasket leak	delta values, % of base	freezer gasket leak	delta values, % of base
evaporator air flow, cfm	60.2	60.2	0 %	60.2	0 %
condenser air flow, cfm	131.0	131.0	0 %	131.0	0 %
h_{air}, condenser, Btu/hr-°F	4.30	4.30	0 %	4.30	0 %
UA_{ff}, W/°F	0.91	1.08	18.7 %	0.91	0 %
UA_{fz}, W/°F	0.97	0.97	0 %	1.17	20.6 %

Note that gasket leaks are load faults, as opposed to the system faults described in the previous three sections, so the only system variable expected to change during these experiments are RunTime and damper position (both are off-line calculations using compartment heater powers). See Appendix D for a complete discussion of the difference between load and system faults.

Note that when ambient air enters a compartment, some moisture is also expected to enter with it. This extra water vapor may result in a higher rate of frost build up on the evaporator. This extra frost is a harmful side effect of a gasket leak, but frost on the evaporator is another one of this project's simulated faults, so the vector equation system should recognize it if its effect is significant.

A.3.2 Air conditioner

All results shown in this section were obtained at the standard conditions: Outdoor temperature = 95°F, indoor temperature = 80°F, indoor relative humidity = 50%. In the air conditioner experiments, COP was calculated with the following equation:

$$COP = Q_{capacity} / W_{compressor} \qquad [A.1]$$

Equation [A.1] was used to analyze whether a test had the desired impact on the system. Relevant parameter values and COP were first calculated for a base case test, then for each fault experiment the calculated COP was compared to the base value. If it was reasonably close to -5% then the experiment was deemed to be successful. An experimental Jacobian matrix was then constructed using changes in parameters and changes in sensor readings just as in the model simulation cases. The data reduction programs used to estimate parameter values are documented by Beaver (1999).

A.3.2.1 Base case experiments

As in the refrigerator case, base case data was taken simply by letting the unit run with no artificial modifications. These test points are meant to represent fault-free operation. Table A.6 shows base case COP and parameter values.

Table A.6 Air conditioner base case

parameters	base case value
COP	4.31
evaporator air flow, cfm	1195
condenser air flow, cfm	2759
% mass flow thru system	100
refrigerant charge, lbm	8.05

These parameter values are what other fault experiment values are to be compared to. The parameter "% mass flow through system" refers to an internal compressor leak, and is explained in Section A.3.2.4.

A.3.2.2 Evaporator air flow

The evaporator fan on the test unit is completely variable, so reducing air flow was done simply by reducing fan speed. Table A.7 shows COP and parameter values.

Table A.7 Air conditioner blocked evaporator air flow

parameters	base case value	reduced evaporator air	delta value, % of base value
COP	4.31	4.11	-4.6 %
evaporator air flow, cfm	1195	1020	-14.6 %
condenser air flow, cfm	2759	2770	0.4 %
% mass flow thru system	100	100	0 %
refrigerant charge, lbm	8.05	8.05	0 %

The -4.6% change in COP is close enough to the target value of -5% for this run to be acceptable. The key parameter here is evaporator air flow, which was reduced by 14.6%. Note that this is vastly different from the value of -62% predicted by the a/c simulation model in Table 2.4. The model's COP value actually increased upon a small reduction in evaporator air flow. This implies that the system simulated by the model has an evaporator flow rate that is not optimized at 1200 cfm. The model would have to be significantly refined before it could be used to generate Jacobians to be used for experimental results.

A.3.2.3 Condenser air flow

The condenser fan used on the test unit is also completely variable, so reducing air flow was done simply by turning down the fan speed. Table A.8 shows COP and parameter values.

Table A.8 Air conditioner blocked condenser air flow

parameters	base case value	reduced condenser air	delta value, % of base value
COP	4.31	4.12	-4.4 %
evaporator air flow, cfm	1195	1195	0 %
condenser air flow, cfm	2759	2187	-20.7 %
% mass flow thru system	100	100	0 %
refrigerant charge, lbm	8.05	8.05	0 %

The -4.4% change in COP is close enough to the target value of -5% for this run to be acceptable. The key parameter here is condenser air flow, which was reduced by 20.7%. This is remarkably close to the model-predicted value of -21% shown in Table 2.4.

A.3.2.4 Compressor leak

This experiment was done to simulate an internal high-to-low side leak in a scroll compressor. The compressor is hermetic, so a realistic leak couldn't actually be created, so a bypass line was installed around the compressor with a needle valve to regulate flow. Figure A.2 below shows a schematic diagram of the line.

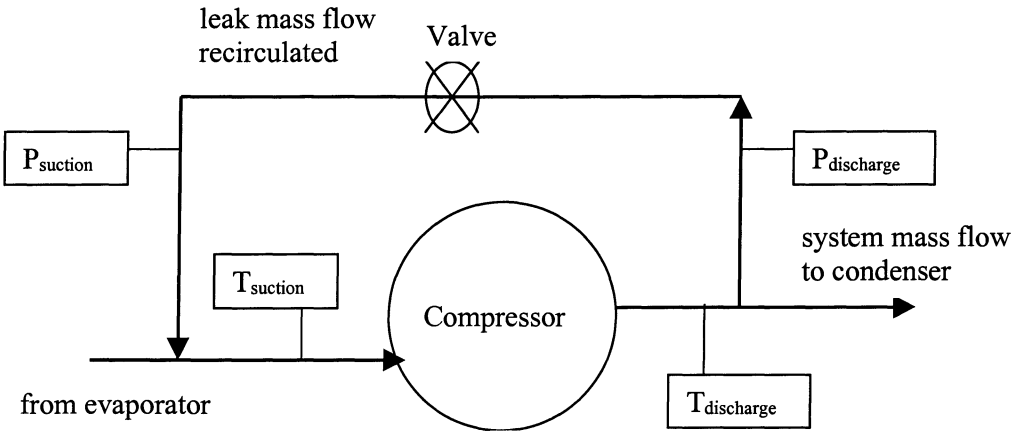

Figure A.2 Air conditioner compressor bypass schematic

When the valve is opened, a portion of the discharge gas recirculates back to the suction side, heats the suction gas, and decreases the compressor's capacity, presumably in a similar fashion as would an actual leak. Table A.9 shows COP and parameter values for this fault.

Table A.9 Air conditioner compressor leak

parameters	base case value	compressor leak value	delta value, % of base value
COP	4.31	4.07	-5.6 %
evaporator air flow, cfm	1195	1194	-0.0 %
condenser air flow, cfm	2759	2775	0.6 %
% mass flow thru system	100	90.0	-10.0 %
refrigerant charge, lbm	8.05	8.05	0 %

The COP decreased by 5.6%, which is sufficiently close to the target value. An explanation for the parameter "% mass flow through system" is as follows: in model runs, the key parameter for this fault is

$$\{(\text{system mass flow}) / (\text{system mass flow} + \text{mass flow}_{leak})\} \times 100\% \qquad [A.2]$$

where "system mass flow" refers to the non-leaking flow that is circulated as normal. So when there is no leak, the parameter value is 100%. Unfortunately with the instrumentation present there is no way to know exactly the mass flow that is bypassed around the compressor, so the above calculation cannot be made exactly. However, a good estimate can be made by assuming that the mass flow rate *through the compressor* is nearly the same in both the base case and fault case. This is not exactly true, as bypassed discharge gas will affect the pressure and temperature of the compressor suction gas. A correction factor was added to account for difference in densities. This way the leak mass flow may be approximated as:

$$\text{mass flow}_{leak} \approx \{\text{sys. mass flow}_{base} - \text{sys. mass flow}_{w/leak}\} \times (\rho_{fault}/\rho_{base}) \qquad [A.3]$$

This way if the mass flow into the compressor is less dense than in the base case, a good estimate is calculated. When a compressor leak was simulated with the model, the calculated parameter change given by equation [A.2] (as in Table 2.4) is -6.8%. The change given by equation [A.3] is -6.5%. These two values are fairly close, so equation [A.3] should be sufficient for experimental purposes.

A.3.2.5 Low charge

This experiment simulates the effect of a slow leak in an actual unit. It was performed by removing a measured mass of charge from the system, and allowing the unit to operate with less than the recommended charge. Table A.10 shows COP and parameter values.

Table A.10 Air conditioner low charge

parameters	base case value	low charge value	delta value, % of base value
COP	4.31	4.13	-4.2 %
evaporator air flow, cfm	1195	1194	-0.0 %
condenser air flow, cfm	2759	2791	1.2 %
% mass flow thru system	0	0	0 %
refrigerant charge, lbm	8.05	6.55	-18.6 %

The -4.2% change in COP is close enough to the target value of -5% for this run to be acceptable.

Appendix B

Jacobian Matrix Issues

B.1 Matrix Construction

The Jacobian matrix used in this project is a matrix of first derivatives. Its size is M × N, where M is the number of sensor locations being used and N is the number of faults to be detected. The matrix takes the form:

$$\begin{bmatrix} J_{1,1} & J_{1,2} & J_{1,3} & \cdot & \cdot & J_{1,N} \\ J_{2,1} & J_{2,2} & \cdot & \cdot & \cdot & \cdot \\ J_{3,1} & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ J_{M,1} & \cdot & \cdot & \cdot & \cdot & J_{M,N} \end{bmatrix}$$

Where each element represents a derivative:

$$J_{m,n} = \frac{\partial x_m}{\partial k_n} \qquad [B.1]$$

Theoretically, a derivative element is known precisely at a single point. In this project, however, such exact dependencies of variables on operational parameters are not known. Therefore point derivatives must be estimated numerically by running a model or experiment at two different operating conditions. So in this project, equation [B.1] becomes

$$J_{m,n} = \frac{\partial x_m}{\partial k_n} \approx \frac{\Delta x_m}{\Delta k_n} \qquad [B.2]$$

For example, in order to approximate the Jacobian element $\partial x_1/\partial k_1$, first the system would be simulated using a base case value of k_1 to obtain a value for x_1. Then k_1 would be changed to k_1' (while all other parameters are held constant) and the system would be simulated again, leading to some value x_1'. Then the derivative would be approximated as

$$\partial x_1/\partial k_1 \approx \Delta x_1/\Delta k_1 = (x_1' - x_1) / (k_1' - k_1) \qquad [B.3]$$

For simulation purposes described in Chapter 2, the values of k_m' are chosen to be values that cause the system energy use to increase by an amount ΔE_{crit} (refrigerator) or system COP to decrease by an amount ΔCOP_{crit} (air conditioner).

Note that an implicit assumption of this approach is that the behavior of the system between these two operating points is linear. Of course, a system with the complexity of a refrigerator or air conditioner is not expected to behave linearly over a large range of operation. It is believed, however, that for the purposes of this project the operational parameter range of interest is small enough that linearity is a reasonable assumption. This assumption was checked using the simulation models and is discussed later in Section B.3.

B.2 Matrix Normalization

As described in the previous section, each Jacobian matrix element is composed of some variable change divided by some parameter change. Unfortunately, simply plugging in values of Δx_m and Δk_n to each Jacobian element can lead to numerical problems, as well as confusion regarding the different units of each element. For example, in order to induce a similar increase in refrigerator energy use, the air flow rate over the condenser must decrease by approximately 60 cfm, whereas the heat transfer coefficient of the condenser must only decrease by 0.9 Btu/hr-°F, a difference of nearly 2 orders of magnitude. When dissimilar numbers like these are placed in a matrix together, an ill-conditioned matrix may result. Some measure of normalization is desired in order to obtain useful numerical results.

A normalization procedure has been chosen that will resolve any numerical problems as well as confusion concerning units. A standard 2σ normally distributed uncertainty is assumed to be known for all variable (x_m) measurements. Any Jacobian element may therefore be expressed in units of:

(# of standard deviations change in x_m) / (% change in k_n)

by using the following formula:

$$J_{m,n} = \{(x_{m,fault} - x_{m,base}) / \sigma_m\} / \{((k_{n,fault} - k_{n,base}) / k_{n,base}) * 100\%\} \qquad [B.4]$$

As an example of how this method helps keep Jacobian elements on the same order, consider an example:

When the air flow over the condenser is lowered by 60 cfm, the condenser outlet temperature rises about 4°F. Plugging in Δ(temperature)/Δ(air flow) gives

4°F / -60 cfm = -0.0667 °F/cfm.

Alternatively, when the heat transfer coefficient of the condenser decreases by 0.9 Btu/hr-°F, the compressor outlet temperature rises 3°F. Plugging in Δ(temperature)/Δ(h.t. coefficient) gives

3°F / -0.9 Btu/hr-°F = -3.333 °F/ Btu/hr-°F.

Placing these two numbers together in a matrix is not recommended, as one is nearly 10^2 larger than the other. Assuming σ for a temperature measurement is 0.5°F, and using equation [B.2] to first calculate the Jacobian element related to condenser air flow,

{(4°F) / 0.5°F} / {(60 - 120)*100% / 120} = -0.16

and next for the condenser heat transfer example,

{(3°F) / 0.5°F} / {(3.29 - 4.19)*100% / 4.19} = -0.28

The quotient of these two answers has been reduced from 50 to less than 2.0! Numbers such as these are more agreeable in computational analysis.

Note that when the Jacobian is normalized with the units demonstrated above, the inverse Jacobian will then have inverted units

(% change in k_n) / (# of standard deviations change in x_m)

Therefore the $\Delta\mathbf{x}$ vector must also be normalized with units of (# of standard deviations change in x_m) in order to make the units of the calculated $\Delta\mathbf{k}$ vector (% change in k_n). The Δk vector could then be normalized in a number of ways, one of which is to divide each k_n term its individual $\Delta k_{n,crit}$ value (shown in Tables 2.2 and 2.4, for example) so that a Δk_n calculation of zero means no fault is present, and a calculation of 1.0 means that the fault has reached a critical level.

B.3 Linearity assumption

As described in section B.1, each Jacobian element is determined with results from two system operating points. These parameter values at the base case level and at the ΔE_{crit} or ΔCOP_{crit} level can be treated essentially as "limiting" values in this project. The system operating point of interest for fault detection about which a derivative should be taken lies somewhere between those two values. The Jacobian matrix is essentially a linear model of the system's behavior, so it is beneficial to know whether the linearity assumption is a good one within the range of interest. Simulation results show that for the most part this assumption is acceptable. As an example, refrigerator model runs were performed allowing the value of the parameter $h_{air,cond}$, to vary from its base case value of 4.2 down to 2.8 Btu/hr-°F. Figure B.1 shows the response of two important sensor locations (condenser refrigerant and air outlet temperatures) within the interval of interest.

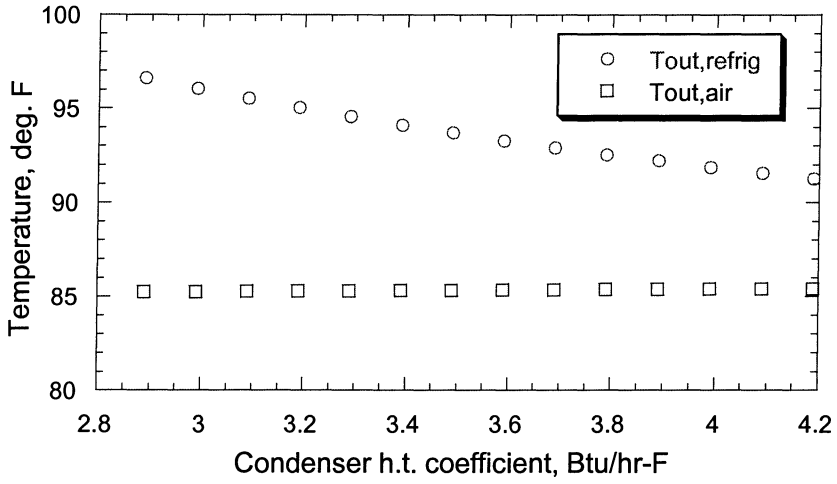

Figure B.1 Linearity demonstration, fouled refrigerator condenser

This plot demonstrates that it is reasonable to approximate these elements $\partial x_m/\partial k_n$ at any point of interest simply as $\Delta x_m/\Delta k_n$ because the plots are nearly linear. This type of trend was observed for most variables as each fault was simulated. There are occasions where this assumption breaks down, however. Figure B.2 shows a similar plot for the air conditioner case of low charge. Charge was varied from its base case value down to a level that caused a 5% decrease in COP.

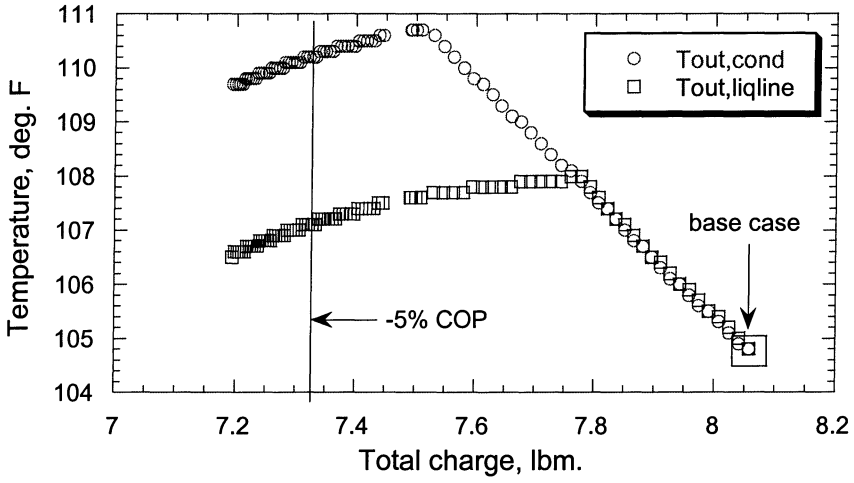

Figure B.2 Nonlinearity demonstration, air conditioner low charge

This plot illustrates an interesting nonlinear situation. Obviously the response of the two temperatures is not linear. An explanation for the "crook" in each temperature response is that as charge is taken out of the system, condenser subcooling decreases. This causes the increase seen in both temperatures, as the liquid line outlet is directly downstream of the condenser exit. Note that during the linear increase, the liquid line outlet temperature is practically equal to the condenser outlet. This is because the liquid line was modeled as adiabatic. At some point as charge is drained from the condenser, subcooling gets close enough to zero so that the wall friction causes the refrigerant to flash somewhere in the 25-ft. liquid line, leaving the liquid line to enter the TXV in the 2-phase state. Once there is 2-phase flow in the liquid line, the resulting pressure drop causes the refrigerant temperature to decrease as it goes through the rest of the line. Of course, as more charge is removed from the system, the condenser exit subcooling eventually goes all the way to zero (at approximately 7.5 lbm. in Figure B.2).

Figure B.2 serves as a caution against blindly assuming that all sensor responses will be nearly linear. There are certain conditions where the system is near some type of operational inflection point (such as a phase change) and a slight change in a parameter will cause it to cross that point.

A similar situation may be encountered in systems that have a clogged filter-drier located in the liquid line upstream of the TXV. At moderate operating conditions the valve would simply open wider to maintain a mass flow rate sufficient to control superheat at the desired setting. At more severe high-load conditions, however, the valve may hit its fully-open position and still be unable to compensate fully for the obstruction in the liquid line. The resulting system response would be easy to detect (increased superheat) but the discontinuous nature of the fault and response would be difficult to handle within our locally-linear FDD methodology.

Appendix C

Uncertainty Analysis

C.1 Introduction

The diagnostic system proposed here relies on measurements of certain system variables which will indicate whether operational parameters are at their normal values or have deviated. Of course in reality every sensor measurement has some inherent uncertainty/error. Of particular interest is whether changes in measured variables are detectable (greater than measurement uncertainty) at a time when energy use has risen beyond its threshold level. In addition to sensor measurements, any Jacobian matrix that is constructed experimentally or empirically must consist of measured variables and parameter changes that are calculated as functions of measured variables. Hence there is also uncertainty associated with the Jacobian (and its inverse). This combination of errors in both the Jacobian, **J**, and the residual vector, $\Delta\mathbf{x}$, ultimately contributes some uncertainty to any parameter change calculation, $\Delta\mathbf{k}$. That is, in the equation

$$\Delta\mathbf{k} = \mathbf{J}^{-1}\,\Delta\mathbf{x} \qquad [C.1]$$

uncertainties in $\mathbf{J}^{-1}$ and $\Delta\mathbf{x}$ mean that there must also be uncertainty in the $\Delta\mathbf{k}$ calculation.

C.2 Error propagation example

This section will illustrate the effect of error propagation by following a single Δk_n calculation from start to finish. For simplicity, model-generated results were used in this example. Refrigerator model output was generated at the following conditions: Ambient temperature = 75°F, freezer temperature = 5°F, fresh food compartment temperature = 45°F. A sensor set known to be of high quality was used to detect the six system faults (again for simplicity, load faults were omitted in this example). The sensor set was chosen based on criteria described in Chapter 3. Table C.1 shows the included sensors, along with the assumed (2σ) errors associated with their measurements.

Table C.1 Variable information for sensor set

Sensor	2σ Uncertainty
T_{Shell}	1.0 °F
$T_{LiqLineOut}$	1.0 °F
T_{EvapIn}	1.0 °F
T_{CompIn}	1.0 °F
$TA_{CondOut}$	1.0 °F
RT_{calc}	1.0 %

The Jacobian matrix used for this example was constructed using the sensor set listed above and the six (system fault) parameter changes listed in Table 2.2.

C.2.1 Jacobian uncertainty

As discussed earlier, every element in the Jacobian matrix has some uncertainty associated with it because each partial derivative element was calculated as $J_{m,n} = \Delta x_m/\Delta k_n$ based on measured (or in this example, simulated) variables. The Jacobian takes the form:

$$\begin{bmatrix} J_{1,1} & J_{1,2} & \cdot & \cdot & \cdot & J_{1,6} \\ J_{2,1} & J_{2,2} & \cdot & \cdot & \cdot & \cdot \\ J_{3,1} & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ J_{6,1} & \cdot & \cdot & \cdot & \cdot & J_{6,6} \end{bmatrix}$$

Note that each element in the Jacobian is calculated from normalized variables and parameters (as discussed in Appendix B), and has units:

(# of standard deviations change in x_m) / (% change in k_n).

There is an applicable statistical formula for finding the uncertainty for this quotient of a Δ(variable) measurement and a Δ(parameter) estimate (based on variable measurements). It is given by Figliola and Beasley (1991) as:

$$u_{J_{m,n}} = \pm\sqrt{\left(\frac{\partial J_{m,n}}{\partial x_m}\cdot u_{x_m}\right)^2 + \left(\frac{\partial J_{m,n}}{\partial k_n}\cdot u_{k_n}\right)^2} \qquad [C.2]$$

where the "u" terms represent 2σ measurement uncertainties. Table C.1 lists the assumed variable uncertainties ($u_{x,m}$). For simplicity, parameter change uncertainties ($u_{k,n}$) were assumed to be ±10% of that particular Δk_n value. For example, according to Table 2.2, for the frosted evaporator fault, airflow was reduced by 20%, from 58.4 to 46.4 cfm (Δk value = -12.0 cfm). The 2σ uncertainty, 10% of that Δ value, is 1.2 cfm, which is 2% of the original 58.4 cfm. Therefore the stochastic value of Δk_n is (-20 ± 2)% of the base case value. This level of accuracy was assumed to be reasonable for carefully controlled experiments.

Continuing the frosted evaporator example, when the airflow was reduced by 20%, the compressor inlet temperature decreased by 2.8°F (= -5.6σ). This particular Jacobian element is calculated as (-5.6σ / -20% = +0.27 σ/%). The partial derivative terms in equation [C.2] were calculated numerically: This particular element has a magnitude of 0.27 σ/%, so by separately perturbing the Δx and Δk terms it can be shown that $\partial J/\partial x = 0.05$ (1/%) and $\partial J/\partial k = 0.0135$ (σ/%²). Plugging these numbers into equation [C.2] results in:

$$u_{J_{m,n}} = \pm\sqrt{(0.05(1/\%)\cdot 2\sigma)^2 + (-0.0135(\sigma/\%^2)\cdot 2\%)^2} \approx \pm 0.10\sigma/\%$$

A Monte Carlo technique was used for further illustration of Jacobian uncertainty. A distribution for the same Jacobian element was found by choosing random normally distributed values of Δx within the range (5.6 ± 2) σ and values of Δk within the range

(20 ± 2) %, and creating a large number of values for that element, thereby producing a distribution centered about the mean value of 0.27, shown below in Figure C.1.

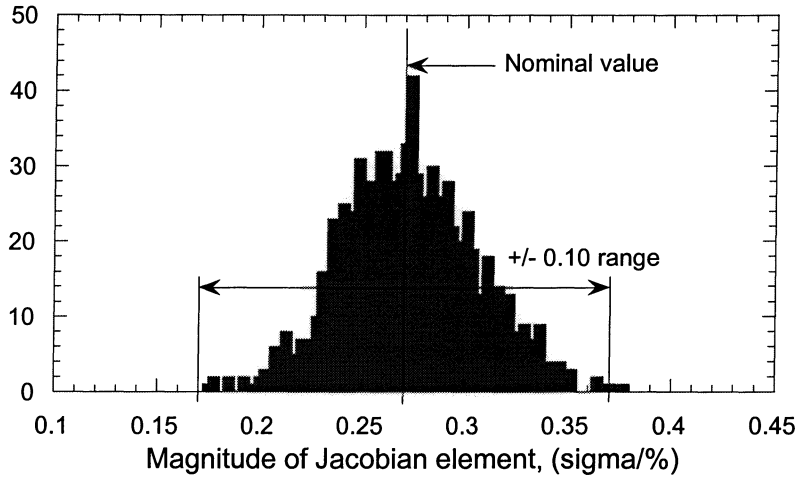

Figure C.1 Jacobian element uncertainty distribution

The figure shows that the distribution is in fact centered about 0.27 and covers a 2σ range of roughly ± 0.10 σ/%, confirming the accuracy of the estimate obtained above. The results of this analysis indicate that the uncertainty of this Jacobian element (and for most elements) is substantial compared to the nominal value.

Of course, the true utility of the Jacobian matrix comes only after it is inverted, as shown in equation [C.1]. Note that each element in the inverse Jacobian has units associated with it that consist of

(% change in k_n) / (# of standard deviations change in x_m)

Each element of the inverse matrix quantifies the percent change in a parameter k_n due to a single standard deviation change in a single variable x_m. Instead of using more detailed statistics equations as in Section C.2.1, a Monte Carlo analysis was the easiest way to determine a distribution for an inverse Jacobian element. This was accomplished by inverting each Jacobian matrix that was created in the previous step. For example, if 1000 random values are chosen for Δx_m and Δk_n, then 1000 values are known for each

Jacobian element, hence 1000 Jacobians can be produced and inverted. Each element of the inverse matrix will then have 1000 different distributed values centered about some nominal. This was in fact done for the Jacobian discussed in this Appendix, then (as described in Section C.2.2) the stochastic inverse was multiplied by a stochastic $\Delta\mathbf{x}$ vector in order to calculate the Δk_n value of interest (in this case evaporator airflow, to continue the previous example).

C.2.2 Parameter calculation uncertainty

Consider an example calculation of a single Δk_n. The deterministic equation is given by:

$$\Delta k_n = J^{-1}_{n,1}\,\Delta x_1 + J^{-1}_{n,2}\,\Delta x_2 + ... + J^{-1}_{n,M}\,\Delta x_M \qquad [C.3]$$

Equation [C.3] doesn't account for any error, though. After the uncertainty distribution for each inverse Jacobian element was found, another Monte Carlo simulation was used to evaluate the Δk_n values from equation [C.1]. The inverse Jacobian matrix must be multiplied by the $\Delta\mathbf{x}$ vector, each element of which is a measured variable and has an inherent uncertainty of 2σ, in order to calculate the $\Delta\mathbf{k}$ vector. Values within the individual $\mathbf{J^{-1}}$ and $\Delta\mathbf{x}$ normal distributions were chosen randomly and multiplied together, thereby creating a distribution for each calculated Δk_n element. To clarify, each Δk_n was calculated using:

$$\Delta k_n = (J^{-1}_{n,1}+\beta_{n,1})(\Delta x_1+\delta_1) + ...+ (J^{-1}_{n,M}+\beta_{n,M})(\Delta x_M+\delta_M) \qquad [C.4]$$

where δ_m is a normally distributed measurement error in Δx_m and $\beta_{n,m}$ is the error in $J^{-1}_{n,m}$. A Δk_n distribution is found by calculating equation [C.4] many times, using independent randomly selected valued of δ_n and $\beta_{n,m}$ each time. Figure C.2 was produced as such. It shows the uncertainty distribution of evaporator air flow using the sensor set listed in Table C.1, where the nominal Δk value is -20%.

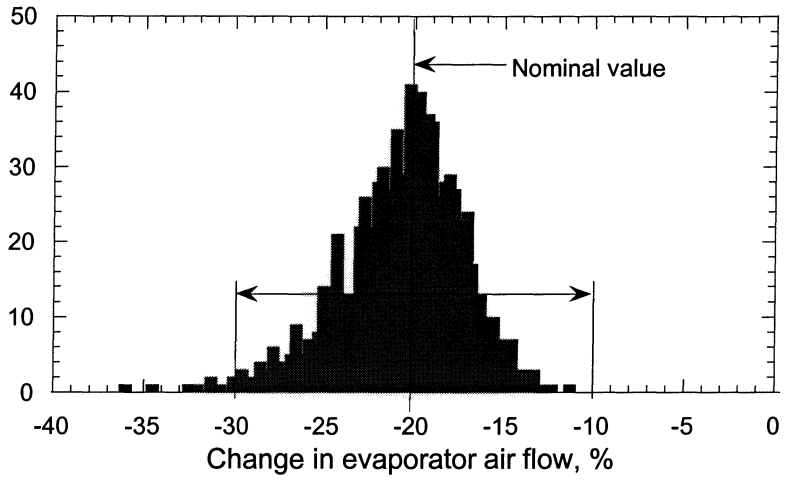

Figure C.2 Uncertainty distribution for calculated Δk element

The plot shows that the calculated fault distribution for the evaporator air flow has a value of approx. (-20 ± 10) % of the base case value. Figure C.2 illustrates the importance of the nominal Δk_n value being greater than the uncertainty range.

Appendix D

Load versus System Faults

D.1 Introduction

Two types of faults were observed during the refrigerator study. The first type of observed faults were system faults. Most of the faults reported on here fall into this category. This type may affect any temperature or pressure along the refrigerant loop, as well as compressor run time. An example of this type of fault is a dirty condenser. The heat transfer ability of the condenser is reduced, forcing condensing temperature to increase, as the system attempts to offset the loss of heat transfer surface area.

The second type, load faults, does not affect any of the temperatures or pressures along the refrigerant loop. This type of fault simply makes the system run longer than normal. The type of load fault considered here is a freezer or fresh food compartment gasket leak in the refrigerator. A load fault may occur in an air conditioning system as well (such as an open door/window or open shades/blinds), but they were not studied here because they are normal occurrences rather than faults. The bad gasket allows more ambient heat to leak into the compartment than normal. The compressor then must run longer to keep that compartment cool, so this fault would show up analytically in the compressor's run time.

The difference between these two types of faults shows up in the Jacobian matrix. The following section describes how.

D.2 Mathematical difference

Section D.1 stated that the two load faults did not affect any of the temperatures or pressures along the refrigerant loop. When a sensor reading does not change upon fault induction, it means that sensor's $\partial x_m/\partial k_n$ element in the Jacobian matrix is zero. Therefore for the two load faults every $\partial x_m/\partial k_{load}$ element in the Jacobian, with the exceptions of ∂(RunTime)/∂k and ∂(damper position)/∂k, is expected to be zero. None of the other sensors listed in Table 2.1 can give any information related to those two faults.

As an illustration, equation [D.1] below shows an expanded version of equation [2.1] and includes all 13 of the candidate refrigerator sensor locations.

$$\begin{bmatrix} J_{1,1} & J_{1,2} & \cdot & \cdot & \cdot & J_{1,6} & 0 & 0 \\ J_{2,1} & J_{2,2} & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ J_{3,1} & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & J_{12,7} & J_{12,8} \\ J_{13,1} & \cdot & \cdot & \cdot & \cdot & J_{13,6} & J_{13,7} & J_{13,8} \end{bmatrix} \cdot \begin{bmatrix} \Delta k_1 \\ \Delta k_2 \\ \Delta k_3 \\ \Delta k_4 \\ \Delta k_5 \\ \Delta k_6 \\ \Delta k_7 \\ \Delta k_8 \end{bmatrix} = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \\ \Delta x_4 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \Delta(RT) \\ \Delta(f_z) \end{bmatrix} \qquad [D.1]$$

Suppose the x_{12} sensor is RunTime, the x_{13} sensor is damper position, and faults k_7 and k_8 represent load faults. Only two elements $\partial(RT)/\partial k$ (= $J_{12,7}$ or $J_{13,7}$) and $\partial(f_z)/\partial k$ (= $J_{12,8}$ or $J_{13,8}$) in columns 7 and 8 of the Jacobian are nonzero. Hence the only sensors to which Δk_7 and Δk_8 can contribute are x_{12} and x_{13}. Consider a situation where, in a set of 8 sensors (to detect 8 faults), RunTime and damper position were not included. In that case, the last two rows of the Jacobian matrix in equation [D.1] would not be present either, meaning that two columns of the Jacobian would consist entirely of zeros (since none of the included sensors would be affected by load faults). Then Δk_7 and Δk_8 would always be calculated as zero, regardless of how badly a gasket was really leaking.

This example shows a benefit of the methods for choosing sensors described in Chapter 3. Checking the condition number of a matrix will guard against the scenario described above. Mathematically speaking, a matrix with an entire column of zeros has an infinite condition number, so it would not be chosen as a feasible set of sensors.

Unfortunately, columns 7 and 8 of equation [D.1] (mostly zeros) tend to increase the condition number of the matrix. An option that has been considered is to treat load faults and system faults separately. That is, in the case of using 8 sensors to detect 8

faults, use a 6×6 Jacobian to detect system faults and a 2×2 Jacobian to detect load faults. In a setup like this, system faults would be checked first. If none were found, the second Jacobian would then be used to find load faults. The problem with this setup is that the ability to detect simultaneous faults would be lost. If both a system and load fault existed at the same time, any effect of the system fault on RunTime and/or damper position would hinder the ability to detect the load fault. Perhaps this idea deserves more consideration and research, but it is disregarded in this report.

Appendix E

Details of RMS Method for Evaluating Sensor Sets

While discussing the value of the RMS "sum-of-squares" measure for sensor contributions, Section 3.3 stated that for a square matrix the lowest possible value is 1. This Appendix shows the proof of that statement. If the RMS value is equal to 1, it implies that every sensor is contributing equally to the detection of each fault. By calculating the RMS value for a sensor set, a user can see how close the set is to this ideal case.

Consider the following matrix:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,M} \\ a_{2,1} & a_{2,2} & & \vdots \\ \vdots & & & \vdots \\ a_{N,1} & \cdots & \cdots & a_{N,M} \end{bmatrix}$$

where each $a_{m,n}$ term is the product of $((\partial k_n/\partial x_m)\Delta x_m)/(\Delta k_{n,crit})$, similar to the terms shown in equation [3.1], but normalized by the $\Delta k_{n,crit}$ term. Therefore the sum of any row m is actually $\Delta k_n / \Delta k_n = 1$. We wish to minimize the quantity (arbitrarily called Ψ):

$$\Psi = \sqrt{\sum_{m=1}^{M}\sum_{n=1}^{N} a_{n,m}^2} \qquad [E.1]$$

subject to the constraints ϕ (N total constraints):

$$\phi = \sum_{m=1}^{M} a_{n,m} - 1 = 0 \qquad [E.2]$$

Equation [E.2] states that each row must sum to 1. This is true because of the normalization of each $a_{m,n}$ term discussed above.

To simplify this problem, we will minimize Ψ^2 (and cancel out the square root symbol in equation [E.1]) rather than Ψ, as the result will be same. Also in the interests of simplification, we will minimize Ψ^2 for only one row, since each constraint is a row constraint. For this task the method of LaGrange multipliers may be used. Stoecker (1989) gives a useful description and demonstration of LaGrange multipliers and their applications. The general form of the LaGrange multiplier equation is:

$$\nabla\Psi - \sum\lambda\nabla\phi = 0 \qquad [E.3]$$

where λ is a LaGrange multiplier of unknown value. From equation [E.1],

$$\nabla(\Psi^2) = \sum 2a_{n,m} \qquad [E.4]$$

and from equation [E.2],

$$\nabla\phi = 1 \qquad [E.5]$$

for all $a_{n,m}$. The accompanying LaGrange equations (M equations per row) are:

$$\begin{aligned} 2a_{n,1} - \lambda(1) &= 0 \\ &\ldots \\ 2a_{n,M} - \lambda(1) &= 0 \end{aligned} \qquad [E.6]$$

Solving equations [E.2] and [E.6] simultaneously gives the minimum value of Ψ^2 as $1/(M^2)$ for a single row, when all of the $a_{n,m}$ elements are equal to 1/M. This illustrates that the minimum value of Ψ is:

$$\Psi_{min} = N*(1/M) \qquad [E.7]$$

Equation [E.7] is generalized for all matrices. In the case of square matrix where N=M, $\Psi_{min} = 1$.

Appendix F

Simulation Model Changes

The model that was used to simulate air conditioner faults was developed by Mullen, et al. (1998). The original model was unable to simulate the two compressor faults considered in this study: low motor efficiency and compressor leakage. Therefore some equations and variables had to be added to accommodate these simulations.

F.1 Low motor efficiency

The motor within the compressor could run inefficiently at times and demand more power than should be necessary to perform the same amount of compression work. The consequence of this fault would be extra heat generated by the motor. This would in turn heat the compressor's plenum gas, oil, and shell. In order to simulate this fault, a parameter was added to an existing equation. The parameter is called "betapwr" and is simply a scaling factor that is multiplied by the compressor power term:

```
PwrComp = pwrmap(tsatincomp,tsatoutcomp,pc)*betapwr
```

where "pwrmap" is function call which calculates compressor power based on compressor inlet and outlet saturation temperatures (which are effectively evaporating and condensing pressures). A model user can leave the compressor power unchanged by setting betapwr = 1.0, or simulate an inefficient motor by setting betapwr > 1.0. When the motor draws more power than normal, it will dissipate more heat than normal, as seen in the following energy balance on the motor:

```
PwrComp = qMotor + Pwrshaft
```

where "Pwrshaft" is the mechanical work needed to turn the shaft and "qMotor" is the extra power dissipated as heat rejected to the plenum gas and oil. Both these terms were added to the model, but they appear in the same equation. Another new equation will be

introduced in Section F.2 to recover both terms. Figure F.1 below shows a diagram that illustrates these motor terms.

F.2 Compressor leak

Appendix A described the compressor bypass line that was installed in the experimental setup. This is an artificial representation of the fault, though. An actual high-to-low side leak in a scroll compressor would actually happen within the sealed compressor shell as shown in Figure F.1.

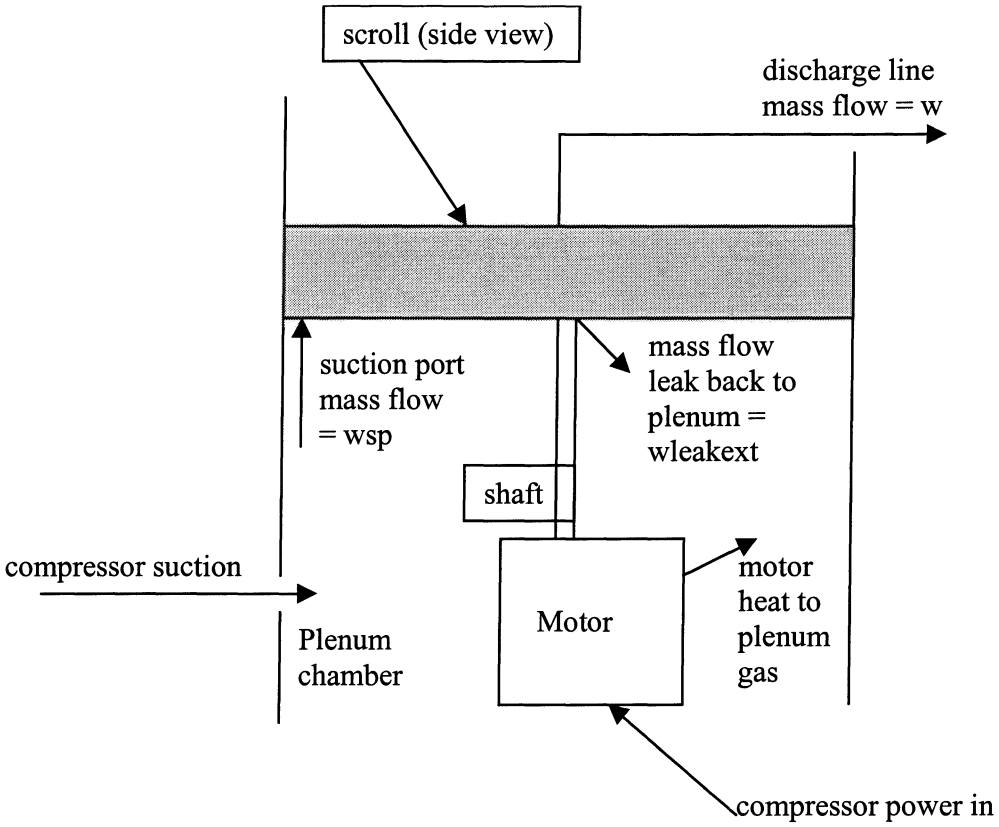

Figure F.1 Compressor simulation schematic

Refrigerant could actually leak past the plate separating the high and low sides of the compressor. This type of leak is difficult to model in both simulation and experiment, so the model simulation refrigerant loop was modified essentially in the same way as the

experimental loop. Figure F.2 shows a schematic of the compressor region of the loop that is simulated by the model. Note the similarity to Figure A.2.

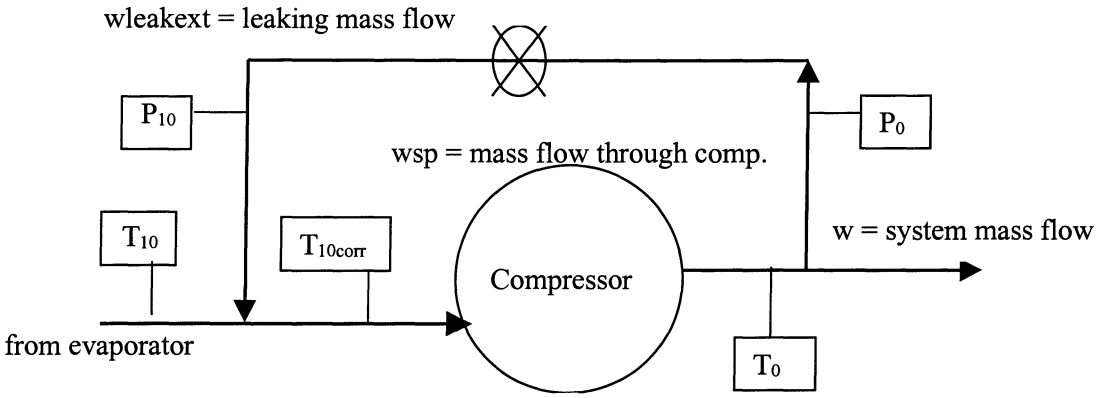

Figure F.2 Compressor simulation bypass line

This fault was modeled simply by specifying how much mass flow was leaking. A parameter called "wleakext" was added to the model. A mass flow balance at either of the bypass line junctions in Figure F.2 gives three mass flow rates: total system mass flow (w), leakage to plenum (wleakext), and the sum of those, mass flow through the compressor (wsp). These three terms combine to form the equation:

```
w = wsp - wleakext
```

where "wsp," the mass flow through the compressor suction port, is a new variable. An energy balance at the same junction gives:

```
w*h10 + wleakext*h0 = wsp*h10corr
```

where "h10corr" represents the corrected compressor suction enthalpy, which is a bit higher than normal due to the leak.

In order to recover the "qMotor" term introduced in Section F.1, an energy balance was done on the plenum gas:

```
qComp + wsp*hsp = qMotor + wsp*h10corr
```

where "hsp" is the enthalpy of the mass flow through the suction port, assumed to be at the temperature of the plenum gas. Since the temperature of the plenum gas is unknown, an assumption was made that it was at the same temperature as the compressor shell and at the same pressure as the suction gas:

```
hsp = hpt(p10,Tshell)
```

This assumption seems reasonable because the inside of the shell is continuously being splattered with oil droplets which also come into contact with the plenum gas, so the heat transfer resistance on the inside to the hot oil and gas is probably much less than that on the outside to the air.

Appendix G

Refrigerator Results

G.1 Simulation results

Chapter 4 discussed reasons that refrigerator results are addresses seperately here in this Appendix. They are presented here just as air conditioner results were presented in Chapter 4.

G.1.1 Single faults

Figures 3.8 and 3.9 showed that the refrigerator Jacobian producing the lowest calculation error (about 25%) has a condition number of approx. 37 and an RMS value of 2.6. Table G.1 below lists the sensor set used to construct this Jacobian.

Table G.1 Refrigerator set of 8 sensors with lowest calculation error

Cond. # = 37 RMS = 2.6
T_{Dis}
$T_{CondOut}$
T_{EvapIn}
$T_{EvapOut}$
T_{CompIn}
$TA_{CondOut}$
RunTime
damper position

It was stated that the average calculation error (described in Section 3.4.3) for this sensor set is approximately 25%. Table G.2 shows the numbers from which that figure (25%) was calculated. It lists the width of each 90% confidence interval (see Figure 3.7 for graphical representation of confidence interval) in the form: Δ(parameter) = [nominal value ± $^1/_2$ width of C.I.] %. Note that all of the nominal values in the table are expected to be exact (the same critical fault magnitudes as listed in Table 2.2). The reason is that the inverse Jacobian $\mathbf{J}^{-1}$ was multiplied by the same set of $\Delta\mathbf{x}$ vectors that was used to create the Jacobian **J** in the first place. The 90% confidence intervals are simply an

indication of how much uncertainty is involved even when the nominal calculated value is perfect.

Table G.2 Set of 90% confidence intervals, best set of 8 sensors

simulated fault:	clogged captube		worn compressor		low motor efficiency		frosted evap		fouled cond		low cond airflow		ff gasket leak		frez gask leak	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-
cap exit area	-16	2	0	2	0	2	0	2	0	2	0	2	0	2	0	2
mdot map	0	13	-8	12	0	10	0	11	0	12	0	11	0	11	0	11
power map	0	5	0	5	6	5	0	5	0	4	0	5	0	5	0	4
evap airflow	0	17	0	14	0	15	-20	14	0	15	0	13	0	14	0	13
cond h_{air}	0	17	0	15	0	14	0	14	-21	16	0	14	0	14	0	15
cond airflow	0	9	0	9	0	9	0	8	0	8	-50	8	0	8	0	7
UA_{ff}	0	25	0	23	0	22	0	22	0	23	0	22	20	22	0	22
UA_{fz}	0	27	0	26	0	24	0	24	0	24	0	23	0	23	7	25

Each column of Table G.2 (low evap air flow, low cond air flow, etc.) represents the result of the inverse Jacobian multiplied by a single $\Delta\mathbf{x}$ vector, generated by the simulation of that particular fault. The left part of each column shows the nominal calculated value of each parameter, the right part shows the [±] value, or ½ the width of the 90% confidence interval. The shaded boxes highlight the parameter calculations that are expected to be nonzero, based on the fault present. Refer to Section 4.1.1 for an example showing how to read the table effectively.

This Jacobian's average calculation error (25%) was calculated by averaging the width of every confidence interval (= 2 × [± value]). Note that, as seen in air conditioner results, the [±] values for each individual calculated parameter are fairly consistent no matter which particular fault has been simulated. For example, regardless of which fault is simulated in Table G.2, the [±] value for the UA_{fz} parameter is always between 23% and 27%. Unfortunately, the nominal value of UA_{fz} in the case of a freezer gasket leak is +7%. This implies that, even when using the best set of 8 sensors, there are two possible negative occurrences: 1) there is no fault present, but due to the width of the confidence interval a false positive is generated; and 2) a freezer gasket leak has reached a critical level, but there is the distinct possibility that the FDD system will not recognize a fault at

all. The same type of conclusion can also be seen in the case of a worn compressor and a fresh food gasket leak. In the case of a fault where $\Delta k_{crit} > [^1/_2$ the confidence interval width], neither of the situations mentioned above would ever happen. There would still be the possibility of early/late detection (as discussed in Section 3.2.3), but in that case the question is not whether a fault is present, but what level of severity it is at.

The next issue of interest is the effect that the addition of extra sensors has on fault detectability. With this goal in mind, an exhaustive search was performed of all possible sets of 9 sensors in search of the one with the lowest average calculation error. Figures G.1 and G.2 below show average calculation error vs. condition number and RMS value, respectively, for sets of 9 sensors.

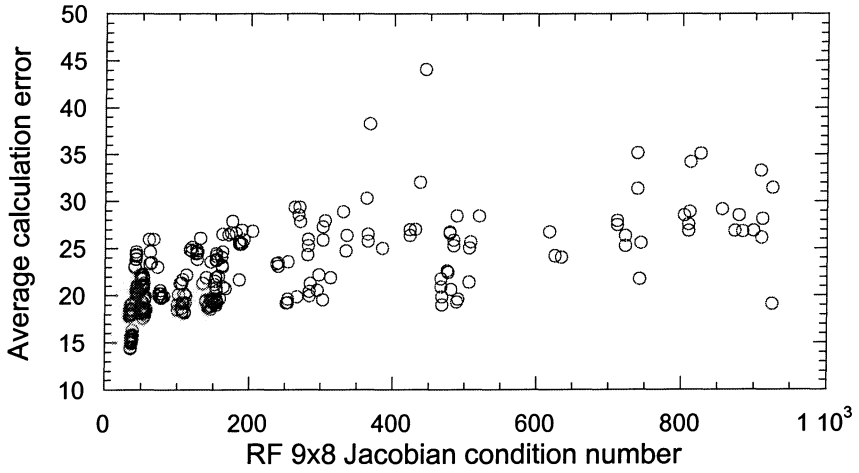

Figure G.1 Average calculation error vs. condition number (refrig, 9 sensors)

The same trend is apparent here as was in Figure 3.8, but more sets seem to be clustered toward the lower left corner of the plot (note that the y-axis is from 0-50, where in Figure 3.8 it was 0-100). This makes sense, as all sensor sets will benefit from the inclusion of an extra sensor.

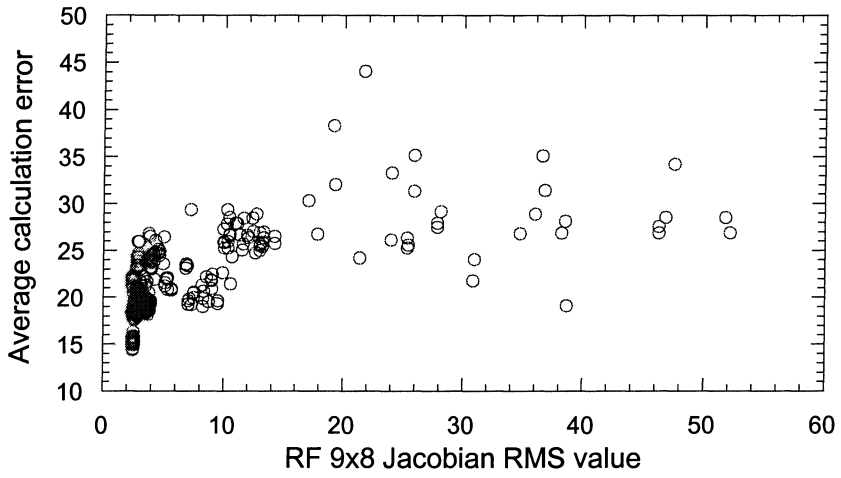

Figure G.2 Average calculation error vs. RMS value (refrig, 9 sensors)

Again, this plot's shape is similar to that of Figure 3.9. The sensor set that shows the lowest calculation error (approximately 14%) is shown in Table G.3 below.

Table G.3 Refrigerator set of 9 sensors with lowest calculation error

Cond. # = 35 RMS = 2.5
T_{Dis}
$T_{CondOut}$
$T_{LiqLineOut}$
T_{EvapIn}
$T_{EvapOut}$
T_{CompIn}
$TA_{CondOut}$
RunTime
damper position

The average error went down about 9% by adding one sensor. Note that the set is the same as that shown in Table G.1, but with the liquid line outlet temperature added. The reason that detection accuracy improved is because the liquid line outlet and condenser outlet temperatures are closely related (they are separated only by the nearly adiabatic liquid line). The addition of this redundant sensor took half of the burden off the condenser outlet sensor, allowing more equal contributions, as discussed in Section

3.5). The addition of one sensor helped detection accuracy quite significantly, thus one might wonder how well the FDD method could perform with even more sensors. To answer this question, all 13 of the candidate sensor locations (listed in Table 2.1) were used to detect faults. When all 13 sensors were used, an average calculation error of 9.1% resulted. It seems that the addition of one extra sensor goes a long way toward increasing detection accuracy, but the value of extra sensors drops quickly after the first, at least for this particular refrigerator.

G.1.2 Multiple faults

The refrigerator sensor set listed in Table G.1 was used as a further test of the ability of the FDD method to detect simultaneous multiple faults. Four separate simulation results are listed below in Table G.4. They were chosen with the purpose of representing simultaneous system faults, simultaneous load faults, and one of each. The four cases listed refer to the following parameter conditions:

Case 1: Evaporator air flow (-17%) and condenser air flow (-50%)
Case 2: Captube exit area (-14%) and freezer UA (+13%)
Case 3: Fresh food compartment UA (+15%) and freezer UA (+13%)
Case 4: Captube exit area (-14%) and condenser air flow (-33%)

Table G.4 Set of 90% confidence intervals, multiple fault cases

Case #:	1		2		3		4	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-	nom	+/-
cap exit area	1	2	-15	3	0	3	-14	3
mass flow map	-1	13	-3	16	-2	22	2	17
power map	-1	6	-1	7	0	8	1	6
evap airflow	-17	17	2	21	-2	24	9	20
cond h_{air}	0	17	-3	22	-2	25	1	22
cond airflow	-48	11	-1	11	-1	14	-23	11
UA_{ff}	-2	26	-2	31	10	38	-4	33
UA_{fz}	-2	30	7	37	11	42	6	37

The two fault values are highlighted in each column of Table G.4 for easier reference. The results show good news and bad. The uncertainty ranges for each fault are similar to those shown in Table G.2 for single faults, so the occurrence of extra faults does not seem to affect the width of the Δk_{calc} 90% confidence interval, with the exception of the two load fault UA parameters. The width of their confidence intervals have grown significantly from the single fault cases of Table G.2.

An interesting case occurs in column 1 of Table G.2. The parameter "evap air flow" shows equal magnitudes for both the nominal and [±] values. In this case there remains a 5% chance of a false positive indication, as the [±] value is based on a 90% confidence interval.

Most of the calculated fault parameter values are close to what was specified (for example, captube exit area in cases 2 and 4, evaporator airflow in case 1). A couple are not as close, such as UA_{fz} in case 2 (under-predicted at +7%, should be +13%) and condenser airflow in case 4 (also under-predicted at -23%, should be -33%). Some of the parameter calculations that are expected to be zero are not as close to zero as in the single fault cases (see Table G.2), for example UA_{fz} and evaporator airflow in case 4.

It seems that in the case of multiple faults some are still as detectable as in the single fault cases, but others (notably load faults, which were not detectable even in the single fault cases) are negatively affected by the presence of other faults and nonlinearity effects.

G.2 Experimental results

A number of the faults that were simulated with models were also induced experimentally in the laboratory. Table G.5 below lists these faults for the refrigerator.

Table G.5 Experimentally induced faults (refrig)

Refrigerator
frosted evaporator
reduced condenser air flow
fouled condenser
fresh food gasket leak
freezer gasket leak

Appendix A describes each fault experiment that was performed and lists the calculated parameter values for each test. Table G.6 below lists the "critical" fault levels (parameter changes) observed in experiments. These parameter values, along with variable values, were used to construct experimental Jacobian matrices in the same fashion as was done with model results.

Table G.6 Experimental critical fault parameter changes (refrig)

Refrigerator	
parameter	Δk_{crit}
evaporator air flow	-20%
condenser air flow	-32%
$h_{air,cond}$	-16%
fresh food UA	19%
freezer UA	21%

Experimental data for refrigerator runs was taken under the following conditions: ambient temperature = 75°F, freezer temperature = 5°F, fresh food compartment temperature = 45°F. These were chosen because they represent a real refrigerator's typical operating condition.

As in the air conditioner case, an exhaustive search was performed on all possible sets of 5 sensors for those with the lowest condition number, RMS value, and average calculation error. The following plots are similar to those seen in Chapter 3. Figure G.3 shows RMS value as a function of condition number.

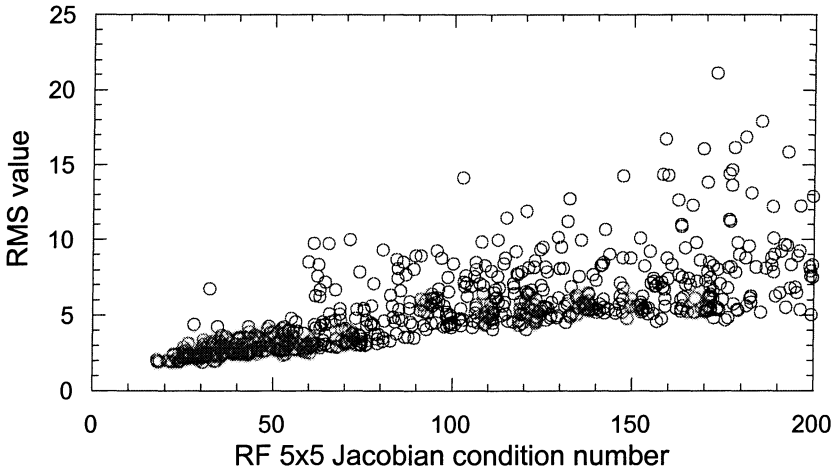

Figure G.3 RMS value vs. condition number (refrig experiments)

It appears that the same type of trend that was seen in simulation results is seen here as well. There is a general trend of the best conditioned matrices also having the lower RMS values, but there are also many sets with large condition numbers that have low RMS values. Figure G.4 shows average calculation error (described in Section 3.4) as a function of Jacobian condition number. Figure G.5 shows average calculation error as a function of RMS value.

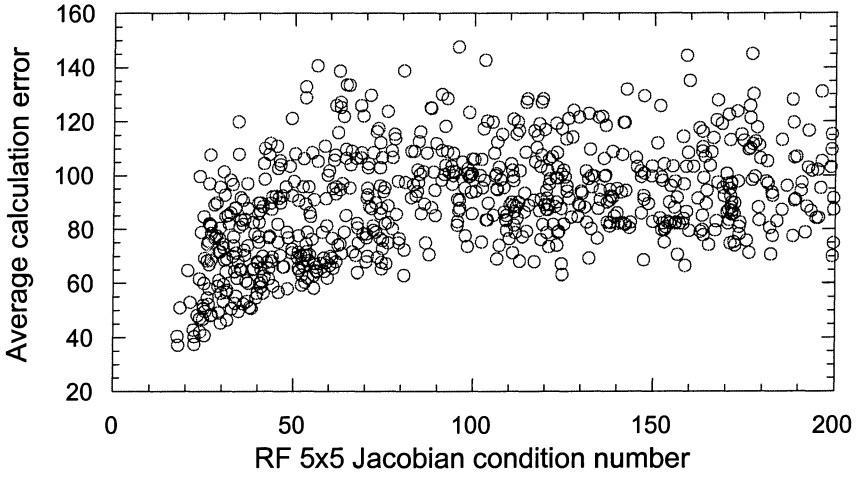

Figure G.4 Average calculation error vs. condition number (refrig experiments)

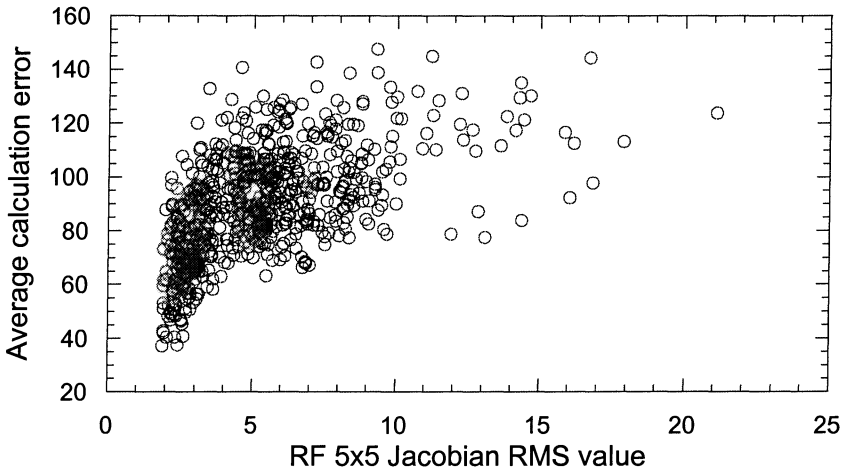

Figure G.5 Average calculation error vs. RMS value (refrig experiments)

Both Figures G.4 and G.5 show results similar to those seen in simulations. Namely, that the matrices with the lowest values of condition number and RMS tend to give more accurate FDD results. These results were used to construct a 5×5 Jacobian for detecting the five refrigerator faults listed in Table G.5. The five sensors used to construct the Jacobian were chosen such that the Jacobian gives the minimum average calculation error. Table G.7 lists the sensor set used.

Table G.7 Experimental refrigerator sensor set

5 refrigerator sensors
Cond # = 24.8, RMS = 2.24
W_{Comp}
T_{Dis}
T_{CompIn}
RunTime
damper position

The detection accuracy of this Jacobian was tested as in Section 4.1, with 90% confidence intervals. Table G.8 below shows how accurately each fault (and non-fault) was detected.

Table G.8 Set of 90% confidence intervals, refrigerator experiments

fault:	frosted evap		low cond airflow		fouled condenser		ff gasket leak		frez gask leak	
calculated Δk_n, %	nom	+/-	nom	+/-	nom	+/-	nom	+/-	nom	+/-
evap airflow	-21	10	0	10	0	9	0	11	1	10
cond airflow	0	6	-32	7	1	7	0	7	0	7
cond h_{air}	0	15	1	15	-16	14	1	16	0	14
UA_{ff}	1	39	0	41	0	42	18	44	1	40
UA_{fz}	0	16	1	17	1	16	1	15	21	14

The width of the confidence intervals for the three system faults look to compare well with those predicted with model results. The load faults, however, have wider distributions than observed in the model case. This may be explained by inconsistency in the RunTime and damper position experimental "sensors." Recall that in the steady-state experiments, actual compressor RunTime was 100% and f_z was constant. Calculations were made off-line using readings from compartment heaters to estimate values of RT_{calc} and $f_{z,calc}$ (see Chapter 2 for a description of the heaters and their purpose). Unfortunately, as discussed by Kelman and Bullard (1999), the compartment heaters used in the refrigerator experiments gave consistently suspect readings. For this reason, no firm conclusions could be drawn, so the question was investigated using the a/c test facility (as reported in Chapter 4), which had more accurate instrumentation.