Development, Validation, and Application of a Refrigerator Simulation Model

R. J. Woodall and C. W. Bullard

ACRC TR-99

June 1996

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 66
Refrigerator Systems Analysis
C. W. Bullard, Principal Investigator

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.
Dayton Thermal Products
Delphi Harrison Thermal Systems
Eaton Corporation
Electric Power Research Institute
Ford Motor Company
Frigidaire Company
General Electric Company
Lennox International, Inc.
Modine Manufacturing Co.
Peerless of America, Inc.
Redwood Microsystems, Inc.
U. S. Army CERL
U. S. Environmental Protection Agency
Whirlpool Corporation

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

This report describes the further development and validation of the Refrigerator/Freezer Simulation (RFSIM) model. The reports also describes the first major application of the model as an analysis tool for new refrigerator designs; several aspects of multi-speed compressor operation were examined with the model. Several improvements were made to the model that facilitated the validation process and the examination of multi-speed compressors: the model was made more general so that it could operate in numerous configurations in addition to the original design and simulation modes; many improvements were made in the modeling logic and robustness of the capillary tube-suction line heat exchanger model; and the equation-of-state-based property routines that calculated the thermodynamic properties were replaced with interpolation routines that were much faster. The RFSIM model, in design and simulation mode, was validated with data from two refrigerators. In both modes, the average model errors were less than ±5% for several important variables such as evaporator capacity and coefficient of performance. The errors of the simulation mode were reduced from the previous model validation primarily by using a different void fraction correlation in the refrigerant charge equations. The results from the validated RFSIM model indicate that a two-speed compressor could yield energy savings of 4% to 14% due to the increased steady-state efficiency at the low speed and an additional 0.5 to 4% savings due to the decreased cycling frequency. The results also showed that the capillary tube-suction line heat exchanger, when designed for the low speed, did not adversely affect the pull-down capacity when the compressor operated at the high speed. Lastly, it was found that a refrigerator operating at low ambient temperatures could actually benefit from a decrease in the condenser fan speed. This change in fan speed increased the evaporator capacity by reallocating charge to the evaporator and subsequently reducing the superheat at the evaporator exit.

Table of Contents

Page

Chapter

Appendix

List of Tables

Table **Page**

List of Figures

Figure **Page**

Chapter 1

Introduction

1.1 Previous model development

The Refrigerator/Freezer Simulation (RFSIM) Model has been developed at the Air Conditioning and Refrigeration Center (ACRC) to facilitate evaluation of new system designs. Manufacturers are making substantial improvements in refrigerator technology to comply with increasingly stringent energy efficiency standards for refrigerator/freezers. A flexible, validated simulation model will allow researchers and designers to examine many aspects of refrigerator/freezer design faster than performing similar laboratory experiments. In this way, the RFSIM model can be used as "screening" tool to identify potential areas of improvement that can then be further investigated with laboratory testing.

The RFSIM model presented in this report has been developed from the steady-state ACRC2 simulation model that was written by Porter and Bullard (1992). That model had two main advantages over other existing refrigerator/freezer simulation models. Most of the other models (Arthur D. Little, 1982 and Merriam, et al., 1993), were limited to running in "design" mode, but the ACRC2 model could run in "design" mode as well as "simulation" mode. The design mode is useful for the rough design of the system components (other than the capillary tube) at a single operating condition, but it is unable to predict the behavior of the refrigerator over a range of operating conditions as does the simulation mode.

The other major advantage of the ACRC2 model was that it used a Newton-Raphson method to solve the equations while the other existing models used a successive-substitution method. Although the successive-substitution method is sometimes faster than Newton-Raphson, it much less flexible. In the Newton-Raphson method, the equations are not entangled with the solution algorithm and can be written in any order. Therefore, the equations are much easier to modify or replace. In addition, the convergence speed and robustness of the Newton-Raphson method can be greatly improved depending on the numerical implementation.

Goodson and Bullard (1994) built upon the foundation of the ACRC2 model and developed the first version of the RFSIM model. The RFSIM model had several advantages over the earlier ACRC2 model: the model equations were solved by an improved Newton-Raphson equation solver (ACRC Solver) which is described by Mullen and Bullard (1994); the empirically-obtained curve fit for the mass flow rate through the capillary tube was replaced by a capillary tube-suction line heat exchanger model based on "first principles"; the charge equations were improved by using a better void fraction correlation and by accounting for the mass of refrigerant in the single-phase components, including the liquid refrigerant dissolved in the compressor oil; and many user-specified parameters describing the heat transfer and pressure drop within the heat exchangers were replaced with correlations from the latest research at the

ACRC (Wattelet and Chato, 1994; Dobson and Chato, 1994; Admiraal and Bullard, 1995; Cavallaro and Bullard, 1995; Souza, et al., 1995).

1.2 Current model development

However, before the RFSIM model could be used to explore new design options, the accuracy of the model in simulation mode had to be improved and the capillary tube-suction line heat exchanger (ct-slhx) model had to be made more robust and flexible.

As reported by Goodson and Bullard (1994), the model in simulation mode had substantial errors in many of the important variables such as mass flow rate, evaporator capacity, and COP (average errors of -22%, -17%, and -10%, respectively). In the current work, the average errors of the simulation mode of the model for data from two refrigerators have been reduced to acceptable limits of ±5%. The process used to identify and eliminate the error in the previous version of the model is described in Chapter 2 and in greater detail in Appendix E. The previous ct-slhx model could handle only a few of the refrigerant-flow modeling scenarios that can occur in refrigerator capillary tubes. Therefore, many improvements were made in the modeling logic and robustness of the capillary tube-suction line heat exchanger (ct-slhx) model. Most of these improvements are described in Appendix C.

During the course of the model validation, other improvements were made to the RFSIM model in terms of flexibility and speed. Instead of being limited to only two modes of operation (design and simulation), the model can now run in a multitude of configurations. These operational modes are described in Chapter 2 and in Appendix A. The refrigerant property routines that were used in the previous version of the RFSIM model were responsible for a large portion of the model's computation time. Therefore, they were replaced with table lookups and interpolation routines that were developed at the ACRC which are inherently faster than the equation-of-state-based property routines. As a result, the computation time of the model was reduced by a factor of five. The new interpolation routines are described in Appendix D.

1.3 Application of the model

All of the improvements mentioned in 1.2 made it possible to use the model to evaluate the promising design option of multi-speed compressors. The validated RFSIM model was used to evaluate several aspects of the operation of a refrigerator having a two-speed compressor. The potential energy savings due to the increased steady-state efficiency at the low compressor speed and the reduction in cycling frequency were calculated with the model. Two other issues are also examined in this report: the "robustness" of the two-speed compressor system and the possibility of varying the speed of one of the heat exchanger fans in conjunction with the compressor speed variation. The modeling methods and results are discussed in Chapter 3.

Chapter 2

ACRC Refrigerator/Freezer Simulation (RFSIM) Model

2.1 The RFSIM model and the ACRC solver

The governing equations and the supporting FORTRAN routines (correlations, refrigerant properties, etc.) that make up the RFSIM model are solved by the ACRC Solver. This solver uses a Newton-Raphson (NR) technique that employs several enhancements in terms of solution speed and convergence robustness. Although the ACRC Solver is a general equation solver, it has features which enable it to handle special problems that occur in thermal system models. The ACRC Solver also provides a simple method of switching variables and parameters within the governing equations. This last feature increases the flexibility of the RFSIM model and allows it to be used either as a design or a pure simulation tool.

The ACRC Solver has been developed over a period of several years by Porter and Bullard (1992), Hahn and Bullard (1993), and by Mullen and Bullard (1994). A detailed description of the ACRC Solver (when used to solve the room air-conditioner model) is given by Mullen and Bullard (1994).

The ACRC Solver has not been changed since the previous report on the RFSIM model (Goodson and Bullard, 1994). Since that report contained an excellent description of the ACRC Solver, the discussion is repeated here (with the consent of the authors) with some modifications that reflect recent improvements in the RFSIM model.

2.1.1 Model-solver relationship

The structure and organization of the ACRC Refrigerator/Freezer Simulation (RFSIM) model as implemented with the ACRC solver is depicted in Figure 3.1. In addition to the actual Newton-Raphson (NR) solution, the ACRC Solver contains separate subroutines for model initialization and checking. Although the checking subroutines can be used as pre- or post-processors, their primary purpose is to provide a means of checking the values of variables or parameters before, during, or after the solution. The checking that takes place before or during the solution is used to set logical flags that are used within the list of governing equations.

For example, the "before" checking will determine, based upon the parameters and the initial guesses of the variables, if the evaporator exit is two-phase or superheated, and a logical flag will be set accordingly. This flag will cause the ACRC Solver to evaluate the correct set of equations related to the exit condition of the evaporator. Likewise, the "during" checking subroutine will determine if the flag indicating the exit state of the evaporator should be changed between NR iterations due to changes in the variables describing the exit condition. The "after" checking is used to see if the values of certain variables are within allowable ranges (e.g. evaporating and condensing temperatures for the compressor maps).

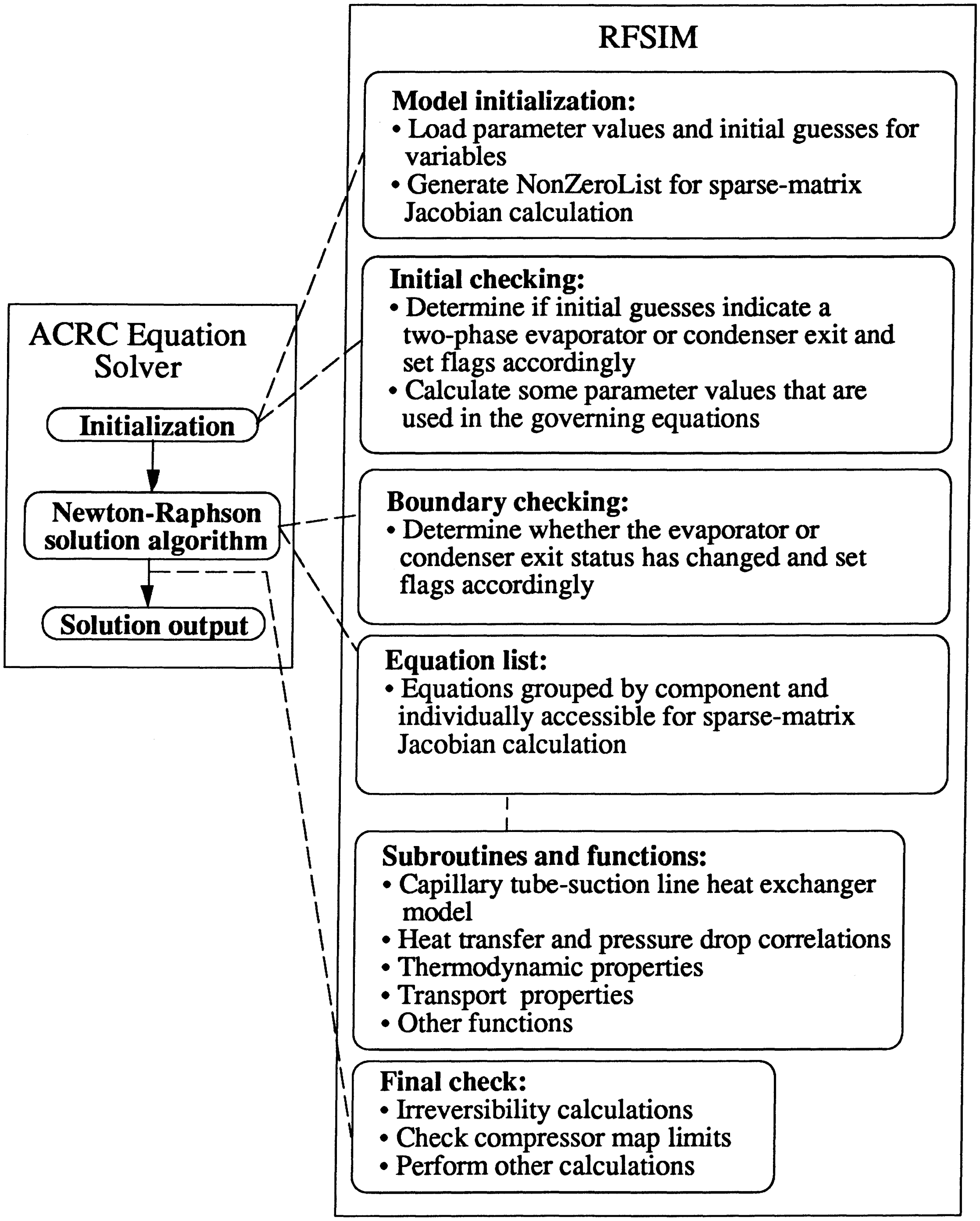

RFSIM
Model initialization:
• Load parameter values and initial guesses for variables
• Generate NonZeroList for sparse-matrix Jacobian calculation
Initial checking:
• Determine if initial guesses indicate a two-phase evaporator or condenser exit and set flags accordingly
• Calculate some parameter values that are used in the governing equations
Boundary checking:
• Determine whether the evaporator or condenser exit status has changed and set flags accordingly
Equation list:
• Equations grouped by component and individually accessible for sparse-matrix Jacobian calculation
Subroutines and functions:
• Capillary tube-suction line heat exchanger model
• Heat transfer and pressure drop correlations
• Thermodynamic properties
• Transport properties
• Other functions
Final check:
• Irreversibility calculations
• Check compressor map limits
• Perform other calculations
ACRC Equation Solver
Initialization
Newton-Raphson solution algorithm
Solution output

Figure 2.1 Organization of RFSIM and the ACRC solver

The governing equations of the RFSIM model are listed separately and in an order-independent fashion. Therefore, it is relatively easy to modify them or to replace an equation with a new one. Each "equation" can actually consist of several lines of intermediate calculations. As illustrated in Figure 2.1, many of these equations also involve subroutine or function calls. Therefore, the user can choose to place the majority of the calculations in the actual governing equations or in the supporting routines.

2.1.2 Switching parameters and variables

The basic requirement of the Newton-Raphson method is that there must be as many governing equations as variables and that the equations be independent and non-singular. Thus, a given variable can become a parameter if a former parameter simultaneously becomes a variable (in order to maintain the same number of equations and variables). Additionally, the equations must remain independent and have no singularities.

For example, Figure 2.2 depicts a set of three equations, requiring three variables for solution. Normally, a designer might specify the evaporator area (A_{evap}) and solve for the COP, but variable-parameter switching allows the NR method to solve for the A_{evap} that will yield a particular COP. In the multi-speed compressor analysis described in Chapter 3, the switching ability was used to "re-optimize" the refrigerator by setting the superheat and subcooling of the heat exchangers as known parameters and solving for the inlet length of the capillary tube and the total system charge required to achieve those "design" operating conditions. Variable-parameter switching can also be used for parameter estimation. For example, measured outlet conditions of a heat exchanger may be specified and the RFSIM model can be used to solve for the required heat transfer coefficient.

The ACRC solver allows switching of a parameter and a variable by simply changing two flags in the input file. There is no need to change the program or recompile, making it simple to change the model from a simulation to a variety of design configurations.

Equation set:

$$COP = \frac{Q_{evap}}{\dot{W}_{comp}}$$

$$Q_{evap} = U \cdot A_{evap} \Delta T$$

$$\dot{W}_{comp} = f(\Delta T)$$

Normal configuration:		After switching:	
Variables:	Parameters:	Variables:	Parameters:
COP	U	A_{evap}	U
Q_{evap}	A_{evap}	Q_{evap}	COP
$\dot{W}_{comp}$	ΔT	$\dot{W}_{comp}$	ΔT

Figure 2.2 Example of parameter-variable switching

2.1.3 Speed enhancements in the model and solver

The RFSIM model for the Whirlpool refrigerator consists of 112 governing equations, many of which involve lengthy calls to subroutines and functions. A straightforward evaluation of the Jacobian matrix for the 112 equations would require considerable execution time for the 12,544 ($=112^2$) partial derivative calculations. However, most of those equations contain only a few variables, so the majority of the partial derivatives are always zero. Such a system of equations is termed "sparse".

To improve the execution time, the non-zero elements of the Jacobian are mapped in advance by the ACRC Solver. That information is used to ensure that only those partial derivatives that may ever be non-zero are evaluated when the Jacobian is calculated. The remainder of the Jacobian elements are always zero and time is not wasted by calculating them.

Similarly, the linear-solution step in the Newton-Raphson method is speeded up by a sparse-matrix Gaussian elimination routine given by Stoecker (1989) that uses full pivoting and linked lists. The results of these speed enhancements are presented in Table 2.1 for the original version of the RFSIM model operating on a Convex C240 machine. Recent execution times have been further reduced from the results shown (20 seconds to about 3 seconds per iteration) due to improvements in the refrigerant property routines and a reduction in the number of simultaneous equations. However, the results in Table 2.1 still indicate the relative magnitudes of improvements obtained via speed enhancement within the ACRC Solver.

A typical simulation run, which uses a previous solution at similar conditions for the initial guesses, is solved in about 12 sec. Actual execution times vary by computer, but it is clearly demonstrated that while the sparse-matrix Gaussian elimination saves some time, the largest enhancement is obtained through the sparse-matrix Jacobian calculation.

Table 2.1 Speed enhancement results

	Sec/iteration
RFSIM with no enhancement:	180
Adding sparse Gaussian elimination:	170
Adding sparse Jacobian calculation:	20

2.1.4 Automated step relaxation to enhance solution robustness

The Newton-Raphson method is not globally convergent because it uses a linear approximation of a non-linear set of equations to "step" the variables towards a solution. It is possible that a NR step will be calculated that does not bring the variables closer to a solution, particularly when the initial guesses are poor. A NR step may even result in an attempt to evaluate a function (e.g. a thermodynamic or transport property) outside of its domain. Common examples include attempting to calculate a refrigerant quality in the superheated region or attempting to raise a negative number to a non-integer power (e.g. in a heat transfer or pressure drop correlation).

If a NR step is taken, and the variables are not brought closer to the solution (as measured by the residual values of the governing equations), then the ACRC Solver will reduce the step size by half and reevaluate the governing equations. This process will be repeated until a step size is found that reduces the residuals or until the number of step reductions exceeds a user-defined limit. This technique greatly increases the model's robustness and somewhat reduces the need for good initial guesses.

2.2 Model description

The RFSIM model is a comprehensive steady-state refrigerator model that is made up of approximately 110 system equations. As discussed in the previous section, these equations are solved simultaneously by the ACRC Solver. There are no restrictions in the order of the equations or how the equations must be written other than the requirement that they be written in residual format as described in B.1.1.1. Each “residual", or system, equation can be preceded by an unlimited number of explicit, intermediate calculations and can include any function or subroutine call that is allowed in FORTRAN.

Although the current version of the RFSIM model is very flexible, it is convenient to discuss its operation in terms of the "design" and "simulation" modes that were presented in the original work (Goodson and Bullard, 1994). When the model is run in design mode, two refrigerant states must be specified (e.g. superheat and subcooling). When the model is run in simulation mode, no refrigerant states need to be specified because the model contains two additional sets of equations: the capillary tube-suction line heat exchanger (ct-slhx) model and the charge conservation equations. These two sets of equations will be discussed in the following sections.

The model can be run in number of other configurations as well. The previous example of variable-parameter switching taken from the multi-speed compressor analysis is an example of such a configuration. The superheat and subcooling are both specified, but the ct-slhx model and charge conservation equations are used as well to solve for the capillary tube length and refrigerant charge required at the design operating condition. The current version of the RFSIM model will allow the use or non-use of the ct-slhx model and the charge equations to be specified simply by changing flags in the input file.

The following sections outline the important equations used with the RFSIM model. To assist in the discussion, a summary of the heat transfer and pressure drop correlations is presented in Table 2.2. A more in-depth discussion of the model equations and supporting FORTRAN routines is presented in Appendix B.

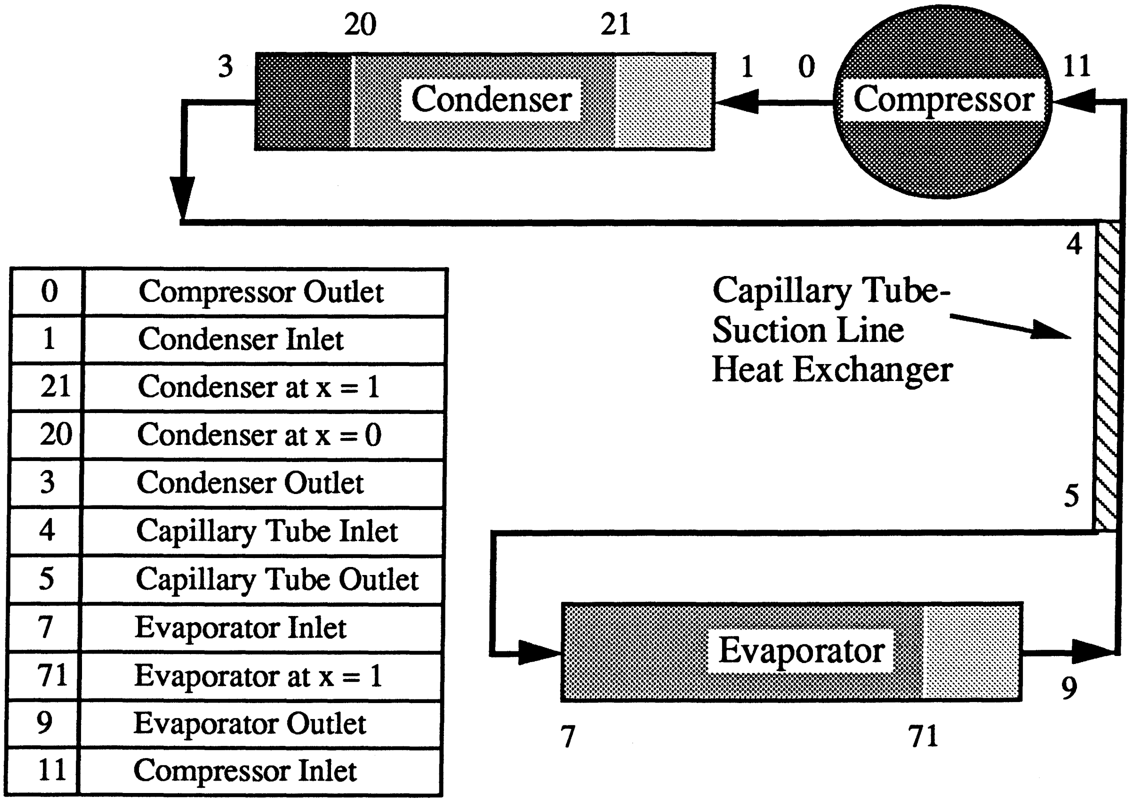
Table 2.2 Correlations used in the RFSIM model

Condenser Heat Transfer	
Superheated	Gnielinski (Incropera and DeWitt, 1990)
Two-phase	Dobson and Chato (1994)
Subcooled	Gnielinski (Incropera and DeWitt, 1990)
Air-side	Cavallaro and Bullard (1994) or user-supplied constant
Evaporator Heat Transfer	
Superheated	Gnielinski (Incropera and DeWitt, 1990)
Two-phase	Wattelet and Chato (1994)

Air-side	Cavallaro and Bullard (1994) or user-supplied constant
Capillary Tube Heat Transfer	
Single-phase	Gnielinski (Incropera and DeWitt, 1990)
Two-phase	Gnielinski (Incropera and DeWitt, 1990)
Suction Line Heat Transfer	
Single-phase	Gnielinski (Incropera and DeWitt, 1990)
Two-phase	Wattelet and Chato (1994)
Evaporator & Condenser Pressure Drop	
Two-phase	Souza et. al. (1995)
Single-phase	Moody friction factor from Haaland(1983)
Single-phase return bends	Ito (1960)
Two-phase return bends	Christofferson et. al. (1993)
Capillary Tube Friction Factors	
Single-phase	Colebrook (Swamee and Jain, 1976)
Two-phase	Colebrook with Dukler's (1964) viscosity

2.2.1 Refrigerant state equations

The numbering scheme used for the state points in the cycle is shown in Figure 2.3.



0	Compressor Outlet
1	Condenser Inlet
21	Condenser at x = 1
20	Condenser at x = 0
3	Condenser Outlet
4	Capillary Tube Inlet
5	Capillary Tube Outlet
7	Evaporator Inlet
71	Evaporator at x = 1
9	Evaporator Outlet
11	Compressor Inlet

Figure 2.3 State points within the RFSIM model

The refrigerant state at every point in the cycle is defined by the group of equations referred to as the "state" equations. This group of equations defines thermodynamic properties at every state based upon the state variables (e.g. pressure, temperature, quality). This group also relates most upstream and downstream states with pressure drop equations and a few heat transfer equations. The pressure drop correlations that are used in the various parts of the cycle are shown in Table 2.2. The simple heat transfer equations are only for components, other than the condenser and evaporator, that might experience some heat transfer (e.g. discharge and liquid lines).

The thermodynamic properties come from a package of interpolation routines that was developed at the ACRC. These interpolation routines decreased the solution time of the RFSIM model by a factor of five when compared to the REFPROP Version 3.0 property package from NIST that was originally used within the model. More information about the interpolation routines is available in Appendix D.

2.2.2 Charge inventory equations

The mass of refrigerant in the single-phase components in the cycle is calculated simply by dividing the component volume by the specific volume of the refrigerant in the component. However, there is a slight complication in the calculation of the single-phase refrigerant in the compressor because there are two contributions to the total charge: single-phase refrigerant vapor that exists in the free volume of the compressor, and liquid refrigerant that is dissolved in the compressor oil. The vapor portion is calculated as the other single phase components, but the amount of liquid refrigerant dissolved in the oil is determined through the use of an empirical model developed at the ACRC by Grebner and Crawford (1992).

However, the calculation of the refrigerant mass in the two-phase portions of the heat exchangers is more difficult. When the heat flux is assumed to be constant, the evaluation of the two-phase mass depends only on the average void fraction, the two-phase volume, and the average liquid and vapor densities. Therefore, a void fraction correlation ($\alpha= f(x)$) is integrated over the quality range to obtain an average value (Equation B.5). With this average value of the void fraction, the two-phase mass can be calculated directly with Equation 2.1 shown below:

$$M_{two-phase} = V \cdot \left[\rho_g \overline{\alpha} + \rho_f (1 - \overline{\alpha})\right] \tag{2.1}$$

In the RFSIM model, the Premoli void fraction correlation (Rice, 1987) is used because it yielded the best agreement with data that was taken from two refrigerators as described in Appendix E.

The last charge conservation equation sets the sum of the refrigerant mass in each component equal to the total refrigerant charge. If the system variable representing the total charge is not specified by the user, then the summation equation simply calculates the charge as an output. However, if the total charge is specified as a known value, then the charge equations constrain the system by affecting the states of the refrigerant in the heat exchangers. For example, if the total charge value is set too high or if the void fraction correlation is underpredicting the two-phase mass in the heat exchangers, then the model solution will contain too much subcooling at the condenser exit. This, in turn, will affect the rest of the system equations.

2.2.3 Compressor equations

In addition to the refrigerant charge equations, the compressor is described by two compressor maps (mass flow and power), a refrigerant-side energy balance, an air-side energy balance, and a rate equation predicting the heat transfer from the compressor shell to the air stream. The refrigerant- and air-side energy balances are direct applications of the first law of thermodynamics. The rate equation, however, is dependent on a empirical relation between the shell temperature and the discharge temperature and a user-specified "UA" for the compressor shell.

The compressor maps are the standard manufacturer-supplied bi-quadratic curve fits of mass flow rate and power data as functions of the evaporating and condensing temperatures corresponding to the inlet and exit compressor pressures, respectively. In the absence of data on multi-speed compressors, however, the values returned by the compressor maps are multiplied by scaling factors. The primary purpose of these factors is two simulate the effect of a change in the compressor speed or size. Of course, the accuracy of such a scaling will be entirely dependent on the user's knowledge of how the mass flow and power vary with the compressor speed or size. These scaling factors were used to generate the multi-speed compressor results discussed in Chapter 3.

2.2.4 Condenser equations

For modeling purposes, the condenser is divided into three zones corresponding to the three phases of refrigerant contained in the condenser: superheated vapor, two-phase mixture, and subcooled liquid. Each of these three zones is treated as a individual heat exchanger with its own inlet and outlet refrigerant and air temperatures. Accordingly, the heat transfer of each zone is described by three equations: a refrigerant-side energy balance, an air-side energy balance, and an effectiveness-NTU rate equation. The refrigerant pressure drop through each zone is also calculated by the equations described in 2.2.1. The RFSIM model has logic that allows the condenser to have only two zones if the condenser has a two-phase refrigerant exit.

The refrigerant-side energy balance equations relate the heat transfer of each zone to the change in total enthalpy of the flowing refrigerant. Likewise, the air-side energy balances relate the heat transfer in each zone to the change in total enthalpy of the air flowing over the zone. Currently, the three outlet air temperatures are averaged to obtain a single condenser outlet air temperature. After leaving the condenser, the air flows past the compressor and then through the condenser fan. There is an air-side energy balance for the compressor (described in 2.2.3) and another one for the condenser fan.

The effectiveness-NTU method predicts the heat transfer in each zone according to Equation 2.2:

$$Q_{zone} = \varepsilon C_{min}\left(T_{ref,in} - T_{air,in}\right) \tag{2.2}$$

Where ε is the effectiveness of the heat exchanger (zone) and C_{min} is the minimum heat capacity of the two fluid streams in the heat exchanger. The effectiveness is calculated from explicit functions that are dependent on the overall heat transfer conductance, “UA”, and the minimum and maximum heat capacities. In general, the exact form of the effectiveness equation is dependent on the flow configuration of the two streams in the heat exchanger. For the wire-on-tube condensers used in domestic refrigerators, a “parallel-counterflow” effectiveness equation is used for the superheated and subcooled zones. In the two-phase zone, effectiveness is independent of flow configuration.

The area in the "UA" term of each zone’s effectiveness-NTU method is calculated from the total condenser area and the fraction of the total condenser filled by each zone. These fractions are also used in the pressure drop equations, and, like many variables in the RFSIM model, are determined implicitly by the Newton-Raphson method. The “U” in the “UA” term is calculated from the resistive-network analogy assuming that the only resistance to heat transfer is the air- and refrigerant-side convection resistances. The summation of the heat transfer resistances is shown in Equation 2.3 below:

$$\frac{1}{U_{air}A_{air}} = \frac{1}{h_{air}A_{air}} + \frac{1}{h_{ref}A_{ref}} \tag{2.3}$$

The areas in Equation 2.3 are known from the zone fractions just described. For the single-phase refrigerant zones, the refrigerant-side heat transfer coefficient is obtained from the single-phase correlation shown in Table 2.2. For the two-phase zone, an average value of the refrigerant-side heat transfer coefficient is obtain by integrating, over the quality range, the local two-phase condensing heat transfer coefficient shown in see Table 2.2. The air-side heat transfer coefficient is either specified by the user or an empirical correlation can be used as shown in Table 2.2.

2.2.5 Capillary tube-suction line heat exchanger equations

There are actually two different sets of equations that can be used within the RFSIM model to describe the capillary tube-suction line heat exchanger (ct-slhx). The simplest set of equations is an effectiveness method which predicts the amount heat transfer from the capillary tube to the suction line. These equations use a user-specified value of the effectiveness (actual heat transfer divided by the maximum heat transfer) and the inlet conditions of the heat exchanger to calculate the actual heat transfer. When the effectiveness method is used, the mass flow rate of refrigerant is determined solely by the compressor map.

The other set of equations is referred to as the capillary tube-suction line heat exchanger model (ct-slhx model). The heat transfer and the mass flow rate are predicted with a finite-difference solution of the governing differential equations. These discretized governing equations are not placed directly within the list of system equations, but they are solved in a subroutine that is called by several system equations. In this subroutine, the discretized governing equations of the ct-slhx are solved in a "sequential" manner. The word "sequential" is placed in quotation marks because the discretized segments are solved in a sequential manner from the end of the ct-slhx (evaporator inlet) to the beginning (condenser outlet), but the equations within each segment are actually solved by a one- or two-variable internal Newton-Raphson routine. The primary reason for placing the solution of the discretized governing equations in a "sequential" subroutine was to reduce the number of equations and required initial guesses present in the system model (simultaneous set of equations).

For modeling purposes, the ct-slhx is divided into three sections: adiabatic inlet section (from the inlet of the capillary tube to the point where the suction line is first soldered to the capillary tube); heat exchanger section (the portion of the capillary tube which is soldered to the suction line); and adiabatic outlet section (from the end of the heat exchanger section to the evaporator inlet). For the segments in the adiabatic inlet and outlet section of the ct-slhx, the governing equations include the mass, momentum, and energy conservation equations. These same equations apply in the heat exchanger section, but the energy equation for the capillary tube contains an extra term representing the heat transfer with the suction line. The governing equations for the heat exchanger section also include the following: a mass conservation equation for the suction line; a rate equation describing the heat transfer from the tube wall to the suction line refrigerant; and a convective heat transfer balance between the capillary tube refrigerant and the suction line refrigerant. A momentum equation for the suction line is not included because the pressure drop for the suction line is assumed to be negligible.

These governing equations just described are the main equations of the ct-slhx model, but there are other equations and assumptions that affect the results of the model. Among the most important of these is the assumption that a choked-flow condition exist at the capillary tube exit. This realistic assumption allows the mass flow rate of the refrigerant in the capillary tube to be calculated from a numerical partial derivative of thermodynamic properties at the exit. The calculation of the choked-flow mass flow rate is simplified somewhat by the assumption of homogeneous equilibrium two-phase flow throughout the capillary tube. There is another equation at the other end of the capillary tube that accounts for the sudden contraction pressure drop that occurs as the refrigerant flows from the large-diameter liquid line to the small-diameter capillary tube. The heat transfer coefficients and friction factors that are used in the governing equations of the ct-slhx model are shown in Table 2.2.

Unlike an adiabatic capillary tube, the refrigerant flow processes in a ct-slhx can be very complicated. For example, refrigerant may enter the capillary tube as a subcooled liquid, flash (begin to vaporize) in the inlet section, recondense in the heat exchanger section, and then flash again in the outlet section. The current ct-slhx model has a tremendous amount of logic that enables it to handle almost any possible combination of flow processes that can occur within the ct-slhx. The details of this logic as well as the solution strategy, governing equations, assumptions, and correlations related to the ct-slhx model are covered in Appendix C.

As mentioned, the ct-slhx model is called by several system equations. When the subroutine is called, it will "solve" the governing equations of the ct-slhx for the set of inputs that it was given. Among these inputs are several system variables (such as the pressure and quality at the exit of the capillary tube). The subroutine will start at the capillary tube exit and determine the choked-flow mass flow rate. It will then solve each of the preceding discretized segments until it reaches the inlet of the capillary tube and returns several outputs. These outputs of the ct-slhx model include the lengths of the three sections (inlet, heat exchanger, and outlet), the mass flow rate, and the pressure and enthalpy at the inlet of the capillary tube. In the set of simultaneous system equations, there are six equations that equate outputs of the ct-slhx model to system variables or parameters. For example, there is an equation that requires the mass flow rate calculated by the ct-slhx model to be equal to the mass flow rate calculated by the compressor map. Therefore, the subroutine solution guarantees that the discretized governing equations of the ct-slhx are satisfied, and the Newton-Raphson solution of the system equations guarantees that the subroutine solution matches the physical situation in the refrigerator.

2.2.6 Evaporator equations

The equations that model the evaporator are very similar to those for the condenser. The evaporator is also divided into zones which are modeled as individual heat exchangers. The primary difference between the condenser and evaporator is that the evaporator will have at most two zones: a two-phase zone and a superheated zone. However, the evaporator will have only one zone if its exit is two-phase. As in the condenser, the heat transfer in each zone is modeled with three equations: a refrigerant-side energy balance, an air-side energy balance, and an effectiveness-NTU rate equation. The theory of these heat transfer equations is the same as described in 2.2.4 except that a different two-phase refrigerant heat transfer correlation is used to calculate the "UA" of the two-phase zone. There is also an air-side energy balance equation for the evaporator fan which is located downstream of the evaporator.

2.2.7 Cabinet and system equations

This group of equations relates the refrigerant system (primarily the evaporator) to the refrigerator cabinets and performs some system performance calculations. Two equations calculate the heat load on the fresh-food and freezer compartments. Each heat load is represented by simple one-dimensional UA(ΔT) expression that accounts for the heat transfer through the walls of the refrigerator, plus a term that represents any additional heat load that the user wishes to specify. Since RFSIM is a steady-state model, the air temperatures in the two compartments do not change with respect to time as a result of the cooling capacity exceeding the heat load. Instead, two equations calculate the ratio of the combined compartment heat loads to the system refrigeration capacity. This ratio can be interpreted as the percentage time that the refrigerator would have to run at steady-state to remove the heat added to each compartment; it is also a measure of the excess or "pull-down" capacity of the evaporator. The last cabinet equation calculates the inlet air temperature to the evaporator based upon the air temperatures in the two compartments and value of the split-air fraction (fraction of evaporator air going to freezer).

As measures of performance, the model calculates the COP of the refrigerator and calculates an estimated value of the yearly energy use. The system COP is the cycle COP adjusted for the evaporator fan power as shown in Equation 2.3:

$$COP = \frac{Q_{evap} - Q_{evap\,fan\,power}}{P_{compressor} + P_{evap\,fan} + P_{cond\,fan}} \qquad (2.3)$$

The second term in the numerator represents the heat added to the compartments by the evaporator fan, which offsets some of the cooling provided by the evaporator. The yearly energy use is calculated assuming the steady-state refrigerator could "cycle" for an entire year according to the ratio mentioned in the previous paragraph. For example, if the ratio of the heat load to cooling capacity is equal to 0.5, then the refrigerator can be imagined to run with a 30 minute on-cycle and a 30 minute off-cycle for the entire year. However, the ratio does not predict the cycle length.

2.3 Validation results

Although most of the equations of RFSIM are "physical" in that they are based upon "first principles" and generally applicable correlations, the entire model cannot be considered "physical" because it depends on several user-supplied parameters. Therefore, even if the RFSIM model has been "validated", its accuracy when used in other situations will still depend on the user-supplied input parameters. In view of this fact, the validation process for a model like RFSIM may be best described as a validation of the modeling procedure itself and not as statement that the model will always predict the variables within some accuracy.

The modeling procedure to be validated includes all the assumptions, physical relations, and correlations contained in the model: two- and three-zone heat exchanger models; finite-difference capillary tube-suction line heat exchanger (ct-slhx) equations; refrigerant-side pressure drop and heat transfer correlations; etc. In other words, the validation of RFSIM involves the evaluation of the model's ability to predict key variables assuming that the user-specified parameter values are accurate.

The original validation of the RFSIM model was performed with data from a 18 cu. ft., top-mount Amana refrigerator (model TC18MBL) and was reported by Goodson and Bullard (1994). Since then, however, many improvements have been made to the RFSIM model. Therefore, the validation of the current version of the RFSIM model was performed with the original Amana data as well as data from a 20 cu. ft., top-mount Whirlpool refrigerator (model ET20PK). The data from both refrigerators was used to evaluate the accuracy of the RFSIM model in design and simulation mode when supplied with accurate parameter values. The process of estimating these parameters for the Whirlpool refrigerator is described by Krause and Bullard (1996). Similar techniques were used to obtain the parameters for the Amana refrigerator. The entire model validation process, along a discussion with the problems that were encountered, is described in Appendix E. In this report, some of results from the Amana and Whirlpool data will be presented that illustrate the major findings.

2.3.1 Amana results

The Amana data consisted of 16 points at the four ambient temperatures of 60 °F, 75 °F, 90 °F, 100 °F. The experimental techniques used to obtain this data are described by Rubas and Bullard (1995). All of the data points were superheated at the evaporator exit, but only seven were subcooled at the condenser exit. Therefore, these seven points were the only ones that were used to evaluate the accuracy of the design mode since it requires that the heat exchanger exit states be specified. The model was run in design mode for the seven subcooled points by specifying the following operational parameters: power dissipated by the fresh-food and freezer heaters, atmospheric pressure, evaporator and condenser fan powers, average fresh-food and freezer compartment temperatures, ambient temperature, and the subcooling and superheat. Figure 2.4, on the next page, shows the error in the model predictions relative to the experimental data for the model in design mode.

The line in the center of the box for each variable shows the median value of the error, and the upper and lower edges of the box show the limits of +/- 25% of the error population. The lines extending from the top and bottom of the box show the maximum and minimum error values for each variable.

As Figure 2.4 illustrates, the model predicts the evaporation and condensation temperatures within 1 °F of the actual values. There is a slight underprediction in the mass flow rate (~ 2.5%) and fairly even distribution (± 3%) of the errors in the evaporation capacity. The main problem, however, with the model in design mode is the underprediction of the system power of about 4% and the subsequent overprediction of the COP of about 3.5%.

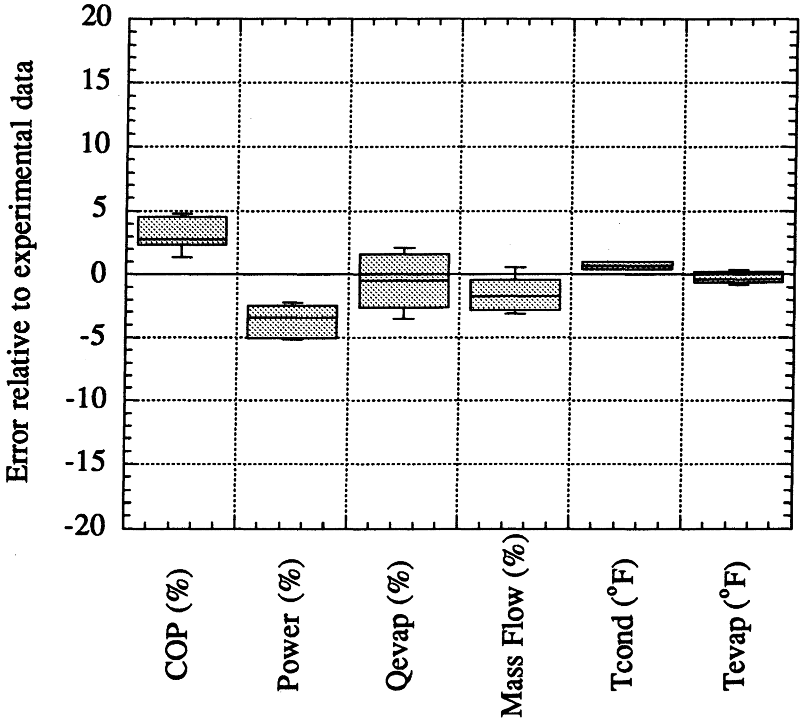


Figure 2.4 Model error in design mode (Amana)

The above results are primarily a measure of the accuracy of the multi-zone heat exchanger models and the compressor maps since the ct-slhx model and the charge conservation equations are not used in the design mode. Since the evaporation and condensing temperatures are predicted very well, the underprediction in the system power in Figure 2.4 is a direct indication that the power compressor map is underpredicting for these data points. The fact that the evaporation capacity is predicted well, and predictions of the refrigerant states within the heat exchangers are very close to the data, supports the validity of the multi-zone heat exchanger models. The scatter in the capacity predictions reflects both the approximations involved in treating complex heat exchanger geometries as simplified geometries (e.g. counterflow or parallel-counterflow with uniform airflow), as well as the scatter in the data used to estimate the various empirical parameters input to the model.

The results of the design mode of the RFSIM model when run with the Amana data indicate that the accuracy of the model is limited by the accuracy of the compressor maps. Since the predictions of this particular map are within the expected range of ±5%, the results shown in Figure 2.4 are probably indicative of what can be expected with the design mode of the RFSIM model.

To examine the need for accurate compressor maps, new compressor maps were generated for the Amana from 13 experimentally-measured values of mass flow and power and the original map points. With the new maps, the model predictions for the same six variables shown above were excellent (see Figure E.2 in Appendix E). The errors in system power and COP were effectively eliminated, and the scatter in the evaporation capacity was reduced.

Although the predictions of the simulation mode of the RFSIM model with the Amana data are somewhat worse than for the design mode, the errors are much lower than reported in the original work (Goodson and Bullard, 1994). These results can be seen in Figure E.3 and E.4 in Appendix E. The primary difference between the results of the current and original validation is the void fraction correlation used in the charge correlations. Originally, the Hughmark correlation was used, but the Premoli correlation is currently used because it gives better agreement with the data for the Amana and the Whirlpool refrigerators. The validation of the simulation mode of the RFSIM model will be illustrated with the Whirlpool data in the next section.

2.3.2 Whirlpool results

The Whirlpool data consisted of 26 points at the four ambient temperatures of 60 °F, 75 °F, 90 °F, 100 °F. The experimental techniques used to obtain this data are described by Krause and Bullard (1996). All of the data points were superheated at the evaporator exit, but only 12 were subcooled at the condenser exit.

When the RFSIM model was run in design model for the 12 subcooled points, the agreement with the data was poor (see Figure E.5 in Appendix E). The mass flow rate was significantly overpredicted (6% to 15%), and, as a result, the evaporation capacity and COP were also overpredicted. When the measured evaporating and condensing temperatures are used as inputs, the manufacturer-supplied mass flow map overpredicted the data by an average of 15%. The results of this analysis confirmed earlier findings (Krause and Bullard, 1996) that the compressor map for the Whirlpool overpredicted mass flow. A simple, suction-gas density correction to the mass flow map was attempted, but the results did not improve appreciably, possibly due to the additional heat transfer to the refrigerant between the suction inlet and suction port.

Since the design mode had already been validated for the Amana test data, the primary purpose of the simulation model validation was to check the accuracy of the capillary tube-suction line heat exchanger model and the charge conservation equations. In order to focus the analysis on the accuracy of these two sets of equations, the errors introduced by the faulty compressor map would have to be reduced to a tolerable level. Therefore, new compressor maps were made that would maintain the general shape of the generic maps but would better predict the experimental data. The new compressor maps significantly improved the model predictions of the design mode (see Figure E.6). The majority of the mass flow rate errors were brought down to the +3% range, and the evaporation capacity and COP errors showed similar improvement.

With the new compressor maps, the simulation mode of the model was run for all 26 data points (subcooled and two-phase condenser exits). The errors between model predictions and the experimental data were calculated for the same six variables and are displayed in Figure 2.5 below.

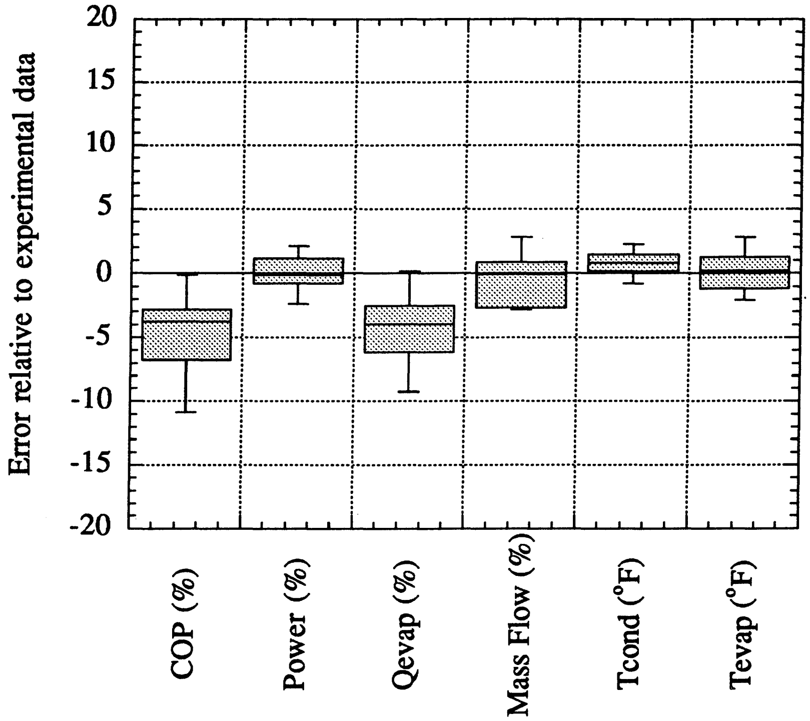


Figure 2.5 Model error in simulation mode with the new maps (Whirlpool)

As Figure 2.5 illustrates, the simulation mode predictions of power, condensing temperature, and evaporating temperature are all scattered about the zero error line in a fairly narrow range. Although the model predictions of the mass flow rate shown in Figure 2.4 are good (within ±3% of data), they are based on only 12 of the 26 data points because experimental mass flow values are only obtainable for data points having subcooled condenser exits.

A closer examination revealed that the largest errors in evaporation capacity (-4% to -9%), COP (-4% to -11%), and the evaporating temperature (-2 °F to 0.5 °F) occur for the two-phase condenser exit points. Based upon this observation, it is reasonable to assume that the mass flow errors, if available for the two-phase condenser exit points, would be worse than shown in Figure 2.5. Conversely, if just the subcooled condenser points are considered (see Figure E.8 in Appendix E), the model predictions in evaporation capacity and COP are very good (within 3% of data).

The majority of the errors associated with the mass flow, evaporation capacity, COP, and evaporating temperature of the condenser two-phase exits points are probably due to the ct-slhx model's tendency to underpredict mass flow. As observed by Liu and Bullard (1996), the underprediction becomes worse as the portion of the capillary tube that contains two-phase refrigerant increases. Therefore, the mass flow predictions for the two-phase capillary tube inlet conditions will be worse than the predictions for the subcooled points. The underpredictions of the mass flow rate causes the evaporator capacity and COP to be underpredicted as well. They also cause the underpredictions in the evaporating temperature that were previously mentioned. In RFSIM, the compressor map must have the same flow rate as the ct-slhx model. Therefore, if the ct-slhx model is underpredicting mass flow, then the evaporating temperature used by the compressor map will become lower in order for the compressor map to have a lower mass flow rate as well.

Although the capillary tube is probably responsible for most of the uncertainty remaining in the simulation mode of the RFSIM model, the revised compressor maps and the charge conservation equations are also partly responsible. It is difficult to identify the cause of the errors because all of the equations in the RFSIM model depend on and affect the other equations in the model. This is especially true when dealing with the relatively small errors present in Figure 2.5. However, it is important to recognize that even if all the data points are used as in Figure 2.5, the average errors of the simulation mode of the model are still within 4%, or 2 °F, for all the variables.

2.3.3 Validation summary

The excellent agreement between the design mode predictions and the experimental data confirm the validity of the modeling procedure used in the design mode of RFSIM. The average errors of the design mode were all less than 4% for the COP, system power, evaporation capacity, and mass flow and were less than 1 °F for the evaporating and condensing temperatures. The largest errors (in power and COP) are a direct consequence of the error in the manufacturer-supplied generic compressor map. Therefore, the accuracy of the RFSIM model in design mode appears to be limited by the accuracy (normally ±5%) of the compressor maps.

Although the agreement between the simulation mode predictions and the experimental data indicate that the ct-slhx model and charge conservation equations used in the simulation mode are valid, there is still room for improvement. The significant errors in the COP and evaporation capacity (and probably mass flow) for the two-phase condenser exit data points are primarily due to the underprediction of mass flow of the ct-slhx model. Experimental investigations into the behavior of capillary tube-suction line heat exchangers are currently underway that will hopefully eliminate some of the remaining errors in the ct-slhx model. However, with the current model, the average errors in COP, power, evaporation capacity, and mass flow rate are less than 4%. In addition, the vast majority of the model predictions of evaporating and condensing temperatures are within 2 °F of the experimental values.

Chapter 3

Variable Speed Compressor Modeling

3.1 Introduction

To meet the tougher energy standards, refrigerator manufacturers are exploring many new options that would reduce the energy consumption of their refrigerators while keeping them economically and aesthetically attractive to customers. One of the options under consideration is the replacement of the current single-speed compressor with a variable- or multi-speed compressor. Although there are several potential benefits of using a multi-speed compressor, they all originate from the multi-speed compressor's ability to better match the loading conditions on the refrigerator.

When manufacturers design the refrigerant-system for a refrigerator, there are two important issues that must be considered: energy efficiency and pull-down capacity. Unfortunately, these two issues are usually in opposition. If a system is designed for a fast pull-down, then the temperature "lift" (difference between the condensing and evaporating temperatures) will be higher and will cause the compressor to operate in a less efficient area of the compressor map. In addition, a fast pull-down system will also cycle more frequently which will result in higher cycling losses (see Krause and Bullard, 1996 and Coulter and Bullard, 1995). At the other extreme, if a system is designed to barely meet the cooling load of a closed-door refrigerator at a 90 °F ambient temperature, then it will not be able to maintain the required food temperature at higher ambient temperatures or if the there is any "shock" to the system. This "shock" could take the form of frequent door openings, warm food, or a defective door gasket. Therefore, designers must compromise and give up a little pull-down capacity and energy efficiency to arrive at a workable design.

A variable- or multi-speed compressor would help in this situation because it would allow the refrigerator to have more than one capacity to better match changing loading conditions. The compressor could operate at a lower speed and lower evaporation capacity for the majority of the time, and operate at a higher speed when the high pull-down capacity is needed (when several bags of groceries are added to the refrigerator). In theory, an infinitely-variable compressor would appear to be the ideal solution to the problem. The speed of such a compressor would be changed by a control system in response to one or more input variables. Cycling losses would effectively be eliminated and the temperature lift of the compressor would always match the capacity requirements. For a variety of reasons that will become apparent later, this may not, in fact, be superior to a simpler two-speed compressor.

In addition to the steady-state (lower temperature lift) and cycling energy savings already discussed, a multi-speed compressor would produce other benefits as well. In fact, some of the energy savings could be traded for smaller heat exchangers, which would allow for more internal storage volume for fixed external cabinet dimensions. The longer cycles that would reduce the cycling losses may also extend the life of compressor because it would cycle at a lower frequency. Since a multi-speed compressor would operate at its lowest speed most of the time, the noise made by the compressor would also be lower.

For the purposes of this study, the multi-speed compressor was assumed to be a two-speed compressor whose high speed would correspond to the current nominal compressor speed of 3600 rpm, and whose low speed would be nominally equal to 2400 rpm. The magnitude of the energy savings that could be realized with a two-speed compressor were determined through the use of the validated RFSIM model. Two other issues were also examined in this study: the "robustness" of the two-speed compressor system and the possibility of varying the speed of one of the heat exchanger fans in conjunction with the compressor speed variation.

3.2 Modeling methods

Before the results are presented, the modeling methods will be briefly discussed. The validated RFSIM model for the Whirlpool refrigerator was used for all of the simulations. The evaporator and the condenser were unchanged during the simulations; only the compressor, capillary tube-suction line heat exchanger, and the total refrigerant mass were varied. The average temperatures for the fresh-food and freezer compartments were set at 40 °F and 5 °F, respectively. These temperatures were chosen to approximate the average cycle temperatures in each compartment during a pull-down at the DOE test condition. Unless otherwise stated, all of the simulations were performed at the above compartment temperatures and an ambient temperature of 90 °F.

3.2.1 Compressor speed variation

The equations used in the RFSIM model are covered in Chapter 2 and Appendix B. However, a few words will be said about the technique used to model the speed variation in the compressor. Since the RFSIM model uses compressor maps to describe the mass flow rate of refrigerant and the power consumed by the compressor, simple scaling factors were introduced to adjust the mass flow and power values returned by the maps. These scaling factors are discussed more detail in A.3.3 and B.1.4. For the high speed case, the scaling factors for the mass flow map and the power map (beta_Wmap and beta_Pmap) were both set at one. At the low speed, the scaling factors for the mass flow map and the power map were set at 0.7305 and 0.7192, respectively (Bockhold, 1995). The information used to generate the scaling factors at the low speed came from one data point from a prototype two-speed compressor (at ~2400 rpm and ~3300 rpm) at the ASHRAE standard rating point (-10 °F evaporating, 130 °F condensing, and 90 °F ambient).

Although it would be desirable to have complete compressor maps at 2400 rpm and 3600 rpm for a production-ready compressor, such information is not available in the open literature due to the proprietary nature of this new technology. Although the capacity seems to be directly proportional to the compressor speed, the power variation is very dependent on the motor that is used with the compressor. The single available data point indicates that the compressor itself is 1.5% more efficient at the low speed. Since this number could increase as motors and compressors are optimized for lower speeds, the simulations were run with several different compressor efficiencies (different beta_Pmap scaling factors) to cover the range of realistic energy savings that could be realized with a two-speed compressor.

3.2.2 Re-designing the refrigerator for steady-state operation

Since one of the primary purposes of this study is to evaluate the energy savings that would occur if the refrigerator compressor operates at the low speed of 2400 rpm, the refrigerator had to be re-optimized for operation at the low speed. In other words, if the original, high-speed compressor is simply "slowed" down, the performance at the low-speed would not be as good as it could be because the rest of the system (e.g. capillary tube and refrigerant charge) is designed for the high speed. The "optimization" of the refrigerator at the low speed is not a optimization in the mathematical sense of the word. Rather, it is a design of the capillary tube and the refrigerant charge that would yield maximum EER performance in the otherwise unaltered refrigerator. Since the heat exchangers were unchanged, the design approach was to specify the refrigerant state at the exit of the heat exchangers and use the RFSIM model to solve for the corresponding inlet adiabatic length of the capillary tube and the total refrigerant charge in the system.

The main question with this type of design is how the exits of the evaporator and condenser should be specified. It was expected that the optimal performance would occur when the refrigerant at both heat exchanger exits was saturated ($x = 1.0$ in the evaporator and $x = 0.0$ in the condenser). This would seem to ensure that the high refrigerant-side heat transfer coefficients of the two-phase refrigerant existed in as much of the evaporator and condenser as possible. To test this hypothesis, the refrigerant state at the exits of both heat exchangers were varied from a two-phase mixture ($x = \sim 0.90$ in the evaporator and $x = \sim 0.06$ in the condenser) to a single-phase fluid (superheat and subcooling of 20 °F). As expected, the COP of the system increased as the two-phase refrigerant exits approached the respective saturation lines. However, the COP continued to increase slightly as the superheat and subcooling increased until a maximum was reached at about 9 °F superheat and subcooling. This optimum COP, in the single-phase heat exchanger exit range, is shown in Figure 3.1 below.

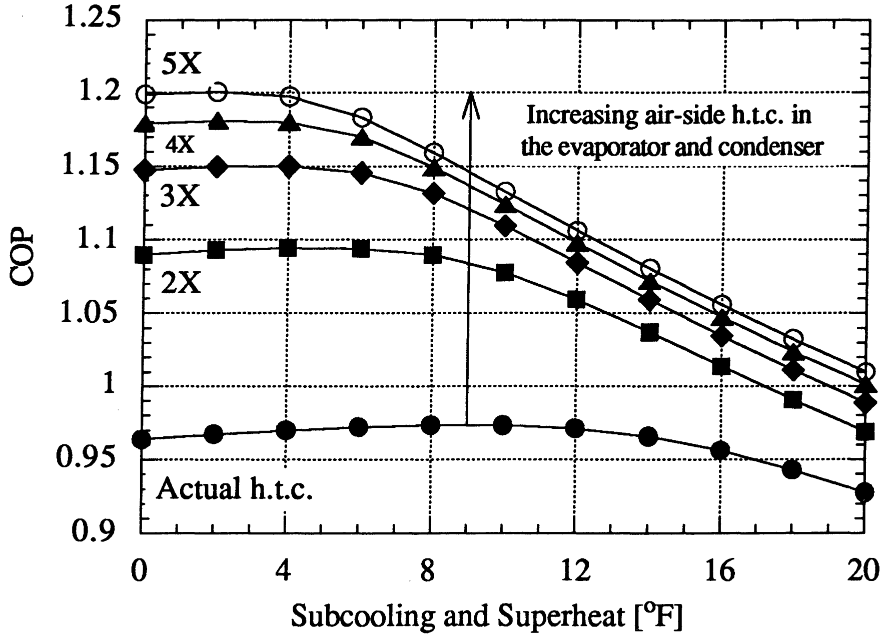


Figure 3.1 COP vs. superheat and subcooling

Although it is not known for sure why the optimal system did not correspond to the saturated heat exchanger exits, one contributing factor seems to be the finite air-side heat transfer coefficients. Since these are much smaller than the refrigerant-side heat transfer coefficients, the overall heat transfer resistance is dominated by the air-side. When the air-side heat transfer coefficients are increased by a factor of 5, the optimum heat exchanger exits "shift" over to the saturation line. It appears that when the air-side heat transfer resistance is effectively eliminated, the original argument for having saturated exits is valid because only the refrigerant-side heat transfer affects the overall heat transfer coefficient (UA) of the heat exchangers.

In any case, a superheat and subcooling level had to chosen as the design optimum. Although Figure 3.1 indicates that the actual optimum is near 9 °F, the superheat and subcooling level chosen was 1 °F. This level was chosen for two reasons: the actual difference in COP between 1 °F and 9 °F is very small; and the increase in the corresponding capillary tube length from 1 °F to 9 °F was rather large (an additional 5 ft).

Once the optimal design procedure was chosen, it became apparent that the original, high-speed compressor system would also have to be re-optimized because the refrigerant at its heat exchanger exits were two-phase mixtures. If it is not re-optimized, then the new, low-speed compressor would have an unfair advantage in the comparison. There are several reasons why the actual Whirlpool geometry does not match the optimum determined by the model: there is still some error in the model (although it has been validated); the real refrigerator was optimized for cycling operation, but the model is a steady-state model; and the average compartment temperatures in the real refrigerator may have been different than the temperatures assumed for this study (40 °F and 5 °F). Essentially, the re-optimization of the original, high-speed system can be thought of as a transformation of the cycling optimal design to a steady-state optimal design.

The low-speed and the high-speed system were both re-optimized by setting the superheat of the evaporator and the subcooling of the condenser equal to 1.0 °F and using the RFSIM model to solve for the inlet adiabatic capillary tube length and the total refrigerant charge. The results of these re-designs are shown in Table 3.1 below along with the original system.

Table 3.1 Original and re-optimzed systems

	Original high-speed compressor system	**Optimized high-speed compressor system**	**New two-speed compressor system**

3600 rpm	Actual Whirlpool design (A)
	Lin = 1.54 ft
	Mtotal = 8.25 oz
		Re-optimized @ the high-speed (B)
		Lin = 5.26 ft
		Mtotal = 8.20 oz
			System C @ 3600 rpm
			Lin = 8.49 ft
			Mtotal = 8.61 oz

2400 rpm	N.A.	N.A.	Re-optimized @ the low-speed (C)
			Lin = 8.49 ft
			Mtotal = 8.61 oz

Systems B and C were the ones that were actually compared in this study. All of the energy use comparisons were made between system B at 3600 rpm and system C at 2400 rpm. As Table 3.1 indicates, system C can operate at 2400 or 3600 rpm. The high-speed operation of system C is compared to the low-speed operation of system C and to system B when the robustness of the new two-speed compressor system (C) is evaluated.

3.3 Simulation results

3.3.1 Energy savings

The energy savings that could obtained by operating a refrigerator at a lower speed like 2400 rpm are divided into two categories: the savings due to the increase in the steady-state efficiency, and the savings due to the reduction in the cycling losses that occurs when operating at a lower evaporation capacity. The steady-state results were obtained directly from the RFSIM model. However, the cycling results were based on the RFSIM model and some experimental data obtained from the Whirlpool refrigerator in cycling operation.

3.3.1.1 Steady-state energy savings at low speed

There are two factors that result in steady-state energy savings when operating at the low speed: the compressor itself can be more efficient at the low speed; and the temperature lift of the compressor at the low speed is smaller. However, the energy savings due to these two effects are partially offset by the fact that the two heat exchanger fans are running longer because the on-cycles are longer. Therefore, the actual energy savings resulting from low-speed operation will depend on the relative magnitudes of all three factors.

Since the on-cycle lengths will be longer with a low-speed compressor, the power consumed by the fans has a drastic effect on the energy savings. The RFSIM model was used to calculate the yearly energy usage of the system B and system C operating at 2400 rpm over a range of fan powers. The results can be seen in Figure 3.2 below.

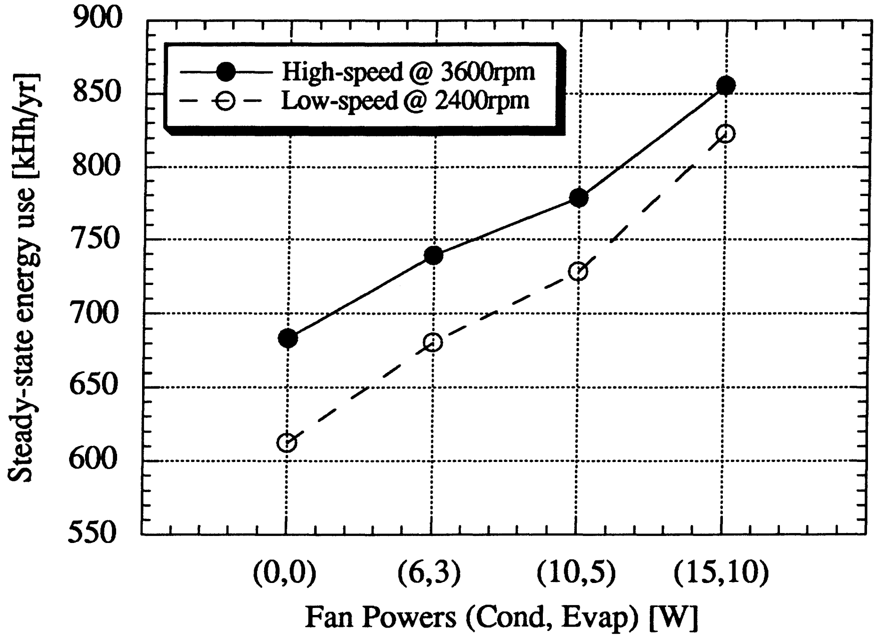


Figure 3.2 Effect of fan power on steady-state energy use

The volumetric flow rate is the same for each set of fans. This graph shows energy savings of only 4 % for the actual Whirlpool test unit, with its standard 15 W condenser fan and 10 W evaporator fan. The limiting case of "zero" fan power is included only for illustration purposes; the practical limit for fans capable of delivering the nominal air flow rate is closer to 6 W and 3 W for the condenser and evaporator, respectively. Energy savings of approximately 7% can be obtained with these efficient fans. These simulations were run with the compressor that is 1.5% more efficient at 2400 rpm than it is at 3600 rpm.

Since the compressor efficiency at the low speed has a strong impact on the overall energy savings, the above simulations were repeated for compressors having better efficiency improvements at the low speed: 3%, 5%, 7%, and 10%. The energy savings for all of the compressor/fan efficiencies were calculated and are shown in Figure 3.3 below.

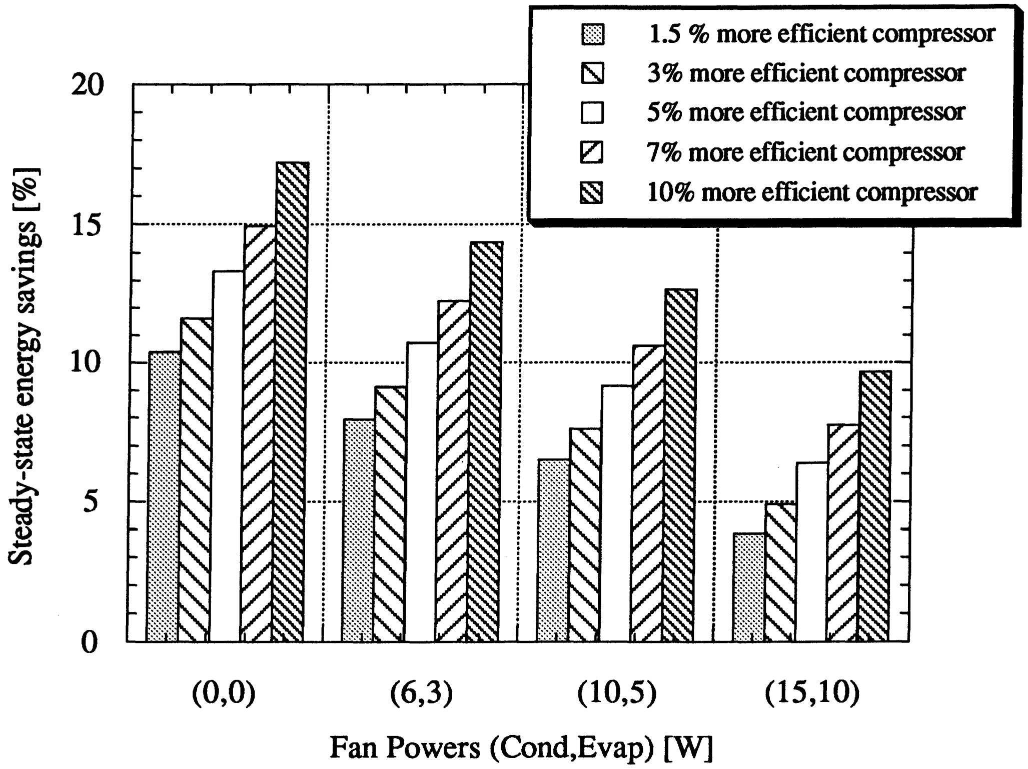


Figure 3.3 Effect of fan powers and compressor efficiencies on steady-state energy use

This graph contains all of the information to necessary to evaluate the potential steady-state energy savings due to low-speed compressor operation. For example, if the compressor used was actually 3.0% more efficient at the low speed and the fans were also efficient (6 and 3 W), then the steady-state energy savings would be approximately 9%.

The group of columns at the far left (zero fan power) are noteworthy because they show just the positive effects of the increased compressor efficiency and the reduced temperature lift. For example, the compressor with the 1.5% efficiency improvement has a overall energy savings of 10.5%. This would indicate that the reduction in temperature lift accounts for the additional 9% energy savings. When compared to the high-speed system (B), the average evaporation temperature rose from -17.8 °F to -14.1 °F and the average condensing temperature fell from 109.2 °F to 105.4 °F in the two-speed system (C) when operating at 2400 rpm. This change in the evaporating and condensing temperature is the reduction in temperature lift that results in a lower power consumption by the compressor. However, some of these energy savings are lost to the fans as shown by the three other groups of columns.

3.3.1.2 Reduction in cycling losses

In addition to the steady-state energy savings, there are energy savings due to the reduction in the cycling losses. Krause and Bullard (1996) and Coulter and Bullard (1995) measured a performance degradation in two extensively-instrumented cycling refrigerators: the difference between a cycling refrigerator and a refrigerator operating at steady state with the same heat exchanger air inlet temperatures that occur throughout the on-cycle.

Cycling losses result in both a reduction in evaporator capacity and an increase in compressor power consumption. Although the magnitude of the cycling losses is greatly dependent on the particular refrigerator, it is directly proportional to the number of cycles. In other words, the cycling loss for a particular refrigerator can be quantified in terms of Btu of capacity per cycle and W-h of power per cycle. Therefore, if the cycling frequency is decreased, there will be a corresponding decrease in the magnitude of the cycling loss for a given time period such as a day or year.

The energy savings due to reducing the cycling losses depend on the thermal capacitance of the refrigerator compartments and the temperature change experienced by the compartment contents during a cycle. An approximation proposed by Dautel (1996) begins with a time-averaged form of the following differential equation:

$$C_f \frac{dT_f}{dt} + C_z \frac{dT_z}{dt} = \dot{Q}_f + \dot{Q}_z - \dot{Q}_{evap} \tag{3.1}$$

Where C_f and C_z are the thermal capacitances (m * C_p) of the fresh-food and freezer compartments, respectively. The three rate terms Q_{frig}, Q_{frez}, and Q_{evap} are the total fresh-food heat load, total freezer heat load, and the net evaporator capacity, respectively. When the Equation 3.1 is integrated over the cycle length and the rate terms are replaced with average values, the equation can be rewritten to give a simple expression for the on- and off-cycle length as shown in Equation 3.2 below:

$$L_{on,off} = \frac{C_f \Delta T_f + C_z \Delta T_z}{\overline{\dot{Q}}_f + \overline{\dot{Q}}_z - \overline{\dot{Q}}_{evap}} \tag{3.2}$$

The two thermal capacitance terms and the change in the compartment temperatures (ΔT_f and ΔT_z) were determined by analyzing actual cycling data from the Whirlpool test unit. The three terms in the denominator of Equation 3.2 are outputs of the RFSIM model. Therefore, if these three terms are assumed to be the average values for a cycle, then the Equation 3.2 will give the approximate on-cycle length. The off-cycle length can be determined by setting the Q_{evap} term to zero and changing the signs of the ΔT_f and ΔT_z terms.

Since the available cycling loss data was reported at the four ambient temperatures of 100 °F, 90 °F, 75 °F, 60 °F, the first step in calculating the additional energy savings was to determine the reduction in the cycling losses at these four temperatures. For example, at an ambient temperature of 90 °F, the total cycle length of system C at 2400 rpm was 84 minutes, and the total cycle length of system B was 53 minutes. Therefore, the cycling frequency of system C at 2400 rpm is 30% less than system B and would have a corresponding reduction in cycling losses of 30%. The results of the cycling loss reduction at the other ambient temperatures are shown below in Table 3.2.

Table 3.2 Reduction of cycling losses for two-speed compressor at 2400 rpm

Ambient temperature (°F)	% Reduction in cycling losses
100	48
90	30
75	17
60	11

The actual energy savings due to the reduction in cycling losses are more difficult to determine because they are very refrigerator-dependent. Using the percentages in Table 3.2 and the "COP loss %" numbers reported by Krause and Bullard (1996) and Coulter and Bullard (1995), the energy savings range from 0.4% to 1.5% for the Whirlpool and from 2.2% to 4.7% for the Amana. The original cycling losses, and thus, the energy savings are higher for the Amana because its high-capacity system is sized for a faster pull-down. In general, the energy savings due to the reduction in cycling losses will be greater for refrigerators which are sized for fast pull-downs.

3.3.2 "Robustness" of the new two-speed compressor system

Although the low speed of the new compressor is more energy-efficient than the original high-speed compressor, there are several questions that remain about the performance of the new system. The main question is if the new system (C), when operated at 3600 rpm, has the same peak capacity as the original system (B) (i.e. the same pull-down performance). The other questions deal with the operation of the new system (C) at both speeds under a variety of loading conditions. A better understanding of system performance at both speeds is needed because some kind of controller will have to switch the compressor from one speed to the other based upon the loading conditions.

3.3.2.1 Capacity at the high speed

As Table 3.1 indicates, the only difference between the original system B and system C operating at 3600 rpm is the capillary tube and the amount of refrigerant charge in the system. Since the compressors are the same (at least in this study), the question of whether the system C at 3600 rpm will have the same capacity as system B is really a question of capillary tube "robustness". The new capillary tube and charge will cause the system to operate at a different point (e.g. evaporation and condensation temperatures) even though the compressor is the same. To answer the question, the RFSIM model was run with both systems over the ambient temperature range likely to be encountered during the high-speed operation. The resulting on-cycle lengths for both systems are shown in Figure 3.4 below:

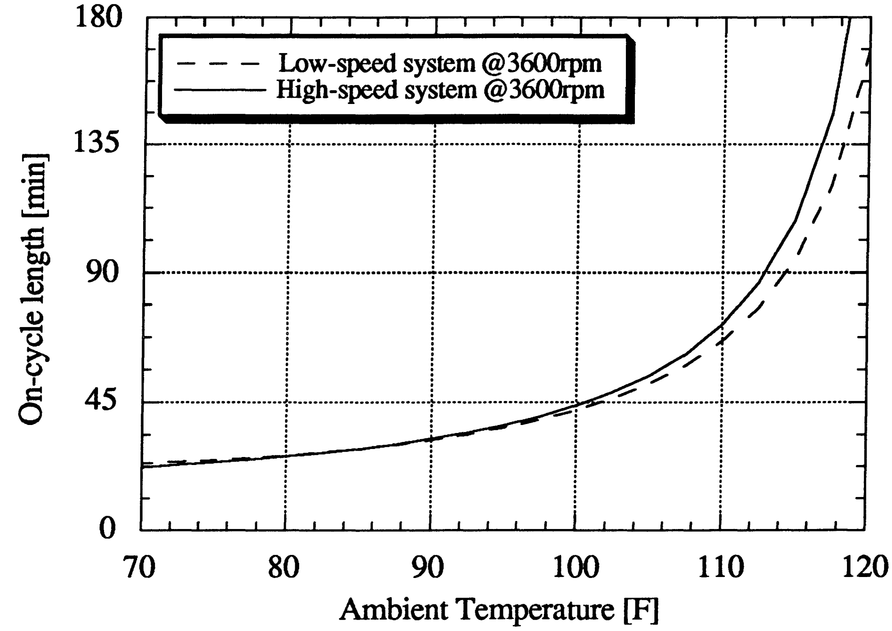


Figure 3.4 On-cycle lengths for system B and system C at 3600 rpm

The on-cycle lengths for system C at 3600 rpm are less than or equal to those for system B for the entire temperature range. This means that the capacity of the new system is slightly better than the original system. These results suggest that the capillary tube design and charge level that were chosen for system C at 2400 produce a system that also provides the required capacity at the high speed.

3.3.2.2 Operation of two-speed compressor under various loadings

The performance at the low and high speed of the new compressor system was examined under different loading conditions: a range of ambient temperature ranges, additional steady-state heat loading, and additional "one-time" heat loading. The ambient temperature is the most significant loading factor since it determines the amount of heat transfer through the refrigerator walls and the length of the on- and off-cycles. Refrigerators are frequently subjected to additional "one-time" heat loads such as warm food that must be cooled down. Although less frequent, it is possible for a refrigerator to experience a additional steady-state loading such as incomplete door closure or a faulty door gasket.

The on-cycle length of both speeds of the new system were calculated for a range of ambient temperatures using the RFSIM model. These results served as the "base" case for the other two loading conditions. The RFSIM model was run again for both speeds over a range of ambient temperatures with an additional steady-state heat input of 20W. The results of these two sets of model runs are shown in Figure 3.5 below:

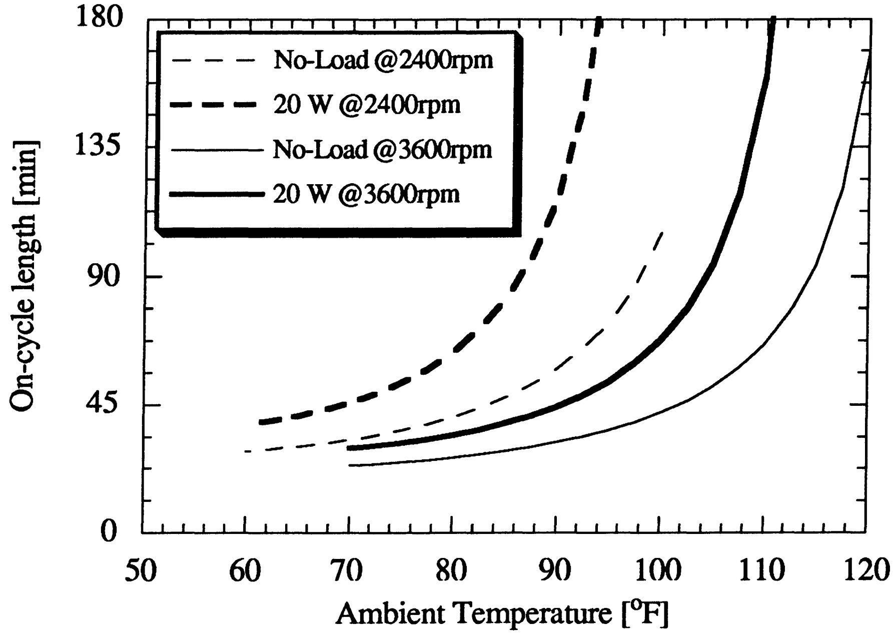


Figure 3.5 On-cycle lengths for the no-load condition and a 20W steady-state load

The dashed lines are for the low-speed compressor operation, and the solid lines are for the high-speed compressor operation. At each speed, the heavier lines represent the 20W load condition. Although the 20W load was somewhat arbitrary, it does illustrate the fact that a steady-state load will increase the cycle length at every ambient temperature. For example, in a closed-door, 90 °F situation (e.g. the DOE energy test), the compressor would operate at the low speed in order to take advantage of the previously described energy savings. However, if a refrigerator with a worn-out door gasket was placed in 90 °F room, the compressor would have to operate at the high speed to avoid excessive cycle lengths. This scenario suggests that one possible control strategy would be to monitor the cycle length. If the compressor were running at the low speed and the on-cycles were longer than some preset time (e.g. 90 or 120 minutes), the compressor would be switched to the high speed. Likewise, if the compressor was running at the high speed and the cycles were too short, the compressor would be switched to the low speed.

The model results for the "base" case and a modified form of Equation 3.2 were used to calculate the on-cycle lengths for the two compressor speeds with a one-time heat load. The heat load of 200 Btu was simply added to the terms in the numerator and the new cycle lengths were calculated. This calculation assumes that the entire 200 Btu load is removed by the refrigerator in one cycle. In a real refrigerator, an additional one-time heat load (such as warm food) would probably not be removed completely during a single cycle. The cycle length for the "base" case and the loaded case are shown for both compressor speeds in Figure 3.6 below:

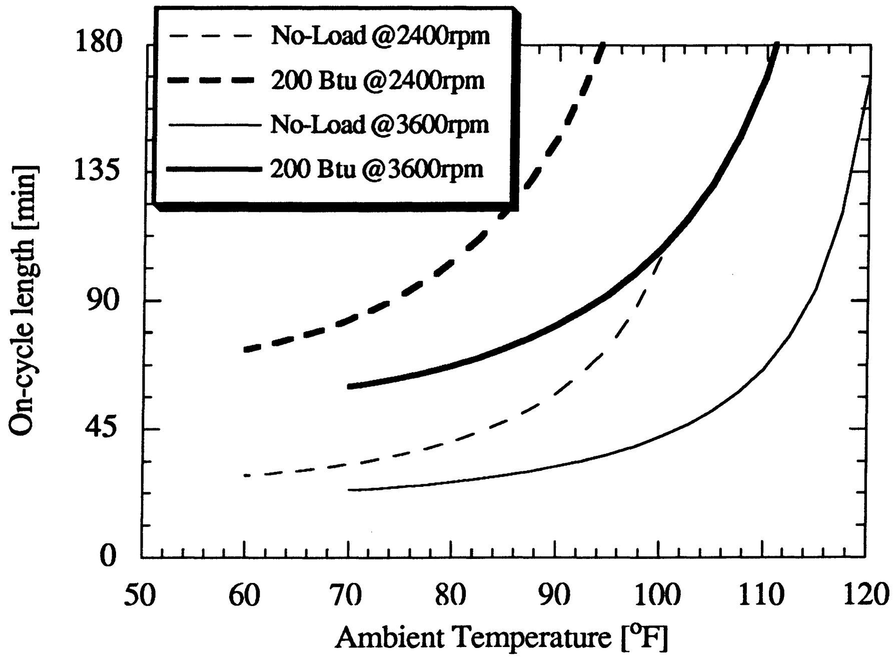


Figure 3.6 Cycle lengths for the no-load condition and a one-time load of 200 Btu

This graph is very similar to Figure 3.5 and illustrates the same basic principle of an increased cycle length due to an extra heat load. However, the difference between the two graphs is that Figure 3.6 represents the difference in the cycle lengths for one cycle. In other words, after the 200 Btu has been removed from the compartments during one cycle, the refrigerator would then begin to cycle according to the "base" case cycle-length line. The difference may seem subtle, but it has implications in regard to the control strategy that would be required to switch from one speed to the other. Due to the inherently transient characteristics of the one-time heat load, the compartment temperatures may exceed specified limits before the control strategy would switch to the high compressor speed. In other words, by the time the control system "realized" that the low speed compressor was inadequate because of a long on-cycle, the temperature of the refrigerator contents may have been too high for too long.

Based upon this observation, a better control strategy might be to monitor one or both of the compartment temperatures and the time since the beginning of the on-cycle. If one of the compartment temperatures did not drop below a set point by a certain length of time (e.g. 5 or 10 minutes), then the compressor would be switched to the high speed. This would ensure (at least within the capacity of the high-speed compressor) that the compartment temperatures would not be out of range for too long. It would also prevent the compressor from being switched to the high speed every time the door was opened; since the low-speed compressor would generally be sufficient to cool the air within the compartments back down to the setpoint.

Once the compressor was switched to the high speed, it would finish that cycle and then remain at the high speed until the higher capacity was no longer needed. The signal to switch to the low speed could be triggered by a cycle length shorter than some preset value. Since a "short" cycle could have different meanings depending on the ambient temperature, a better method might be to count the number of cycles in which the compartment temperature was brought below the setpoint within the allotted time. This would give some indication of when the refrigerator was returning to normal operation. Keeping the compressor at the high speed as long as necessary would prevent the compartment temperatures from going out of range again and again if there was a large heat load (like warm groceries) that would affect several consecutive cycles.

3.3.3 Effects of changing fan speed along with compressor speed

Another way to change the capacity of refrigerator is to vary the speed of one or both of the heat exchanger fans. According the "fan laws", the volumetric flow rate of air blown by a fan through a fixed duct system is directly proportional to the speed of a fan (ASHRAE, 1992). In other words, if the speed is increased by 50%, then the volumetric flow rate will increase by 50%. In the refrigerator, any change in the volumetric flow rate will change the velocity of the air flowing over the heat exchanger, and consequently, the air-side heat transfer coefficient as well. Since the majority of the heat transfer resistance of the heat exchanger is on the air side, any change in the air-side heat transfer coefficient will have a significant impact on the overall heat transfer coefficient of the heat exchanger.

In addition to changing the air-side heat transfer coefficient, a change in fan speed will be accompanied by a change in the power required to rotate the shaft. According the "fan laws", the shaft power of a fan moving air through a fixed duct system is proportional to the speed ratio raised to the third power as shown in Equation 3.3 below:

$$\frac{Power_{new}}{Power_{nominal}} = \left(\frac{N_{new}}{N_{nominal}}\right)^{3} \tag{3.3}$$

How the actual electrical power drawn by the fan motor varies with the speed ratio will depend on the efficiency of the motor over the speed range. If the motor efficiency is assumed to be constant over the speed range of interest, then the ratio of the electrical powers will equal to the ratio of the shaft powers. Because of the power of "3" in the fan law, even a slight change in the fan speeds can result in a dramatic change in the fan powers.

Therefore, any change in the speed of the evaporator or condenser fan will be accompanied by two opposing forces. If the speed is increased, the heat transfer coefficient will increase, but the electrical fan power will also increase (depending upon the motor efficiency assumption). If the speed is decreased, then the situation will be reversed. In either case, the change in heat transfer coefficient will affect the refrigerant states within the heat exchangers as well as the heat transfer. Thus, it is possible for a change in fan speeds to change evaporating and condensing temperatures, superheat and subcooling, and other system variables throughout the system. All of these factors will determine the magnitude of heat transfer in the evaporator and condenser, the compressor power consumption, and the overall efficiency of the refrigerator.

3.3.3.1 Modeling methods and assumptions

The RFSIM model was used to evaluate the feasibility of varying the fan speed in addition to the compressor speed. The intention of this study was not to explore every possible scenario in which the speed of one or both of the fans could be varied. Instead, the intent was to run a sufficient number of simulations that would hopefully identify areas of operation (e.g. low or high compressor speed, ambient temperature) that would benefit from a change in one of the fan speeds. The benefits that were sought were reductions in energy use or increased evaporator capacity.

In addition to gaining a better understanding of the possible benefits of varying the fan speeds, the model results were used to answer a few specific questions about the two-speed compressor system operating at off-design conditions: can the evaporator fan speed be slowed down at the low compressor speed to obtain additional energy savings at low ambient temperatures; when the compressor is running at the low speed, can a increase in speed of either fan be substituted for a switch to the high compressor speed; and can the capacity of the high-speed compressor be significantly increased, without too much energy penalty, by increasing the speed of either fan.

It should be noted that the results of this study, and any similar study, are very dependent upon the assumptions that were made and the modeling methods that were used. First, the nominal fan powers were chosen to be 6W for the condenser fan and 3W for the evaporator fan. The choice of low fan powers was based upon the assumption that variable speed fans would probably already have efficient motors. These low nominal fan powers affect the results because there is not as much "room" for energy improvement by reducing the fan speeds, and not as much energy penalty for increasing them. The second major assumption was that the motor efficiency remained constant over the speed range simulated. To keep the number of simulations to a reasonable level, only one magnitude of speed adjustment, ± 25%, was considered for the evaporator and condenser fans, and it was applied to one fan at a time for the low-speed and high-speed compressor. The resulting test matrix made it possible to examine the effects, at both compressor speeds, of individually slowing down and speeding up the evaporator and condenser fans over a range of ambient temperatures (see Appendix H).

3.3.3.2 Model results

The first question of whether the evaporator fan could be slowed down at the low compressor speed to obtain additional energy savings was answered by the modeling results. At an ambient temperature of 60 °F, the annual energy use would virtually remain constant if the evaporator fan speed were reduced by 25%. However, the evaporator capacity would decrease

4.5%, from 441 Btu/hr to 421 Btu/hr. Apparently, the decrease in the fan power (~1.7W) was completely offset by the increased on-cycle length that results from the lower capacity. Similar results were obtained at all of the ambient temperatures between 60 °F and 100 °F. Therefore, it does not appear that reducing the evaporator fan speed at low-load conditions will yield net energy savings.

Although it is reasonable to assume that the evaporator capacity will increase when the speed of the evaporator fan is increased, it was not known *a priori* whether the increase would be sufficient to substitute for a switch to the high compressor speed. The simulation results from the model show that when the compressor is running at the low speed in an ambient temperature of 100 °F, the speed-up of the evaporator fan can increase the evaporator capacity by 18 Btu/hr, from 429 to 447 Btu/hr. This 4.2% increase is due primarily to the 17% increase in air-side heat transfer coefficient which causes a 14% increase in the "UA" of the evaporator. However, approximately 10 of the additional 18 Btu/hr would be used to remove the 3 extra Watts of fan power entering the compartments. Therefore, only an additional 8 Btu/hr is actually available the cool the contents of the refrigerator. Since there is also an energy penalty of about 1.5%, increasing the speed of the evaporator fan to obtain additional capacity at the low compressor speed does not seem promising. A greater increase in fan speed would probably increase capacity, but the fan power rejected to the compartments will increase at a faster rate.

The results of increasing the evaporator fan speed at the high compressor speed showed similar results. At ambient temperatures between 90 °F and 120 °F, the evaporator capacity can be increased by a fairly constant 23 Btu/hr when the evaporator fan speed is increased by 25%. As before, this increase in evaporator capacity is due to the 17% increase in air-side heat transfer coefficient. Also as before, there is an additional 10 Btu/hr from the fan that must be removed from the compartments. Although the energy penalty is less than 1% for the entire temperature range, it is doubtful that an additional 13 Btu/hr of cooling capacity would be worth the extra expense of a variable-speed evaporator fan.

Regardless of the compressor speed or ambient temperature, increasing the condenser fan speed by 25% does not produce any beneficial results. Invariably, the energy consumption increases and the evaporation capacity decreases. However, the results are much more promising for the case of reducing the condenser fan speed. At the lower ambient temperatures for both compressor speeds, the energy use decreases and evaporation capacity increases when the condenser fan speed is reduced by 25%. For example, at an ambient temperature of 60 °F with the low compressor speed, the energy use decreases 2.7% while the evaporator capacity increases by 2.1 %, or 9 Btu/hr. Although the energy reduction is expected, the increase in evaporator capacity is somewhat surprising.

A closer examination revealed the cause of the capacity increase. As mentioned earlier, changing the air-side heat transfer coefficients of one of the heat exchangers will affect the state of the refrigerant within the heat exchanger. This, in turn, will affect the operation of the other heat exchanger and the rest of the system components. The lower condenser air-side heat transfer coefficient associated with the decrease in the fan speed raises the condensing temperature and decreases the size of the subcooled region of the condenser. Since the subcooled region contains a large percentage of the condenser's refrigerant charge, the lower fan speed reduces the mass of refrigerant within the condenser. For all practical purposes, the decrease in condenser refrigerant mass will result in an equal increase in the evaporator mass. The increase of the refrigerant mass in the evaporator causes the two-phase zone to become larger, the superheated zone to become smaller, and the evaporation temperature to rise (~1 °F). The relative increase in the two-phase zone, which has a higher "UA", produces an overall increase in the evaporator heat transfer.

The key to this beneficial fan speed adjustment is the high superheat level that originally exists in the evaporator (~18 °F). Since the evaporation temperature increases, the air-refrigerant temperature difference decreases. Therefore, the higher "UA" of the evaporator had to overcome the lower ΔT before it could increase the heat transfer. When the superheat level is already low, the increase in the evaporator "UA" is not enough to overcome the reduction in the heat exchanger ΔT. This appears to be the reason why the benefit of a lower condenser fan speed is only observed at the low ambient temperatures (high superheat and subcooling).

The dependence of the energy savings on the ambient temperature can be seen more clearly when COP is plotted verses condenser fan speed as in Figure 3. 6 below:

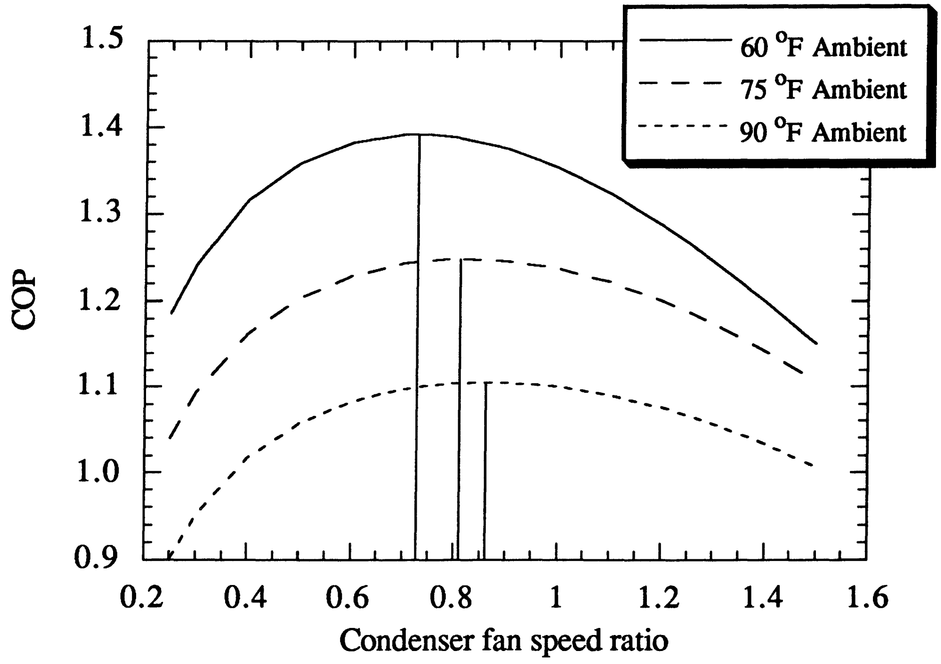


Figure 3.6 Optimum condenser fan speed vs. ambient temperature

The fan speed ratios are simply the ratios of the new fan speed to the nominal fan speed. For example, the 25% reduction in fan speeds already discussed would correspond to a fan speed ratio of 0.75. For each ambient temperature curve, vertical lines have been drawn at the optimum values of the condenser fan speed ratio.

At 90 °F ambient, there is practically no benefit in decreasing the speed of the fan from the nominal setting. At this temperature there is maximum in COP when the condenser fan speed is decreased to ~86% of its nominal value, but the increase in COP from the nominal value is negligible. At 75 °F ambient, the optimum condenser fan speed appears to be at 80% of its nominal value. Although it is difficult to tell from the graph, there is 1% increase in the COP when compared to the nominal value. At 60 °F ambient, the results are similar to what was reported in the previous paragraph because the optimum fan speed ratio shown in Figure 3.6 is very close to the 25% decrease in speed that was originally examined. There is a 2.8% increase in the COP at the optimum fan speed ratio of 0.73.

The results of Figure 3.6 suggest that a two-speed condenser fan might be beneficial. The high fan speed would correspond to the current nominal speed and would be used at ambient temperatures higher than 75 °F. The low fan speed would operate at about 75% of the high speed and would operate at ambients lower than 75 °F. Operation at the low fan speed would result in a energy savings and a noise reduction for many refrigerators.

3.4 Conclusion

With the validated RFSIM model, many aspects of the operation of a two-speed compressor refrigerator were examined. From the results of the analysis, it appears that there are substantial energy savings to be obtained from the increased steady-state efficiency of the new system when operating at the low speed. Depending on the efficiency improvement of the compressor at the low speed and the fan powers, these steady-state energy savings could realistically range from 4 to 14%. The results of the analysis also indicate that an additional 0.5 to 4% energy savings might be obtained from the reduction in the cycling frequency. It is not clear whether energy savings of this magnitude could justify the incremental cost of a 2-speed compressor. More substantial benefits might include the reduction in compressor-related noise and increased reliability resulting a from a lower cycling rate.

In addition to the energy results, several aspects of the robustness of the new compressor system were examined using the simulation model. The most important result was that a system optimized for low-speed operation, when operating at the high speed, would have as much capacity the original high-speed system. Although the analysis of the new system's response to internal loads was somewhat qualitative, it provided a crude basis for recommending a possible control strategy for two-speed compressors. That control system might include at least one temperature measurement, a timer, and possibly a counter. If the temperature did not fall below a certain setpoint in certain amount of time, the compressor would switch to the high speed. Similarly, the compressor would switch back to the low speed when the cycles became too short.

The effects of varying the evaporator or condenser fans at both compressor speeds were examined over a range of ambient temperatures. Several operating scenarios that were expected to benefit from fan speed adjustment were analyzed with the matrix of model runs. None of the scenarios seemed to benefit significantly from the appropriate fan speed adjustments. However, it was found that slowing down the condenser fan at low ambient temperatures (60 °F to 75 °F) would yield an energy savings as well as a capacity increase. Basically, the lower condenser fan speed transferred refrigerant charge from the condenser to the evaporator where it reduced the superheat and, thereby, increased the "UA" of the evaporator. This option seems even more attractive when the reduction in fan noise is considered.

Chapter 4

Summary and Conclusions

4.1 RFSIM model

The governing equations and the supporting FORTRAN routines that make up the RFSIM model are solved by the ACRC Solver. This solver uses a Newton-Raphson technique that employs several enhancements in terms of solution speed and convergence robustness. Because the equations are solved with the Newton-Raphson technique, they can be written in any order. Therefore, the equations are relatively easy to modify or replace; for example, the equations can be modified to model a heat exchanger in greater detail or to evaluate components of substantially different design. The ACRC Solver has features which allow RFSIM to correctly model, without user intervention, single- and two-phase heat exchanger exit conditions. The ACRC Solver also provides a simple method of switching variables and parameters within the governing equations; for example, variables and parameters can be switched to solve for the geometric parameters required to achieve a specified energy efficiency. This switching feature increases the flexibility of the RFSIM model and allows it to be used as a design or a simulation tool.

The RFSIM model is a comprehensive steady-state refrigerator model that is made up of approximately 110 governing equations. Although most of the equations of RFSIM are “physical” in that they are based upon "first principles" and generally applicable correlations, the entire model cannot be considered “physical” because it depends on several user-supplied parameters. Therefore, the validation process for RFSIM may be best described as a validation of the modeling procedure itself and not as statement that the model will always predict the variables within some accuracy.

The modeling procedure used within the RFSIM model has been validated using data from two refrigerators. The accuracy of the model is better in design mode because fewer equations and more user-specified parameters are used. With the data from the Amana refrigerator, the average errors in design mode were all less than 4% for the COP, system power, evaporation capacity, and mass flow and were less than 1 °F for the evaporating and condensing temperatures. The accuracy of the design mode of the model appears to be limited by the accuracy of the manufacturer-supplied compressor maps.

Although there was more scatter in the accuracy of the simulation mode results, the modeling procedure within the simulation mode can also be considered valid. With the data from the Whirlpool refrigerator, the average errors of the simulation mode were all less than 4% for the COP, power, evaporation capacity, and mass flow rate, and the vast majority of the errors in the evaporating and condensing temperatures were less than 2 °F. Although the average errors were acceptably small, there were still some fairly large underpredictions of some of the variables that were due primarily to error remaining in the capillary tube-suction line heat exchanger model.

4.2 Multi-speed compressor

From the results of the multi-speed compressor analysis, it appears that the energy savings to be obtained from the increased steady-state efficiency at the low compressor speed could realistically range from 4 to 14%. The results also indicate that an additional 0.5 to 4% energy savings might be obtained from the reduction in the cycling frequency of the refrigerator.

Several aspects of the robustness of the capillary tube-suction line heat exchanger design for the two-speed compressor system were examined with the simulation model. It was shown that a system optimized for low-speed operation, when operating at the high speed, could have as much capacity the original base-case high-speed system. A relatively simple control strategy was proposed; one that requires measurement of on-cycle time and one or two compartment air temperatures.

The effects of varying the speed of the evaporator or condenser fans at both compressor speeds were examined over a range of ambient temperatures. One energy-saving scenario was identified: decreasing the condenser fan speed for refrigerators operating at low ambient temperatures. By affecting the distribution of refrigerant charge throughout the system, the decrease in condenser fan speed reduces the superheat in the evaporator and increases the overall "UA" of the evaporator. The resulting increase in evaporator capacity more than offsets the decrease in condenser "UA", and the energy use of the refrigerator is decreased.

References

ASHRAE, ASHRAE Handbook HVAC Systems and Equipment, S18.4, Atlanta GA, 1992.

Bockhold, D., personal communication, University of Illinois at Urbana-Champaign, 1996.

Cavallaro, A. R., and C. W. Bullard, "Effects of Varying Fan Speed on a Refrigerator/Freezer System," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 101:2, in press, 1995.

Christoffersen, B.R. and J.C. Chato, "Heat Transfer and Flow Characteristics of R-22, R-32/R-125 and R-134a in Smooth and Micro-Fin Tubes," ACRC TR-47, University of Illinois at Urbana-Champaign, 1993.

Coulter, W.H., and C.W. Bullard, "An Experimental Analysis of Cycling Losses in Domestic Refrigerator-Freezers," ACRC TR-77, University of Illinois at Urbana-Champaign, 1995.

Dautel, T., "Possible Control Strategies for Variables-Speed Compressor Refrigerators," ME-393 project, University of Illinois at Urbana-Champaign, 1996.

Dobson, M.K. and J.C. Chato, "Heat Transfer and Flow Regimes During Condensation in Horizontal Tubes," ACRC TR-57, University of Illinois at Urbana-Champaign, 1994.

Dukler, A.E., et al., "Pressure Drop and Hold Up in Two-Phase Flow," A. I. Ch. Journal, Vol. 10, pp. 38-51, 1964.

Goodson, M.P., and Bullard, C.W., "Modeling a Refrigerator/Freezer System," ACRC TR-61, University of Illinois at Urbana-Champaign, 1994.

Grebner, J.J. and R.R. Crawford, "The Effects of Oil on the Thermodynamic Properties of Dichlorodifluoromethane (R-12) and Tetrafluoroethane (R-134a)," ACRC TR-13, University of Illinois at Urbana-Champaign, 1992.

Haaland, S.E., "Simple and Explicit Formulas for the Friction Factor in Turbulent Pipe Flow," J. Fluids Eng., March 1983, pp. 89-90.

Hahn, G. W., and C. W. Bullard, "Modeling Room Air Conditioner Performance," ACRC TR-40, University of Illinois at Urbana-Champaign, 1993.

Incropera, F., and DeWitt, D., Fundamentals of Heat and Mass Transfer, John Wiley and Sons, New York, 1990.

Ito, H., "Pressure Losses in Smooth Pipe Bends," Journal of Basic Engineering: Transactions of the ASME, March 1960, p.135.

Krause, P. E., and C. W. Bullard, "Cycling and Quasi-Steady Behavior of a Refrigerator," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 102:1, in press, 1996.

Liu, L. and C.W. Bullard, forthcoming M.S. Thesis, University of Illinois at Urbana-Champaign, 1996.

Mullen, C. E. and C. W. Bullard, "Room Air Conditioner System Modeling," ACRC TR-60,University of Illinois at Urbana-Champaign, 1994.

Porter, K. J., and C. W. Bullard, "Modeling and Sensitivity Analysis of a Refrigerator/Freezer System," ACRC TR-31, University of Illinois at Urbana-Champaign, 1992.

Rice, C.K., “The Effect of Void Fraction Correlation and Heat Flux Assumption on Refrigerant Charge Inventory Predictions,” ASHRAE Transactions, Vol. 93, Part 1, 1987, pp. 341-367.

Rubas, P. J., and C. W. Bullard, "Factors Contributing to Refrigerator Cycling Losses," Intl. Journ. of Refrigeration, 18:3, 1995.

Souza, A.L., and Pimenta, M.M.,"Prediction of Pressure Drop During Horizontal Two-Phase Flow of Pure and Mixed Refrigerants," ASME Symposium on Cavitation and Multiphase Flow, 210, pp. 161-171, 1995.

Stoecker, W. F., Design of Thermal Systems, 3rd ed., New York: McGraw-Hill, 1989.

Swamee, P.K., and Jain, A.K., "Explicit Equations for Pipe-Flow Problems," Proceedings of the ASCE, Journal of the Hydraulics Division, 102, HY5, p.657-664, May 1976.

Wattelet, J.P. and J.C. Chato, "Heat Transfer Flow Regimes of Refrigerants in a Horizontal-Tube Evaporator," ACRC TR-55, University of Illinois at Urbana-Champaign, 1994.

Appendix A

Refrigerator/Freezer Simulation (RFSIM) Model User's Reference

The purpose of this document is to acquaint any potential users with the operation of the RFSIM model. Since the ACRC Solver and the RFSIM model are intricately linked, some of the information covered in the ACRC Equation Solver User's Reference (Appendix A of Mullen and Bullard, 1994) will be repeated here. There will be several instances where the reader will be referred to Appendix A of Mullen and Bullard (1994) for more information about the actual solver. This document will focus primarily on topics which are unique to the operation of the RFSIM model.

A.1 XK input file

The "XK" file is the primary way that the user interacts with the RFSIM model. It serves as the input file for each model run, and the output of the model is written in the same form in the "XK.out" file. This makes it very easy to use to output of one model run as the basis for the input for another model run. The "XK" file is list of all the system variables (residuals), many of the system parameters, and many values that are calculated after the Newton-Raphson solution is complete. These three groups of variables are denoted by a "X", "K", and a "C" within the XK file, respectively. Originally, the XK file only contained Xs and Ks, and therefore, was named the "XK" file (Goodson and Bullard, 1994). The best way to illustrate the various features of the XK file is to look at an example. Figure A.1 below shows excerpts of an actual XK file used with the RFSIM model.

```
** XK initialization file: initializes variable guesses and parameter values.
**   Output Flag specifies if variable is printed to spreadsheet readable file
**         ( 1 = Print, 0 = Don't Print )
**   Parameters are flagged with "K" and variables are flagged with "X."
**   The units are delimited with '[ ]'.
**   The last number signifies the number of decimal places (0-10).
** The ORDER of the input lines CANNOT CHANGE without program modification.
Output Flag Name          XK#                     Value  Units      # of digit
***********  DO NOT DELETE THESE FIRST NINE LINES!    ***************
:
1  X  powercomp      = XK( 56) =            102.767 [Watts]             3
1  X  qcond          = XK( 57) =            542.366 [Btu/hr]            3
1  X  qsupcond       = XK( 58) =             95.499 [Btu/hr]            3
1  X  q2phcond       = XK( 59) =            444.915 [Btu/hr]            3
1  X  qsubcond       = XK( 60) =              1.952 [Btu/hr]            3
1  X  qevap          = XK( 61) =            455.143 [Btu/hr]            3
1  X  q2phevap       = XK( 62) =            452.623 [Btu/hr]            3
1  X  qsupevap       = XK( 63) =              2.520 [Btu/hr]            3
1  X  qcomp          = XK( 64) =            263.626 [Btu/hr]            3
:
1  K  tamb           = XK(172) =              90.00 [F]                 2
1  K  tafrig         = XK(173) =             40.000 [F]                 3
```

```
1  K  tafrez         = XK(174) =        5.000 [F]               3
1  K  UAf            = XK(175) =        0.855 [Btu/hr-F]        3
1  K  UAz            = XK(176) =        0.502 [Btu/hr-F]        3
1  K  vdotcond       = XK(177) =      105.134 [ft^3/min]        3
1  K  vdotevap       = XK(178) =       51.615 [ft^3/min]        3
:
1  C  Itot           = XK(183) =        246.8 [Btu/hr]          1
1  C  Icomp          = XK(184) =        138.3 [Btu/hr]          1
1  C  Icond          = XK(185) =         49.6 [Btu/hr]          1
1  C  Ievap          = XK(186) =         48.1 [Btu/hr]          1
1  C  Ipipes         = XK(187) =         10.8 [Btu/hr]          1
1  C  TcondAvg       = XK(188) =      105.548 [F]               3
1  C  TevapAvg       = XK(189) =      -14.576 [F]               3
```

Figure A.1 Portion of an XK file

Since there are other intermediate variables within the governing equations, the Xs will be referred to as "residual variables" or just "variables" while the intermediate variables (not in the XK file) will be referred to as "non-residual variables". This distinction will become clearer as both types are later discussed. The second quantity on each line is the XK flag. This flag tells the program whether each quantity is a X, K, or a C. As in this XK file, the variables are usually listed first, followed by the parameters, and then the calculated values. Within each group, the quantities are listed in approximate alphabetical order. The order is not really important and is done primarily to aid the user in finding a particular quantity in the XK file. Although the order is totally arbitrary, it must match the order of the variables, parameters, and calculated values in the file EQUIVLNT.INC.

A.1.1 Variables (Xs)

The Xs are sometimes referred to as residual variables because these are the variables that are actually solved for in the residual, or governing, equations. The Newton-Raphson (NR) routine adjusts these variables until all the residual equations are satisfied to within a specified tolerance. The non-residual, or intermediate, variables appear in the governing equations, but they are calculated directly from residual variables or parameters. Although the non-residual variables do not appear in the XK file, the user should be aware of their existence.

The number of variables (Xs) in the XK file must match the number of residual equations located in EQNS.f. In addition, there must be at least one variable in every equation in order for the solver's NR method to work. If one or more equations does not have any variables, then the system will have a singularity. In other words, the residual values of one of the equations will not have any dependence on the variables, and therefore, the NR method will not work.

Any variable (X) may be exchanged, or "swapped", for a parameter (K) in the XK file as long as the number of variables remains the same. Swapping an X and K simply means that some variable (X) will become a parameter (K) and some parameter (K) will become a variable (X). This is shown schematically below for a variable and a parameter from the above XK file.

```
1  X  qcond          = XK( 57) =          542.366 [Btu/hr]        3
1  K  tamb           = XK(172) =            90.00 [F]             2

Will become

1  K  qcond          = XK( 57) =          542.366 [Btu/hr]        3
1  X  tamb           = XK(172) =            90.00 [F]             2
```

In this example, the condenser heat transfer (qcond) was originally a variable and the ambient temperature (tamb) was a known parameter. If someone, instead, wanted to solve for the ambient temperature that would yield a specified condenser heat transfer, the XK flags for the two quantities would be switched by editing the XK file as shown above. This would let the solver know that the value given for qcond was fixed, and the value for tamb could be varied by the NR method until the governing equations were solved. There is the additional requirement that the equations must remain independent and non-singular. There are also some restrictions as to which parameters (Ks) can become Xs, but this will be discussed in the following section. One issue that was discussed in Appendix A of Mullen and Bullard (1994) is the re-building of the NonZeroList. Every time an X and a K are switched, the NonZeroList has to be re-built because of the sparse-matrix Jacobian calculation that is implemented in the model. The NonZeroList is essentially a "map" of the dependence of all the residual equations on all of the variables (Xs). This map allows for the quick calculation of the Jacobian matrix (partial derivatives of all the equations with respect to all the variables), which is a part of the NR method. If an X and a K are switched, the existing NonZeroList may no longer be valid, so it has to be rebuilt.

A.1.2 Parameters (Ks)

As the name implies, parameters are values that are constant for given solution. Parameters are typically measurable quantities such as tube diameters and heat exchanger areas as well as estimated quantities such as heat transfer coefficients and volumetric air flow rates. Parameters are included in the XK file to make the model more flexible and useful. If all the known parameters for a given refrigerator were placed directly in the governing equations, the model would work for that particular case, but it would be difficult to modify the model for another refrigerator or even for other operating conditions. In the RFSIM model, most of the parameters in the governing equations are actually "variables", in the FORTRAN sense, whose values are set elsewhere. This allows the user to easily change the values in the XK file without having to recompile the FORTRAN. With the "MULTIPLE" run ability described in Appendix A of Mullen and Bullard (1994), the values of one or more parameters can be changed over a specified range, and the model will be solved at every point. For example, the user can examine the effect, on the rest of the system, of changing the ambient temperature from 60 °F to 100 °F.

Since parameters are just constants used within the governing equations, there is no limit to the number of parameters that can be in the XK file (other than array size limits specified within DIMENSN.INC). As mentioned in A.1.1, the variables (Xs) can be switched with the parameters (Ks) for a given solution. However, there are some parameters (Ks) that cannot be switched. In general, these non-switchable parameters are Ks whose numerical values are used as "flags" in the governing equations. For example, there is a parameter named "CompNum" whose value (1, 2, 3, 4, etc.) determines which compressor maps should be used by the model. Below in Figure A.2 are the parameters (Ks) that cannot be switched with variables (Xs).

```
1  K  CaptubeOutput  = XK(119) =             0.0 [ ]          1
1  K  CompNum        = XK(120) =              4. [ ]          0
1  K  ectslhx        = XK(132) =           0.748 [ ]          3
1  K  hcondNum       = XK(138) =              0. [ ]          0
1  K  hevapNum       = XK(140) =              0. [ ]          0
0  K  numDPin        = XK(159) =              4. [ ]          0
0  K  numDPout       = XK(160) =              5. [ ]          0
0  K  numDTsl        = XK(161) =              6. [ ]          0
```

Figure A.2 Non-switchable parameters (Ks)

With the exception of "ectslhx", all the parameters in Figure A.2 fall under the category described above. The effectiveness of the captube-suction line heat exchanger (ectslhx) cannot be made an unknown because of the way the captube-suction line heat exchanger equations are handled within the RFSIM model. If the captube model is used, ectslhx will automatically become a C (without user intervention) since ectslhx will be calculated after the NR solution is complete. If the captube model is not used, ectslhx must be a K since it describes the heat transfer taking place in the captube-suction line heat exchanger. Therefore, ectslhx can never be made an X in the XK file.

A.1.3 Calculated values (Cs)

As mentioned earlier, the third category of calculated values did not exist in the earlier version of the RFSIM model. Instead, the few values that were calculated outside of the governing equations were flagged with a K. This lead to some confusion since new users would not know whether a "parameter" in the XK file was really a known constant or something that was calculated after the NR solution was complete. With this new category, the user will know from each XK solution file what was a parameter and what was a calculated value.

As alluded to in A.1.1, the calculated values (Cs) cannot be switched with the variables (Xs) or the parameters (Ks). Actually, Cs are similar to the Xs in that they are both outputs of the model. Although not in the "switching" procedure previously discussed, some of the Xs in the current model could become Cs and some of the Cs could become Xs <u>if</u> the number of residual equations was changed. Instead of calculating some variable by means of a residual equation in the NR equation set, the calculation could be done after the NR solution is complete. The primary motivation for calculating a quantity as a C instead of an X is to reduce the number of residual equations necessary in the model. In other words, it is much quicker to obtain some of the desired outputs of the model by explicit calculations than it is by iterative methods such as the Newton-Raphson method.

However, when a calculation is placed outside of the residual equations, the user loses flexibility as to what can be specified as an input and an output (through the use of the X-K switching). If the user is confident that a certain quantity will always be an output and it can always be calculated explicitly, then it would be better if that calculation was done in the subroutine "FC" located in the file CHECKMOD.f. This subroutine is called after the solution is complete and can perform as many explicit calculations as required by the model. A good example of these type of calculations are the irreversibility calculations performed in the RFSIM model. These values, shown as Cs in Figure A.1, will always be outputs of the model, and therefore, do not need to be in the residual equations.

Although the Cs are outputs of the model, there situations where a governing equation will depend on one of their values. In this case, the Cs have to be calculated before the solution takes place in the subroutine "IC" located in the file CHECKMOD.f. If the Cs are functions only of parameters (Ks), then this one calculation will enough. However, it is possible for a C to be a function of parameters (Ks) and variables (Xs). Since the variables' values will change during the iterations, the Cs have to be re-calculated every iteration. This is done in the subroutine "BC" located in CHECKMOD.f.

Therefore, it is possible to have calculated values (Cs) that have to be calculated in all three subroutines (IC, BC, FC) of CHECKMOD.f. An example of this type of variable is the condenser internal volume. This volume is used in a number of residual equations, but it is a function only of the length and internal diameter of the condenser. Since either the length or diameter could be a variable (X), the volume is calculated before, during, and after the NR solution. In any case, the Cs will always be calculated from variables or parameters.

A.2 Overview of the "modes" of the model

A.2.1 Previous version of the RFSIM model

In the previous version of the model, there were two modes of operation: design and simulation (Goodson and Bullard, 1994). In design mode, the user was required to specify the superheat at the evaporator exit and the subcooling at the condenser exit. The capillary tube and total refrigerant charge in the system were assumed to be correct for that situation. In the simulation mode, on the other hand, the user did not have to specify the exits of the two heat exchangers because two extra constraints were added in the form of the capillary tube-suction line heat exchanger model and the charge conservation equations. Strictly speaking, the charge conservation equations were used in both modes, but the total charge was not specified (a K) in the design mode, but it was specified in the simulation mode.

This method of categorizing the model had two major shortcomings. First, the method for switching back and forth between the two modes was relatively confusing. Second, and perhaps more important, was the inherent lack of flexibility of the model. This was most apparent in the validation process reported by Goodson and Bullard (1994). The errors in the design mode of the model were very small, but the errors in the simulation mode were substantially higher. Unfortunately, since the capillary tube model and the charge conservation equation were the difference between the two modes, it was difficult to identify the cause of the additional error. It was impossible to determine whether the majority of the error was caused by the captube model or charge conservation equations. This problem was the major stimulus for the changes in the operational "modes" of the previous version of the model.

A.2.2 Current version of the RFSIM model

The current version of the RFSIM model will run in exactly the same way as the design and simulation modes of the old model. However, the model will run in number of other configurations as well. Although the names "design" and "simulation" can still used to describe the model when it is run as described in A.2.1, convenient names do not exist for all the new ways to run the model. Instead, the model operation is described in terms of whether the capillary tube model is used and total charge is specified. The design mode corresponds to the case when neither the capillary tube model or total charge are specified as parameters and the exits of the heat exchangers are specified. The simulation mode corresponds to the case when both are specified as parameters and the heat exchanger exits are not specified. In the new version, the capillary tube model and the total charge can be specified independently just as any other parameters in the XK file. For example, the model can be used with the capillary tube model and the superheat at the evaporator exit specified. Instead of trying to describe all the possible ways to run the model, a brief description will be given for three of the more important "modes" of model operation.

If the model is run without the capillary tube model or the total charge being specified (design mode), then the model will only give useful information about the compressor and the two heat exchangers. However, since the total charge will be calculated, the design mode is useful for examining the accuracy of the void fraction correlations that are critical to the charge equations. For example, if several data points are available that have superheated evaporator and subcooled condenser exits, then the design mode can be used to calculate the charge that should be in the refrigerator system. These calculated values can then be compared to the actual amount of refrigerant in the system, and some preliminary conclusions can be drawn about the accuracy of the void fraction correlation used in the simulation runs.

If the capillary tube model is used and the total charge is specified and no refrigerant information is specified (simulation mode), then the model can be used as a pure simulation tool. In other words, for a particular refrigerator described by a set of parameters, this mode of the model will solve for all of the system variables such as refrigerant states and heat transfers. This method of running the model is obviously very important because it can be used to evaluate the effects on the system performance of changing one or more of the parameters.

The third mode is actually a broad category that includes most of the new ways that the model can be run. In this mode, the capillary tube model is used, the total charge may or may not be specified, and at least one piece of information about the system is specified. This one piece of information may be the evaporator superheat, condenser subcooling, evaporation capacity, COP, or anything else that would be of interest to the user. Depending on the exact implementation, this method of operation can be used to solve for one or more parameters of the system. In effect, the model can be used as a true design tool to design one or more of the components in the refrigerator. For example, if the capillary tube model is used and the exits of both heat exchangers are specified, then the captube inlet length and the total charge could be determined. This model run would give the design of the capillary tube and the refrigerant charge required to achieve the desired heat exchanger exits. There are countless other variations of this method that could design part of the system in order to meet a certain evaporation capacity or COP requirement.

All of the operational modes of the model depend on whether the capillary tube-suction line heat exchanger model and the total charge are specified. Therefore, some of the details of what happens in the system model when each of these two items are specified and when they are not specified will be discussed.

A.2.2.1 Capillary tube-suction line heat exchanger specification

The capillary tube-suction line heat exchanger model calculates the mass flow rate through the capillary tube and the temperature rise in the suction line. When the capillary tube model is not used, the component is described only by an effectiveness equation that describes the heat transfer from the capillary tube to the suction line. The inlet enthalpy to the evaporator is calculated in both cases. Therefore, the major difference between using the capillary tube model and not using the model is the calculation of the mass flow rate through the capillary tube.

Since there is no mass accumulation in a steady-state situation, the mass flow rate through the capillary tube must equal the compressor mass flow rate. Therefore, when the capillary tube model is used, there is another calculation of mass flow rate that is set equal the compressor mass flow rate. This extra calculation provides the additional constraint on the system that allows some parameter (K) like superheat or subcooling to become a variable (X).

In addition, the temperature gain in the suction predicted by the capillary tube model is usually more accurate than the simple effectiveness equation. The prediction of inlet evaporator enthalpy is also more accurate when using the capillary tube model. This is true because the effectiveness method uses an energy balance based upon the predicted heat transfer, while the capillary tube model actually solves for the exit state of the capillary tube. The enthalpy at this exit state is equal to the enthalpy at the evaporator inlet due to the isenthalpic expansion from the choked-flow capillary tube exit. Since the capillary tube model involves a finite-difference solution, it has the added ability of providing detailed refrigerant information along the capillary tube and suction line.

When the capillary tube model is used, the effectiveness of the capillary tube-suction line heat exchanger is calculated after the NR solution is complete as a calculated value (C). If the RFSIM model is then run without the capillary tube model, this C will automatically become a K and its value will be used to calculate the heat transfer to the suction line. By providing accurate values of the effectiveness on the previous solution, the capillary tube model can make the RFSIM model more accurate even when it is not being used. In much the same way, the RFSIM model, when not using the capillary tube, will provide reasonable estimates for the initial guesses for variables required by the capillary tube model. Then, if the RFSIM model is run with the capillary tube model, there is a better chance that it will converge.

A.2.2.2 Total charge specification

One of the residual, or governing, equations in the RFSIM model adds up the mass of refrigerant in all of the components and sets it equal to the total charge in the system. The calculation of refrigerant mass in the single-phase components is fairly straightforward if the volumes are known. However, there is significant uncertainty in the calculation of the two-phase refrigerant mass in the heat exchangers. This uncertainty is primarily due to the fact that the void fraction (vapor cross-sectional area divided by the total cross-sectional area) can be different at the same quality within the heat exchanger depending on the assumptions made. These assumptions are quantified in the form of a void fraction correlation and can be used to calculate the average void fraction in a two-phase region of a heat exchanger. The average void fraction can then be used to calculate the mass in the two-phase region.

If the total charge is a variable (X), then the charge conservation equation simply calculates the total charge in the system. Therefore, if there is any error in the volume of the components or in the void fraction correlation, the total charge value will be erroneous, but there will be no other effects on the RFSIM model since the total charge variable is not used in any other equation.

However, if the total charge is specified as a parameter (K), then the individual contributions to the total charge will be constrained. Since the mass in the heat exchangers is relatively large compared to the single-phase components, any errors in the volumes or the void fraction correlations will cause errors in the heat exchangers' refrigerant mass. These errors cause the refrigerant states within the heat exchangers to be incorrectly predicted (primarily the condenser exit and the evaporator inlet). The errors in the refrigerant states within the heat exchangers then affect everything else in the RFSIM model. Therefore, it is very important to know if the void fraction correlation being used works well. If the charge equations do not predict the total mass well when superheat and subcooling are specified, as described in A.2.2, then the RFSIM model will probably not work well when the total charge is specified.

A.2.3 How to use the various modes of operation in the RFSIM model

As indicated in A.2.1, switching between modes of operation in the current version of the RFSIM model is easier than it was in the old version. The capillary tube-suction line heat exchanger model can be "turned on" or "turned off" simply by changing the XK flag of the variable or parameter named "CaptubeModel". As shown below in Figure A.3, the value of this variable will always be equal to 1.0 regardless of whether the capillary tube model is used or not.

```
1  K  CaptubeModel   = XK(118) =              1.0 [ ]               1
1  K  CaptubeOutput  = XK(119) =              0.0 [ ]               1
```

Figure A.3 Capillary tube model variables and parameters

To start using the capillary tube model, the XK flag of "CaptubeModel" has to be set to K, and some other parameter in the model has to be made a variable (X). As mentioned in A.2.2.1, the RFSIM model, when not using the capillary tube model, provides reasonable estimates for five variables that are used by the captube model subroutine: "pcrit", "xcrit", "DPout", "DTsl", and "DPin". The first two variables are the pressure and quality at the capillary tube exit, and the last three variables are the pressure and temperature steps used to discretize the capillary tube in the finite difference solution.

Sometimes the estimates provided by the RFSIM are not good enough, and the RFSIM model will not converge when the capillary tube is first "turned on". In this case, the user can either adjust the initial guesses or start with another solution XK file that used the capillary tube model and change the necessary parameters. Since it takes some time to get a feel for the necessary initial guesses, it is recommended that at least one solution XK file be kept for the purpose of providing a starting point for future model runs that use the capillary tube model. To stop using the capillary tube model, the XK flag for CaptubeModel has to be set to an X, and some other variable (X) has to made a parameter (K). There are rarely any problems encountered when a solution that used the capillary tube model is used as the starting point for model run without the capillary tube model.

The second line of Figure A.3 contains the parameter "CaptubeOutput". This parameter was also shown in Figure A.2 as one of the parameters that cannot become a variable (X). The value of this parameter dictates how much output from the capillary tube-suction line heat exchanger model is desired while running the RFSIM model. If CaptubeOutput equals 0.0, then there will be no output other than a printed statement that the capillary tube model subroutine has been called by the overall system model. If CaptubeOutput equals 1.0, then the property profile in the capillary tube and suction line will printed to the screen at the end of the solution. The property profile will look something like Figure A.4 shown below.

```
 #  C-T Temp  Wal Temp  S-L Temp  C-T Pres  Enthalpy   Quality  SubCol  Length
 4     63.26      0.00      0.00     87.28    31.234   0.00000    7.67   0.000
 5     63.26      0.00      0.00     83.17    31.234   0.00000    4.84   1.249
 6     63.26      0.00      0.00     79.05    31.234   0.00000    1.90   2.499
 7     63.26      0.00      0.00     76.48    31.234   0.00000    0.00   3.281
 8     62.11      0.00      0.00     74.94    31.233   0.00483    0.00   3.668
 9     58.92     57.03     45.02     70.82    31.231   0.01794    0.00   4.453
10     53.52     51.25     36.70     64.24    29.573   0.01907    0.00   5.614
11     48.29     45.63     28.37     58.32    27.921   0.01938    0.00   6.601
12     43.21     40.13     20.05     52.97    26.276   0.01905    0.00   7.457
13     38.28     34.78     11.72     48.15    24.647   0.01818    0.00   8.206
14     33.50     29.58      3.40     43.81    23.025   0.01668    0.00   8.875
15     29.54     25.11     -4.93     40.45    21.563   0.01398    0.00   9.417
16     25.07      0.00      0.00     36.89    21.554   0.02996    0.00   9.719
17     20.24      0.00      0.00     33.33    21.538   0.04663    0.00   9.902
18     15.00      0.00      0.00     29.78    21.507   0.06410    0.00  10.010
19      9.25      0.00      0.00     26.22    21.451   0.08244    0.00  10.067
20      2.84      0.00      0.00     22.66    21.346   0.10173    0.00  10.083
```

Figure A.4 Capillary tube property profile

With the exception of "Wal Temp" and "S-L Temp", all of the column headings refer to the capillary tube. These two columns refer to the suction line, and therefore, only have non-zero data in the heat exchanger portion of the capillary tube. If CaptubeOutput equals 2.0, then the above property profile will be printed at the end of every iteration.

The specification or non-specification of the total charge is also straightforward. To require the mass of refrigerant in the system add up to a specified amount, then the XK flag of "Mtotal" should be set to a K, and some other parameter (K) must be made a variable (X). Whenever the XK flag of Mtotal is changed from an X to a K, the user should not change its value at the same time because this can prevent the model from converging. For example, a case may exist where the RFSIM model will predict a refrigerant mass of 0.6 lbm, but the actual mass may be 0.5 lbm. If the user simultaneously changes Mtotal from an X to a K and its value from 0.6 lbm to 0.5 lbm, the RFSIM model will probably crash or at least not converge. If the user really wants to see what happens when the total charge is constrained to 0.5 lbm, he/she should first let the model solve at 0.6 lbm and then change its value gradually down to 0.5 lbm through the use of the MULTIPLE run or several SINGLE runs.

To remove the refrigerant charge constraint from the system, the XK flag of Mtotal should be set to an X, and some other variable (X) must become a parameter (K). As with the use of the capillary tube model, there are no problems associated with the changing of Mtotal from a parameter (K) to a variable (X). In the new solution, the value of Mtotal will simply be calculated based upon all of the parameter values.

A.3 Important options within the model

Since there are many governing equations (112 at the time of writing), numerous subroutines that are called in the governing equations, and many parameters in the XK file, the RFSIM model can appear very intimidating to new users. To help new users become familiar with the RFSIM model, some of the more important issues and options will be discussed. Since someone modifying the model will encounter most of these same issues and options, this section will also serve as a "checklist" for reconfiguring the model for a new refrigerator.

A.3.1 Refrigerant type

The most important option within the RFSIM model is the choice of refrigerant to be used in the simulation. Each time the model is run, a statement will be printed to the screen, like the one below, that declares the refrigerant selected for all the refrigerant-dependent functions within the model.

```
*********************************************************************
Refrigerant selected ==>  R12
See REFRIG.INC for refrigerant code or to change refrigerant type.
*********************************************************************
```

In this case, the refrigerant being used is R-12. If this is not the refrigerant that is desired, it is a simple matter to change refrigerants if the RFSIM model has the capability. Since thermodynamic properties and transport properties are required to model a refrigerant within the RFSIM model, there are currently only two refrigerants to chose from: R-12 and R-134a. However, the capability to model any refrigerant can be added to the RFSIM model. For details on how to switch or add refrigerants, see the Appendix D.

A.3.2 Refrigerant mass calculations

There are two important issues involving the calculation of refrigerant charge in the system. First, There are three residual equations that calculate the mass in single-phase components which may need to be modified for different refrigerators. Second, the two equations that calculate the mass in the heat exchangers are very dependent on the void fraction correlation that is used within the subroutines. Both of these issues will be discuss in the following sections.

A.3.2.1 Refrigerator-dependent single-phase components

Two of the three single-phase equations calculate the refrigerant in the compressor, and the third equation calculates the charge in the accumulator. The two compressor equations may need to be changed because of the differences between low- and high-side sump compressors or because of the oil-refrigerant combination used in the model. The first of these compressor equations is shown below.

```
C***** The compressor sump (vapor) refrigerant mass *********************

c       ** The two volumes should be consistent with the location of the
c          compressor sump. (i.e. v11 = low side, v0 = high side )

1100    R(mass+0) = MCompvap - MassSingle(volcomp,v0,v0)
         GOTO 1
```

As the comment statements indicate, the volume arguments in the "MassSingle" function must be consistent with the location of the compressor sump. In this case, the volumes are equal to v0 because a high-side sump is being modeled. If a low-side sump is being modeled, the volumes would be set to v11.

The second compressor equation calculates the amount of refrigerant dissolved in the compressor oil. To perform this calculation, the RFSIM model uses the refrigerant-oil solubility equations developed at the ACRC by Grebner and Crawford (1992). These equations depend on seven experimentally determined constants that are included in the XK file and are shown below.

```
0  K  K1oil         = XK(141) =        -0.0059927652 [ ]                   10
0  K  K2oil         = XK(142) =         0.0416615100 [ ]                   10
0  K  K3oil         = XK(143) =         0.0020046597 [ ]                   10
0  K  K4oil         = XK(144) =        -0.0032682848 [ ]                   10
0  K  K5oil         = XK(145) =         0.0017368443 [ ]                   10
0  K  K6oil         = XK(146) =        -0.0002855223 [ ]                   10
0  K  K7oil         = XK(147) =         0.0000160929 [ ]                   10
```

These seven constants are specific to a particular refrigerant-oil combination. Therefore, if the refrigerant or oil is changed, new constants need to be placed in the XK file. For more details about the refrigerant-oil combinations available, see Grebner and Crawford (1992). If the particular combination is not available, then the “closest” combination should be used or additional constants should be developed from other data.

Although there are actually two equations that implement the refrigerant-oil solubility calculations, only one of the two equations might require modification. This equation relates the liquid mass fraction of the refrigerant in the oil (Woil) to the seven parameters shown above and the compressor operating conditions. The equation is shown below as it appears in the EQNS.f file.

```
C**** The mass of refrigerant dissolved in the oil (high-side sump)  ***

c       ** Two of the variables are dependent upon whether the
c          compressor is a low- or high-side sump:
c          The pressure in the "Tstar" equation (p11=low,p0=high)
c          The saturation temperature in the residual equation
c               (Tsat11=low, Tsat0=high)
```

```
1280    Aoil = K1oil + (K2oil/Woil**(1.0/2.0))
        Boil = K3oil + (K4oil/Woil**(1.0/2.0)) + (K5oil/Woil) +
     &            (K6oil/Woil**(3.0/2.0)) + (K7oil/Woil**2.0)
        Tstar = (1 - Woil)*(Aoil + Boil*p0)

        R(mass+9) = ((t0+460) - (Tsat0+460))/(Tsat0+460) - Tstar
         GOTO 1
```

The three variables "Aoil", "Boil", and "Tstar" are examples of the non-residual variables that were discussed in A.1.1. Tstar is a function of the pressure of the refrigerant-oil mixture shown in boldface in the above equations. In this case, the compressor is a rotary compressor with a high-side sump. Therefore, the refrigerant-oil mixture experiences the high discharge (p0) pressure of the compressor. If this were a low-side sump compressor, the pressure would be equal to inlet compressor pressure (p11). Similarly, the saturation temperature at the sump pressure (Tsat0) is also set for a high-side sump and would be changed to Tsat11 if a low-side sump is being used.

The third equation that may have to be changed calculates the charge in the accumulator. If a refrigerator has an accumulator, it will be located in the suction line either before or after the capillary tube-suction line heat exchanger. Since the suction gas temperature changes across the suction line heat exchanger, the specific volumes for the two possible accumulator locations are different. Therefore, the volume argument in this equation should reflect the correct location as shown below.

```
C***** The accumulator refrigerant mass ********************************

c       ** The two volumes should be consistent with the location of the
c          accumulator. (i.e. v9 = before suction line hx, v11 = after
c          suction line hx)  **

1240    R(mass+7) = MAccum - MassSingle(volaccum,v11,v11)
         GOTO 1
```

In this case, the accumulator is located downstream of the suction line heat exchanger, and thus, the volume at that location (v11) is used within the function call. As the comment statements indicate, the volumes would be set to v9 if the accumulator was located before the suction line heat exchanger.

A.3.2.2 Void fraction correlations

The importance of the void fraction correlations was already discussed in A.2.2.2. Several correlations were examined by running the RFSIM model with known values of evaporator superheat and condenser subcooling. The known data values were from the top-mount Amana and the Whirlpool refrigerators. The results of this analysis indicated that Premoli's correlation is the best void fraction correlation for the operating conditions in the evaporators and condensers of domestic refrigerators. As a result, Premoli's void fraction correlation is currently used in the RFSIM model.

The routines that use the void fraction correlations have been written in a manner that would allow the easy switching of correlations. If necessary, the user may switch to another void fraction correlation by editing the file EQNS.f and changing one of the arguments in the function calls to "MassCond" and "MassEvap". As an example, the function call to MassEvap is shown below.

```
MEvap = MassEvap(Volevap,fsupevap,f2phevap,v70,v71,v9,xie,xoe,
                t7,t71,w,Devap,30,3)
```

The last argument in the function call, "3" in this case, is the parameter that determines which void fraction correlation is used to calculated the two-phase mass in the evaporator. The legend showing which correlation corresponds to which number can be found in the comment statements of the MassEvap and MassCond functions located in EQNSUBS.f. Since there are two functions, the void fraction correlations can be specified separately for the evaporator and condenser. The choice of correlations was purposefully "hidden" from the user and not placed in the XK file for two reasons: the void fraction will probably not have to be changed too often; and to keep the number of correlation flags in the XK file to a minimum.

A.3.3 Compressor issues

In addition to refrigerant charge calculations already discussed, there are several issues dealing with the compressor modeling within the RFSIM model. Probably the most important option is the choice of compressor maps. These maps are nine-parameter curve fits that give the mass flow through the compressor and the power consumed by the compressor as functions of the evaporating and condensing temperatures corresponding to the pressures seen by the compressor. There are several compressor maps currently available in the RFSIM model. For the top-mount Amana and Whirlpool refrigerators, there is a map made just from the compressor manufacturer's data and there is another map made with the manufacturer's data and calorimetry data from the refrigerators taken at the ACRC. There is also a single compressor map available for the new Amana side-by-side refrigerator. The user can select a particular compressor map just by changing the value of the parameter "CompNum" in the XK file according to the following legend:

1	Top-mount Amana (manufacturer's data)
2	Modified top-mount Amana (manufacturer's data and calorimetry data)
3	Whirlpool (manufacturer's data)
4	Modified Whirlpool (manufacturer's data and calorimetry data)
5	Side-by-Side Amana (Manufacturer's data)

If a new compressor map needs to be added to the list, this can be done by modifying the functions "wf" and "Pcompf" located in EQNSUBS.f. The new coefficients for the mass flow and power map need to be added to the "IF-ELSE IF-END IF" structure in each function and a new CompNum value needs to be assigned to the coefficients.

To assist in the modeling of variable-speed compressors, two parameters have been added to the RFSIM model: beta_wmap and beta_Pmap. These two parameters are factors that are multiplied by the mass flow and power values returned by the compressor maps, respectively. These factors can be used to simulate the effect of changing the speed of the compressor if the user knows how the mass flow rate and power change with respect to the speed. If they are both equal to one, then the compressor maps are not adjusted at all.

It should be noted that the adjusted values (mass flow and power) will not always be equal to the original values multiplied by the "beta" factors. This is because the "beta" factors will change the mass flow and power, which will change the evaporating and condensing temperatures that the compressor maps are dependent upon. In other words, the ratio of the new to old values will not necessarily equal the "beta" ratios because the evaporating and condensing temperatures at the new condition can be different than they were at the old condition.

Another issue dealing with the compressor modeling is the curve fit that relates the compressor shell temperature to the discharge (refrigerant at the compressor exit) temperature. This curve fit is needed for the "hA ΔT" equation that calculates the heat transfer from the compressor. The ΔT is the temperature difference between the shell temperature and the temperature of the air stream flowing over the compressor. The two equations (relating shell to discharge temperatures) that have been developed for the top-mount Amana and Whirlpool refrigerators are very similar, and therefore, it would seem that either curve fit could be used for any refrigerator if nothing else is available. The curve fit from the Amana might work better with reciprocating compressors and the expression from the Whirlpool might work better with rotary compressors. If a new curve fit is available, it can be inserted in the governing equations located in EQNS.f. The hA of the compressor shell in the above equation is also needed, but it will be discussed in a following section.

A.3.4 Model parameters

Although the RFSIM model is general enough to be used for most domestic refrigerators, the parameters within the XK file define the particular refrigerator that is actually being modeled. Therefore, it is important for the user to be aware of the parameters that are unique to a given refrigerator simulation. All of the constants that define a refrigerator are stored as parameters (Ks) in the XK file. Most of the parameters in the XK file can be place into one of three groups based upon their function within the RFSIM model. However, some of the parameters could legitimately be placed in more than one of the groups depending on how the model is being used.

The purpose of this section is not to give a definition of all the parameters in the model, but rather, to give users a better understanding of the general nature of the parameters required to run the model.

A.3.4.1 Directly measurable parameters

This group includes all the parameters that define the directly measurable characteristics of the model refrigerator. If the purpose is to simulate an existing refrigerator, the majority of these parameters can be easily measured and placed into the XK file. However, if the model is being used as design tool for a new refrigerator, then some of these parameters will instead be specified by the user. Most of these parameters involve the physical dimensions of the refrigerant system. Roughly speaking, this group includes the following parameters: evaporator and condenser fan powers; internal volumes of the compressor, filter drier, and accumulator; lengths and diameters of all the tubing in the refrigerant system; and several details about the geometry of the condenser and evaporator such as the number of return bends, number of equivalent circuits, etc. For a comprehensive list of all the measurable parameters in the system, the reader is referred to the Appendix G.

A.3.4.2 Empirically determined parameters

These parameters also define the model refrigerator, but they are usually determined from some kind of experiment or empirical correlations. As with most thermal models, some of the most difficult parameters to obtain for the RFSIM model are those dealing with the heat transfer in the system. There are nine such parameters that fall into this category:

UAf	Overall heat transfer conductance of the fresh-food section
UAz	Overall heat transfer conductance of the freezer section
haircond	Air-side convection coefficient of the condenser
hairevap	Air-side convection coefficient of the evaporator
hAcomp	Overall heat transfer conductance (compressor shell to the air stream)
vdotcond	Volumetric air flow rate over the condenser
vdotevap	Volumetric air flow rate over the evaporator
frecirc	Air recirculation fraction from the condenser exit to the condenser inlet

The values of these numbers could come from experimentation, correlations, manufacturer's estimates, or previous experience with refrigerators. Obviously, it would be desirable to have accurate values for all these parameters, but if some are not available, then the user's best estimates will have to suffice.

Currently, the user can specify a value for haircond and hairevap or can use a particular air-side convection correlation instead. This choice is made through the use of the hcondNum and hevapNum parameters in the XK file. If the value for either of these two flags is set to “0”, then the RFSIM model will use the values given in the XK file for haircond or hairevap. If either flag value is greater than “0”, then a particular air-side heat transfer correlation will be used and the value of haircond and hairevap will be calculated and marked as a “C” in the XK file.

Currently, there are only two correlations available in the RFSIM model. The first, hcondNum and hevapNum equal to “1”, is a curve fit of the air-side convection coefficient for the Amana top-mount refrigerator verses air velocity (Cavallaro and Bullard, 1995). It was found that the air-side convection coefficient could be correlated to the air velocity according to Equation A.1 shown below.

$$h_{air} = A \bullet V^B \qquad \text{(A.1)}$$

Where V is the velocity of the air [ft/sec], and A and B are experimentally determined constants. In the RFSIM model, the velocity of the air is determined from the above volumetric flow rates and the heat exchanger’s frontal flow area. The second correlation, hcondNum and hevapNum equal to “2”, is a curve fit of the same form for the Whirlpool refrigerator. However, the correlation was generated by assuming that the B constant in Equation A.1 would be the same for the Whirlpool as it was for the Amana. The A constant was then determined from the one data value of air-side convection coefficient that was estimated from data.

Since the two correlations were developed for specific refrigerators, it is not recommended that they be used for other refrigerators unless a new value of the coefficient A is determined. This could be done as it was for the Whirlpool if at least one value of the air-side convection coefficient at a known volumetric flow rate is available. The user can add this type of correlation, or any other, by modifying the functions “haircnd” and “hairevp” in the file EQNSUBS.f. The new correlations will have to be assigned a new number and then placed within the “IF-ELSE IF- END IF” structure in both functions. The list of arguments passed to the functions will have to modified if the new correlations are dependent on more than volumetric flow rate and frontal flow area.

A.3.4.3 Operational parameters

For a given refrigerator, there can be a large number of operating conditions that the user wishes to simulate. Strictly speaking, the “load” imposed on the refrigerator is defined by the following five parameters:

tamb	Ambient temperature
tafrez	Average air temperature in the freezer compartment

tafrig	Average air temperature in the fresh-food compartment
FrezHeater	Heater (or other load) in the freezer compartment
FrigHeater	Heater (or other load) in the fresh-food compartment

The ambient temperature, plus any correction due to the recirculation fraction, sets the temperature of the air entering the condenser, and therefore, it affects the condensing temperature. Together with the two compartment temperatures, the ambient temperature also determines the heat transfer through the walls of the refrigerator. The two compartment temperatures determine the inlet air temperature to the evaporator, and thus, they affect the evaporating temperature.

The last two parameters, FrezHeater and FrigHeater, are used primarily when the model's ability to match steady-state data is tested. Experimentally, heat is added to both compartments to maintain the steady-state conditions, and therefore, heat has to be added to the steady-state model as well. These parameters can also be used to simulate the effect of any additional heat input that may be in either compartment.

There are four other parameters that could be added to this group since they also define the operating condition: beta_wmap, beta_Pmap, beta_condfan and beta_evapfan. The first two parameters are related to the speed of the compressor and have already been discussed. Beta_condfan and beta_evapfan are the ratios of the new fan speeds to their nominal values for the condenser and evaporator, respectively. These ratios are used to determine the new values of the volumetric flow rates through the fans and the power consumed by the fans. According to the fan laws, the ratios of the new to nominal volumetric flow rates will be equal to the "beta" factors. Also according to the fan laws, the ratios of the new to nominal shaft powers will be equal to the "beta" factors raised to the 3rd power.

As with the compressor "beta" parameters, these fan "betas" have been added to the model to assist with the simulation of variable speed systems. The difference between the compressor's and the fans' "betas" is that the nominal values are calculated for the compressor, but nominal values have to be given for the volumetric flow rates and fan powers. If the fan "betas" are set to equal to one, then the RFSIM model will just use the nominal values given for the volumetric flow rates and powers.

A.3.5 Heat exchanger geometry

Probably the most difficult issues within the model to describe are the heat exchanger geometries. It is difficult because there are a number of modeling geometries that can occur in domestic refrigerators. Since the heat transfer of each "zone" of the heat exchangers are modeled with the effectiveness-NTU equations, the following information is needed for each zone: the inlet air temperature, the inlet refrigerant temperature, and the configuration of the particular zone (i.e. parallel flow, counterflow, etc.). Although every evaporator will be modeled with two zones, for example, the configuration of the zones and the calculation of the inlet refrigerant and inlet air temperature for both zones may be different for each case.

In the evaporator of the top-mount Amana, the two-phase and superheated zones were both in a counterflow configuration with the air stream (see Figure D.3 in Goodson and Bullard, 1994). In this case, the air first flowed over the superheated zone and then over the two-phase zone. Therefore, the air temperature at the inlet of the two-phase zone could be calculated directly by an air-side energy balance across the superheated zone.

In the Whirlpool, however, half of the evaporator is in a counterflow configuration and the other half is in a parallel flow configuration (Appendix B). Therefore, some of the two-phase zone is in a counterflow configuration, and the other part of the two-phase zone and the superheated zone is in a parallel flow configuration. Although the effectiveness of a two-phase zone is independent of the flow configuration, the effectiveness of the superheated zone does depend on the configuration. The heat transfer in the superheated zone also depends on the inlet air temperature which has to be calculated from an energy balance on the parallel-flow portion of the two-phase zone. Therefore, if the Amana evaporator equations had been used to model the Whirlpool evaporator, there would be some error in the results.

The preceding example was used to illustrate the potential problems in the heat exchanger modeling within the RFSIM model. If the user wants to model a refrigerator with a different evaporator or condenser geometry than currently used, he/she may have to modify the governing equations, add or delete governing equations, or just use the existing configuration and live with the errors. Since the single phase zones account for a small fraction of the heat transfer in the two heat exchangers, the error of an incorrect configuration may be negligible. If the governing equations are to be modified, then the user should use the existing FORTRAN code and the information in Appendix B as examples of how the heat transfer is modeled.

A.4 Running the RFSIM model

Although much of the following information is covered within the Appendix A of Mullen and Bullard (1994), it will be repeated here in the context of the RFSIM model. The intent of this section is cover, with some detail, things that are unique to the operation of the RFSIM model while referring the reader to Appendix A of Mullen and Bullard (1994) for the information that is common to any model solved by the ACRC solver.

A.4.1 Compiling the FORTRAN code

The first step in running the RFSIM model is compile all the FORTRAN files that make up the model and the solver. Since there are several files containing source code and "Include" files, it is recommended that the model be compiled through the use of a "make" file. Although the exact form of this file will vary depending on the platform used to run the model, the basic idea to link together all the source code and create an executable program. The benefit of using a "make" file is that the entire model FORTRAN code will not have to be recompiled every time a change is made to one of the subroutines or functions. Only the file containing that routine will have to be recompiled. However, if the user is unable to compile the program in this way, he/she can always merge all of the FORTRAN source code (*.f files) together into one file and then compile the file using the particular command of the FORTRAN compiler in use.

A.4.2 Initializing the RFSIM model

Once the model is compiled, the next step is to make sure that the desired values of the parameters and initial guesses for the variables are in the XK file named "XK" in the model directory. As mentioned in A.1, this "XK" file serves as the input for the model. The next step is to make sure that the solver options and the type of model run desired are correct as listed in the files "SLVERSET" and "INSTR", respectively. The file SLVERSET contains the settings for various Newton-Raphson parameters such as the convergence criteria and number of iterations, and it also contains information specifying the type of model output. The INSTR file tells the solver whether it is to perform a "SINGLE", "MULTIPLE", "SENSITIVITY", or "UNCERTAINTY" analysis. The details about the SLVERSET file and the analyses options in the INSTR file can be found in Appendix A of Mullen and Bullard (1994).

A.4.3 Starting the model solution

When the above steps are completed, the RFSIM model is ready to be run. Currently in the Convex C240 (UXH) environment, this is done by typing "RFSIM" at the prompt. When the program is executed, the following text will be printed to the screen:

```
Single Run using extender: out

Rebuild the NonZeroList?
 "y" if residual equations have been modified or if parameters and/or
  variables have been swapped (y/n)?
```

The concept of the NonZeroList has already been discussed in A.1.1 and in Appendix A of Mullen and Bullard (1994). Basically, the user should type "y" if an X and a K have been swapped since the last model run or if the governing equations have been modified by changing the number of variables (Xs). If the user is not sure, then he/she should rebuild the NonZeroList. It takes a little more time to run the model when the NonZeroList is rebuilt, but it is better to be safe than sorry.

A.4.4 Checking routines

A.4.4.1 Initial checking

Whether the user types "y" or "n" in response the above question, the model will perform an initial check of the model run defined by the XK file to set the values of several important logical flags. When it does so, it will print messages to the screen similar to the following:

```
The evaporator has a superheated exit.
The condenser has a two-phase exit.
The captube-suction line hx model IS NOT being used.
```

These three statements let the user know the exit conditions of the evaporator and condenser and whether the capillary tube-suction line heat exchanger model is being used. The checking of the heat exchanger exit conditions is done in the subroutine "IC" located in the CHECKMOD.f file. This subroutine not only prints statements to the screen like the ones above, but it also set the values of three logical flags (Evap2phX, Cond2phX, and CTSLHXSIM) that are used in various parts of the model. The above case would correspond to the following flag values: Evap2phX = "false", Cond2phX = "true", and CTSLHXSIM = "false". Since the possible scenarios for exit conditions of the evaporator and condenser are identical, they will be presented here for the evaporator only.

In the evaporator and condenser, only one of the two variables (i.e. superheat or exit quality) may be specified as a known value. Although this may seem counter-intuitive since a quality exit dictates a superheat of zero, this is the way that it has to be done in the RFSIM model. The above statement for the evaporator indicates that the user specified the variable "superheat" as a K and gave it a value greater than zero. If the superheat was a K and the value was equal to or less than zero then the following warning statement would appear instead of the one above:

```
The superheat must be greater than zero !
```

In this case, the flag Evap2phX would still be "false" and superheat would be given a value of 1.0 °F so the model would not crash. Likewise, the statement for the evaporator exit would be one of the following if the quality at the exit of the evaporator, "xoe", was a K:

```
The evaporator has a two-phase exit.
```

```
or
```

```
The xoe must be between 0.0 and 1.0 !
```

In the second case, the value of xoe was outside of its allowable range, so the model reassigned it a value of 1.0. In either case, the flag Evap2phX set to a value of "true". If xoe and superheat were both flagged as Ks, then the model would print the following warning to the screen:

```
xoe and superheat cannot both be known! Check XK file!
```

In this case, the user should stop the program and modify the XK file since the program will probably crash anyway. The only other possibilities for the evaporator exit are when both xoe and superheat are variables (Xs). In this case, the IC subroutine will determine whether the exit is two-phase or superheated based upon the values of the two variables. If superheat was greater than zero and xoe was equal to 1.0, the following statement would appear on the screen:

```
Initially, the evaporator has an superheated exit.
```

Notice the word "Initially" at the front of the statement. This indicates the superheated exit is not fixed and can change to a two-phase exit during the simulation as will be described later. However, the flag Evap2phX will be given a value of "false" as before. Likewise, if the value of xoe is in between 0.0 and 1.0 and superheat is equal to 0.0, then the following statement will appear on the screen:

```
Initially, the evaporator has an two-phase exit.
```

As with the previous case, this indicates that the two-phase exit is not fixed and can change to a superheated exit during the simulation. To start with, the flag Evap2phX is given a value of "true". If the values of superheat and xoe do not fall into one of the two scenarios just described, then the IC subroutine will print the following warning:

```
The guesses for superheat and xoe were inconsistent!
```

This simply means that the subroutine cannot tell whether the user intended that the initial exit condition of the evaporator be two-phase or superheated. An example of this would be when the initial values were 0.98 for xoe and 2.0 for superheat. One of these guesses would indicate a two-phase exit, but the other guess would indicate a superheated exit. As a default, the subroutine will change the value of xoe to 1.0 and the value of superheat to 0.0, and the flag Evap2phX will be given a "true" value. It really does not matter whether the subroutine assumed that the initial exit was two-phase or superheated since the actual exit condition will be determined during the solution.

A.4.4.2 Boundary checking

Boundary checking is done after every iteration of the RFSIM model. The boundary checking only has meaning if the state of one or both of the heat exchanger exits is unknown. As before, only the evaporator exit will be discussed here because the situation at the condenser exit is identical.

In the Newton-Raphson method, the variables are changed each iteration in a way that makes the entire variable set closer to satisfying the governing equations. During one iteration, it is possible that one or more of the variables associated with the exit of the evaporator may be changed to a value that is physically impossible because the solution "wants" to go to the other exit condition. For example, if the evaporator initially has a two-phase exit, and the value of xoe is changed to a number greater 1.0, this indicates that the exit is probably superheated. Actually, this example is one of the scenarios that happens quite often when running the model. When it does happen, the following message will be printed to the screen:

```
Two-phase evaporator with xoe > 1
#Switching to superheated evaporator
```

When the evaporator exit is switched from two-phase to superheated, the following events occur within the model: Evap2phX is given a value of "false"; superheat is given a value of 1 °F; xoe is given a value of 1.0; and several other variables associated with the superheated zone of the evaporator are given reasonable initial guess values.

If the evaporator initially has a superheated exit, there are at least two different indications that the exit is really two-phase: the value of superheat is changed to a number less than or equal to 0.0; the value of the superheated faction of the evaporator, "fsupevap", is changed to number less than or equal to 0.0. Although these two "warnings" usually occur together, they are both included in the boundary checking routine. Depending on which one of these events occur, the subroutine will print one of the following messages to the screen:

```
Superheated with superheat <= 0.0
#Switching to two-phase evaporator
```

or

```
Superheated with fsupevap <= 0.0
#Switching to two-phase evaporator
```

Regardless of which warning triggers the switch, the same events will occur within the model when it happens: Evap2phX is given a value of "true"; xoe is given a value of 0.99; superheat is given a value of 0.0; fsupevap is given a value of 0.0; and the two-phase fraction of the evaporator, "f2phevap", is given a value of 1.0.

On occasion, the one of the heat exchanger exits will switch back and forth from two-phase to a single-phase. Sometimes the model eventually settles on a two-phase or single-phase exit, but sometimes it does not. It will keep switching exits until the number of iterations specified in SLVERSET has expired. Although the switching problem seems to occur when the actual exit should be very close to saturation, it is not known how it can be fixed. Several attempts to solve the problem were unsuccessful. The only advice that can be given if this occurs during the simulation is to slightly change one or more of the parameter values. This seems to be enough to get the model away from whatever problem causes the hang-up.

A.4.4.3 Final checking

The only checking done in the RFSIM model after the solution is complete is on the saturation temperatures used in the compressor maps. The curve fits that describe the mass flow rate and power consumption of the compressor are functions of the evaporating and condensing temperatures. Since these curve fits were made with data that has a fairly limited range, it is possible that one or both of the temperatures may be out of the data range, and therefore, the compressor map values will be extrapolations. If this occurs, one or both of the following warnings will be printed to the screen:

```
The condensing temperature (tsat0) is outside map.
```

or

```
The evaporating temperature (tsat11) is outside map.
```

If either of the temperatures is out of range of the map data, nothing happens other than the printed warning. Since it is not really known how much error is involved in the extrapolations, it is up to the user to decide if the results from such a simulation are to be considered valid.

A.5 Helpful hints for using the model

As with any program, there are some aspects of the RFSIM model that cannot be learned without running the model. The purpose of this section is pass on some of that experience that the author has acquired through the extensive use and modification of the model. As with some of the other sections in this document, some of the information presented here has already been covered elsewhere. The information about the use and modification of the model will be presented in numbered format to save space and time.

A.5.1 Using the model

1. Once a solution is obtained as an “XK.out” file, it should be save somewhere else in the directory under a name other than “XK” or “XK.out”. If several previously obtained, unaltered solutions exist, the user will have a much easier time of obtaining new solutions.
2. To obtain a solution that is far away from an existing solution in “parameter space”, two things can be done:

a) Manually increment the parameters, solve the model, and then copy the "XK.out" solution onto the "XK" input file for the next run. Repeat this process until the final solution is reached.

b) Use the MULTIPLE run analysis described in Appendix A of Mullen and Bullard (1994). This is the better option since the model can be used to automatically increment the parameters and march towards the final solution.

3. The model is very sensitive to changes in certain parameters like "Mtotal" and "Dct". If these parameters have to be changed over a wide range, then the MULTIPLE run analysis with several intermediate steps is the best way to do it.

4. There are times when the model is not able to solve a particular point (a solution for a given set of parameters). There are a couple of recommendations for these problems depending on the situation:

   a) If the problem point is a SINGLE run or if it is a "real" (as opposed to an intermediate) point in a MULTIPLE run, the point can be approached from a different direction. For example, a MULTIPLE run is being used to vary the ambient temperature from 100 °F to 60 °F in steps of five degrees, and the model cannot solve the point corresponding to a temperature of 75 °F. The model <u>may</u> solve if the point is approached from a lower temperature instead.

   b) If the problem point is a intermediate point in a MULTIPLE run that is affecting the real points later in the run, the number of intermediate steps between each real point can be changed. This will cause the model to not attempt that exact point, and thus, the problem point may be "jumped" over in the process.

5. If Xs and Ks are switched, rebuild the NonZeroList, and run the model before any numerical changes are made to the parameters. Once the solution has been obtained in the new configuration, then the parameters can be adjusted, and the model can be solved. Sometimes the model cannot handle the "shock" of an X-K switch and a change in one or more of the parameters at the same time.

6. Although it was already recommended in A.4.3, the NonZeroList should be rebuilt if the user has any doubt about the necessity of doing so. If the model is run, and there are error messages about a "singularity" within one or more residual equations, then the NonZeroList probably needs to be rebuilt. Also, if the model is run, and the residuals do not seem to be going down as fast as normal, the NonZeroList may need to be rebuilt.

<u>A.5.2 Modifying the model</u>

A comprehensive checklist for the modification of any model solved by the ACRC solver can be found in Appendix B of Mullen and Bullard (1994). The list presented below will be an abridged version of the comprehensive checklist that will be sufficient for most situations.

1. If variables (Xs), parameters (Ks), or calculated values (Cs) are added to the XK file, the following has to be done:
   a) Their order must agree with the order of the Xs, Ks, and Cs within EQUIVLNT.INC.
   b) All of the Xs, Ks, and Cs should also be declared as double precision FORTRAN variables in EQUIVLNT.INC.
   c) The number of variables (Xs) must match the number given by the FORTRAN variable "NumVar" located in INITMOD.f. The sum of number of parameters (Ks) and calculated values (Cs) should match the number given by the FORTRAN variable "NumPar" located in INITMOD.f.
   d) Obviously, if any variables (Xs) are added, then the same number of residual equations has to be added to EQNS.f.
2. If any of the residual, or governing, equations are modified, then the NonZeroList has to be rebuilt the next time the model is run.
3. If any residual equations are added to or removed from EQNS.f, the following has to be done:
   a) The computed GOTO numbers in the FORTRAN must match the number labels for each residual equation.
   b) The indices of the "R( )" array must be continuous for all the residual equations.
   c) The number of residual equations must match the number of variables (Xs) in the XK file and the number given by the FORTRAN variable "NumVar" located in INITMOD.f.

References

Cavallaro, A. R., and C. W. Bullard, "Effects of Varying Fan Speed on a Refrigerator/Freezer System," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 101:2, in press, 1995.

Goodson, M.P., and Bullard, C.W., "Modeling a Refrigerator/Freezer System," ACRC TR-61, University of Illinois at Urbana-Champaign, 1994.

Grebner, J.J. and R.R. Crawford, "The Effects of Oil on the Thermodynamic Properties of Dichlorodifluoromethane (R-12) and Tetrafluoroethane (R-134a)," University of Illinois at Urbana-Champaign, ACRC TR-13, 1992.

Mullen, C. E. and C. W. Bullard, "Room Air Conditioner System Modeling," University of Illinois at Urbana-Champaign, ACRC TR-60, 1994.

Appendix B

ACRC Refrigerator/Freezer Model (RFSIM) Description

The purpose of this document is twofold: to present the theory behind the governing equations and some of the supporting routines used in the Refrigerator/Freezer Simulation (RFSIM) Model; and to describe the model, in sufficient detail, so that users with access to the FORTRAN code will be able to understand the existing model and make modifications that will certainly arise during the use of the model.

Some of the information presented here is covered elsewhere due to the close relationship between the ACRC Solver and the RFSIM model. This document is similar to Appendix D of Goodson and Bullard (1994) since much of the information has not changed in the new version of the model. In order to have a single, current model description, some sections have been replicated (either conceptually or exactly) from the previous appendix with the consent of the authors.

B.1 Governing Equations

There are several topics concerning the format of the governing equations that have to be covered before the theory of the equations themselves can be explained. Throughout this document, the governing equations will also be referred to as "residual" equations for reasons that will soon become clear. All of the governing, or residual, equations are located in the FORTRAN file "EQNS.f".

B.1.1 Form of the equations

B.1.1.1 Residual Format

The Newton-Raphson (NR) method that is employed by the ACRC Solver to solve the system of non-linear equations requires that the equations be written in residual format. This means that all of the variables and parameters on one side of the equation must be moved to the other side of the equation. In algebra, the contents of one side of the equation would then be equal to zero. In the NR method, however, the contents are set equal to a "Residual" value that will change with every iteration. The function of the NR method is to change the variables within the residual equation so that the residual value becomes smaller and smaller. Once the residual value goes below some specified tolerance (near zero), the equation is considered to be solved. As an example, an equation written in algebraic form would appear as follows:

$x^2 + 7 = 15$

However, in residual format the equation would appear as follows:

```
Res(1) = 15 - (x² + 7)
```

The above equation would be solved when the "Res(1)" value was less than some specified tolerance. The residual value was written in an array, or subscripted, form because most numerical implementations of the NR method, including the ACRC Solver, will designate a single array that will contain all of the residual values. Although all of the governing equations will appear in the above residual format, there are some additional characteristics of the EQNS.f file that will be discussed.

B.1.1.2 Sparse Matrix Jacobian Calculation

B.1.1.2.1 Theory

A detailed description of the NR method can be found in Stoecker (1989), and a somewhat abridged description can be found in Appendix A of Mullen and Bullard (1994). One of the steps in the NR method is the calculation of the Jacobian matrix which is used to solve for the variable increments (ΔX) through the linear algebra shown in Equation B.1.

$$[\mathrm{J}][\Delta\mathrm{X}]=[\mathrm{R}] \tag{B.1}$$

Where J is the Jacobian matrix and R is the residual array. The above set of linear equations has to be solved at every iteration in order to increment the variables toward a solution. The Jacobian matrix is a NxN matrix of partial derivatives of all N number of residual equations with respect to all N number of variables. Since many of the residual equations are functions of only a few variables, many of the partial derivative "spots" in the matrix will always be zero. If the entire Jacobian (NxN partial derivatives) is calculated every iteration, the computation time can become quite excessive. Therefore, it is very beneficial to only calculate the partial derivatives that will ever be non-zero during the iterations. This technique is known as a "sparse-matrix" Jacobian calculation, and it affects the format of the residual equations in EQNS.f.

B.1.1.2.2 Implementation in the ACRC Solver

In order to calculate just the necessary partial derivatives, the residual equations are placed within a computed GOTO structure. A sample computed GOTO statement is shown below:

```
GOTO (10, 20, 30), NUM
```

If NUM in the above statement equals 1, the program will shift control to line number 10. If NUM equals 2 or 3, the program will go to line number 20 or 30, respectively. Although the line numbers here have been listed in ascending order, there are no restrictions on the line numbers.

It is the position within the parenthesis, not the value of the line numbers, that determines which line number the program shifts control to.

The computed GOTO structure containing the residual equations can be illustrated by a simple example.

```
      ELEMENT = 1
1     EQNUM = NonZeroList(ELEMENT,VariableNum)
      ELEMENT = ELEMENT + 1

      GOTO(10, 20, 30),EQNUM
      IF(EQNUM.EQ.0) GOTO 99

10    R(1) = x + y - 7
         Goto 1
20    R(2) = x^2+ z^2 - 16
         Goto 1
30    R(3) = y + z^2 - 9
         Goto 1

99    END
```

In this three-equation set, the integer variable EQNUM would determine which residual equation was evaluated. The value of EQNUM is obtained by reading a particular memory location from the two-dimensional array NonZeroList. The NonZeroList is a list that stores the numbers all of the residual equations that have to be evaluated in order to calculate the partial derivatives for each variable. For example, the variable "x" affects the 1st and 2nd residual equations only. Therefore, the NonZeroList would show that only these two residuals would need to be evaluated for the purpose of calculating the partial derivatives with respect to the variable "x".

After every residual equation evaluation, the program shifts control back to line number 1 so the EQNUM can be assigned a new number and the next appropriate residual equation can be evaluated. When all of the necessary residual equations for a particular variable's partial derivative calculations have been completed, the value of EQNUM will be set to zero and the residual values will be returned to the subroutine that called the above sample code.

As in the above example, all of the actual residual equations in EQNS.f need to preceded by a unique line number that is also in the computed GOTO statement. The residual equations should be followed by the "GOTO 1" statement as well. The order of the residual equations is arbitrary because the ACRC Solver will determine the correct residual numbers for each variable when it builds the NonZeroList. In other words, if the 2nd and 3rd residual equations were switched, the new NonZeroList would show that the 1st and 3rd residual equations would have to be evaluated for the partial derivative calculations with respect to the variable "x". For more information about the NonZeroList, see Appendix A.

B.1.1.3 Equation Switching

It is sometime necessary, in a model like the RFSIM model, to have the ability to switch between two or more residual equations that describe the same component. For example, a certain one-variable curve fit may be included as a residual equation, and it may have a different form depending on the value of one of its independent variable. If the variable is less than 100, the curve fit may be linear, and if the variable is greater than 100, the curve fit may be a quadratic. Although there are no such curve fits in the RFSIM model, there is a need for the equation switching ability just described because of the form of the evaporator and condenser models and the implementation of the capillary tube-suction line heat exchanger model.

Since the heat exchangers are modeled in zones (subcooled, two-phase, superheated), they may be some solutions where the condenser, for example, has a two-phase exit and others were it has a subcooled exit. If there is a two-phase exit, the entire set of subcooled equations (heat transfer, pressure drop, etc.) will not have any meaning. If equation switching were not available, then separate models for every possible combination of heat exchanger exits (i.e. subcooled condenser and two-phase evaporator) would have to be developed. The user would essentially have to know the heat exchanger exits before the model was solved. This arrangement would obviously be cumbersome and not very useful.

The equation switching for both heat exchanger exits are performed with the help of the logical flags "Evap2phX" and "Cond2phX". These two flags are both "true" if the respective heat exchanger exits are two-phase. The values for these flags are set in the initial checking (IC) and boundary checking (BC) subroutines in the CHECKMOD.f file, and the process is described in section A.4.4.1, A.4.4.2, B.2.1, and B.2.2. Therefore, when the model is run and the residual equations involving the exit states of both heat exchangers are to be evaluated, the above flags tell the solver which equations to use. For example, the refrigerant-side heat transfer in the subcooled zone of the condenser is described by the following residual equation:

```
IF (Cond2phX) THEN
   R(cond+10) = 0 - qsubcond
ELSE
   R(cond+10) = w*(h20 - h3) - qsubcond
END IF
```

If the condenser has a two-phase exit (Cond2phX = "true"), then heat transfer in the subcooled region is set equal to zero. However, if the condenser has a subcooled exit (Cond2phX = "false"), then the heat transfer is equal to the mass flow rate (w) times the difference in enthalpy at states 20 and 3.

There are several other residual equations describing the capillary tube-suction line heat exchanger (ct-slhx) that require a similar switching ability. Each residual equation contains two possible equations that can be used depending on whether the ct-slhx model is used within the RFSIM model. The flag that indicates whether or not the capillary tube model is being used is

"CTSLHXSIM" and its value is also set in the initial checking (IC) subroutine. If the capillary tube-suction line heat exchanger model is being used, the CTSLHXSIM flag is given a "true" value. Unfortunately, it is somewhat more difficult to see the "equivalence" between the two equations that are switched by means of the logical flag value. Therefore, these equations will not be shown here, but they will be presented when the modeling of the ct-slhx is discussed.

B.1.1.4 NonZeroFlag

The NonZeroFlag is a logical flag used to indicate when the NonZeroList is being calculated. There are some instances where it must be used because of the equation switching capabilities which were just explained in section B.1.1.3. There are other instances when it is very useful, but optional, because it can greatly reduce the amount of time required to rebuild the NonZeroList. As an example of a situation where it must be used, the equation concerning the heat transfer in the subcooled zone of the condenser will be discussed again.

```
1640    IF (.not.NonZeroFlag) THEN
           IF (Cond2phX) THEN
              R(cond+10) = 0 - qsubcond
           ELSE
              R(cond+10) = w*(h20 - h3) - qsubcond
           END IF
        ELSE
           R(cond+10) = qsubcond + w + h20 + h3
        END IF
         GOTO 1
```

As discussed earlier, the equation flag Cond2phX dictates which equation is to be used. However, the possibility of two equations poses a problem for the sparse matrix Jacobian calculation. The NonZeroList shows all the residual equations that a particular variable can possibly affect. Since the NonZeroList is only calculated once, at the beginning of a solution, it is possible for the list to be incorrect if the equations are switched during the solution.

In the above equation, only one variable (qsubcond) appears if there is a two-phase exit, but four variables (qsubcond, w, h20, h3) appear if there is a subcooled exit. Therefore, if the NonZeroList were built when the exit was two-phase, only one variable (qsubcond) would have an effect on this particular residual equation. If the exit condition of the heat exchanger changed during the iterations, the NonZeroList would no longer be correct and the necessary partial derivatives (with respect to w, h20, and h3) would not be taken.

To remedy this problem, an extra equation and logical flag has been added to all the equations involving the exit states of the heat exchanger. When the NonZeroFlag is "true", a dummy residual equation is evaluated that includes every variable that occurs in both of the real equations. The form of this equation is irrelevant since the NonZeroList only checks to see if the residual changes with a change in the variable. Therefore, it is easiest to simply add up the variables as shown above in the example. It is a good idea to also include any parameters (Ks) that might be in the either equation in the dummy equation since parameters can easily become variables through the X-K swapping discussed in A.1.1and A.1.2. When the NonZeroFlag is "false", one of the actual residual equations is evaluated.

Sometimes it may beneficial to add the extra "IF-ELSE IF-END IF" logic and the dummy residual equation even when it is not absolutely necessary. The reason for doing this is because every partial derivative location in the Jacobian matrix is calculated when the NonZeroList is built. Even with simple equations, this can be computational expensive, but building the NonZeroList can be extremely slow when some of the residual equations contain lengthy subroutines and function calls.

In the RFSIM model, the prime example of this is the capillary tube model subroutine. Since it is called from the residual equations, the capillary tube model subroutine is called 112 times (112 partial derivative calculations of one of the residual equations w.r.t. every variable). To eliminate this excessive computation time, a dummy residual equation was added that contained all the variables and parameters that might ever affect those particular residual equations. This technique is used in a few other residual equations that involve relatively complicated routine calls. However, the user must be careful not to overlook any variables and parameters that might affect the residual equation since this will result in an incorrect NonZeroList.

B.1.1.5 Equation Counters

One of the advantages of using a Newton-Raphson solver is that the equations do not have to be in any particular order. For convenience, however, the RFSIM model places the residual equations into groups by component or sub-model. If it is necessary to insert an equation into the middle of the residual equations, then the indexes of all of the residual values following that one will need to be renumbered by adding one to their current number.

To make modifying the model a little easier, integer counters were developed to mark the first location in each group of equations. For example, if the first condenser equation is the 71st residual equation, then the counter for the condenser group (cond) will be equal to 71 and the residual value of that equation will be denoted R(cond+0). All of the other residual values of the condenser equations will be denoted R(cond+n) where n = 1, 2, 3, 4, etc. Therefore, if an equation is added in the condenser group, then only the "n" part of the index of the residual values of the condenser equations that follow and the integer counters of the groups that follow will have to be increased by one.

B.1.2 Refrigerant state equations

The refrigerant state at every point in the cycle is defined by the group of equations referred to as the "state" equations. This group of equations includes thermodynamic property equations, pressure drop equations, and a few heat transfer equations. Typically, in the single-phase portions of the refrigerant cycle, the pressure and temperature are considered the independent variables and the specific enthalpy and volume are obtained by the appropriate property calls to the interpolation package described in Appendix D. In the two-phase portions of the cycle, the pressure or temperature and the quality are considered the independent variables and the specific enthalpy, specific volume, and the temperature or pressure are obtained by the appropriate property calls to the interpolation package.

The pressures at every state point, with the exception of the compressor inlet, are related to the pressure at the next state by a pressure drop equation. The refrigerant temperatures in the two-phase portions of the cycle can be directly calculated from the pressures. However, it is somewhat harder to see where the single-phase temperature values come from. Since the ACRC Solver uses the NR method to solve the equations, it is unnecessary to have an explicit expression for every variable in the model. In other words, the single-phase temperatures are implicitly included in several equations in the model and their values are determined when the entire model is solved. The enthalpies that are calculated through the thermodynamic property routines are used in the refrigerant side energy balances. The volumes are used in the calculation of the refrigerant mass and in the pressure drop equations.

The calculation of these thermodynamic properties can be best seen by looking at an example. For each segment in the refrigerant cycle (i.e. superheated region of the condenser), the states at the endpoints are defined and the pressure drop in the segment is calculated. Below are the equations for the superheated portion of the condenser as they appear in EQNS.f.

```
C***********************  Condenser Inlet  ****************************C

140     R(prop+6) = hpt(p1,t1) - h1
         GOTO 1

160     R(prop+7) = vpt(p1,t1) - v1
         GOTO 1

C*********  Pressure drop in the superheated condenser  ***************C

180     R(prop+8) = DpSupCond - dpspHX(Dcond,STC,NSECTC,w,
     &              RTBCND*fsupcond,v1,v21,t1,t21,DZC*fsupcond,rough,0)
         GOTO 1

200     R(prop+9) = p1 - p21 - DpSupCond
         GOTO 1

C******************* Condenser at saturated vapor *******************C

220     R(prop+10) = TsatP(p21,1.0d0) - t21
         GOTO 1

240     R(prop+11) = htx(t21,1.0D0) - h21
         GOTO 1

260     R(prop+12) = vtx(t21,1.0D0) - v21
```

```
      GOTO 1
```

At state 1 (condenser inlet), the enthalpy and volume are calculated by the single-phase property functions "hpt" and "vpt" which return the properties as a function of pressure and temperature. The pressure drop in the superheated portion of the condenser is returned by the function "dpspHX". This pressure drop (DpSupCond) is then set equal to the difference of inlet and outlet pressure at states 1 and 21. At state 21 (saturated vapor), the temperature (t21) is calculated by though the use of the "TsatP" function call. This function call requires the pressure and the quality because it was written general enough to handle zeotropes where the temperature would depend on quality as well as the pressure. The enthalpy and volume at state 21 are calculated through the use of the "htx" and "vtx" functions and the saturation temperature (t21). These could have just as easily been calculated through the use of the "hpx" and "vpx" functions and the pressure (p21).

It should be noted that the heat transfer in the superheated condenser zone is calculated, but it is done elsewhere in the group of condenser equations. The current method of dividing the equations into groups is somewhat arbitrary, but the intention is to make the state equations as independent and general as possible. In other words, if a different condenser geometry was modeled in the RFSIM model, only the equations in the condenser group might have to be modified. The state equations would remain the same as long as there was a superheated zone.

There are also a few equations which are in the model to account for the heat transfer in the refrigerant tubing other than the condenser and evaporator. There is a heat transfer equation for the discharge line (compressor outlet to condenser inlet) and the liquid line (condenser outlet to capillary tube inlet). Currently, these equations assume that there is no heat transfer in either of these components by setting the inlet enthalpy equal to the outlet enthalpy as shown below.

```
60     R(prop+2) = h0 - h1
        GOTO 1
```

This equation requires the enthalpy at the compressor discharge (h0) to equal the enthalpy at the condenser inlet (h1). The equations have been kept in the model so that future users could easily modify the model by adding terms to account for heat transfer in the discharge and liquid line. There are also two equations that deal with the heat transfer in the capillary tube-suction line heat exchanger. However, since these equations depend on the whether the capillary tube model is used, they will be discussed with the other capillary tube equations.

B.1.3 Charge inventory equations

The importance of the charge inventory calculations on the RFSIM model has been presented in section A.2.2.2 and A.3.3 will not repeated here. Instead, the theory of the equations used to calculate the refrigerant mass in the single- and two-phase components will be presented.

The calculation of the refrigerant mass in the single-phase components is a relatively straightforward matter. The mass can be calculated through the use of the Equation B.2 shown below.

$$M_{single-phase} = \frac{V_{component}}{\upsilon_{refrigerant}} \tag{B.2}$$

This calculation is done in the "MassSingle" function call located in EQNSUBS.f. An example of a residual equation that uses this function call is shown below for the suction line.

```
C***** The suction line refrigerant mass   ******************************

1220   R(mass+6) = MSuctLine - MassSingle(volsuctline,v9,v11)
        GOTO 1
```

The MassSingle function only requires the volume of the component, the inlet specific volume and the outlet specific volume. These two volumes are averaged and then used in Equation B.2 to calculate the single-phase refrigerant mass.

There are two contributions to the total refrigerant mass that are contained in the compressor: a single-phase vapor portion that exist in the free volume of the compressor sump, and a liquid portion that is dissolved in the oil. The single-phase vapor refrigerant is calculated, as with any other single-phase component, through Equation B.2. However, the calculation of the liquid refrigerant dissolved in the oil is handled through the use of empirically determined relations that predict the liquid fraction of the refrigerant in the oil. The two residual equations that calculate the liquid refrigerant mass in the compressor oil are shown below.

```
C**** The mass of refrigerant dissolved in the oil (high-side sump)  ****

1260   R(mass+8) = Mcompoil/(Mcompoil + Moil) - Woil
        GOTO 1

1280   Aoil = K1oil + (K2oil/Woil**(1.0/2.0))
       Boil = K3oil + (K4oil/Woil**(1.0/2.0)) + (K5oil/Woil) +
     &            (K6oil/Woil**(3.0/2.0)) + (K7oil/Woil**2.0)
       Tstar = (1 - Woil)*(Aoil + Boil*p0)

       R(mass+9) = ((t0+460) - (Tsat0+460))/(Tsat0+460) - Tstar
        GOTO 1
```

Together, these two equations define the liquid refrigerant mass fraction (Woil) and the amount of refrigerant (Mcompoil) in the oil. The details of the user specified oil-refrigerant solubility constants (K1oil through K7oil), sump pressure (p0), and saturation temperature (Tsat0) are discussed in A.3.2. This empirical model was developed by Grebner and Crawford (1992).

The single- and two-phase refrigerant contained in the condenser and evaporator is calculated through the function calls "MassCond" and "MassEvap", respectively. The amount of refrigerant in the single-phase portions of the heat exchangers are calculated in the same way as the other single phase components, but the calculation of the two-phase refrigerant mass is significantly more complicated. Although the single-phase mass calculation is straightforward, the function calls do contain logic so that the single-phase mass at the condenser and evaporator exit will only be calculated if those phases exist. The actual residual equation that calls the MassCond function is shown below.

```
C***** The condenser refrigerant mass ***********************************

1140   IF(.not.NonZeroFlag) THEN
         R(mass+2) = MCond - MassCond(Volcond,fsupcond,f2phcond,fsubcond,
     &      v1,v21,v20,v3,xoc,t21,t20,w,Dcond,50,3)
       ELSE
         R(mass+2) = Mcond+Volcond+fsupcond+f2phcond+fsubcond+v1+v21+
     &      v20+v3+xoc+t21+t20+w+Dcond
       END IF
        GOTO 1
```

The first equation is the real residual equation and the second equation is an example of the optional dummy residual equation that is used for the purposes of speeding up the evaluation of the NonZeroList. The real equation sets the value returned under the name MassCond equal to the system variable for the mass in the condenser, Mcond.

There are two primary factors that complicate the prediction of two-phase refrigerant mass in evaporating or condensing flow: how the void fraction (ratio of the vapor cross-sectional area to the total cross-sectional area) varies with quality; and how the quality varies with the length of the heat exchanger tubing. The void fraction can be different for a given quality because the slip (ratio of the vapor velocity to liquid velocity) can be different for a given quality. The quality variation with length depends on the heat flux. If the heat flux is constant, then the quality will vary linearly with the heat exchanger length.

The amount of refrigerant mass in a two-phase region can be calculated from Equation B.3 shown below.

$$M_{two-phase} = \frac{V \cdot \left[\rho_g \int_0^L \alpha \, dl + \rho_f \int_0^L (1-\alpha) \, dl \right]}{\int_0^L dl} \tag{B.3}$$

The above expression for the two-phase mass can be rewritten as Equation B.4.

$$M_{two-phase} = V \cdot \left[\rho_g W_g + \rho_f \left(1 - W_g \right) \right] \tag{B.4}$$

The variable W_g includes the integrals in Equation B.3 and is described by Rice (1987) as the "heat-flux-averaged void fraction over a given quality range". Rice (1987) also found that the heat flux assumption was unimportant for the calculation of the evaporator mass and probably less important than choice of void fraction correlation for the calculation of the condenser mass.

For this reason and a lack of knowledge about the actual heat flux in the two-phase portion of the condenser, the heat flux is assumed to be constant. When this assumption is made, the expression for W_g can be written as Equation B.5 shown below.

$$W_g = \frac{1}{(x_o - x_i)} \int_{x_i}^{x_o} \alpha(x) dx \qquad \text{(B.5)}$$

Thus, the problem of calculating the two-phase mass in the heat exchangers is essentially a problem of choosing the correct void fraction correlation to integrate in Equation B.5. As explained in A.3.2.2, the Premoli void fraction correlation is currently used in the RFSIM model. In the MassCond and MassEvap function calls, the integral in B.5 is evaluated by numerical integration (trapezoid rule) of the chosen void fraction correlation over the correct quality range. With the value of W_g (average void fraction), the two-phase mass is directly calculated with Equation B.4.

The last refrigerant inventory equation is the charge conservation equation. As shown below, all of the contributions to the total mass are summed and set equal to the total mass (Mtotal).

```
c**** The total charge in the system ***********************************

1300   R(mass+10) = MCond + MEvap + MDisLine + MLiqLine + MAccum
     &              + MSuctLine + MCompvap + MCapTube + Mcompoil  - Mtotal
        GOTO 1
```

This is the equation that really constrains the system when the Mtotal is specified as a known parameter (K) as discussed in A.2.2.2.

B.1.4 Compressor equations

There are five equations, other than the refrigerant charge equations, that are used to model the compressor: mass flow rate through the compressor; power consumed by the compressor; refrigerant-side energy balance about the compressor; air-side energy balance about the compressor; and a rate equation describing the heat transfer from the compressor shell to the air stream. Since there are several choices that the user must make in the compressor modeling, many of the equations to follow have already been discussed in A.3.2.

The mass flow through the compressor and the power consumed by the compressor are described by nine-parameter curve fits known as compressor “maps”. The mass flow and power consumption are calculated as functions of the saturation temperatures corresponding to the inlet and outlet pressures of the compressor. These relations, or the data necessary to make them, are usually available from the compressor manufacturers. These empirical relations have not been replaced by more physical compressor models because compressor maps are widely accepted and understood within the refrigeration industry.

The several maps that are currently available within the RFSIM model are described in A.3.2. With the exception of the first map (top-mount Amana with manufacturer's data), all of the maps are specific to a particular compressor-refrigerant combination. Within EQNS.f, the two residual equations involving the compressor maps appear as follows:

```
c****   The compressor map mass flow rate ********************************

1320    R(comp+0) = beta_Wmap * wf(tsat0,tsat11,CompNum) - w
          GOTO 1

c****   The compressor map power ******************************************

1340    R(comp+1) = beta_Pmap * Pcompf(tsat0,tsat11,CompNum) - powercomp
          GOTO 1
```

The two map values of mass flow rate and power are returned by the two function calls "wf" and "Pcompf", respectively. The variables tsat0 and tsat11 are the two saturation temperatures, and the parameter CompNum is the number of the compressor map to be used. These returned values are then multiplied by two parameters, beta_Wmap and beta_Pmap, before being set equal to the system values of mass flow rate and compressor power consumption. Since these parameters are discussed in detail in A.3.2, only a brief description will be given here. Their basic purpose is to "scale" the compressor maps to simulate the effect of a change in compressor speed or compressor size. However, the accuracy of such a scaling will be dependent on the user's knowledge of how the mass flow and power vary with the compressor speed or size.

The refrigerant-side energy balance about the compressor is a classic application of the 1st Law of Thermodynamics for a control volume and can be seen below.

```
c****   Refrigerant-side energy balance for the compressor ***************

1360    R(comp+2) = w*(h0-h11) - (BTU(powercomp) - qcomp)
          GOTO 1
```

The residual equation requires the change in total enthalpy of the refrigerant flowing through the compressor to be equal to the work (in Btu/hr) put into the compressor minus the heat transfer from the compressor.

The 1st Law of Thermodynamics is also applied to air stream flowing over the compressor.

```
C****   Air-side energy balance for the compressor ***********************

1380    mdotacond = vdotcond_calc*60/va(patm,tacondfanin)
        R(comp+3) = mdotacond*(ha(tacondfanin) - ha(tacondout)) - qcomp
          GOTO 1
```

This residual equation requires the change in total enthalpy of the air stream to be equal to the heat transfer to the air. The mass flow of air (mdotacond) is an intermediate variable calculated from the volumetric flow rate and the specific volume.

The rate of heat transfer from the compressor can also be described through the use of a convection heat transfer relation.

```
c****  Rate equation for the heat transfer from the shell to the air **

1400   ts = -3.4407 + 0.88355*t0
       R(comp+4) = hAcomp*(ts - tacondout) - qcomp

         GOTO 1
```

This residual equation requires the heat transfer from the compressor shell to be equal the appropriate heat transfer coefficient (hAcomp) times the temperature difference between the shell (ts) and the inlet air stream (tacondout). In an actual compressor, the shell temperature is dependent upon the operating conditions. Since the compressor model (i.e. the maps) is not a physical model that predicts the shell temperature, this temperature needs to be related to some other variable that is related to the operating conditions. It has been found by Cavallaro and Bullard (1994) that the compressor shell temperature (ts) is directly proportional to the compressor discharge temperature (t0). Therefore, the above equation actually relates the rate of heat transfer from the compressor to the discharge temperature and the inlet air temperature.

B.1.5 Condenser equations

The condenser equations model the heat transfer from the refrigerant to the air within the condenser by breaking the condenser into three zones: superheated, two-phase, and subcooled. Each of these three zones can be thought of as an individual heat exchanger that has its own inlet and outlet refrigerant and air streams. Therefore, each zone can be modeled with three equations: a refrigerant-side energy balance, an air-side energy balance, and an effectiveness-NTU rate equation.

As a "by-product" of this type of modeling, the fractions of the entire condenser occupied by each phase are also obtained. These fractions (fsupcond, f2phcond, and fsubcond) are used in the heat transfer equations as well as the pressure drop equations discussed in B.1.2. The fact that condenser zone fractions are obtain indirectly is good example of the benefits of using a method like Newton-Raphson to solve the equations. It would be very difficult, if not impossible, to solve the equations in the RFSIM model with a technique such as successive substitution because it would be difficult to obtain explicit expressions for all the variables such as the condenser zone fractions.

This method of modeling a multi-phase heat exchanger can be illustrated by the condenser schematic shown in Figure B.1 below.

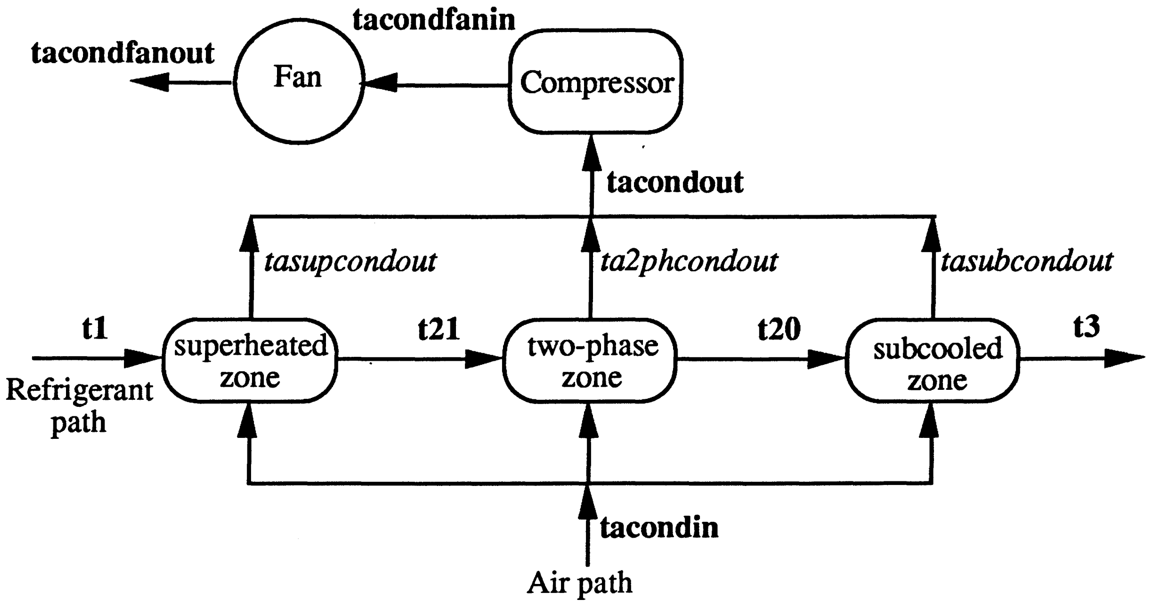


Figure B.1 Multi-zone condenser modeling schematic

B.1.5.1 Refrigerant-side energy balance

Since the equations for all three zones are similar, only the superheated zone will be discussed in detail. The simplest of the three heat transfer equations is the refrigerant side energy balance.

```
C       ** Refrigerant-side energy balance **

1520    R(cond+4) = w*(h1 - h21) - qsupcond
         GOTO 1
```

This residual equation requires that the change in total enthalpy of the refrigerant stream flowing through the superheated condenser zone (states 1 to 21) be equal to the heat transfer in that zone (qsupcond).

B.1.5.2 Effectiveness-NTU equations

The effectiveness-NTU relations provides a means of calculating the heat transfer in a heat exchanger based upon information about the two fluid streams and the characteristics of the heat exchanger. One of the pieces of information needed for the effectiveness-NTU method is the "UA" for the particular heat exchanger to be modeled. Although this calculation could be done as an intermediate calculation, a separate residual equation is used so that the user can specify the "UA" value or the air-side heat transfer coefficient within the heat exchangers. The calculation of the "U" for the superheated condenser zone is shown below.

```
c       ** Overall convection coefficient U **
```

```
1560   if(int(hcondNum).ge.1) then
          haircond = haircnd(vdotcond_calc,AAFC,hcondNum)
       end if

       Call UsCond(1,w,(t1+t21)/2.,(t20+t21)/2.,(t20+t3)/2.,xoc,Dcond,
     &  crtmult,NSECTC,alphacond,haircond,Usubcondc,U2phcondc,Usupcondc)

       R(cond+6) = usupcondc - usupcond
        GOTO 1
```

The initial "IF THEN-END IF" statement is required because the user can specify a numerical value for the air-side heat transfer coefficient (hcondNum=0) or a particular air-side heat transfer correlation (hcondNum>0). If a correlation is specified, the air-side heat transfer coefficient (haircond) will be returned by the function call "haircnd" located in the EQNSUBS.f file. In any case, the "UsCond" subroutine, located in the EQNSUBS.f file, will return the overall heat transfer coefficient (usupcondc) based upon Equation B.6 shown below:

$$\frac{1}{U_{air}A_{air}} = \frac{1}{h_{air}A_{air}} + \frac{1}{h_{ref}A_{ref}} \tag{B.6}$$

Equation B.6 assumes that there is no resistance to heat transfer within the condenser material itself. The heat transfer coefficient on refrigerant side (h_{ref}) is determined within UsCond by a function call to single-phase heat transfer correlation. If this were the two-phase zone, the h_{ref} would be determined by a call the "h2phcondACRC" function located in the FUNCTION.f file. This function returns an average heat transfer coefficient by integrating, over the specified quality range, the local two-phase condensing heat transfer correlation developed at the ACRC by Dobson and Chato (1994). With the two heat transfer coefficients (h_{air} and h_{ref}) and the ratio of external to internal areas (A_{air}/A_{ref}), Equation B.6 can be rewritten as Equation B.7 and used to solve explicitly for U_{air}..

$$U_{air} = \left[\frac{1}{h_{air}} + \frac{\left(A_{air} / A_{ref}\right)}{h_{ref}}\right]^{-1} \tag{B.7}$$

As a last step, the above residual equation sets the returned value of U_{air} (usupcondc) equal to the residual variable for the U_{air} value (usupcond), which is used in the effectiveness-NTU equation shown below.

```
c      ** Effectiveness-NTU equation **

1540   asupcond = acond * fsupcond
       cminsupcond = min(qsupcond/(t1-t21),Cacond*fsupcond)
       cmaxsupcond =  max(qsupcond/(t1-t21),Cacond*fsupcond)
       esupcond = epc(usupcond*asupcond,cminsupcond,cmaxsupcond)

       R(cond+5) = esupcond*cminsupcond*(t1 - tacondin)  - qsupcond
        GOTO 1
```

The effectiveness residual equation involves several intermediate calculations in addition to the actual effectiveness equation. First, the area of the superheated condenser zone is determined from the total area (acond) and the superheated fraction (fsupcond). Then, the minimum and maximum heat capacities of the two fluid streams in the heat exchanger are determined. The heat capacity for the refrigerant stream is calculated by dividing the superheated heat transfer by the corresponding change in temperature. The heat capacity for the air stream is calculated by multiplying the total heat capacity of the condenser air stream (Cacond) by the fraction that actually flows over the superheated portion. The effectiveness value is calculated by a function call to "epc" located in the FUNCTION.f file. This particular function call is used because of the "parallel-counterflow" geometry that exist within the condenser. Lastly, the heat transfer in the superheated condenser is calculated by multiplying the effectiveness by the minimum heat capacity and the temperature difference between the two inlet streams.

B.1.5.3 Air-side energy balance

There is not a separate residual equation involving the air-side energy balance for each of the three condenser zones because of the geometry of this particular condenser. As shown in Figure B.1, the heat transfer in all three zones contribute to the outlet condenser air temperature. Since the individual outlet air temperatures for each zone do not have physical significance, the air-side energy balances for each zone were done as intermediate calculations for the purpose of obtaining a mixed-air outlet temperature.

```
c**** The weighted average of the outlet condenser air temps ***********

1700   tasupcondout = qsupcond/(Cacond*fsupcond) + tacondin
       ta2phcondout = q2phcond/(Cacond*f2phcond) + tacondin

       if(.not.Cond2phX) then
         tasubcondout = qsubcond/(Cacond*fsubcond) + tacondin
       else
         tasubcondout = 0.0
       end if

       R(cond+13) = fsupcond*tasupcondout + f2phcond*ta2phcondout +
     &   fsubcond*tasubcondout - tacondout
        GOTO 1
```

The first two lines of this residual equation calculate the outlet air temperature of the superheated and two-phase zones (tasupcondout and ta2phcondout, respectively). The "IF THEN-ELSE-END IF" logic used for the subcooled air outlet temperature (tasubcondout) is necessary because the subcooled zone might not exist. Lastly, the mixed air temperature is approximated by fraction-weighted average of these three individual temperatures.

B.1.5.4 Miscellaneous equations

In addition to the equations for the two-phase and subcooled zones, there are several other equations used to model the condenser. One equation calculates the external area of the condenser and is shown below.

```
c       ** Air-side area of the condenser **
1420    R(cond+0) = alphacond*pi()*Dcond*Lcond - acond
         GOTO 1
```

Another equation requires the three condenser zone fractions to add up to 1.0. This is equivalent to requiring the areas of the three zone to add up the total area.

```
c       ** The fractions of the three zones have to add to one **
1460    R(cond+1) = fsubcond + f2phcond + fsupcond - 1.0
         GOTO 1
```

A similar equation sums the heat transfers of each zone and sets them equal to the total condenser heat transfer (qcond).

```
C**** Overall heat transfer in the condenser ****************************
1500    R(cond+3) = qsubcond + q2phcond + qsupcond - qcond
         GOTO 1
```

There are three other equations that calculate quantities related to the condenser air stream. The first equation calculates the heat capacity that is used in all of the effectiveness equations and the air-side energy balance.

```
c**** Heat capacity of the condenser air stream ************************
1480    mdotacond = vdotcond_calc*60/va(patm,tacondfanin)
        R(cond+2) = mdotacond*cpa(tacondin) - Cacond
         GOTO 1
```

The second equation is an air-side energy balance about the condenser fan. In this particular refrigerator, the fan is located downstream of the condenser and the compressor, and therefore, the outlet air temperature of the fan (tacondfanout) is the temperature of the air that is returned to the environment.

```
c**** Energy balance about the condenser fan ***************************
1720    mdotacond = vdotcond_calc*60/va(patm,tacondfanin)
        R(cond+14) = mdotacond*(ha(tacondfanout)-ha(tacondfanin)) -
     &   BTU(pcondfan_calc)
         GOTO 1
```

The third and last equation calculates the inlet temperature to the condenser based upon the exiting temperature (tacondfanout), the ambient temperature (tamb), and the given recirculation fraction.

```
c**** The weighted average of the inlet condenser air *****************
1740   R(cond+15) = frecirc*tacondfanout + (1-frecirc)*tamb - tacondin
          GOTO 1
```

B.1.5.5 Modeling other condensers

The user should be aware that the exact form of the preceding condenser equations are unique to the refrigerator being model (the Whirlpool in this case). However, the general form of the equations will not change for other refrigerators. The condensers can always be modeled as multi-zone heat exchangers, each described by three equations: a refrigerant-side energy balance, an air-side energy balance, and an effectiveness-NTU equation. The exact form of these equations may differ from the ones shown here because the geometry of the heat exchanger, compressor, and the condenser fan may be different. For example, it was unnecessary to have three separate residual equations for the air-side energy balances in this heat exchanger because three zones all "saw" the same inlet air temperature and were not downstream of each other. There are other, more serious, differences in condenser layout that can affect the way that the equations are written. For example, one of the zones (typically the two-phase zone) can be split into two halves, each with a different flow configuration or air inlet temperature. This is exactly what happened with the condenser of the top-mount Amana (Goodson and Bullard, 1994) and the evaporator in the current Whirlpool refrigerator. Suggestions for dealing with this modeling issue will be discussed in the evaporator section.

B.1.6 Capillary tube-suction line heat exchanger equations

The residual equations that describe the behavior of the capillary tube-suction line heat exchanger (ct-slhx) are substantially different from the other groups of equations. There are actually two different sets of equations, or sub-models, that can be used to model the ct-slhx within the RFSIM model. One sub-model is based upon a finite-difference solution of the governing equations for refrigerant flow through the ct-slhx. This method accounts for the mass flow rate and the heat transfer that takes place within the component. The other sub-model is a simple method that addresses only the heat transfer in the ct-slhx through the use of a effectiveness equation. This second method is equivalent to the constant enthalpy statement ($h_{in} = h_{out}$) that is used in thermodynamic text books to describe the adiabatic capillary tube found in vapor compression cycles. Unless otherwise specified, the "capillary tube-suction line heat exchanger (ct-slhx) model" refers to the sub-model that uses the finite-difference method. More details of the differences between the two sub-models are presented in A.2.2.1.

One of the problems with implementing two sub-models of the same component that have different complexity is the resulting difference in the number of required residual equations. The simple effectiveness method requires only two equations, but the finite-difference approach requires eight equations. In other words, the current number of residual equations required to model the Whirlpool refrigerator could be reduced from 112 to 106 if only the effectiveness method was used to describe the capillary tube. Since the ACRC Solver requires a fixed number of residual equations for given model, and either sub-model can be used within the RFSIM model, 112 residual equations are needed. There are a total of eight residual equations related to the capillary tube-suction line heat exchanger. There are actually two possible equations for each of these eight residual equations that can be switched as described in B.1.1.3. However, six of these equations will be "dummy" equations if the effectiveness method is used to model the capillary tube.

B.1.6.1 Effectiveness based sub-model

There are only two equations that really describe the capillary tube-suction line heat exchanger when the effectiveness method is used. The first equation predicts the amount of the heat transfer from the hot refrigerant in the capillary tube to the colder refrigerant in the suction line based upon the user-supplied value of the effectiveness. The effectiveness of the heat exchanger is defined as shown in Equation B.8 below.

$$\varepsilon_{ct-slhx} = \frac{h_{suction,out} - h_{suction,in}}{\left[h\left(P_{suction,out}, T_{captube,in}\right) - h_{suction,in}\right]} \qquad \text{(B.8)}$$

Where the numerator of the fraction represents the actual heat transfer (enthalpy difference), and the denominator represents the maximum heat transfer that could take place. This maximum heat transfer would take place when the refrigerant at the suction line outlet reaches the temperature of the refrigerant at the capillary tube inlet.

This definition of effectiveness will have a slight error when the refrigerant is a two-phase mixture in any portion of the adiabatic inlet section of the capillary tube. Since the refrigerant is two-phase, its temperature will drop as the pressure drops due to the throttling effect of the capillary tube. As a result, the refrigerant temperature in the capillary tube at the heat exchanger inlet will be lower than the $T_{captube,in}$ used in Equation B.8. In other words, the refrigerant in the suction line could never reach $T_{captube,in}$ because the highest temperature in heat exchanger on the capillary tube side is lower than $T_{captube,in}$.

The residual equation that implements the effectiveness equation is shown below.

```
1020   IF (CTSLHXSIM) then
         R(prop+50) = numDTsl*DTsl + t9 - t11
       ELSE
         R(prop+50) = ectslhx*(hpt(p11,t4) - h9) - (h11 - h9)
       END IF
        GOTO 1
```

Actually, the second equation is the one that is evaluated when the effectiveness method is used (CTSLHXSIM = "false"). The first equation will be discussed in the following section since it is evaluated when the capillary tube-suction line heat exchanger model is used. The dummy equation for the building of the NonZeroList that was described in B.1.1.4 is not needed here because the CTSLHXSIM flag will never be switched during a solution.

The second equation that describes the capillary tube-suction line heat exchanger when the effectiveness method is used is an energy balance for the component. It is assumed that there is no heat transfer from the capillary tube or suction line to the environment. Therefore, all the heat that leaves the capillary tube is added to the suction line. The residual equation that implements this constraint is shown below.

```
600     IF (CTSLHXSIM) then
           R(prop+29) = hpx(pcrit,xcrit) - hcrit
        ELSE
           R(prop+29) = (h4 - h7) - (h11 - h9)
        END IF
         GOTO 1
```

Again, the second equation is the one that is evaluated when the effectiveness method is employed. It sets the change in enthalpy across the capillary tube (h4-h7) equal to the change in enthalpy across the suction line (h11-h9). As before, the first equation of this group is evaluated when the capillary tube-suction line heat exchanger model is used and will be discussed in the following section.

B.1.6.2 Capillary tube-suction line heat exchanger model

The capillary tube-suction line heat exchanger (ct-slhx) model is a physical model that accounts for the heat transfer, pressure drop, and mass flow in the ct-slhx through the use of a finite-difference solution. The ct-slhx is discretized into several segments (~15), and the appropriate governing equations are solved for each segment. However, the governing equations for each segment are not placed with the other residual equations located in the EQNS.f file. Instead, they are placed in a subroutine ("ctslhx" in CAPTUBE.f) where they are solved in a sequential manner. The sequential solution actually involves the iterative solution of one- and two-variable implicit equations at each segment. The primary reason for sequentially solving the governing equations for each of the discretized segments in a subroutine was to reduce the number of equations and required initial guesses present in the system model. If the discretized equations were place directly into the system model with the other residual equations, the total number of simultaneous equations could easily double, and the user would have to give initial guess values for all the variables (Xs) within the ct-slhx.

Because of the reasons just mentioned, the approach taken in the RFSIM model was to perform the majority of the calculations associated with the ct-slhx in the "ctslhx" subroutine and to reduce the number of ct-slhx residual equations in the system model. Like most subroutines, the sequential solution requires several input values, and it will return several output values. Regardless of the inputs, the subroutine will solve the governing equations for each segment. However, the subroutine outputs may not match the actual situation present within the refrigerator. To remedy this problem, several residual equations (six) are present in the system model that call the "ctslhx" subroutine and then compare the outputs with parameters (Ks) or variables (Xs). In this way, the sequential solution guarantees that the governing equations are satisfied and the residual equations (and the ACRC Solver) guarantee that the "solution" from the subroutine matches the physical situation within the refrigerator. Although this approach may seem confusing when first encountered, the following discussion of the six residual equations should make it clearer.

The remaining discussion of the ct-slhx model will focus on the residual equations that call the "ctslhx" subroutine in the RFSIM rather than on the subroutine itself. A complete discussion of the modeling of the capillary tube-suction line heat exchanger, including the "ctslhx" subroutine, can be found in Appendix A. In order to better explain the residual equations, a summary of the inputs and outputs of the "ctslhx" subroutine will be presented.

The sequential solution depends upon the following variables (in the FORTRAN sense): pcrit, xcrit, DPout, DTsl, DPin, t9, p9, xoe, Dct, Dsuctline, numDPout, numDTsl, and numDPin. The definitions of these variables are shown below.

Pcrit	pressure at the choked exit of the capillary tube
xcrit	quality at the choked exit of the capillary tube
DPout	pressure steps used to discretize the adiabatic outlet section
DTsl	suction line temperature steps used to discretize the heat exchanger section
DPin	pressure steps used to discretize the adiabatic inlet section
T9	refrigerant temperature at the suction line inlet
p9	refrigerant pressure at the suction line inlet
xoe	refrigerant quality at the suction line inlet
Dct	diameter of the capillary tube
Dsuctline	diameter of the suction line
numDPout	number of pressure steps in the adiabatic outlet section
numDTsl	number of suction line temperature steps in the heat exchanger section
numDPin	number of pressure steps in the adiabatic inlet section

Depending on how the RFSIM model is being used, some the above variables will be actually be known constants (Ks) and some will be residual variables (Xs).

Based upon the above input values, the subroutine will use the governing equations to calculate the following quantities: w_cap, Lout_cap, Lhx_cap, Lin_cap, p4_cap, and h4_cap. Again, the definitions of these variables are shown below.

w_cap	mass flow rate through the capillary tube and suction line
Lout_cap	length of the outlet adiabatic section of the capillary tube

Lhx_cap	length of the heat exchanger section (capillary tube and suction line)
Lin_cap	length of the inlet adiabatic region of the capillary tube
p4_cap	pressure at the outlet of the liquid line from the condenser
h4_cap	enthalpy at the inlet to the capillary tube

The suffix of “cap” is present on all the variables to indicate that they were calculated by the “ctslhx” subroutine and to distinguish them from the similar system variables. These six variables are used in the six residual equations that ensure that the solution returned by the “ctslhx” subroutine is consistent with the actual situation in the refrigerator.

In addition to the six residual equations just mentioned, there are two others that are related to the refrigerant state variables. These two equations were shown above in section B.1.6.1 during the discussion of the effectiveness method. The first of these two equations is shown again below.

```
1020   IF (CTSLHXSIM) then
         R(prop+50) = numDTsl*DTsl + t9 - t11
       ELSE
         R(prop+50) = ectslhx*(hpt(p11,t4) - h9) - (h11 - h9)
       END IF
        GOTO 1
```

This time, however, the CTSLHXSIM flag is true and the boldfaced equation is used. The equation serves as an alternative to the effectiveness equation and requires the temperature gain in the suction line (t11-t9) to be equal to number of DTsl segments (numDTsl) times the temperature step in the suction line (DTsl). The second of the two residual equations is also shown again below.

```
600    IF (CTSLHXSIM) then
           R(prop+29) = hpx(pcrit,xcrit) - hcrit
       ELSE
          R(prop+29) = (h4 - h7) - (h11 - h9)
       END IF
        GOTO 1
```

Again, the boldfaced equation is used when the CTSLHXSIM flag is true. This equation defines the enthalpy at the choked exit of the capillary tube based upon the critical pressure and quality. This enthalpy, in turn, is used to set the enthalpy at the inlet to the evaporator (h7).

The first of the six residual equations that uses the outputs from the “ctslhx” subroutine involves the mass flow rate of refrigerant through the system. It sets the mass flow rate calculated by the subroutine equal to the system mass flow (from the compressor map). This equation is shown below.

```
1760   IF (CTSLHXSIM) then
c      ===============================================================
         if(.not.NonZeroFlag) then
```

```
c       ------------------------------------------------------------
          Call ctslhx(pcrit,xcrit,DPout,DTsl,DPin,Dct,Dsuctline,t9,
     &       xoe,p9,numDTsl,numDPin,numDPout,Lin_cap,Lhx_cap,Lout_cap,
     &       p4_cap,h4_cap,w_cap,Tct,Tw,Tsl,P,h,Qual,Z,q,j)

c         ** The map mass flow must equal the captube mass flow when
c            CaptubeModel equals one **

          R(cap+0) = w_cap*CaptubeModel - w
c       ------------------------------------------------------------
        else
c       ------------------------------------------------------------
c         **This equation is used when the NonZeroList is being made.**

          R(cap+0) = pcrit+xcrit+DPout+DTsl+DPin+Dct+Dsuctline+t9+xoe+
     &     p9 + w + Lin + Lhx + Lout + h4 + p4
c       ------------------------------------------------------------
        end if
c     ==============================================================
      ELSE
c     ==============================================================
c       ** Dummy equation to keep CapTubeModel set to 1 when CTSLHXSIM
c          is false **
        R(cap+0) = CaptubeModel - 1.0
c     ==============================================================
      END IF
       GOTO 1
```

This residual equation is slightly more confusing than the other equations because it contains two "IF THEN-ELSE-END IF" statements. These statements dictates which equation is actually evaluated for the purpose of calculating the residual.

If the ct-slhx model is not being used (CTSLHXSIM = "false"), the third equation is used. This dummy equation simply keeps the value of the CaptubeModel variable equal to 1.0. If the ct-slhx model is being used (CTSLHXSIM = "true"), then either the first or second equation is used. The second equation is the dummy equation used when the NonZeroList is being built (NonZeroFlag = "true"). This equation allows the ACRC Solver to take partial derivatives much quicker than would be the case if the "ctslhx" subroutine was actually called. Therefore, much less time is required to build the NonZeroList. At all other times when the ct-slhx model is being used, the first equation is used. This equation, shown in boldface, requires the system mass flow rate (w) to be equal to the mass flow rate calculated by the "ctslhx" subroutine (w_cap) times the CaptubeModel parameter. Since this parameter should always be equal to 1.0, the equation is really setting the two mass flow rates equal to each other. The parameter CaptubeModel is only included in this equation and the third equation to provide a convenient means of switching between the effectiveness method and the ct-slhx model. The procedure of switching between the two methods of modeling the ct-slhx is discussed in A.2.3.

The other five residual equations also compare an output of the "ctslhx" subroutine with a system variable or parameter. The second residual equation is shown below.

```
1780   IF (CTSLHXSIM) THEN
c      ============================================================
         if(.not.NonZeroFlag) then
c        ----------------------------------------------------------
           Call ctslhx(pcrit,xcrit,DPout,DTsl,DPin,Dct,Dsuctline,t9,
     &       xoe,p9,numDTsl,numDPin,numDPout,Lin_cap,Lhx_cap,Lout_cap,
     &       p4_cap,h4_cap,w_cap,Tct,Tw,Tsl,P,h,Qual,Z,q,j)

c          ** The inlet length calculated by the ctslhx subroutine must
c             equal the actual inlet length **
           R(cap+1) = Lin - Lin_cap
c        ----------------------------------------------------------
         else
c        ----------------------------------------------------------
c          **This equation is used when the NonZeroList is being made.**
           R(cap+1) = pcrit+xcrit+DPout+DTsl+DPin+Dct+Dsuctline+t9+xoe+
     &       p9 + w + Lin + Lhx + Lout + h4 + p4
         end if
c        ----------------------------------------------------------
c      ============================================================
       ELSE
c      ============================================================
c        **  Dummy equation to give initial guess value to DPin when
c            the captube model isn't being used **
        R(cap+1) = ( (0.75*Lin/(Lin+Lhx+Lout))*(p4-pcrit) )/numDPin -
     &     DPin
c      ============================================================
       END IF
        GOTO 1
```

This equation, shown in boldface, requires the actual inlet length (Lin) to be equal to the inlet length calculated by the "ctslhx" subroutine (Lin_cap). Although the inlet length (Lin) is usually specified as a parameter (K), it can also be a variable (X). When the ct-slhx model is not used, the third equation is used for the residual evaluation. This equation gives a reasonable value to the variable DPin that will serve as the initial guess for the next time the ct-slhx model is used.

The remaining four residual equations are very similar the one just discussed. They all have one "real" equation, one equation for the NonZeroList, and one equation to provide an initial guess for one of the input variables of the "ctslhx" subroutine (pcrit, xcrit, DPout, DTsl, or DPin). To save space, only the "real" equations will be shown below.

```
R(cap+2) = Lhx - Lhx_cap
R(cap+3) = Lout - Lout_cap
R(cap+4) = h4 - h4_cap
R(cap+5) = p4 - p4_cap
```

As with the previous two residual equations, these residual equations require the values of the variables calculated by the "ctslhx" subroutine to be equal to the values present in the system model (either parameters or residual variables). When these six residual equations are satisfied, the input variables (pcrit, xcrit, DPout, DTsl, and DPin) describe the actual operation of the capillary tube-suction line heat exchanger within the refrigerator.

B.1.7 Evaporator equations

The equations that describe the evaporator are very similar to those for the condenser. The evaporator is also broken up into zones which are modeled as individual heat exchangers. The primary difference between the condenser and evaporator is that the latter will have at most two zones: a two-phase zone and a superheated zone. As with the condenser zones, the heat transfer in these two zones will be modeled with three equations: a refrigerant-side energy balance, an air-side energy balance, and an effectiveness-NTU rate equation. However, there is some added complexity in the implementation of these equations because of the geometry of evaporator in the Whirlpool refrigerator as mentioned in B.1.5.5. The need for the added complexity can be explained with the help of Figure B.2 which shows the geometry of the evaporator.

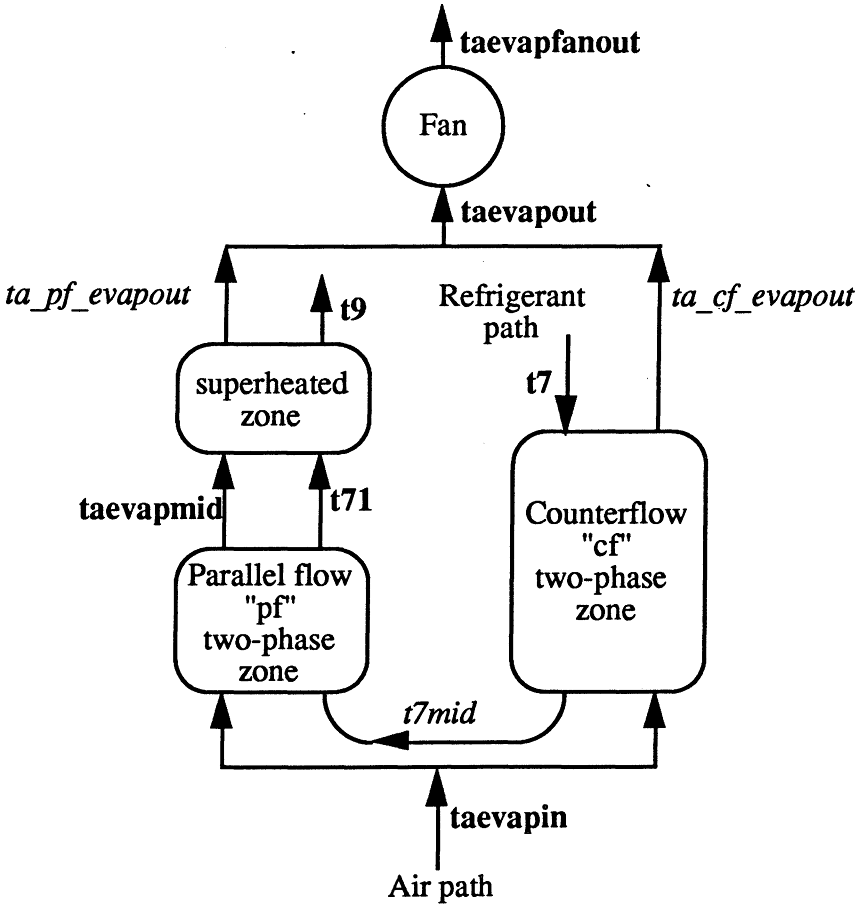


Figure B.2 Multi-zone evaporator modeling schematic

The difficulty with modeling this evaporator is primarily due to the fact that there are 2 two-phase zones: one in a counterflow configuration with the air stream, and the other in a parallel flow configuration with the air stream. Although the effectiveness of a two-phase zone is independent of the flow configuration, the fact that the two-phase zone exist on both "sides" of the evaporator does affect the air-side energy balances. For example, the temperature of the air entering the superheated zone (taevapmid), is obtained from an air-side energy balance around the parallel-flow two-phase zone only. Likewise, the temperature of the air leaving the counterflow side of the evaporator (ta_cf_evapout) is obtained from an air-side energy balance around the counterflow two-phase zone only. In the current model, the heat transfer in the 2 two-phase zones is calculated separately although there is only a single two-phase zone in the evaporator. This apparent contradiction will be better understood when the equations are discussed.

There is an alternative approach to modeling the evaporator in this situation. The two parts of the two-phase zone could be treated as individual zones, each with its set of three equations. This was the way that the condenser was modeled in the original version of RFSIM (Goodson and Bullard, 1994). That approach will work, but there is the undesirable "side-effect" of having an additional set of "two-phase" variables and equations. The main problem with having extra variables and equations is that the model becomes more difficult to modify for a new refrigerator that has different heat exchanger geometries. With the current approach, any new heat exchanger geometry would probably only require the modification of the existing residual equations. However, the older approach would almost certainly require the addition or subtraction of residual equations unless the new heat exchanger geometries were identical to the old ones. Although it is not impossible to add or subtract residual equations from the RFSIM model, it is definitely more difficult than modifying the existing equations. The intention of adopting the current heat exchanger modeling convention was to make the RFSIM model as standard as possible for future users.

Because of the added complexity of the current evaporator, all of the equations for each zone will be presented. However, the equations will be covered in less detail since much of the theory was already presented in the condenser section.

B.1.7.1 Refrigerant-side energy balances

The refrigerant-side energy balance for the two-phase zone is shown below.

```
C      ** Refrigerant side energy balance **
1980   IF (.not.NonZeroFlag) THEN
          IF (Evap2phX) THEN
             R(evap+4) = w*(h9 - h7) - q2phevap
          ELSE
             R(evap+4) = w*(h71 - h7) - q2phevap
          END IF
       ELSE
          R(evap+4) = w + h9 + h7 + h71 + q2phevap
       END IF
        GOTO 1
```

Whether the first or second equation is used will depend on whether the exit of the evaporator is two-phase or superheated. Both equations require the heat transfer to the refrigerant in the two-phase zone (q2phevap) to be equal to the mass flow rate (w) times the appropriate enthalpy difference. The refrigerant-side energy balance for the superheated zone has a similar form and is shown below.

```
C       ** Refrigerant-side energy balance **
2060    IF (.not.NonZeroFlag) THEN
           IF (Evap2phX) THEN
              R(evap+8) = 0.0 - qsupevap
           ELSE
              R(evap+8) = w*(h9 - h71) - qsupevap
           END IF
        ELSE
           R(evap+8) = w + h9 + h71 + qsupevap
        END IF
         GOTO 1
```

Again, the equation used will depend on the exit condition of the evaporator. If there is a two-phase exit (the first equation), then the heat transfer to the refrigerant in the superheated zone is set to zero. Otherwise, the heat transfer is set equal to the change in total enthalpy of the refrigerant stream. Both of the refrigerant-side energy balances (two-phase and superheated zones) will probably never have to be changed due to a change in the evaporator geometry since they involve only the refrigerant state points.

B.1.7.2 Effectiveness-NTU equations

The effectiveness-NTU equations for the two-phase zone are quite involved because of the two parts of the zone. For the remaining discussion, the counterflow and parallel flow part of the two-phase zone will each be referred to as a "zone" although there is really only one two-phase zone. The heat transfer from the air to the refrigerant for each zone is calculated through the use of the effectiveness-NTU equations and then added together to obtain the total two-phase heat transfer. All of these intermediate calculations and the residual equation are shown below.

```
2000
c       *Assume that the counterflow (cf) 2ph fraction will always be
c        equal to 0.5*
        f2ph_cf_evap = 0.5
        a2ph_cf_evap = aevap*f2ph_cf_evap
        f2ph_pf_evap = f2phevap - f2ph_cf_evap
        a2ph_pf_evap = f2ph_pf_evap*aevap

c       *The heat capacity of the cf and pf air streams will be
c        equal to 0.5 times the total heat capacity*
        cmin2phevap = 0.5*Caevap

c       *The effectivenesses for the cf and pf zones*
```

```
      e2ph_cf_evap = e2p(u2phevap*a2ph_cf_evap,cmin2phevap)
      e2ph_pf_evap = e2p(u2phevap*a2ph_pf_evap,cmin2phevap)

c     *The eff-NTU equation for the cf zone*
      q2ph_cf_evap = e2ph_cf_evap*cmin2phevap*(taevapin - t7)

c     *Assume that the 2ph pressure drop in the cf zone is
c      proportional to the cf fraction*
      p7mid = p7 - Dp2phEvap*(f2ph_cf_evap/f2phevap)

c     *Enthalpy at the end of the 2ph cf zone*
      h7mid = (q2ph_cf_evap/w) + h7

c     *Inlet refrigerant temperature for the pf 2ph zone*
      call TPHiter(p7mid,h7mid,t7mid)

c     *The eff-NTU equation for the pf zone*
      q2ph_pf_evap = e2ph_pf_evap*cmin2phevap*(taevapin - t7mid)

c     *The total 2ph heat transfer*
      R(evap+5) = q2ph_cf_evap + q2ph_pf_evap - q2phevap

       GOTO 1
```

The reduce the amount of text required to explain the equations, the actual comment statements present in the FORTRAN have been left with the equations. The first block of equations calculates the fractions and associated areas of the counterflow and parallel flow two-phase zones. It has been assumed that the fraction of the counterflow two-phase zone (f2ph_cf_evap) will always be equal to 0.5. This is equivalent to assuming that the counterflow side of the evaporator will always be two-phase.

The next equation is simply a statement that the minimum heat capacity of the two fluid streams (cmin2phevap) in the two-phase zones will always be equal to the one-half of the heat capacity of the air stream. The factor of 0.5 is present because half of the air flow rate will flow over each side of the evaporator.

The next two equations are calculations of the effectiveness of each two-phase zone (e2ph_cf_evap and e2ph_pf_evap) through the use of the "e2p" function call. The overall heat transfer coefficient (u2phevap) is calculated in another residual equation that calls the "UsEvap" subroutine. As described in the condenser section, the overall heat transfer coefficient for a zone depends on the refrigerant- and air-side heat transfer coefficients as well as the ratio of the air-side to refrigerant-side area. The "UsEvap" subroutine obtains the refrigerant-side coefficient through the use of the "h2phevapACRC" function call. This function returns an average heat transfer coefficient by integrating, over the specified quality range, the local two-phase evaporating heat transfer correlation developed at the ACRC by Wattelet and Chato (1994). Because u2phevap is calculated once and used for both two-phase zones (which have different quality ranges), there is probably a slight error in the two effectivenesses due to the dependence of the refrigerant-side heat transfer coefficient on the quality range chosen for the correlation.

The next equation calculates the heat transfer in the counterflow two-phase zone (q2ph_cf_evap) with the effectiveness-NTU equation. It is possible to perform this calculation at this point because the effectiveness (e2ph_cf_evap) and the temperatures of the inlet streams (t7 and taevapin) are known. However, the heat transfer in the parallel flow two-phase zone cannot be calculated because the inlet refrigerant temperature is not yet known. It would not be a bad assumption to set the temperature of the refrigerant at the inlet of the parallel flow two-phase zone (t7mid) equal to the temperature of the refrigerant at the inlet to the evaporator (t7). However, in order to make the equations general enough to account for any temperature glide due to pressure drop or the use of a zeotropic refrigerant, the temperature t7mid is calculated through the next three equations.

The first of these three equations calculates the refrigerant pressure (p7mid) by assuming that the pressure drop in the counterflow two-phase zone is equal to the fraction of the counterflow zone times the total two-phase pressure drop. The enthalpy at the midpoint (h7mid) is calculated with a refrigerant-side energy balance about the counterflow two-phase zone. With these values of pressure and enthalpy, the "TPHiter" subroutine is used to calculate the temperature of the refrigerant (t7mid) at the midpoint.

With this temperature, the next equation calculates the heat transfer from the air to the refrigerant in the parallel flow two-phase zone (q2ph_pf_evap) through the use of the effectiveness equation. Once this is done, the actual residual equation can be evaluated. This equation sets the heat transfer of the 2 two-phase zones equal to the total two-phase heat transfer (q2phevap).

The effectiveness-NTU equations are much simpler for the superheated region. The required intermediate calculations and the residual equation are shown below.

```
2080    asupevap = aevap*fsupevap

        if (Evap2phX) then
            cminsupevap = 0.9
        else
            cminsupevap = min((qsupevap/superheat),(0.5*Caevap))
        end if

        if (Evap2phX) then
            cmaxsupevap = 1.0
        else
            cmaxsupevap = max((qsupevap/superheat),(0.5*Caevap))
        end if

        esupevap =  ep(usupevap*asupevap,cminsupevap,cmaxsupevap)

        R(evap+9) = esupevap*cminsupevap*(taevapmid - t71) - qsupevap
         GOTO 1
```

As with the other examples shown, the first step is to calculate the area of the superheated zone (asupevap). Next, the minimum and maximum heat capacities of the two fluid streams in the superheated zone must be determined. If there is no superheated zone (Evap2phX = "true"), then the minimum and maximum heat capacity values are given nonzero dummy values so the effectiveness function calls (like "ep" in this case) will not crash. As with the two-phase zone, the value of the overall heat transfer coefficient (usupevap) is calculated in another residual equation through the used of the "UsEvap" subroutine. The effectiveness calculation for a parallel flow configuration is done by the "ep" function call located in the FUNCTION.f file. The residual equation requires the heat transfer predicted by the effectiveness-NTU method to be equal to the heat transfer from the air to the refrigerant in the superheated zone (qsupevap).

B.1.7.3 Air-side energy balances

The air-side energy balances for the parallel flow and counterflow two-phase zones actually require the use of the effectiveness-NTU method to obtain the heat transfer values. Essentially, either some or all of the intermediate calculations, shown in B.1.7.2, used to predict the entire two-phase heat transfer are repeated in these two energy balances. This is necessary because there are no residual variables that define the individual heat transfer for each two-phase zone like qsupevap does for the superheated zone.

The first such air-side energy balance is performed about the parallel flow two-phase zone for the purposes of calculating the inlet air temperature for the superheated zone (taevapmid). The intermediate calculations and residual equation are shown below.

```
2040    f2ph_cf_evap = 0.5
        f2ph_pf_evap = f2phevap - f2ph_cf_evap
        a2ph_cf_evap = aevap*f2ph_cf_evap
        a2ph_pf_evap = aevap*f2ph_pf_evap
        cmin2phevap = 0.5*Caevap
        e2ph_cf_evap = e2p(u2phevap*a2ph_cf_evap,cmin2phevap)
        e2ph_pf_evap = e2p(u2phevap*a2ph_pf_evap,cmin2phevap)
        q2ph_cf_evap = e2ph_cf_evap*cmin2phevap*(taevapin - t7)
        p7mid = p7 - Dp2phEvap*(f2ph_cf_evap/f2phevap)
        h7mid = (q2ph_cf_evap/w) + h7
        call TPHiter(p7mid,h7mid,t7mid)
        q2ph_pf_evap = e2ph_pf_evap*cmin2phevap*(taevapin - t7mid)

        R(evap+7) = 0.5*Caevap*(taevapin-taevapmid) - q2ph_pf_evap
         GOTO 1
```

The intermediate calculations are the exact same ones that were shown in B.1.7.2. The whole purpose of their repetition is to obtain a value for the heat transfer from the air to the refrigerant in the parallel flow two-phase zone (q2ph_pf_evap). Although this calculation was done in the other residual equation, the current value of q2ph_pf_evap is not available to this residual equation because of the sparse-Jacobian technique used by the ACRC Solver. In other words, every residual equation is considered a separate equation whose only inputs are the residual variables (Xs), parameters (Ks), and calculated values (Cs). This type of equation repetition is the "price" that has to be paid if a heat exchanger zone that really has two parts is to be modeled with one set of equations and variables. In any case, the calculated value of q2ph_pf_evap is set equal to the change in energy of the air stream as it flows across the parallel flow two-phase zone in the residual equation.

The other residual equation involving air-side energy balances is similar to the equation in the condenser because intermediate temperatures are calculated which are then averaged to get an outlet temperature of the entire heat exchanger. This time, however, the two intermediate temperatures come from air-side energy balances about the superheated zone and the counterflow two-phase zone. The intermediate calculations and the residual equation that define the outlet air temperature of the evaporator are shown below.

```
2120    a2ph_cf_evap = 0.5*aevap
        cmin2phevap = 0.5*Caevap
        e2ph_cf_evap = e2p(u2phevap*a2ph_cf_evap,cmin2phevap)
        q2ph_cf_evap = e2ph_cf_evap*cmin2phevap*(taevapin - t7)

        ta_cf_evapout = taevapin - q2ph_cf_evap/(0.5*Caevap)

        ta_pf_evapout = taevapmid - qsupevap/(0.5*Caevap)

        R(evap+11) = 0.5*ta_cf_evapout + 0.5*ta_pf_evapout - taevapout

         GOTO 1
```

The first four equations are repetitions that calculate the heat transfer from the air to the refrigerant in the counterflow two-phase zone (q2ph_cf_evap). This value is then used in an air-side energy balance to calculate the outlet air temperature from the counterflow side of the evaporator (ta_cf_evapout). The outlet air temperature from the parallel flow side of the evaporator (ta_pf_evapout) is also calculated from an air-side energy balance. The residual equation requires the outlet air temperature of the evaporator (taevapout) to be equal to the average of the two intermediate outlet temperatures.

B.1.7.4 Miscellaneous equations

As with the condenser, there are several additional equations used to model the evaporator: a calculation of the air-side evaporator area; a constraint on the two-phase and superheated fractions; a summation of the total evaporator heat transfer; a calculation of the heat capacity of the evaporator air stream; and an air-side energy balance about the evaporator fan. These equations are very similar to the condenser equations, and therefore, will not be shown here. However, a few words will be said about the evaporator fan energy balance. Since the fan is immediately downstream of the evaporator, the inlet temperature is the outlet temperature of the evaporator (taevapout). The temperature of the air leaving the evaporator fan (taevapfanout) is the temperature of the air that provides the cooling in the fresh food and freezer compartments as will be explained in the next section.

B.1.8 Cabinet and system equations

There are seven residual equations within the RFSIM model which relate the refrigerant system to the cabinets of the refrigerator and perform some system performance calculations. The first two equations, shown below, calculate the heat load on each of the compartments.

```
c**** The total heat transfer into the freezer compartment *************

2160    R(cab+0) = BTU(UAz)*(tamb - tafrez) + BTU(FrezHeater) - Qfrez
         GOTO 1

c**** The total heat transfer into the fresh food compartment **********

2180    R(cab+1) = BTU(UAf)*(tamb - tafrig) + BTU(FrigHeater) - Qfrig
         GOTO 1
```

The first term in each equation is the heat transfer through the cabinet walls represented by the "UA (ΔT)" expression. The second term in each equation represents any internal load that the user wishes to specify. As described in A.3.4.3, these terms are used primarily when the model's ability to match steady-state experimental data is tested.

If the RFSIM model was a dynamic model, then the above heat loads of each compartment (Qfrez and Qfrig) would be part of the differential equations that would describe how the temperature of each compartment (air plus contents) varied with time. Such equations would also depend on the amount of cooling that the evaporator provided to each compartment. However, since RFSIM is steady-state model, the two compartment air temperatures do not change with time. Therefore, the cabinet equations within the RFSIM model only match the physical situation in a real refrigerator when the cooling provided by the evaporator to each compartment is equal to the heat load of each compartment. Since this situation does not normally occur in a refrigerator, equations are used to calculate the relative sizes of the cooling capacity and heat load. The two equations that perform this function are shown below.

```
c**** The fraction of time that the system has to run to remove all the
c     heat added to the freezer compartment ****************************

2200   mdotaevap = vdotevap_calc*60/va(patm,taevapout)
       R(cab+2) = Qfrez/(mdotaevap*fz*(ha(tafrez)-ha(taevapfanout))) -
     &   RunTime
        GOTO 1

c**** The fraction of time that the system has to run to remove all the
c     heat added to the fresh food compartment *************************

2220   mdotaevap = vdotevap_calc*60/va(patm,taevapout)
       R(cab+3)=Qfrig/(mdotaevap*(1-fz)*(ha(tafrig)-ha(taevapfanout)))-
     &   RunTime
        GOTO 1
```

Together, these equations calculate the RunTime variable. The numerator of the first term in both equations is the heat load on the individual compartment. The denominator of the first term in both equations is the cooling rate for each compartment as defined by the appropriate mass flow of air times the change in enthalpy of the air stream. The appropriate mass flow rate of air into each compartment is determined by the split air fraction (fz). This variable is equal to the fraction of the total air stream that flows through the freezer compartment.

As the comment statements indicate, the RunTime is the fraction of time that the refrigerator must run at steady-state to remove the heat added to each compartment. It could also be thought of as a measure of the excess capacity of the evaporator. For example, if the RunTime was equal to 0.5, that would indicate that the cooling provided by the evaporator to each compartment was twice as large as the respective heat loads. Since the load ratios of both compartments are required to equal the same fraction (RunTime), the variable fz is also determined by these two equations. In other words, the "model" damper that controls the fraction of the total air stream that goes to each compartment is adjusted until the RunTime of each compartment is equal.

If other information is known about the refrigerator (such as the thermal mass of each compartment), then the RunTime variable could be used to extrapolate the steady-state results of the RFSIM model to actual cycling performance. For instance, the RunTime could be used to obtain a rough estimate of the time rate of change of the compartment temperatures or the cycle length.

The value of the split air fraction (fz) is used to calculate the temperature of the air entering the evaporator according to the following equation.

```
c**** The inlet evaporator air temperature *****************************

2240   R(cab+4) = fz*ha(tafrez) + (1-fz)*ha(tafrig) - ha(taevapin)
        GOTO 1
```

The implicit assumption made in this and the previous two equations is that the temperature of the return air from each compartment is equal to the average air temperature of each compartment. However, the effects of stratification within either compartment could be modeled by adding two additional parameters (Ks) that define the return air temperatures. The only problem with this additional complexity is that the user must specify two more parameters that may not be known.

The last two equations calculate variables that indicate the performance of the refrigerator system. The first equation, shown below, calculates the coefficient of performance (COP) of the system according to the classic equation from thermodynamics.

```
c**** Calculation of COP *********************************************
2260   R(cab+5)=(qevap-BTU(pevapfan_calc))/BTU(pcondfan_calc+
     &   pevapfan_calc+powercomp) - COP
        GOTO 1
```

The numerator of the first term is the net heat transfer from the air to the refrigerant in the evaporator. The heat of the fan is included because it dumps heat into the air stream after the air has been cooled by the evaporator. The denominator is equal to the total power (in terms of Btu/hr) that is required to run the refrigerator. The last equation is an approximation that calculates how much energy is used in a year's time if the steady-state refrigerator could "cycle" by turning on and off according to the RunTime variable. For example, if RunTime was equal to 0.5, then the model refrigerator could be imagined to run for 30 minutes and then shut off for 30 minutes for the entire year. Of course, the RunTime variable does not predict the cycle length, but it is still a useful measure of what fraction of the time the refrigerator would run. The energy equation is shown below.

```
c**** Calculation of the yearly steady-state energy usage **************

2280   R(cab+6) = (pcondfan_calc+pevapfan_calc+powercomp)*RunTime*365*
     &   24/1000 - Energy
        GOTO 1
```

The energy consumed in an entire year (Energy in kWh) is equal to the total power required to run the refrigerator times the RunTime times the number of hours in a year.

B.2 Model checking routines

The ACRC Solver provides a method of performing model checking or calculations before, during, and after the Newton-Raphson solution process. These functions are performed before, during, and after the solution process by the "IC", "BC", and "FC" subroutines located in the CHECKMOD.f file, respectively. Although these subroutines have to be present for the ACRC Solver to work, there are considered part of the "model" since they are model specific. If they are not needed, the subroutines can be left empty.

The primary purpose of these subroutines is to check the values of variables (X) and parameters (K) and set logical "flags" accordingly. This utility is especially useful in thermal system modeling involving heat exchangers since the phase of a heat exchanger exit (i.e. two-phase or superheated) can usually change depending on the operating conditions. In the RFSIM model, knowledge of the exit conditions of both heat exchangers is needed in order to use the right set of equations. The setting of logical flags such as the exit condition of a heat exchanger can be performed in the initial (IC) and/or boundary (BC) subroutines. The setting of a flag will not need to be done in the BC subroutine unless the particular flag can possibly change due to the change in variables that occurs during iterations. For example, the flag that indicates whether the capillary tube-suction line heat exchanger model is used within the RFSIM model is only set in the IC subroutine because it will not change during the solution process. The final checking (FC) subroutine can be used to warn the user if a particular variable value is outside of a desired range.

These checking subroutines are also very useful in performing explicit calculations or calling other routines. Where these calculations are performed (IC, BC, FC), will depend on the purpose of the calculations. If the calculated values are inputs to the model equations, they will be performed in the IC and/or BC subroutines. If the calculated values are outputs of the model, they will usually be performed in the FC subroutine.

The operation of these checking routines, from a users point-of-view, is covered in A.4.4. In the following sections, the mechanics of the checking routines will be covered in an outline format. The actual FORTRAN code will not be shown, but rather, the general logic and contents of each subroutine will be covered. The reader is referred to the FORTRAN listing for details of the implementation of the logic. Ample comment statements within the FORTRAN listing, along with following information, should make the checking routines understandable.

B.2.1 Initial checking (IC)

1. Information is obtained about the variables and parameters in the XK file.
   a) The XKflag value ("X", "K", or "C") is obtained for several quantities in the XK file. This information is used in the setting of logical flags in the IC subroutine.
   b) The indexes of the XK array (position within the XK file) are obtained for several variables. This information allows the subroutine to change the XKflag value of certain quantities in the XK file if necessary. For example, if the capillary tube-suction line heat exchanger model is being used, the XKflag of the effectiveness (ectslhx) is changed to a "C" because it will be calculated in the FC subroutine.
2. The value of the logical flag concerning the evaporator exit (Evap2phX) is set.
   a) If the superheat at the evaporator exit (superheat) is a "K" and the quality at the evaporator exit (xoe) is an "X", then the evaporator exit is <u>superheated</u> and <u>Evap2phX is given a value of "false"</u>. If the value of superheat is not greater than 0.0 °F, then it will be assigned a value of 1.0 °F.
   b) If superheat is a "X" and xoe is an "K", then the evaporator exit is <u>two-phase</u> and <u>Evap2phX is given a value of "true"</u>. If the value of xoe is less than 0.0 or greater than 1.0, then it will be assigned a value of 1.0.
   c) If superheat and xoe are both "Ks", then an <u>error</u> message is printed to the screen because the model will not operate correctly. Although it does not matter, Evap2phX is given a value of "true".
   d) If superheat and xoe are both "Xs", then the exit condition will be determined based upon their initial guess values.
      i) If superheat is greater than 0.0 °F and xoe equals 1.0, then the evaporator exit is <u>superheated</u> and <u>Evap2phX is given a value of "false"</u>.
      ii) If superheat equals 0.0 °F and xoe is between 0.0 and 1.0, then the evaporator exit is <u>two-phase</u> and <u>Evap2phX is given a value of "true"</u>.

iii) If the values of superheat and xoe do not fall into one of these two categories, a warning message is printed to the screen. The values of superheat and xoe are changed to 0.0 °F and 1.0, respectively. The evaporator exit is assumed to be two-phase and Evap2phX is given a value of "true".

3. The value of the logical flag concerning the condenser exit (Cond2phX) is set.

a) If the subcooling at the condenser exit (subcool) is a "K" and the quality at the condenser exit (xoc) is an "X", then the condenser exit is subcooled and Cond2phX is given a value of "false". If the value of subcool is not greater than 0.0 °F, then it will be assigned a value of 1.0 °F.

b) If subcool is a "X" and xoc is an "K", then the condenser exit is two-phase and Cond2phX is given a value of "true". If the value of xoc is less than 0.0 or greater than 1.0, then it will be assigned a value of 0.0.

c) If subcool and xoc are both "Ks", then an error message is printed to the screen because the model will not operate correctly. Although it does not matter, Cond2phX is given a value of "true".

d) If subcool and xoc are both "Xs", then the exit condition will be determined based upon their initial guess values.

i) If subcool is greater than 0.0 °F and xoc equals 0.0, then the condenser exit is subcooled and Cond2phX is given a value of "false".

ii) If subcool equals 0.0 °F and xoc is between 0.0 and 1.0, then the condenser exit is two-phase and Cond2phX is given a value of "true".

iii) If the values of subcool and xoc do not fall into one of these two categories, a warning message is printed to the screen. The values of subcool and xoc are changed to 0.0 °F and 0.0, respectively. The condenser exit is assumed to be two-phase and Cond2phX is given a value of "true".

4. The value of the logical flag indicating whether or not the capillary tube-suction line heat exchanger (ct-slhx) model is going to be used (CTSLHXSIM) is set.

a) If the XKflag of CaptubeModel is a "K", then the ct-slhx model will be used and CTSLHXSIM is given a value of "true". The XKflag of the effectiveness of the ct-slhx (ectslhx) is given a value of "C" since it will be calculated in the FC subroutine.

b) If the XKflag of CaptubeModel is a "X", then the ct-slhx model will not be used and CTSLHXSIM is given a value of "false". The XKflag of the effectiveness of the ct-slhx (ectslhx) is given a value of "K" since it will be used as the basis for the effectiveness method described in B.1.6.1.

5. The XKflag values (K or C) for the air-side convection coefficients of the condenser (haircond) and evaporator (hairevap) will be assigned.
   a) If the XKflag of haircond is not an "X", then it will be assigned a value of "K" or "C" depending on the value of the variable hcondNum. This variable indicates whether the model will use the user specified value of haircond (hcondNum=0) or use a correlation (hcondNum>0).
      i) If hcondNum is equal to 0, then the XKflag of haircond will be given a value of "K" since the value of haircond will be specified by the user.
      ii) If hcondNum is greater than 0, then the XKflag of haircond will be given a value of "C" since the value of haircond will be calculated by a correlation.
   b) If the XKflag of hairevap is not an "X", then it will be assigned a value of "K" or "C" depending on the value of the variable hevapNum. This variable indicates whether the model will use the user specified value of hairevap (hevapNum=0) or use a correlation (hevapNum>0).
      i) If hevapNum is equal to 0, then the XKflag of hairevap will be given a value of "K" since the value of hairevap will be specified by the user.
      ii) If hevapNum is greater than 0, then the XKflag of hairevap will be given a value of "C" since the value of hairevap will be calculated by a correlation.
6. Several calculated values (Cs) that appear in the XK file and are used in the residual equation are calculated. Typically, these values will only depend on parameters (Ks). However, if any of these calculated values depend on residual variables (Xs), then the values calculated here in the IC subroutine represent initial values since the residual variables will be changing every iteration.
7. The status of the three logical flags will be printed to the screen and to a file. This process is described in A.4.4.

B.2.2 Boundary checking (BC)

1. The XKflag value ("X", "K", or "C") is obtained for several quantities in the XK file. This information is used in the resetting of logical flags in the BC subroutine.
2. Some of the variables (Xs) related to the evaporator exit are checked to see if the evaporator exit logical flag (Evap2phX) should be changed. If the logical flag is changed, a message will be printed to the screen and to a file.
   a) If the evaporator currently has a two-phase exit (Evap2phX = "true"), and the value of xoe is greater than 1.0, and xoe is an "X", then the exit condition will be switched to a superheated exit. Evap2phX will be given a value of "false", and

several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

b) If the evaporator currently has a superheated exit (Evap2phX = "false"), and the value of superheat is less than or equal to 0.0 °F , and superheat is an "X", then the exit condition will be switched to a two-phase exit. Evap2phX will be given a value of "true", and several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

c) If the evaporator currently has a superheated exit (Evap2phX = "false"), and the value of the superheated fraction (fsupevap) is less than or equal to 0.0, and fsupevap is an "X", then the exit condition will be switched to a two-phase exit. Evap2phX will be given a value of "true", and several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

3. Some of the variables (Xs) related to the condenser exit are checked to see if the condenser exit logical flag (Cond2phX) should be changed. If the logical flag is changed, a message will be printed to the screen and to a file.

   a) If the condenser currently has a two-phase exit (Cond2phX = "true"), and the value of xoc is less than 0.0, and xoc is an "X", then the exit condition will be switched to a subcooled exit. Cond2phX will be given a value of "false", and several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

   b) If the condenser currently has a subcooled exit (Cond2phX = "false"), and the value of subcool is less than or equal to 0.0 °F , and subcool is an "X", then the exit condition will be switched to a two-phase exit. Cond2phX will be given a value of "true", and several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

   c) If the condenser currently has a subcooled exit (Cond2phX = "false"), and the value of the subcooled fraction (fsubcond) is less than or equal to 0.0, and fsubcond is an "X", then the exit condition will be switched to a two-phase exit. Cond2phX will be given a value of "true", and several other variables (Xs) will be assigned new values to help with the transition to the new exit condition.

4. Several calculated values (Cs) that appear in the XK file and are used in the residual equation are calculated. Although these values were calculated in the IC subroutine, some of them may need to be recalculated because they may be functions of residual variables (Xs) whose values can change every iteration.

B.2.3 Final checking (FC)

1. The saturation temperatures (tsat11 and tsat0) corresponding to the inlet and exit compressor pressures are compared with the range of evaporating and condensing temperatures used to

make the compressor map. If they (tsat11 and tsat0) are out of the range covered by the map data, a warning is printed to the screen.

2. The subroutine that calculates the irreversibilities of the major system components (Irrev in the EQNSUBS.f file) is called. The results of this subroutine appear as calculated values (Cs) in the XK file.
3. If the capillary tube-suction line heat exchanger (ct-slhx) model is used (CTSLHXSIM = "true"), the effectiveness of the ct-slhx is calculated. The purpose of the calculation is to provide a reasonable initial guess for the effectiveness (ectslhx) in case the ct-slhx model is not used at a later time. Of course, the user can always specify a value for ectslhx instead of using the value calculated by the ct-slhx model.
4. The average condensing (TcondAvg) and evaporating (TevapAvg) temperatures are calculated. These values appear as Cs in the XK file.
5. If specified by the user, the capillary tube profile data (temperature, pressure, quality, etc. vs. length) be printed to the screen after the solution is complete. The user can specify this output by giving the parameter CaptubeOutput a value of 1.0.

B.3 Supporting files

There are FORTRAN several files which contain subroutines and functions that are called by the residual equations in the EQNS.f file or the checking subroutines in the CHECKMOD.f file. Many of the routines contained in these files call other routines and are themselves called by other routines. These files are considered to be part of the model, not the solver, because they are specific to the RFSIM model. Currently, there are four such files which contain subroutines and functions: EQNSUBS.f, CAPTUBE.f, FUNCTION.f, and REFPROP.f. Although the division of the routines into the four files is somewhat arbitrary, the intent is to group routines together that have a similar purpose within the RFSIM model.

Some of the more important routines have already been discussed during the description of the governing equations in B.1. The following discussion of the routines contained in the four files will be limited to a brief description of the routine and a reference to another source of information if available. In particular, the contents of CAPTUBE.f and REFPROP.f are discussed in detail in other Appendixes. The FORTRAN listing of the all routines contain extensive comment statements describing the variables (input, intermediate, and output) and the general logic of the routine.

B.3.1 EQNSUBS.f

This file contains 10 routines (three subroutines and seven functions) that are used for calculations that are specific to the refrigerators modeled within ACRC Project 66.

Table B.1 Routines contained in EQNSUBS.f

Name	Description and Reference	Type
wf	Calculates the compressor map mass flow rate (lbm/hr)	Function
Pcompf	Calculates the compressor map power (Watts)	Function
UsCond	Returns the overall heat transfer coefficient (the "U" in "UA") for the zones in the condenser (Btu / hr-ft^2-°F) (Admiraal and Bullard, 1995)	Subroutine
UsEvap	Returns the overall heat transfer coefficient (the "U" in "UA") for the zones in the evaporator (Btu / hr-ft^2-°F) (Admiraal and Bullard, 1995)	Subroutine
haircnd	Calculates the air-side heat transfer coefficient of the condenser as a function of air velocity (Btu / hr-ft^2-°F) (Cavallaro and Bullard, 1994)	Function
hairevp	Calculates the air-side heat transfer coefficient of the evaporator as a function of air velocity (Btu / hr-ft^2-°F) (Cavallaro and Bullard, 1994)	Function
MassSingle	Calculates the refrigerant mass contained in single-phase component (lbm)	Function
MassCond	Calculates the refrigerant mass in the condenser (lbm) (Rice, 1987)	Function
MassEvap	Calculates the refrigerant mass in the evaporator (lbm) (Rice, 1987)	Function
Irrev	Returns the irreversibility, or lost work, of the major components in the refrigeration cycle	Subroutine

B.3.2 CAPTUBE.f

This file contains all the routines (subroutines) that are required to use the capillary tube-suction line heat exchanger model with the RFSIM model. More information about the following subroutines can be found in Appendix A.

Table B.2 Subroutines contained in CAPTUBE.f

Name	Description and Reference
ctslhx	This subroutine solves the discretized governing equations for the outlet, heat exchanger, and inlet sections of a capillary tube-suction line heat exchanger. It returns several variables that are used in residual equations in a "captube" model or an overall "system" model.
hxsolver	This subroutine solves the segments in the heat exchanger section of the capillary tube. It uses a two-dimensional Newton-Raphson routine to solve the equations. To evaluate the residuals, it calls three different subroutines depending on the state of the refrigerant in the captube and the suction line.
hxres2ph	Returns the residuals of the two equations needed in hxsolver when the capillary tube contains two-phase refrigerant and the suction line contains superheated vapor

hxresliq	Returns the residuals of the two equations needed in hxsolver when the capillary tube contains subcooled liquid refrigerant and the suction line contains superheated vapor
hxres2phsuct	Returns the residuals of the two equations needed in hxsolver when the capillary tube contains two-phase refrigerant and the suction line contains two-phase refrigerant
tpact	Solves an adiabatic capillary tube segment which contains only two-phase refrigerant (quality > 0.0)
ract	Solves an adiabatic capillary tube segment which contains saturated liquid at the inlet (quality = 0.0) and two-phase refrigerant at the exit (quality > 0.0)
spact	Solves an adiabatic capillary tube segment which contains only subcooled liquid
CaptubePrint	Prints, to the screen, the capillary tube-suction line heat exchanger variables as a function of length.

B.3.3 FUNCTION.f

This file contains several routines (all functions) that are used in the RFSIM model. These functions are more general than the routines in EQNSUBS.f and CAPTUBE.f and would be useful for other thermal models as well.

Table B.3. Functions contained in FUNCTION.f

Name	**Description and Reference**
BTU	Converts watts to BTU/hr
pi	Returns the number pi
epc	Calculates the effectiveness of a parallel-counterflow heat exchanger with the shell fluid mixed (Kays and London, 1984)
ec	Calculates the effectiveness of a counterflow heat exchanger (Incropera and DeWitt, 1990)
ep	Calculates the effectiveness of a parallel flow heat exchanger (Incropera and DeWitt, 1990)
e2p	Calculates the effectiveness of two-phase heat exchanger (Incropera and DeWitt, 1990)
cpa	Calculates the specific heat of air (Btu/lbm-°F)
ha	Calculates enthalpy of air (Btu/lbm)
SurfTen	Returns the surface tension of refrigerant (dynes/cm)
cpl	Calculates the specific heat of liquid refrigerant (Btu/lbm-°F)
cpv	Calculates the specific heat of vapor refrigerant (Btu/lbm-°F)
cpsup	Calculates the specific heat of superheated vapor refrigerant (Btu/lbm-°F)
ka	Calculates the thermal conductivity of air (Btu/hr-ft-°F)
kl	Calculates the thermal conductivity of liquid refrigerant (Btu/hr-ft-°F)
kv	Calculates the thermal conductivity of vapor refrigerant (Btu/hr-ft-°F)
mua	Calculates the viscosity of air (lbm/ft-hr)
mul	Calculates the viscosity of liquid refrigerant (lbm/ft-hr)
muv	Calculates the viscosity of vapor refrigerant (lbm/ft-hr)
prair	Calculates the Prandtl number of air
prl	Calculates the Prandtl number of liquid refrigerant
prv	Calculates the Prandtl number of vapor refrigerant

Pcritical	Returns the critical pressure of refrigerants (psia)
Mweight	Returns the molecular weight of refrigerants (g/mol)
reair	Calculates the Reynolds number of air
rel	Calculates the Reynolds number of liquid refrigerant
rev	Calculates the Reynolds number of vapor refrigerant
va	Calculates the specific volume of air (ft^3/lbm)
dpspHX	Calculates the single-phase pressure drop in a heat exchanger or refrigerant line (from the ORNL heat pump model - Fisher and Rice, 1983)
dp2phACRC	Calculates the two-phase pressure drop using correlations developed at the ACRC (Souza, 1995) (Christoffersen, 1993)
dptpHX	Calculates the two-phase pressure drop in a heat exchanger (from the ORNL heat pump model - Fisher and Rice, 1983)
dpsuct	Calculates the pressure drop in the suction line (from the ORNL heat pump model - Fisher and Rice, 1983)
Moody	Calculates the Moody friction factor for turbulent flow in rough pipes (Haaland, 1983)
Xtt	Calculates the Lockhart-Martinelli parameter (X_{tt})
Z	Calculates a frictional multiplier used in dptpHX (from the ORNL heat pump model - Fisher and Rice, 1983)
ZZ	Calculates a frictional multiplier used in dptpHX (from the ORNL heat pump model - Fisher and Rice, 1983)
fdeSouza	Calculates a “two-phase” Darcy turbulent-flow friction factor using the de Souza correlation (Souza, 1995)
fColebrook	Calculates a “two-phase” Darcy turbulent-flow friction factor by using the Colebrook correlation and a quality weighted average of the liquid and vapor friction factors. (Swamee and Jain, 1976)
fColebrookd	Calculates a “two-phase” Darcy turbulent-flow friction factor by using the Colebrook correlation and Dukler’s definition of “two-phase” viscosity. Colebrook(Swamee and Jain, 1976) with Dukler's (1964) viscosity
fBlasius	Calculates a single-phase Darcy friction factor for turbulent flow in smooth tubes using the Blasius correlation
hliq	Calculates the single-phase liquid heat transfer coefficient using the Dittus-Boelter correlation (Incropera and DeWitt, 1990)
hvap	Calculates the single-phase vapor heat transfer coefficient using the Dittus-Boelter correlation (Incropera and DeWitt, 1990)
hliq2	Calculates the single-phase liquid heat transfer coefficient using the Gnielinski correlation (Incropera and DeWitt, 1990)
hvap2	Calculates the single-phase vapor heat transfer coefficient using the Gnielinski correlation (Incropera and DeWitt, 1990)
h2phcondACRC	Calculates the average two-phase refrigerant condensing heat transfer coefficient using correlations developed at the ACRC (Dobson and Chato, 1994)
htpcondwavy	Calculates the local two-phase refrigerant condensing heat transfer coefficient in wavy-stratified flow for h2phcondACRC

h2phEvapACRC	Calculates the average two-phase refrigerant evaporating heat transfer coefficient using correlations developed at the ACRC (Wattelet and Chato, 1994)
Zivi	Calculates the local void fraction of two-phase flow using Zivi's correlation (Rice, 1987)
Hughmark	Calculates the local void fraction of two-phase flow using Hughmark's correlation (Rice, 1987)
Premoli	Calculates the local void fraction of two-phase flow using Premoli's correlation (Rice, 1987)
Razzak_Xtt	Calculates the local void fraction of two-phase flow using Razzak's Xtt-correlated void fraction correlation (Razzak, et. al. , 1995)
Razzak_drift	Calculates the local void fraction of two-phase flow using Razzak's drift-flux void fraction correlation (Razzak, et. al. , 1995)
KH	Calculates the flow parameter (Z) for the Hughmark void fraction correlation (Rice, 1987)

B.3.4 REFPROP.f

This file, along with REFPROP.INC and REFRIG.INC, contain all the FORTRAN necessary for the evaluation of thermodynamic properties of refrigerants in the RFSIM model. The theory and use of these thermodynamic property routines are covered in Appendix D.

Table B.4 Routines contained in REFPROP.f

Name	Description	Type
ref_init	Initializes the refrigerant data arrays	Subroutine
refdata_read	Reads the data in the refrigerant tables into data arrays	Subroutine
SaturationInt	Returns several saturation properties as a function of pressure	Subroutine
PsatT	Returns the saturation pressure (psia) as a function of temperature (°F) and quality (quality is required because of the possibility of zeotropes)	Function
TsatP	Returns the saturation temperature (°F) as a function of pressure (psia) and quality (quality is required because of the possibility of zeotropes)	Function
hpt	Returns the enthalpy (Btu/lbm) as a function of pressure (psia) and temperature (°F)	Function
hpx	Returns the enthalpy (Btu/lbm) as a function of pressure (psia) and quality	Function
htx	Returns the enthalpy (Btu/lbm) as a function of temperature (°F) and quality	Function
vpt	Returns the volume (ft^3/lbm) as a function of pressure (psia) and temperature (°F)	Function
vpx	Returns the volume (ft^3/lbm) as a function of pressure (psia) and quality	Function

vtx	Returns the volume (ft^3/lbm) as a function of temperature (°F) and quality	Function
vps2ph	Returns the two-phase volume (ft^3/lbm) as a function of the pressure (psia) and entropy (Btu/lbm-°F)	Function
spt	Returns the entropy (Btu/lbm-°F) as a function of pressure (psia) and temperature (°F)	Function
spx	Returns the entropy (Btu/lbm-°F) as a function of pressure (psia) and quality	Function
stx	Returns the entropy (Btu/lbm-°F) as a function of temperature (°F) and quality	Function
xhp	Returns the quality as a function of enthalpy (Btu/lbm) and pressure (psia)	Function
TPHint	Returns the temperature (°F) as a function of pressure (psia) and enthalpy (Btu/lbm) by inverse 2-D interpolation	Subroutine
TPHiter	Returns the temperature (°F) as a function of pressure (psia) and enthalpy (Btu/lbm) by iteration	Subroutine
TPSiter	Returns the temperature (°F) as a function of pressure (psia) and entropy (Btu/lbm-°F) by iteration	Subroutine
Terpl	Performs linear interpolation between two 1-D arrays	Subroutine
Terpd	Performs linear interpolation between two "columns" in a 2-D array	Subroutine
Terp2D	Performs two-way linear interpolation in a 2-D array	Subroutine
Invertpl2D	Performs inverse two-way linear interpolation in a 2-D array	Subroutine

References

Admiraal, D. M., and C. W. Bullard, "Heat Transfer in Refrigerator Condensers and Evaporators," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 101:1, 1995.

Cavallaro, A. R., and C. W. Bullard, "Effects of Varying Fan Speed on a Refrigerator/Freezer System," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 101:2, in press, 1995.

Christoffersen, B.R. and J.C. Chato, "Heat Transfer and Flow Characteristics of R-22, R-32/R-125 and R-134a in Smooth and Micro-Fin Tubes", ACRC TR-47, University of Illinois at Urbana-Champaign, 1993.

Dobson, M.K. and J.C. Chato, "Heat Transfer and Flow Regimes During Condensation in Horizontal Tubes", ACRC TR-57, University of Illinois at Urbana-Champaign, 1994.

Dukler, A.E., et al., "Pressure Drop and Hold Up in Two-Phase Flow," A. I. Ch. Journal, Vol. 10, pp. 38-51, 1964.

Fischer, S.K. and C.K. Rice, The Oak Ridge Heat Pump Models, Oak Ridge National Laboratory, ORNL/CON-80/R1, 1983.

Goodson, M.P., and Bullard, C.W., "Modeling a Refrigerator/Freezer System", ACRC TR-61, University of Illinois at Urbana-Champaign, 1994.

Grebner, J.J. and R.R. Crawford, "The Effects of Oil on the Thermodynamic Properties of Dichlorodifluoromethane (R-12) and Tetrafluoroethane (R-134a)", ACRC TR-13, University of Illinois at Urbana-Champaign, 1992.

Haaland, S.E., "Simple and Explicit Formulas for the Friction Factor in Turbulent Pipe Flow," J. Fluids Eng., March 1983, pp. 89-90.

Incropera, F., and DeWitt, D., Fundamentals of Heat and Mass Transfer, John Wiley and Sons, New York, 1990.

Kays, W.M. and A.L. London, Compact Heat Exchangers, New York: McGraw Hill Book Co., 1984.

Mullen, C. E. and C. W. Bullard, "Room Air Conditioner System Modeling", ACRC TR-60,University of Illinois at Urbana-Champaign, 1994.

Abdul-Razzak, A., et al.,"Characteristics of Refrigerant R-134a Liquid-Vapor Two-Phase Flow in a Horizontal Pipe," ASHRAE Transactions: Symposia, 12:1, 1995

Rice, C.K., "The Effect of Void Fraction Correlation and Heat Flux Assumption on Refrigerant Charge Inventory Predictions," ASHRAE Transactions, Vol. 93, Part 1, 1987, pp. 341-367.

Souza, A.L., and Pimenta, M.M.,"Prediction of Pressure Drop During Horizontal Two-Phase Flow of Pure and Mixed Refrigerants," ASME Symposium on Cavitation and Multiphase Flow, 210, pp. 161-171, 1995.

Stoecker, W. F., Design of Thermal Systems, 3rd ed., New York: McGraw-Hill, 1989.

Swamee, P.K., and Jain, A.K., "Explicit Equations for Pipe-Flow Problems," Proceedings of the ASCE, Journal of the Hydraulics Division, 102, HY5, p.657-664, May 1976.

Wattelet, J.P. and J.C. Chato, "Heat Transfer Flow Regimes of Refrigerants in a Horizontal-Tube Evaporator", ACRC TR-55, University of Illinois at Urbana-Champaign, 1994.

Appendix C

Capillary Tube - Suction Line Heat Exchanger Model

C.1 Introduction

The capillary tube-suction line heat exchanger (ct-slhx), shown in Figure C.1, is one of the four major components of the refrigeration system being modeled. This counterflow heat exchanger consists of a capillary tube that is soldered to the outside of the larger diameter suction line to the compressor. The capillary tube is broken up into three distinct sections for modeling purposes. First is the adiabatic inlet section, followed by the heat exchanger section with the suction line, and finally the outlet adiabatic section. The capillary tube is a very important part of the system because it must: 1) reduce the pressure of the liquid refrigerant, and, 2) regulate the flow of refrigerant to the evaporator. (Stoecker, 1986)

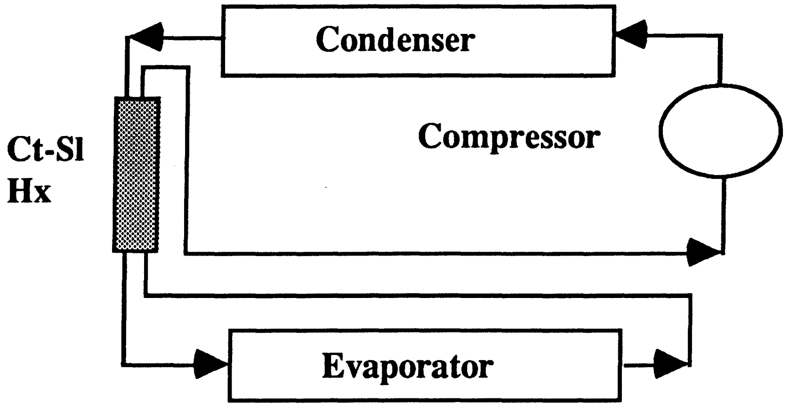


Figure C.1 Vapor compression cycle with ct-slhx

Capillary tube-suction line heat exchangers are very common in household refrigerators because they can increase system capacity by using the cold refrigerant in the suction line to lower the enthalpy of the fluid entering the evaporator, with only a modest increase in compressor power. Thus, there is a slight improvement in the system COP. It also has the added benefit of preventing liquid refrigerant from entering the compressor.

When operating in design mode, the ct-slhx model is not used because the system model assumes that the total amount of refrigerant and the sizing of the capillary tube are correct for that operating condition. However, the ct-slhx model is very important when running the system model in simulation mode. It enables the system model to be solved at a range of operating temperatures, with a fixed amount of charge, by calculating the mass flow rate through the capillary tube and the heat transfer to the suction line at off-design conditions.

C.2 Description of process

Despite its simple appearance, the capillary tube-suction line heat exchanger is a very difficult component to model. The refrigerant in the suction line is typically superheated vapor and is easily modeled as a single-phase fluid. The capillary tube, however, is difficult to model because some portion of it is non-adiabatic. Depending on the location of the heat exchanger and the operating conditions, there can be several different processes occurring in the capillary tube. For instance, if the refrigerant entering the capillary tube is subcooled, it might flash (begin to vaporize) in the inlet section, recondense in the heat exchanger section, and then flash again in the outlet section. Therefore, a model must have sufficient logic to solve operating points at off-design conditions in which different combinations of processes may be occurring. This section will explain all the possible processes which are accounted for as the refrigerant travels through each section of the capillary tube.

First consider the adiabatic inlet section of the capillary tube. The entering refrigerant, from the condenser, is either a two-phase mixture or a subcooled liquid. Regardless of the inlet state, the refrigerant pressure will decrease as it flows through the inlet section. For a two-phase inlet, this means that the temperature will also decrease, and the refrigerant quality will increase. If this is the case, then the entire adiabatic inlet section can be modeled as a two-phase mixture. If there is a subcooled inlet, the refrigerant temperature will remain constant, but the subcooling will decrease as the refrigerant pressure approaches the saturation pressure. At this point, the refrigerant begins to vaporize, and the remainder of the inlet section must be modeled as a two-phase region. The inlet section will usually have some two-phase portion because domestic refrigerators do not operate with large amounts of subcooling at the exit of the condenser when properly charged. However, it is the amount of subcooling and the length of the inlet section that determine whether the inlet section is totally or partially subcooled. Therefore, the model was written handle to handle any situation that might occur in the inlet section.

Therefore, the refrigerant entering the heat exchanger section may be a two-phase mixture or subcooled liquid. In addition to the pressure drop caused by the flow friction, heat is also being transferred from the refrigerant in the capillary tube to the refrigerant in the suction line. Because of the heat transfer, the stagnation enthalpy (static plus kinetic) of the capillary tube refrigerant will decrease. These two mechanisms, pressure drop and heat loss, are in opposition throughout the heat exchanger section. The pressure drop of the refrigerant tends to increase the quality (decrease subcooling) while the heat transfer tends to decrease the quality (increase subcooling). Because of these opposing forces, there are at least five possible scenarios in the heat exchanger section: 1) the refrigerant enters as two-phase mixture and it stays two-phase; 2) the refrigerant enters as two-phase mixture and it recondenses and exits as subcooled liquid; 3) the refrigerant enters as subcooled liquid and it stays subcooled; 4) the refrigerant enters as subcooled liquid and it flashes and exits as a two-phase mixture; and 5) the refrigerant enters as subcooled liquid and it flashes and then recondenses downstream and exits as a subcooled liquid. This last scenario is the most complicated and can occur in a situation where the pressure drop effect (e.g. small capillary tube diameter) is dominant in the upstream part of the heat exchanger and the heat transfer effect (e.g. very cold inlet suction gas) is dominant in the downstream part. All five of these scenarios can be handled by the current model.

Although it was previously stated that the refrigerant in the suction line is typically superheated vapor, it is possible that it may be a two-phase mixture for some portion of the suction line near the evaporator. Typically, refrigerator evaporators will only have a two-phase exit if the condenser has a two-phase exit. Furthermore, if the refrigerant entering the capillary tube is two-phase, it will enter the heat exchanger section with an even higher quality and will likely remain two-phase throughout the heat exchanger section. For this reason, and the increased complexity of the required logic, only one scenario for a two-phase inlet to the suction line is currently available in the model: the capillary tube contains two-phase refrigerant for the portion of the heat exchanger section occupied by the evaporating suction line refrigerant. This will be explained in more detail in later sections.

Finally, the refrigerant enters the adiabatic outlet section. This section is identical to the inlet except that the refrigerant is always exiting as a two-phase mixture at choked flow conditions. If the refrigerant enters the outlet section as a subcooled liquid, its subcooling will decrease to zero, it will flash, and then its quality will increase for the rest of the tube length. If the refrigerant is already two-phase when it enters the outlet section, its quality will increase for the duration of the section. When the quality of the refrigerant increases, its specific volume will also increase. Because the mass flow is constant, this increase in specific volume will cause an increase in the velocity of the refrigerant. Previous experiments on domestic refrigerators have shown that the velocity normally increases until critical flow is reached at the exit. At a fixed condenser pressure, further reductions of the evaporator pressure below this point will not increase the mass flow rate. Thus, it is assumed that there is a condition of choked flow at the exit of the outlet section. If the refrigerant flow is choked at the exit, there will be a discontinuity between the critical pressure at the capillary tube exit and the pressure at the inlet to the evaporator.

C.3 Description of model

The ct-slhx simulation model can be used as a stand-alone model for design purposes or as a component model within the overall refrigerator model. The slight differences in the two uses of the model will be described later in section C.4. The modeling equations used for this simulation code were taken from work which was done on capillary tubes at the ACRC (Peixoto and Bullard, 1994). The approach taken here builds on most of the assumptions and correlations that have been verified in the literature, and employs a solution technique that makes it unnecessary to assume a linear quality profile as other authors have done. The following sections state the assumptions, define the variables, explain the relationship between the model inputs and outputs, and list the governing equations.

C.3.1 Assumptions

The assumptions for adiabatic flow are:

a) negligible heat exchange with the ambient;

b) steady state, pure refrigerant one-dimensional flow;

c) homogeneous equilibrium two-phase flow;

d) critical conditions reached when Mach number of the homogeneous liquid and vapor mixture at exit of the outlet section is equal 1.0.

The additional assumptions for the heat exchanger section are:

e) negligible axial heat conduction in the capillary tube and suction line walls;

f) negligible thermal resistance in the capillary tube and suction line walls,

g) radially and axisymetrically isothermal capillary tube and suction line walls.

C.3.2 Diagram of model with variables and parameters defined

Figure C.2 defines the parameters and variables which are present in the ct-slhx model. Which of the shown variables are actually unknown and which are parameters is determined by how the model is used. The flashing point is shown to lie in the inlet section, but it could located in either of the other two sections or could be absent altogether in the case of a two-phase inlet condition.

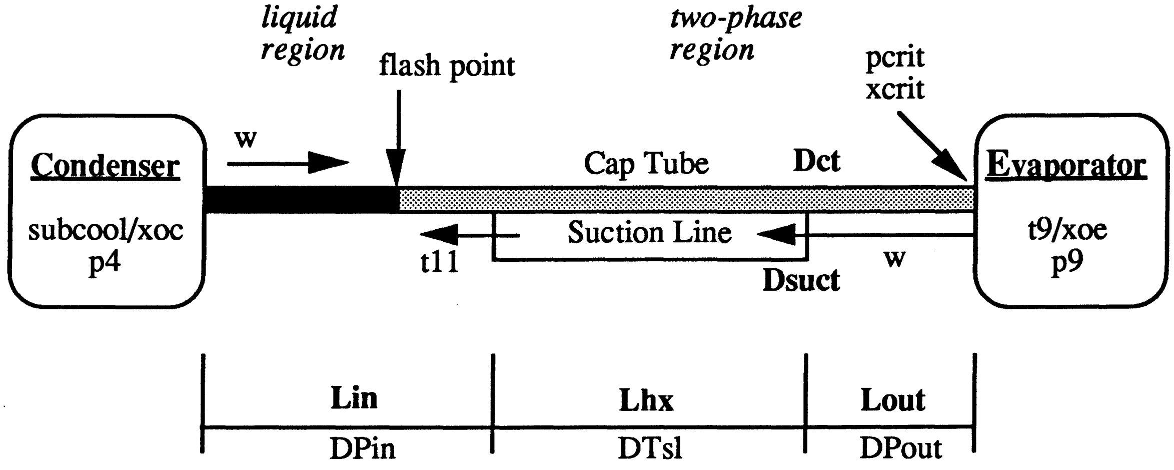


Figure C.2 Variables and parameters used in the ct-slhx model

Where the parameters and variables are defined as shown below:

Dct	Diameter of the capillary tube

Dsuct	Diameter of the suction line
Lin	Length of the adiabatic inlet section of capillary tube
Lhx	Length of the heat exchanger section of the ct-slhx
Lout	Length of the adiabatic outlet section of the capillary tube
DPin	Pressure steps in inlet section of capillary tube
DTsl	Temperature steps in suction line of heat exchanger section
DPout	Pressure steps in the outlet section of capillary tube
numDPin	Number of pressure steps to use in inlet section
numDTsl	Number of temperature steps to use in the heat exchanger section
numDPout	Number of pressure steps to use in the outlet section
subcool	Degrees of subcooling at exit of condenser
p4	Pressure at the exit of the condenser (or liquid line)
p9	Pressure at inlet of suction line
pcrit	Pressure at exit of capillary tube (choked flow)
t9	Temperature at inlet of suction line
t11	Temperature at the exit of the suction line
w_cap	Mass flow rate calculated by ct-slhx subroutine
xcrit	Quality at exit of capillary tube (choked flow)
xoc	Quality of refrigerant at the exit of the condenser
xoe	Quality of refrigerant at the exit of the evaporator

C.3.3 Inputs and outputs of the model

Before the governing equations or solution process are discussed, the inputs and outputs of the model will be presented as shown in Figure C.3 below.

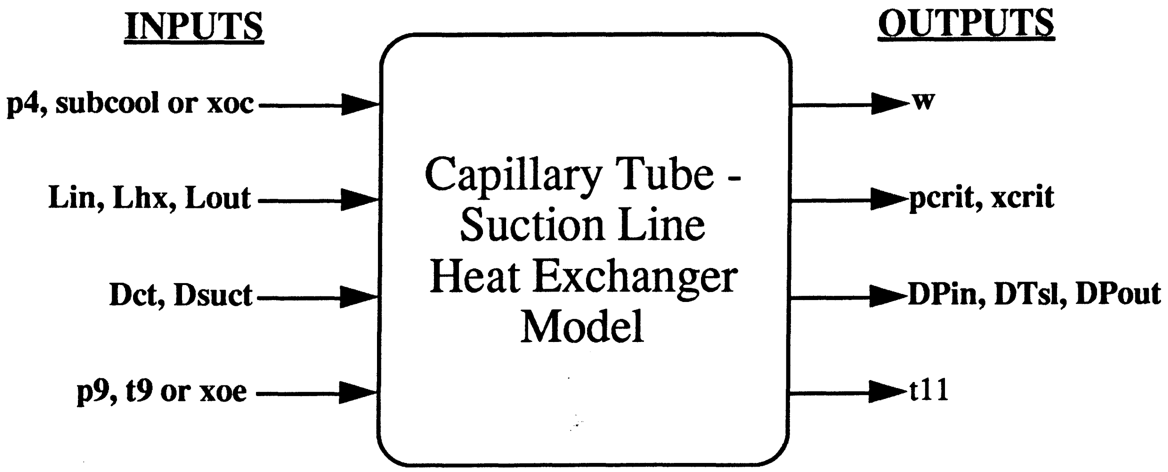


Figure C.3 Ct-slhx model inputs and outputs

This schematic shows the usual relationship between the model inputs and outputs. Typically, the refrigerant states at the inlet to the capillary tube (p4, subcool or xoc) and suction line (p9, t9 or xoe) as well as the geometry of the ct-slhx (Lin, Lhx, Lout, Dct, Dsuct) are considered inputs. The mass flow rate through the capillary tube (w), the refrigerant state at the capillary tube exit (pcrit and xcrit), and the exit temperature of the suction line (t11) are considered outputs.

Although not shown in Figure C.3, the number of steps in each section (numDPin, numDTsl, and numDPout) are also inputs to the model. The number of steps in each section times the solution values of DPin, DTsl, DPout give the total pressure change in the inlet and outlet sections and the temperature change in the suction line. For example, the outlet suction line temperature (t11) is equal to the inlet temperature (t9) plus numDTsl times DTsl. The use of the step variables DPin, DTsl, and DPout in the solution process will become clearer in Section C.5 when the solution of each segment is discussed.

Because the model is solved by the ACRC Newton-Raphson Solver, any of the inputs can be exchanged with any of the outputs shown in boldface. For example, the mass flow rate (w) can be given as an input and the model can be used to solve for the length of the heat exchanger (Lhx). However, the variable t11 can not exchanged with any of the inputs because none of the six simultaneous equations sent to the Newton-Raphson solver depends on this variable. In other words, t11 is calculated after the simultaneous Newton-Raphson solution is complete.

The previous discussion holds true whether the ct-slhx model is used by itself or with the overall system model, but there is a difference in the way that the inputs are defined. When the model is used by itself, the "Inputs" are user-supplied constants. However, when the ct-slhx model is used with the entire system model, the "Inputs" are system variables whose values will change every iteration until the ACRC Solver converges.

C.3.4 Governing equations

The equations which were used for this model are presented in this section. The governing equations for the adiabatic inlet and outlet sections of the capillary tube are identical. They consist of the mass, momentum and energy conservation equations, presented below:

$$\frac{\dot{m}_r}{A_{ct}} = G_{ct} = const. \tag{C.1}$$

$$\frac{-dp}{dx} = \frac{f\upsilon G_{ct}^2}{2D_{ct}} + G_{ct}^2 \frac{d\upsilon}{dx} \tag{C.2}$$

$$\frac{dh}{dx} = \frac{-G_{ct}^2}{2} \frac{d\left(\upsilon^2\right)}{dx} \tag{C.3}$$

Next are the governing equations used in the heat exchanger section. Once again the mass (C.4), momentum (C.6), and energy conservation (C.7) equations for the capillary tube are used. However, there is the addition of the mass (C.5) and energy rate equation (C.8) for the suction line, and a convective heat transfer balance for the two fluid streams (C.9). These six governing equations are listed below:

$$\frac{\dot{m}_r}{A_{ct}} = G_{ct} = const. \tag{C.4}$$

$$\frac{\dot{m}_r}{A_{sl}} = G_{sl} = const. \tag{C.5}$$

$$\frac{-dp}{dx} = \frac{f\upsilon G_{ct}^2}{2D_{ct}} + G_{ct}^2 \frac{d\upsilon}{dx} \tag{C.6}$$

$$\frac{dh}{dx} + \frac{G_{ct}^2}{2}\frac{d(u^2)}{dx} = \frac{dh_{sl}}{dx} \tag{C.7}$$

$$\dot{m}_r \frac{dh_{sl}}{dx} = -h_{sl}\pi D_{sl}(T_w - T_{sl}) \tag{C.8}$$

$$h_{ct}\pi D_{ct}(T_{ct} - T_w) - h_{sl}\pi D_{sl}(T_w - T_{sl}) = 0 \tag{C.9}$$

There is a pressure drop at the entrance of the capillary tube due to the abrupt contraction in tube size from the larger diameter liquid line tube to the capillary tube. It is calculated with Equation C.10. The entrance loss factor, K, is equal to 0.5 (Melo, 1992).

$$Dp = (1+K)\frac{V_{in}^2}{2\upsilon_{in}} \tag{C.10}$$

The thermodynamic properties for the refrigerant in the capillary tube and suction line are obtained by interpolation from tables. The table format, interpolation routines, and property routines were developed earlier at the ACRC for the purpose of detailed evaporator modeling. The property routines have been slightly modified by the author to make them more suitable to refrigerator modeling. The primary difference is in how the property routines respond when a property is requested outside of the table range. This happens more frequently in the capillary tube-suction line heat exchanger model than in general thermal system modeling. The current routines are designed to return values that will keep the model from crashing during iterations even if the values are not exactly correct. The intention is that as the iterations proceed, the property requests will return to the range covered by the tables.

Currently, the subcooled liquid enthalpy and volume are considered functions of temperature only and are evaluated as saturated liquid properties. The two-phase properties returned by the property routines are obtained by the standard quality-weighted average of the vapor and liquid properties shown in Equations C.11 to C.13. The superheated vapor properties used in the suction line are obtained by double interpolation the from the appropriate tables. The new interpolation routines are about five times as fast as the NIST REFPROP property routines that were originally used in the model. This is especially important in the ct-slhx model since there at least several hundred property calls every iteration.

$$h = (1-x)h_f + xh_g \tag{C.11}$$

$$s = (1-x)s_f + xs_g \tag{C.12}$$

$$\upsilon = (1-x)\upsilon_f + x\upsilon_g \tag{C.13}$$

The are some simplifying assumptions made in the heat transfer and pressure drop calculations in the capillary tube and suction line. Regardless of the assumptions made or the correlations used, all of the heat transfer and pressure drop calculations depend on the transport properties of the fluids being modeled. In the ct-slhx model, the required values of thermal conductivity, specific heat, and viscosity are calculated from cubic curve fits developed from data in the ASHRAE Fundamentals Handbook (1993).

Although there is usually single- and two-phase refrigerant in the capillary tube, the refrigerant in the capillary tube is modeled differently for the heat transfer and the pressure drop calculations. As far as heat transfer is concerned, the refrigerant in the capillary tube is modeled as a single-phase liquid. For the pressure drop calculations, on the other hand, an attempt was made to correctly model the difference between the single- and two-phase portions of the refrigerant flow in the capillary tube.

As mentioned, the refrigerant was assumed to be liquid for the heat transfer calculations in the capillary tube. This is done because most two-phase heat transfer correlations are not valid for qualities less than 0.05, which is the quality range for most of the two-phase refrigerant in heat exchanger section. In addition, the heat transfer coefficient of the capillary tube side is much higher than the suction line side so the overall heat transfer resistance will be dominated by the suction line side. Therefore, it does not really matter whether a single-phase liquid or a two-phase heat transfer correlation is used for the capillary tube. Although it was not critical, the Gnielinski correlation (C.16) was chosen as the single-phase heat transfer correlation because it was used in the suction line.

In order to correctly model the pressure drop in the capillary tube, an assumption had to be made about how to model the two-phase pressure drop. Although there are many different ways to do this, the approach taken in the ct-slhx model is to reduce the problem down to the choosing of a two-phase friction factor. The first term on the right-hand-side of the momentum equation (C.6) is a function of the friction factor. Almost invariably, the friction factor is dependent on the Reynolds number which in turn is a function of some viscosity. As will be discussed, one of the ways this can be done is to define a "two-phase" viscosity and then to use existing single-phase friction factor correlations.

The single-phase friction factor that is used in the momentum equations is obtained from Equation C.14. It is an explicit curve fit that approximates the transcendental Colebrook friction factor correlation for single-phase turbulent flow (Swamee and Jain, 1976).

$$f = 0.25\left[\log\left(\frac{e/D}{3.7} + \frac{5.74}{\mathrm{Re}^{0.9}}\right)\right]^{-2} \qquad (C.14)$$

Several different two-phase friction factors were examined as possible candidates for use within the model. For a complete discussion about the two-phase friction factors, see Appendix F. It was determined, from comparison with experimental data, that the Colebrook/Dukler two-phase friction factor correlation should be used in the ct-slhx model. The Colebrook/Dukler correlation uses Equation C.14, but the Reynolds number is determined from a "two-phase viscosity". This viscosity is determined from a void-fraction weighting of the liquid and vapor viscosities as proposed by Dukler (1964). The void-fraction weighting procedure can be seen in Equation C.15.

$$\mu = \frac{x v_v \mu_v + (1-x) v_l \mu_l}{v} \qquad (C.15)$$

Thus, the only thing different about the pressure drop in a single- and two-phase segment of the capillary tube is the friction factor. In a single-phase segment, the friction factor comes from Equation C.14 and the actual viscosity and Reynolds number. In a two-phase segment, the friction factor comes from Equation C.14 and the "two-phase" viscosity (C.15) and the resulting Reynolds number.

The situation in the suction line is just the opposite from the capillary tube in that the heat transfer calculations are more complicated and the pressure drop calculations are less complicated. The complication of the heat transfer comes from the fact that there is a significant difference between the heat transfer characteristics of high-quality evaporating flow and the superheated vapor that can exist in the suction line. The superheated portion and any two-phase portion of the suction line having quality greater than 0.95 is described by a single-phase heat transfer correlation. Any two-phase portion of the suction line having quality less than 0.95 is described by a two-phase evaporating heat transfer correlation.

The heat transfer in the capillary tube and in the superheated portion of the suction line is modeled as single-phase heat transfer. The single-phase heat transfer coefficients were calculated using the Gnielinski equation (Incropera and DeWitt, 1990) in Equation C.16. This correlation was chosen over the Dittus-Boelter equation (Incropera and DeWitt, 1990) and a heat transfer coefficient correlation by Sleicher and Rouse (1975). All three equations provided similar results, but the Gnielinski equation was chosen because it is purportedly more accurate in the ct-slhx model's range of operating conditions and it provided slightly better results when compared to experimental data.

$$Nu_D = \frac{(f/8)(\mathrm{Re}_D - 1000)\mathrm{Pr}}{1 + 12.7(f/8)^{0.5}\left(\mathrm{Pr}^{2/3} - 1\right)} \qquad \text{(C.16)}$$

$$f = \left(0.79 \ln \mathrm{Re}_D - 1.64\right)^{-2}$$

The heat transfer coefficient for the two-phase portion of the suction line is the same evaporation heat transfer correlation used for the refrigerator evaporator in the system model. It was developed from the work done by Wattelet and Chato (1994) at the ACRC on the evaporation heat transfer and pressure drop characteristics of existing and alternative refrigerants.

The modeling of pressure drop in the suction line is very simple because it is assumed that there is no pressure drop. This is not a bad assumption since the actual pressure drop in the entire suction line is on the order of 0.5 psi. It should be noted, however, that if the ct-slhx model is used as part of the overall system model, the pressure drop in the entire suction line is calculated elsewhere. However, this slight change in pressure is not included in the ct-slhx model to describe the state of the superheated vapor refrigerant in the portion of the suction line soldered to the capillary tube. In other words, for the purposes of the above mentioned heat transfer calculations, the state of the refrigerant in the suction line depends only on the local temperature and the "constant" pressure at the evaporator exit.

As mentioned, the refrigerant flow is assumed to be choked at the exit of the capillary tube. Based upon this assumption, the mass flow rate through the capillary tube can be calculated from the state of the refrigerant at the exit. From the momentum equation, it can be shown that choked or critical flow occurs when Equation C.17 is satisfied (Whalley, 1987).

$$1 + G_c^2\left(\frac{\partial v}{\partial p}\right)_s = 0 \qquad \text{(C.17)}$$

This equation is equivalent to the relation defining the speed of sound. When the equation is written explicitly in terms of G_C, as in Equation C.18, it can be clearly seen that the problem of determining the critical mass flux is one of proper evaluation of the partial derivative.

$$G_c = \left[-\left(\frac{\partial v}{\partial p}\right)_s\right]^{-0.5} \qquad \text{(C.18)}$$

How the partial derivative is evaluated depends upon the assumption made about the choked flow. The simplest assumption, and the one made in the current work, is that the flow at the exit of the capillary tube is homogeneous equilibrium flow. This allows the above partial derivative to be evaluated numerically with a standard thermodynamic property routine of the form shown in Equation C.19.

$$Volume = f(\mathrm{Pr}essure, Entropy) \tag{C.19}$$

One the critical mass flux is found, it is a simple matter to calculate the mass flow rate through the capillary tube and suction line according to Equation C.20.

$$\dot{m}_r = G_{crit} A_{cap} \tag{C.20}$$

C.4 Solution strategy for ct-slhx model

One possible solution method for the ct-slhx model is to divide the capillary tube into several segments and to solve all of the governing differential equations in a Newton-Raphson (NR) equation solver. However, there are two significant problems that are encountered if these discretized equations are placed in the system model along with the other component equations. Depending on the number of segments used to discretize the ct-slhx, the number of equations within the system model would increase dramatically. This would increase the number of equations that the equation solver must handle and significantly increase the execution time. Second, and more importantly, the solution would not be robust and would be awkward from a user's point of view. If all the discretized equations were solved simultaneously, initial guesses would be required to specify the refrigerant states along the capillary tube as well as the wall and refrigerant temperatures in the heat exchanger. These initial guesses would have to be provided by the user. This large number of Newton-Raphson variables would increase the chances that the model would not converge because of bad guesses.

In order to decrease the solution time and increase the robustness of the ct-slhx model, a solution process was developed that requires only six initial guesses. A schematic of this solution process is shown in Figure C.4. The ct-slhx is discretized and the governing differential equations for each segment are solved, in a subroutine, as sequentially as possible based upon some parameters and initial guess values for the Newton-Raphson variables. A Newton-Raphson routine is used to solve six residual equations that compare several of the results from the sequential solution with user specified parameters or other system variables. In other words, the subroutine "solution" guarantees that the governing differential equations are satisfied, and the Newton-Raphson guarantees that the "solution" from the subroutine matches the physical situation. The X's in Figure C.4 are the most common group of Newton-Raphson variables. As mentioned before, however, one or more of the X's shown in Figure C.4 could be switched with a known parameter such as Lin or p4.

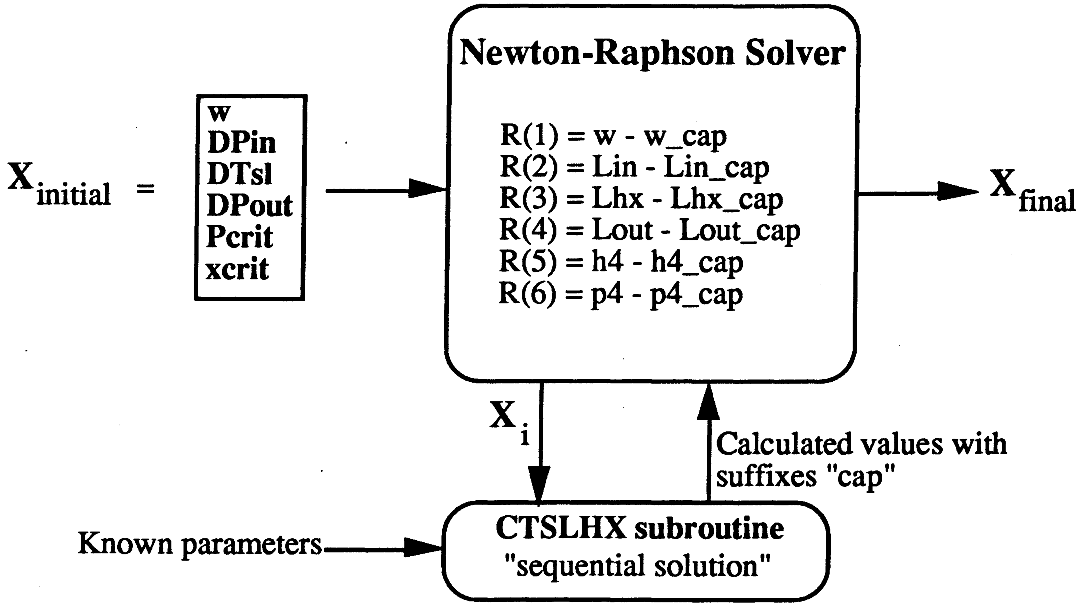


Figure C.4 The solution strategy for the ct-slhx model

The "sequential" subroutine uses five of the NR variables shown above (pcrit, xcrit, DPin, DTsl and DPout), along with some of the known parameters (Dct, Dsuct, p9 and t9 or xoe) to march backwards through the ct-slhx and solve each segment sequentially. The subroutine is able to do this because the starting state and mass flow rate are known from pcrit and xcrit, and the steps used to discretize the governing differential equations in the three sections are known from DPout, DTsl, and DPin. The "sequential" solution of the governing differential equations in the inlet and outlet sections require the iterative solution of one implicit equation, and the solution for the heat exchanger segments requires the iterative solution of two implicit equations. The "sequential" solution process will be described in detail in later sections. Because of the sequential solution, all of the refrigerant states within the capillary tube and suction line are explicit intermediate variables that do not require initial guesses.

The ctslhx subroutine returns several values that are used in the residual equations: w_cap, Lin_cap, Lhx_cap, Lout_cap, P4_cap, and h4_cap. The suffix "cap" indicate values calculated by the ctslhx subroutine. The dependence of these six variables on the NR variables and parameters is shown symbolically by the following:

$w\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

$Lin\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

$Lhx\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

$Lout\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

$P4\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

$h4\_cap = f_{ctslhx}(p9,t9,xoe,Dct,Dsuct,pcrit,xcrit,DPin,DTsl,DPout)$

Each of the six residual equations is a comparison of the above ctslhx subroutine outputs to either user-specified parameters or variables present in the system model. The first equation compares the refrigerant mass flow rate predicted by the capillary tube (w_cap) to the system mass flow rate variable (w).

```
R(1) = w_cap - w
```

This is a very crucial equation in the system model since the ct-slhx model, along with the compressor model, determine the mass flow rate of the entire system. There is only one value for mass flow rate where operation is steady-state because there is no mass accumulation anywhere in the system. The precise method in which mass flow rate is calculated is explained in section C.5.1.

It should be noted, however, that this residual equation does not affect the solution when the six residual equations are used as a stand alone capillary tube model and the mass flow rate is an unknown. In this situation, the NR solver makes the above residual equation go to zero by changing the value of w. Every iteration, w_cap will change because pcrit and xcrit will change, and the value of w will be adjusted to something close to w_cap. This process will continue until w_cap stops changing, and w equals w_cap. Essentially, this residual equation is unnecessary when solving for w in the stand alone ct-slhx model. The problem could be solved by a five-variable NR instead of a six-variable NR. The mass flow rate, w, would could be determined explicitly from the solution values of pcrit and xcrit. However, the equation is necessary when the ct-slhx model is used as part of the overall refrigerator model, and it allows the model to be used in stand-alone mode as a design tool. For example, the heat exchanger length, Lhx, needed for a specified mass flow rate and inlet conditions can be determined by making w a known and Lhx an unknown. This would not be possible if w were not included in the simultaneous set.

The second equation is a comparison of the user-defined length of the inlet section (Lin) to the length which is calculated by the subroutine (Lin_cap).

```
R(2) = Lin - Lin_cap
```

Depending on the values of the subroutine input variables, the ct-slhx model calculates a length for this section, but it does not necessarily match the actual length. Therefore, the input variables are adjusted by the NR solver until the calculated length is equal to the actual length of the inlet section.

Similar to the last equation, the third and fourth equations are length comparisons, but this time they are for the heat exchanger and outlet sections, respectively.

```
R(3) = Lhx - Lhx_cap
```

```
R(4) = Lout - Lout_cap
```

Where Lhx and Lout are the user-defined lengths and Lhx_cap and Lout_cap are the calculated lengths.

The fifth and sixth equations are more easily explained together since they both involve the refrigerant state at the outlet of the condenser. This state will be slightly different than the state at the inlet to the capillary tube because of the sudden contraction pressure drop described by Equation C.10. Although the fifth residual equation doesn't change because of the exit condition of the condenser, the method of calculating the enthalpy (h4) does change. If the refrigerant exiting the condenser is a two-phase mixture, then Cond2phX is true and enthalpy is a function of pressure and quality. If the condenser exit is subcooled, then Cond2phX is false and enthalpy is a function only of the temperature. When the ct-slhx model is used within the system model, this logic is unnecessary because the enthalpy (h4) is calculated elsewhere in the residual equations.

```
IF (Cond2phX) THEN
   h4 = hpx(P4,xoc)
   R(5) = h4 - h4g
ELSE
   t4 = TsatP(P4,0.0d0) - subcool
   h4 = htx(t4,0.0d0)
   R(5) = h4 - h4g
END IF

R(6) = P4 - P4_cap
```

The sixth residual equation compares the exit pressure of the condenser that is calculated by the ctslhx subroutine (P4_cap) with the actual exit pressure of the condenser (P4). Together, the fifth and sixth residual equations compare the predicted refrigerant state at the capillary tube inlet with the state prescribed by the user or the system model.

Therefore, the governing differential equations for the ct-slhx are located in a subroutine rather than in the system model with the other governing equations. This helps to keep the number of equations in the system model to a manageable size and minimizes the number of initial guesses required by the Newton-Raphson (NR) equation solver.

C.5 Solution algorithm for the ctslhx subroutine

The manner in which the ctslhx subroutine "sequentially" solves the governing differential equations for each segment is explained in more detail below for each of the three sections of the ct-slhx. Since there are many possible modes of operation for the ct-slhx, as mentioned in section C.2, this discussion will assume a common mode of operation where the refrigerant flashes in the inlet section, recondenses in the heat exchanger section, and then flashes again in the outlet section. Also, the number of segments in each section will be fixed at three for explanation purposes. In general, four or five steps are used for each section, but this can be changed (by changing the numDPin, numDTsl, and numDPout parameters in the XK file) if warranted by the geometry of the capillary tube.

The solution begins by starting at the outlet of the capillary tube and solving for the state of the refrigerant at the beginning of each preceding segment by taking pressure steps backwards towards the heat exchanger section. Once in the heat exchanger section, the states at each segment are found by taking temperature steps along the suction line. Finally, pressure steps are once again taken until the state of the refrigerant at the entrance of the capillary tube is calculated.

C.5.1 Determination of mass flow rate

The first objective is to determine the mass flow rate of refrigerant, which will facilitate the solution of each segment. Referring back to the governing equations in section C.3.4, a choked flow condition is assumed to be present at the exit of the capillary tube. By using the initial guesses for the state of the refrigerant at this critical condition (pcrit and xcrit), Equation C.20 can be used to calculate the mass flow rate as a function of the critical mass flux (G_{crit}), calculated in Equation C.18. These choked flow equations assume that the two refrigerant phases are in homogenous equilibrium at the capillary tube exit. The mass flow rate that is calculated by these equations is the w_cap variable that is used in the first residual equation.

Now that the mass flow rate has been determined, it is possible to begin marching backwards up the capillary tube to solve each individual segment. In the overall system model, the state of the refrigerant at the exit of the capillary tube is related to the state of the refrigerant at the entrance to the evaporator by the assumption that there is an isenthalpic (constant enthalpy) expansion between the two states. Therefore, the pressure at the exit of the capillary tube (pcrit) will always be higher than the evaporating pressure, and the quality at the exit of the capillary tube (xcrit) will be lower than the quality at the entrance to the evaporator, since enthalpy remains constant.

C.5.2 Adiabatic outlet section

When the solution process starts at the capillary tube exit, there is no way to predict whether the outlet section will be completely two phase or if there will be a subcooled portion. In the present case shown below in Figure C.5, the refrigerant recondenses in the heat exchanger section and flashes in the outlet section. The ctslhx subroutine has to be able to distinguish between the two-phase portion and the liquid portion in the outlet section because each segment must be either totally two-phase or subcooled from a modeling point of view. The model does this by solving for the point of zero quality within the outlet section. The DPout segment in which this point is found is divided into two smaller segments: a two-phase portion and a single phase portion. However, the combined pressure drop of these two smaller segments will equal the DPout value. In this case were the refrigerant flashes in the outlet section, there will be four total segments in the outlet section, but only three DPout segments. This will become clearer as the solution process is further explained.

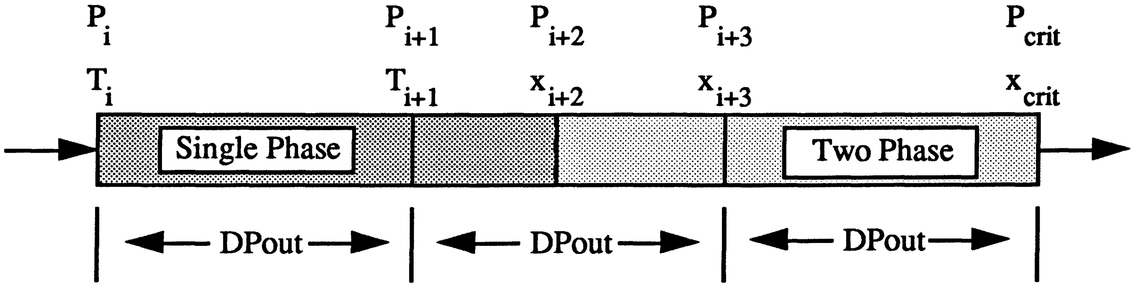


Figure C.5 The adiabatic outlet section of the capillary tube

Every segment, regardless whether it is two phase or subcooled, is described by the three governing equations listed in section C.3.4 (Equations C.1 - C.3). Before the subroutine solves any of the DPout segments, it must determine if there is transition between subcooled and two-phase refrigerant in the outlet section. This can be done by using the discretized form of the energy equation as shown in Equation C.21.

$$(h_{crit} - h_{i+2}) = \frac{-G_{ct}^2}{2}(v_{crit}^2 - v_{i+2}^2) \qquad \text{(C.21)}$$

The energy equation is a function only of the thermodynamic states at the endpoints of the segment. For the first calculation, the segment extends from the end of the capillary tube to the point of zero quality. Therefore, the pressure corresponding to this zero quality can be determined by solving the energy equation with a one-dimensional Newton-Raphson routine, assuming that the critical exit condition is known. Once this pressure is found, the segment in which the transition occurred can be determined. In this situation, if P_{i+2} (the zero-quality pressure) is greater than P_{i+3} (P_{crit} + DPout) and less than P_{i+1} (P_{crit} + 2*DPout), then the flashing occurs in the second DPout segment. If the zero-quality pressure is greater than P_i (P_{crit} + 3*DPout), then there is no transition in the outlet section and all the segments are solved as two-phase segments.

Since the flashing occurred in the second DPout segment, the first DPout segment will be solved as two-phase segment. "Solving" involves determining the segment inlet conditions (P_{i+3} and x_{i+3}) and the length from the discretized momentum and energy equations shown in Equations C.22 and C.23, respectively. The inlet pressure (P_{i+3}) is known from the critical pressure and the DPout pressure step. The DPout variable is known from the initial guesses during the first iteration, and its value is changed every iteration by the ACRC Solver. The quality (x_{i+3}) is found as P_{i+2} was found before with the energy equation. The solution method is the same, but x_{i+3} is the variable that the Newton-Raphson routine iterates until the equation is satisfied.

$$(P_{i+3} - P_{crit}) = \frac{L_{i+3} f_{avg} \upsilon_{avg} G_{ct}^2}{2D_{ct}} + G_{ct}^2(\upsilon_{crit} - \upsilon_{i+3}) \qquad \text{(C.22)}$$

Where L_{i+3} equals the length from the i+3 to the exit.

$$(h_{crit} - h_{i+3}) = \frac{-G_{ct}^2}{2}(\upsilon_{crit}^2 - \upsilon_{i+3}^2) \qquad \text{(C.23)}$$

Once the inlet state (P_{i+3} and x_{i+3}) is known, the length of the segment is determined explicitly from the momentum equation (C.22). This length calculation will be added to the lengths of the other segments of the outlet section to produce the Lout_cap variable that is used in one of the residual equations.

The two-phase portion (i+2 to i+3) of the second DPout segment will be solved by determining the pressure P_{i+2} and the length of the segment as mentioned before. Although this pressure that corresponds to zero quality was determined before, the purpose at that time was to find which, if any, DPout segment contained the flash point. The current numerical value for this pressure will be slightly different from the previous value because the discretized form of the differential equations extends from i+2 to i+3 instead of i+2 to the critical point. In other words, the accuracy of the numerical solution will increase as the length of the discretized segment decreases.

The remainder of the second DPout segment (i+1 to i+2) will be solved as a subcooled segment using the discretized momentum and energy equations as shown in Equations C.24 and C.25, respectively.

$$(P_{i+1} - P_{i+2}) = \frac{L_{i+1} f \upsilon G_{ct}^2}{2D_{ct}} \qquad \text{(C.24)}$$

$$(\mathrm{h}_{i+2} - \mathrm{h}_{i+1}) = 0 \qquad \text{(C.25)}$$

The specific volume terms seen in Equations C.22 and C.23 have been eliminated because the subcooled liquid is incompressible. The energy equation can be used to solve explicitly for the temperature (T_{i+1}) using the assumption that the enthalpy of subcooled liquid close to the saturated liquid line is a function only of temperature. Once again, the length will be obtained explicitly from the momentum equation. The last DPout segment (i to i+1) is also subcooled and will be solved in exactly the same way.

When all the segments in the outlet section (2 subcooled and 2 two-phase) are solved, the lengths of every segment are summed to get Lout_cap. This length is calculated based upon the assumed values of DPout, Pcrit, and xcrit. The residual equation that compares Lout_cap and

Lout is part of the overall Newton-Raphson solver that determines what the new guesses should be in order that Lout_cap approach Lout.

Now that the inlet state to the outlet section is calculated it is possible to continue to the next section. Unfortunately, the heat exchanger section is a somewhat more complicated matter.

C.5.3 Heat exchanger section

The solution strategy for the heat exchanger section is similar to that for the adiabatic outlet section. The solution starts at the end of the heat exchanger section and solves the segments backwards until the beginning of the section. The primary difference between the two sections is that the suction line temperature (DTsl) is used discretize the governing differential equations instead of pressure, for reasons that will soon become clear. In the present case shown below in Figure C.6, the refrigerant enters the heat exchanger section as two-phase mixture and then recondenses because of the heat transfer to the suction line.

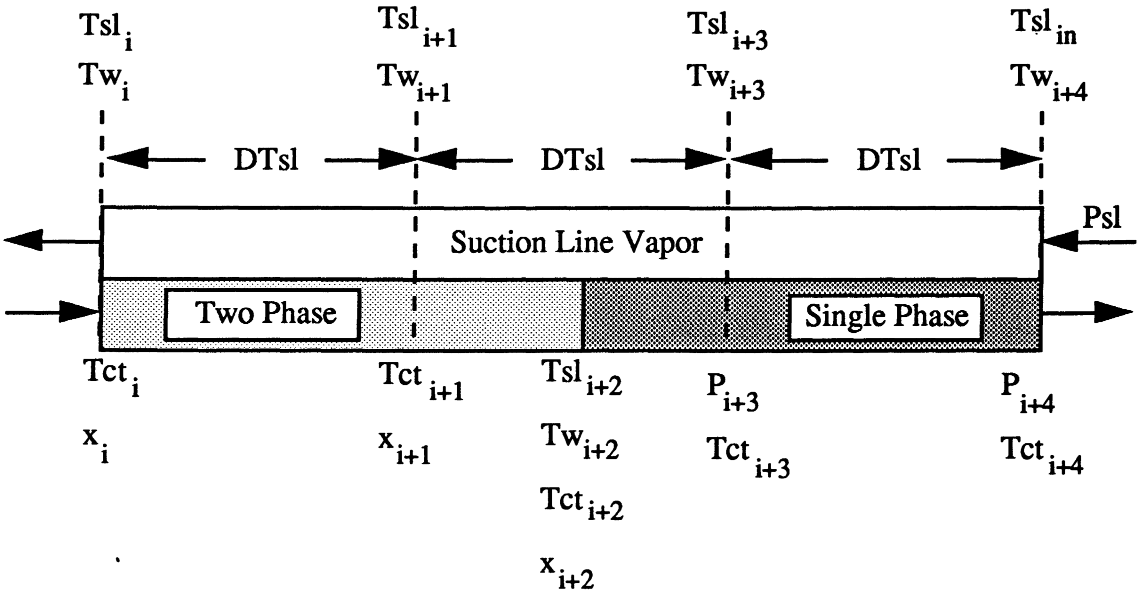


Figure C.6 The heat exchanger section of the ct-slhx

As in the outlet section, the model does not know if there is a transition between two-phase and subcooled refrigerant within the heat exchanger section of the capillary tube. In this case, all that is known from the outlet section solution is that the refrigerant is a subcooled liquid at the heat exchanger exit (P_{i+4} and Tct_{i+4}). The inlet suction line temperature (Tsl_{in} = t9) and pressure (Psl = p9) is also known from the exit conditions of the evaporator. As was previously stated in the governing equations (section C.3.4), the refrigerant in the suction line is assumed to be predominantly superheated vapor with negligible pressure drop. Therefore, the pressure (P_{sl}) is constant throughout the suction line, and only the temperature is needed to specify the state in the superheated portion of the suction line.

Because there is negligible thermal resistance in the capillary tube or suction line wall, there is only one variable used to describe the wall temperature at a point (Tw). Therefore, at a particular point along the ct-slhx, the refrigerant in both streams "see" the same wall temperature around their entire circumferences. This wall temperature varies along the heat exchanger section just as the suction line and capillary tube temperatures vary. The mass flow rate in the suction line is the same as that in the capillary tube because there is no mass accumulation in the evaporator during steady-state operation.

Before the model "solves" any segments in the heat exchanger section, it must first determine if, and where, there is a transition from the subcooled liquid at the exit to a two-phase mixture upstream. All of the variables at the exit (i+4) are known except Tw_{i+4}. This unknown is determined from the discretized form of Equation C.9 shown below as Equation C.26.

$$hct_{i+4}\pi D_{ct}(Tct_{i+4} - Tw_{i+4}) = hsl_{i+4}\pi D_{sl}(Tw_{i+4} - Tsl_{i+4}) \qquad \text{(C.26)}$$

The hct and hsl terms are convection heat transfer coefficients, not enthalpies. This equation states that the heat transfer from a differential area of the capillary tube, near the exit of the heat exchanger section, is equal to the heat transfer to the suction line over the same differential area. Since everything else is known in the equation, Tw_{i+4} can be explicitly determined. Although the refrigerant at the exit is a subcooled liquid, it is not known if the entire DTsl segment (i+3 to i+4) is liquid. To find out if the zero-quality point occurs in this last DTsl segment, the suction line temperature associated with the zero-quality point (Tsl_{i+2}) is determined. If this temperature is less than Tsl_{i+3} (Tsl_{in} + DTsl) then the recondensation occurs in the last segment. In the present case, Tsl_{i+2} is greater than the Tsl_{i+3}, so the transition between the two-phase and liquid refrigerant did not occur in last DTsl segment (i+3 to i+4). The temperature, Tsl_{i+2}, is determined by "solving" the segment from i+2 to i+4.

"Solving" the all-liquid segment (i+2 to i+4) involves determining the four unknowns associated with this segment: Tsl_{i+2}, Tw_{i+2}, Tct_{i+2}, and the length of the segment. State i+3 is temporarily disregarded, as was done in the algorithm for the outlet adiabatic section. These four unknowns are determined from a two-dimensional Newton-Raphson routine involving Equations C.6 through C.9. Of the four equations in the iterative solution routine, only two have to be written in residual format. The other two equations can be used to solve explicitly for Tw_{i+2} and L_{i+2} based upon the initial guesses for the other two variables. Initial guesses are provided for these two iterated variables (Tct_{i+2} and Tsl_{i+2}), which are used to determine the values for Tw_{i+2} and L_{i+2} according to Equations C. 27 and C.28.

$$hct_{i+2}\pi D_{ct}(Tct_{i+2} - Tw_{i+2}) = hsl_{i+2}\pi D_{sl}(Tw_{i+2} - Tsl_{i+2}) \qquad \text{(C.27)}$$

$$(P_{i+2} - P_{i+4}) = \frac{L_{i+2} f_{avg} \upsilon_{avg} G_{ct}^2}{2D_{ct}} + G_{ct}^2(\upsilon_{i+4} - \upsilon_{i+2}) \qquad \text{(C.28)}$$

Equation C.27 states that the convection heat transfer from the capillary tube over a differential area equals the convective heat transfer to the suction line. Equation C.28 is the same momentum equation for the capillary tube that was discussed in the outlet section. Everything in these two equations is either known or can be calculated from the guessed values of Tct_{i+2} and Tsl_{i+2}. Once Tw_{i+2} and L_{i+2} are known, the two residual equations shown in Equations C.29 and C.30 can be evaluated.

$$R(1) = (h_{i+4} - h_{i+2}) + \frac{G_{ct}^2}{2}(\upsilon_{i+4}^2 - \upsilon_{i+2}^2) - Cp_{sl}(Tsl_{i+4} - Tsl_{i+2}) \qquad (C.29)$$

$$R(2) = \dot{m}_r Cp_{sl}(Tsl_{i+4} - Tsl_{i+2}) + hsl_{avg}\pi D_{sl}L(Tw_{avg} - Tsl_{avg}) \qquad (C.30)$$

The first equation is the energy equation for the capillary tube and the second equation is a rate equation for the suction line. The internal Newton-Raphson routine iterates Tct_{i+2} and Tsl_{i+2} until the above residuals go to zero. At that time, the corresponding values of Tw_{i+2} and L_{i+2} are also correct for that segment. In this case where the above equations were used to "solve" a subcooled liquid segment with zero quality at its inlet, the terms involving the difference of specific volumes in the momentum and energy equations are actually zero.

Next, the value of Tsl_{i+2} is compared with the value of Tsl_{i+3} (Tsl_{in} + DTsl). Since Tsl_{i+2} is larger, the recondensation did not occur in this last DTsl segment. Therefore, the last segment (i+3 to i+4) is solved as a totally liquid segment. The solution process is identical to the one outlined above except that the suction line temperature (Tsl_{i+3}) is no longer an unknown. Instead, the capillary tube temperature (Tct_{i+3}) and pressure (P_{i+3}) are the iterated variables. Equations C.27 and C.28 are still used to determine the wall temperature (Tw_{i+3}) and length (L_{i+3}) used in residual Equations C.29 and C.30.

After the last DTsl segment (i+3 to i+4) has been solved, the model again determines the suction line temperature (Tsl_{i+2}) associated with the zero-quality point. Although this was done before when the segment from i+2 to i+4 was solved, the new value will be obtained by solving the segment from i+2 to i+3. In general, these values for Tsl_{i+2} will differ slightly due to the different discretization lengths and the nonlinear nature of the equations. Once the temperature Tsl_{i+2} is determined, it is compared to Tsl_{i+1} (Tsl_{i+3} + DTsl). If Tsl_{i+2} is lower than Tsl_{i+1}, as in this case, then recondensation occurred in the second segment (i+1 to i+3). The solution just obtained when Tsl_{i+2} was determined will be used, so everything up to the i+2 point is known.

With the information from the i+2 point and the suction line temperature Tsl_{i+1} (Tsl_{i+3} + DTsl), the segment from i+1 to i+2 can be solved as a two-phase segment. The solution process is the same as mentioned above except that the iterated variables are the capillary tube temperature (Tct_{i+1}) and quality (x_{i+1}). Again, Equations C.27 and C.28 are used to determine the wall temperature (Tw_{i+1}) and length (L_{i+1}) used in residual Equations C.29 and C.30. The totally two-phase DTsl segment upstream (i to i+1) is solved in exactly the same manner. With the known suction line temperature Tsl_i (Tsl_{i+1} + DTsl) and Equations C.27 through C.30, the values of Tct_i, x_i, L_i, and Tw_i can be determined.

The state of the refrigerant at the entrance to the heat exchanger is used as the starting point for the solution process of the inlet described in the next section. The lengths of the four segments (2 two-phase and 2 subcooled) are summed to get a total length of the heat exchanger, Lhx_cap. This length is calculated based upon the assumed values of DPout, DTsl, Pcrit, and xcrit. As with the outlet section, there is a residual equation that compares Lhx_cap and Lhx that is part of the overall Newton-Raphson solver that determines what the new guesses should be in order that Lhx_cap approach Lhx.

C.5.4 Two-phase inlet to the suction line

Although the following discussion does not apply to the current hypothetical solution, the model can also handle a two-phase inlet to the suction line. Originally, it was thought unnecessary to have a model that was able to handle a two-phase inlet to the suction line because most of the experimental data available had superheated inlets. However, during the validation process for the overall system model, it became evident that it was quite possible for an optimally-designed refrigerator to operate with a two-phase evaporator exit.

The ct-slhx model was not able to handle these two-phase suction line inlets and the prospects for changing the model did not look good because of the solution method employed in the heat exchanger section. Since the model discretized the section using temperature steps in the suction line (DTsl), it was unclear how a two-phase suction line portion with a constant temperature could be incorporated into the heat exchanger section solution.

The approach taken was to "fix" the model rather than give it a complete "overhaul". The obvious benefit of this approach was that it took much less time than a complete re-write of the heat exchanger section solution would have taken. The basic approach of the fix was to solve the two-phase portion of the suction line heat exchanger first and then to solve the rest of the section with the temperature-step approach described in detail in Section C.5.3.

The current model assumes that the refrigerant in the capillary tube side of the heat exchanger is also a two-phase mixture for any portion of the suction line heat exchanger that contains two-phase refrigerant. This assumption was made because the evaporator of most domestic refrigerators will have a two-phase refrigerant exit only when the condenser also has a two-phase refrigerant exit. If the condenser has a two-phase exit, it is more likely that the refrigerant will be two-phase at the end of the heat exchanger section of the capillary tube. However, it is possible that there will be some occasions during simulation for which the stated assumption will not be valid. For such cases, the current model will not work.

As mentioned, two-phase inlets to the suction line are handled by solving the two-phase portions of the suction line before the temperature steps are taken. In other words, the quality of the refrigerant in the suction line is brought up to 1.0 before the "regular" solution process takes place. Since it is unlikely that the quality of the refrigerant entering the suction line is significantly lower than 1.0, the model will use either one or two segments for the two-phase portion. If the entering quality is greater than or equal to 0.95, then only one segment will be used. If the entering quality is less than 0.95, then two segments will be used. A schematic of the solution scheme is shown below in Figure C.7 for the case of two-phase refrigerant inlet (quality less than 0.95) to the suction line.

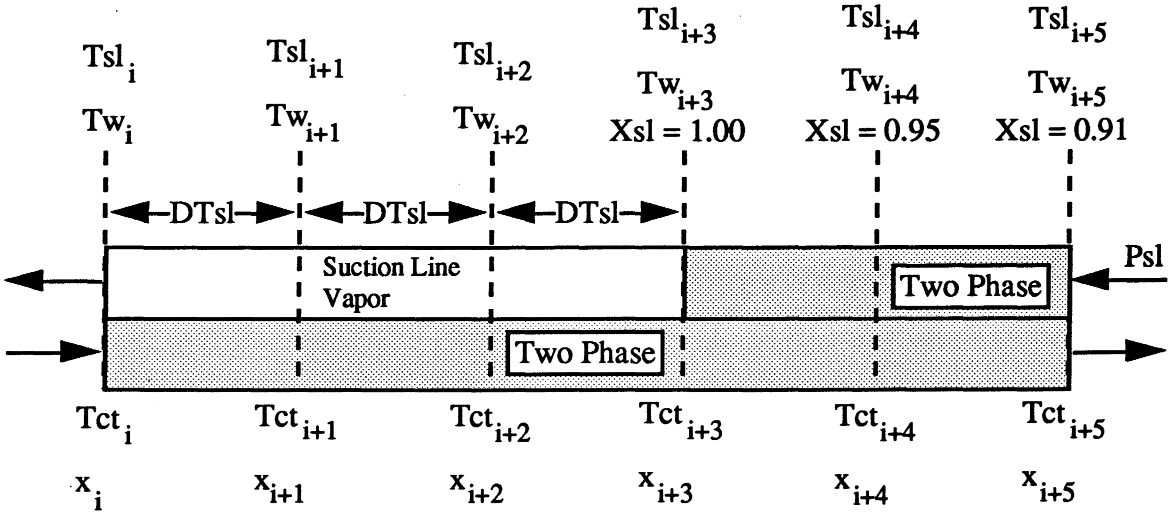


Figure C.7 The heat exchanger section in a two-phase inlet case

The two-phase suction line segments (i+4 to i+5 and i+3 to i+4) are solved with the same equations mentioned in Section C.5.3. However, the suction line heat transfer coefficient comes from the two-phase evaporating heat transfer correlation mentioned in C.3.4. rather than the single-phase correlation used in the superheated portion of the suction line.

At the exit point (i+5), everything except Tw_{i+5} is known from the inputs or the previous solution. As before, the wall temperature is determined explicitly from Equation C.9. The last segment (i+4 to i+5) is solved by determining the capillary tube temperature and quality at i+4 that solve the governing equations. The equations used to solve the segment (i+4 to i+5) are identical to those listed in Equations C.27 through C.30 except that the subscripts are different for the problem at hand. Since the suction line contains two-phase refrigerant and the pressure is known, the suction line refrigerant temperature at i+4 is known. The quality of the refrigerant in the suction line at i+4 is set equal to 0.95. The first two equations (C.27 and C.28) are used to explicitly determine Tw_{i+4} and the length between i+4 and i+5. With these values, the two residual equations (C.29 and C.30) can be evaluated. The internal Newton-Raphson routine iterates Tct_{i+4} and x_{i+4} until the residual equations go to zero.

This process, with the exception that the downstream suction line quality is set to 1.00 instead of 0.95, is repeated to solve the next segment (i+3 to i+4). After the solution of this segment, the solution process for the remaining heat exchanger segments proceeds as outlined in section C.5.3.

There is obviously room for improvement in the solution method of these two-phase refrigerant inlets to the suction line. The number of segments could be increased beyond two if the entering quality level is much lower than 0.90. The model could be made more general so that it could handle a two-phase inlet to the suction line when the refrigerant in the capillary tube is subcooled liquid. This last option would require a significant amount of logic because there would probably be recondensation in the capillary tube opposite the two-phase suction line. The recondensation point would have to be found in a manner similar to that described in Section C.5.3. Whether these, or other improvements, are needed in the heat exchanger section solution depends on whether these operating conditions would ever be encountered in the model.

C.5.5 Adiabatic inlet section

The solution method for the inlet section is identical to that for the outlet section since both sections are governed by the same differential equations. In this case, however, the exit state of the outlet section is obtained from the inlet state of the heat exchanger section, and the variable used to discretize the differential equations is DPin instead of DPout. If Figure C.8 represents the present situation in the inlet section, the solution would proceed as follows:

1) The model would use the exit conditions of the inlet section (hx inlet) and the discretized energy equation to iteratively solve for the zero-quality pressure (P_{i+1}).

2) Since P_{i+1} is greater than P_{i+2} and less than P_i, the model would determine that the flashing point occurred in the first DPin segment.

3) With this information, the model would "solve" the third and second totally two-phase DPin segments (i+3 to hx inlet and i+2 to i+3). For each segment, the discretized energy equation would be used to solve iteratively for the quality and then the momentum equation would be used to solve explicitly for the length of each segment.

4) As mentioned in section C.5.2, the zero-quality pressure P_{i+1} is determined again for the purposes of "solving" the two-phase portion of the first DPin segment. Once the pressure is known, the momentum equation can be used as before to explicitly solve for the length of this two-phase segment (i+1 to i+2).

5) The remainder of the first DPin segment is "solved" as a subcooled liquid segment. Because the liquid is incompressible, the energy equation can be used to explicitly solve for the temperature T_i, and, as before, the momentum equation can be used to solve for the length of the single-phase segment (i to i+1).

6) The lengths of the four segments (1 subcooled and 3 two-phase) are added to obtain the variable Lin_cap which will be compared with the actual length Lin in the second residual equation.

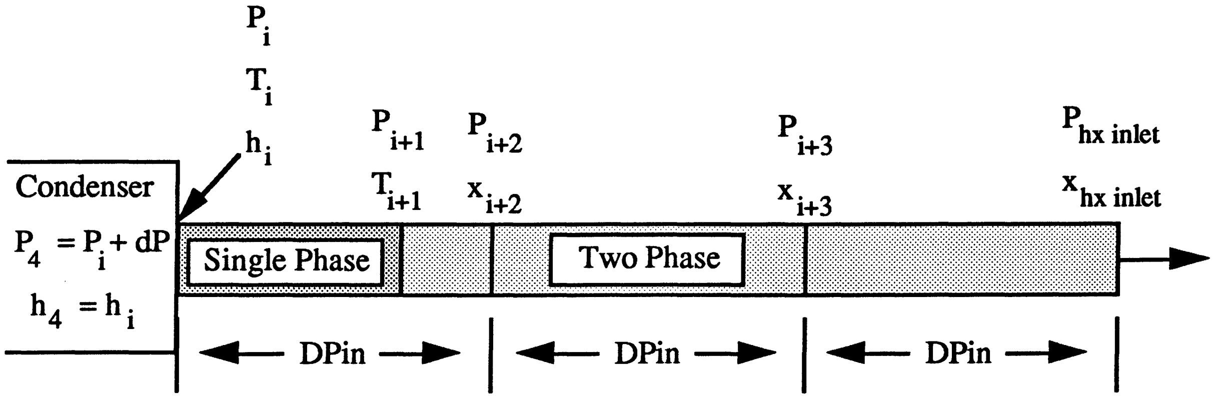


Figure C.8 The adiabatic inlet section of the capillary tube

The primary difference between the inlet and outlet sections is that the inlet section is connected to the condenser exit. As mentioned in Section C.4, residual equations 5 and 6 involve the refrigerant state at the condenser exit rather than the capillary tube inlet. The fifth residual equation compares the calculated enthalpy at the condenser exit with enthalpy given by the user or from the system model. Since it is assumed that the enthalpy is constant across the sudden contraction pressure drop, the enthalpy calculated at the capillary tube entrance (h_i) can be used in the comparison in the fifth residual equation. However, the pressure does change across the sudden contraction according to Equation C.10. Therefore, after the first DPin segment is solved, the pressure drop is calculated and added to P_i to obtain P4_cap which is compared to the system model variable P4. At this point, the ctslhx subroutine is finished and the appropriate variables are returned to the main program to be used in the residual equations.

C.6 Conclusion

The capillary tube-suction line heat exchanger appears to be a very simple device, but in reality, it is really the most complex component in the refrigeration system to model. The multiple combinations of the processes which may occur require a robust model which can operate in several modes. The approach taken here places the governing equations of the capillary tube model in a subroutine which is solved "sequentially" based upon the current values of six Newton-Raphson variables. These six variables are iterated until the six residual equations that compare "sequential" solution outputs to known parameters are solved. If warranted, there are several possibilities for further development on the model: examination of other two-phase friction factors; examination of choked flow equations (other than homogeneous equilibrium flow); examination of metastable regions within the capillary tube; and the improvement to the heat exchanger solution when the inlet to the suction line is two-phase refrigerant

References

ASHRAE, ASHRAE Handbook of Fundamentals, Atlanta GA, 1993.

Dukler, A.E., et al., "Pressure Drop and Hold Up in Two-Phase Flow," A. I. Ch. Journal, Vol. 10, pp. 38-51, 1964.

Incropera, F.P. and D.P. DeWitt, Fundamentals of Heat and Mass Transfer, 3rd ed., John Wiley & Sons, Inc., New York, 1990.

Melo, C., R.T.S. Ferreira and R. H. Pereira, "Modeling Adiabatic Capillary Tubes: A Critical Analysis," Proceedings of the International Refrigeration Conference, Purdue University, p. 118, 1992.

Peixoto, R.A. and C.W. Bullard, "A Simulation and Design Model for Capillary Tube-Suction Line Heat Exchangers," Proceedings of the International Refrigeration Conference, Purdue University, p.335-340, 1994.

Sleicher, C.A. and M.W. Rouse, "A Convenient Correlation for Heat Transfer to Constant and Variable Property Fluids in Turbulent Pipe Flow," International Journal of Heat and Mass Transfer, Vol. 18, p.677-683, 1975.

Stoecker, W.F., Refrigeration and Air Conditioning, Second Edition, McGraw-Hill, New York, 1986.

Swamee, P.K., and Jain, A.K., "Explicit Equations for Pipe-Flow Problems," Proceedings of the ASCE, Journal of the Hydraulics Division, 102, HY5, p.657-664, May 1976.

Wattelet, J.P. and J.C. Chato, "Heat Transfer Flow Regimes of Refrigerants in a Horizontal-Tube Evaporator," ACRC TR-55, University of Illinois at Urbana-Champaign, 1994.

Whalley, P. B., Boiling, Condensation, and Gas-Liquid Flow, Oxford University Press, New York, 1990.

Appendix D

RFSIM Refrigerant Property Interpolation Routines

D.1 Introduction

Although there are many ways to write thermodynamic property subroutines for refrigerants, one of the often overlooked methods is interpolation from data tables. To some researchers and engineers, it may seem archaic to interpolate from tables when computers can use equations of state and thermodynamic relations to calculate refrigerant properties. This may seem especially true considering that the property packages that employ these techniques can minimize memory requirements for a large number of refrigerants simply by using different coefficients in a single equation of state. Although these property packages certainly have advantages in their generality, interpolation from tables can still be the best choice in many situations. In particular, thermodynamic property routines may be unsuitable for use in large thermal system modeling because of the time required to obtain the many properties required during a solution. Because of the computation time and other issues, the property package routine REFPROP from NIST has been replaced in the Refrigerator/Freezer Simulation (RFSIM) Model with interpolation routines written at the ACRC. The following sections will explain the benefits of using interpolation routines in RFSIM, and the specific details of the particular interpolation routines that are now used will be discussed.

D.2 Benefits of interpolation

D.2.1 Speed

When modeling a thermal system such as a refrigerator, it is important to have reliable thermodynamic property data. As thermal models become larger and larger, it is also important to obtain the property data as quickly as possible during the simulation. This is especially true for thermal modeling where an iterative solution scheme, like Newton-Raphson, is employed since the number of property calls in the governing equations is multiplied by the number of iterations. The primary reason for switching from NIST's REFPROP property package to an interpolation routine in the RFSIM model was a desire for a shorter solution time.

When the switch was made to the interpolation routines, the computation time required to solve the model decreased by a factor of five or six. This incredible difference between the two methods is partly due to the large number of property calls in the capillary tube-suction line heat exchanger model that is present within the RFSIM model. This subroutine solves the discretized governing equations in the capillary tube-suction line heat exchanger for approximately 15 segments. Each of these segments is solved by a one- or two-dimensional Newton-Raphson routine. The captube subroutine itself is called 11 times per iteration of the RFSIM model. As a result, there can be anywhere from several hundred to over thousand property calls during one iteration of RFSIM because of the capillary tube.

Even though the difference may not be as great for other thermal system models, the interpolation routines are definitely faster. It seems obvious that the property package would be slower since they have to perform many calculations that could involve numerical integration and differentiation to obtain just one value. The interpolation tables, on the other hand, have only to find the appropriate location in the table and then compute a weighted average of the appropriate values. It seems rational, therefore, to use a property routine like NIST's REFPROP to obtain thermodynamic data when only a few properties are needed (i.e. to make a property table), and to use an interpolation routine to when the properties have to be accessed repeatedly in a thermal system model.

D.2.2 Flexibility

Another benefit of the interpolation routines that are used in RFSIM is that the refrigerant data can come from any source. This is very useful because some refrigerant properties may be readily available from a property package like REFROP or Engineering Equation Solver (EES), while others may only be found in one of the ASHRAE handbooks or from the manufacturer's own data. With the interpolation routines, it doesn't matter where the data comes from as long as there is enough data for single- and two-phase tables. How much data is enough will depend on the user's desired property range in terms of temperatures and pressures.

The flexibility in table size also gives the user the ability to have as much accuracy in thermodynamic properties as needed over a given range of temperatures and pressures. Thus, the user can customize tables for the particular needs of their application. For example, if someone were modeling a R-22 evaporator for an air conditioner, they could use tables that would give very accurate property data in the relatively small range of operating conditions likely to be encountered in the simulations. On the other hand, someone who is modeling an entire refrigerator would need to have accurate properties over a much larger range. Of course, there may be some upper limit to the table sizes imposed by the FORTRAN compiler or the computer memory, but this would probably only happen in rare situations.

In addition to size flexibility, the interpolation routines were designed so that either pure or mixed refrigerants could be used. However, there is one assumption made, however, about the two-phase tables for mixed refrigerants. If they are zeotropic mixtures, it is assumed that the temperature "glide" is linear. The temperature "glide" is the change in temperature that occurs as the mixed refrigerant is changing phase (the difference between the bubble point and the dew point). In other words, at a given pressure, the temperature of the refrigerant can be calculated as any other property (enthalpy, entropy, volume) by a quality weighted average of the bubble (saturated liquid) and dew (saturated vapor) temperature. This will become clearer when the two-phase table format is discussed later.

D.2.3 Insertable into thermal system models

One advantage of the interpolation routines used in RFSIM is the ease of inserting them into existing thermal system models. Many of the commercially available property packages are proprietary or are not suited for insertion into thermal system models developed in-house at universities or companies. The interpolation routines used in RFSIM are very portable to other models as well.

The "interpolation routines" consist of one FORTRAN file (REFPROP.f) comprised of about 23 subroutines and functions, two "include" files (REFPROP.INC and REFRIG.INC), and at least one single- and two-phase table of refrigerant data. The FORTRAN file contains the routines that initialize the data arrays with the correct refrigerant data, the routines that actually return the property information to the thermal system model, and the interpolation routines. The "include" files are just pieces of FORTRAN code that are included in the routines of REFPROP.f and anywhere else in model where information about the refrigerant type is needed.

The interpolation package can be used in another model by placing REFPROP.f and the "include files" with the model FORTRAN (either directly or linked through the use of a "makefile") and by putting the refrigerant data tables in the model directory. The initialization routine (ref_init) in REFPROP.f will need to be called once by the model and any thermodynamic property calls in the model must match those in REFPROP.f. Once this has been done, the entire model (including the interpolation routines) must be compiled in whatever fashion required on the particular platform (i.e. PC, Unix, Convex, etc.) used for the simulation. The preceding discussion is not intended to cover every detail of the process of using the interpolation routines with a new model; rather, it is meant to convey the relative simplicity of inserting the "interpolation package" into any thermal model written in FORTRAN.

D.2.4 Easy to understand

From an engineer's or programmer's perspective, the new interpolation routines have the added benefit of being relatively easy to understand. Since interpolation, even 2-D interpolation, is something that most mechanical engineers learn in thermodynamics, it is fairly easy to look at the routines that return the thermodynamic properties as well the interpolation routines and understand what is "going on". Of course, most users of these routines will never have to look into them, but it is nice to have the ability to modify the routines if necessary.

For example, in the capillary tube-suction line heat exchanger model the calculation of mass flow is based upon the assumption of choked flow condition at the exit. This calculation requires the evaluation of a numerical partial derivative of volume with respect to pressure at constant entropy. This evaluation is straightforward if the property routines can return volume as function of pressure and entropy. Unfortunately, the interface written for NIST's REFPROP that was used in the original version of the RFSIM model did not have that particular property call. As a result, the partial derivative was obtained indirectly by taking four other partial derivatives and using thermodynamic relations and the chain rule. This method was both computationally more expensive and very difficult to understand and explain. With the new interpolation routines, a new property call has been written that returns the two-phase volume as a function of pressure and entropy. This allows for the direct and understandable evaluation of the partial derivative in question.

Although the existing property routines contained in the interpolation "package" will probably be sufficient for most thermal modeling applications, it is possible that someone will need to add or modify a routine. It is a relatively simple matter to add a routine by using the existing property routines as examples.

D.3 Using the interpolation package with the RFSIM model

D.3.1 Changing refrigerant types for which tables already exist

It is a relatively simple matter to switch between refrigerants if the tables already exist for that particular refrigerant. At the time of this writing, extensive tables required for modeling refrigerators, have been created for R-12 and R-134a. Two other tables, used for air conditioner evaporator modeling, have been created for R-22 and R-407C, a zeotropic mixture of R134a/R32/R125 (52/23/25%). However, since the refrigerant transport properties are also needed, only R-12 and R-134a can currently be used with the RFSIM model.

Only one file, REFRIG.INC, needs to be modified in order to change refrigerant types. The following block of code is how REFRIG.INC appears in the program.

```
C**********************************************************************
C   This block of code serves to declare the refrigerant type.
c
C   reftype - Switching parameter that declares refrigerant type.
C             Use the table below. (i.e. for R12 set reftype = 1)
c
c     Pure refrigerants currently available
C
C      1 - R12
c      2 - R22
c      3 - R134a
c
c     Refrigerant mixtures currently available
c
C     11 - R32/R125/R134a (23%/52%/25%)
C**********************************************************************

      integer reftype
      parameter(reftype = 1)
c**********************************************************************
```

In this situation, the refrigerant selected is R12. To change to R134a, the parameter "reftype" would be set equal to 3. After saving the changes made in REFRIG.INC, the entire

RFSIM model FORTRAN would have to be recompiled. In the Convex C240 environment (UXH), this involves deleting all the "*.o" files and recompiling the program through the use of the "Makefile". In other programming environments, the user should make sure that all the code is recompiled however that can be accomplished. Some of the FORTRAN code has to be recompiled because the file REFRIG.INC is included in several routines in the model code. If those routines are not recompiled, the changes made to the refrigerant type in REFRIG.INC will not take effect and the old refrigerant type will still be used. To be safe, it is better to recompile the entire model code.

D.3.2 Adding refrigerants

If the user wants use the RFSIM model, or any other thermal model, with a refrigerant not currently available in the interpolation package, there are several steps that have to be taken. The first step is to add a new refrigerant number in the file REFPROP.INC shown above. For example, if the user wanted to add R152a to the package, he/she would make the following changes (shown in boldface) to the code:

```
C**************************************************************************
C   This block of code serves to declare the refrigerant type.
c
C   reftype - Switching parameter that declares refrigerant type.
C             Use the table below. (i.e. for R12 set reftype = 1)
c
c     Pure refrigerants currently available
C
C      1 - R12
c      2 - R22
c      3 - R134a
c      4 - R152a
c
c     Refrigerant mixtures currently available
c
C     11 - R32/R125/R134a (23%/52%/25%)
C**************************************************************************

      integer reftype
      parameter(reftype = 4)
c**************************************************************************
```

After saving the changes made to this file, the next step would be to edit the subroutine "ref_init" in the file REFPROP.f. Again, the necessary changes have be shown in boldface below.

```
      SUBROUTINE ref_init
c**************************************************************************
      implicit none
      character sp_data$*12, tp_data$*12
      character ref_name$*35
      include 'REFPROP.INC'
      include 'REFRIG.INC'

c     ** Assign variable names to refrigerant data tables**

      if (reftype.eq.1) then
        ref_name$ = 'R12'
        sp_data$ = 'sp_R12_EES'
        tp_data$ = 'tp_R12_EES'
      else if(reftype.eq.2) then
        ref_name$ = 'R22'
        sp_data$ = 'sp_R22xx_4.0'
        tp_data$ = 'tp_R22xx_4.0'
      else if(reftype.eq.3) then
        ref_name$ = 'R134a'
        sp_data$ = 'sp_R134a_EES'
        tp_data$ = 'tp_R134a_EES'

      else if(reftype.eq.4) then
       ref_name$ = 'R152a'
       sp_data$ = 'sp_R152a'
       tp_data$ = 'tp_R152a'

      else if (reftype.eq.11) then
        ref_name$ = 'R32/R125/R134a (23%/52%/25%)'
        sp_data$ = 'sp_zeo52_4.0'
        tp_data$ = 'tp_zeo52_4.0'
      else
        write(*,*) "---ref_init: invalid refrigerant type"
      endif

c     ** Read refrigerant data tables **
      call refdata_read(sp_data$, tp_data$)

      return
      end
c***********************************************************************
```

The character strings sp_data$ and tp_data$ are the names of the single- and two-phase refrigerant data files, respectively. In this example, the two files have been named "sp_R152a" and "tp_R152a". The files containing the thermodynamic data in the model directory have to match the file names in this subroutine. The data is read into the storage arrays through the subroutine "refdata_read" which is called on the last line of the above subroutine. The data files, listed in "ref_init", that actually exist have an additional extender of "_EES" or "_4.0" indicating that they were made from EES and NIST's REFPROP Version 3.0 programs, respectively. Although there is no requirement that this naming convention be followed with future refrigerants, it would probably be a good idea if it was followed.

The two changes just described are all that is needed to prepare the way for new refrigerant tables, but there are other changes needed in the RFSIM model and probably other models as well. As alluded to before, there are several routines that return transport properties and other information of the chosen refrigerant. These routines usually consist of curve fits of the particular property as a function of temperature. In the RFSIM model, these routines are all located in the FUNCTION.f file and consist of the following routines:

SurfTen	Surface tension of the refrigerant (liquid/vapor)
cpl	Specific heat of the liquid refrigerant
cpv	Specific heat of the vapor refrigerant
kl	Thermal conductivity of the liquid refrigerant
kv	Thermal conductivity of the vapor refrigerant
mul	Absolute viscosity of the liquid refrigerant
muv	Absolute viscosity of the vapor refrigerant
Pcritical	Critical pressure of the refrigerant
Mweight	Molecular weight of the refrigerant

Since the above properties are different for each refrigerant, the routines have to have a separate curve fit for each refrigerant. This is accomplished through the use of an "IF-ELSE IF-END IF" structure as illustrated below for the "cpl" function.

```
DOUBLE PRECISION FUNCTION  cpl(F)
c************************************************************************
      IMPLICIT NONE
      DOUBLE PRECISION F,M0,M1,M2,M3
      include 'REFRIG.INC'

      IF (reftype.eq.1) THEN
c        ** R12 **
         M0=0.21496503138
         M1=0.00018209023076
         M2=4.4274578058e-07
         M3=6.7137579036e-09
      ELSE IF (reftype.eq.2) THEN
c        ** R22 **
         M0=0.31423914658
         M1=-0.0046030127722
         M2=0.00019235518087
         M3=-3.5212997016e-06
      ELSE IF (reftype.eq.3) THEN
c        ** R134a **
         M0=0.30870348389
         M1=0.0003159532427
         M2=-4.6969569421e-07
         M3=2.0012620579e-08

     ELSE IF (reftype.eq.4) THEN
c        ** R152a **
       M0= ???
       M1= ???
       M2= ???
       M3= ???
      END IF

      cpl = M0 + M1*F +M2*F**2 + M3*F**3

      END
C************************************************************************
```

Again, the required changes have been shown in boldface. The constants M0, M1, M2, and M3 for R-152a would have to be obtained from a source such as the ASHRAE Fundamentals handbook. After this type of editing has been performed on all nine routines shown above, all of the required changes to the RFSIM model code will have been completed.

The only remaining step before the RFSIM model could be run with the new refrigerant is to actually make the single- and two-phase refrigerant tables. Since this topic is at the heart of the interpolation package, it will be discussed in detail in the following section.

D.4 Property routines and refrigerant tables in the interpolation package

D.4.1 Property routines

The original thermodynamic property interpolation package was developed at the ACRC by Franco Raggazi and Tim Nygaard. They inherited the interpolation routines from Dr. Curt Pedersen and then developed the table format, initialization routines, and property routines that return the thermodynamic properties. With the exception of the property routines, the current interpolation package is the same as before. In the property routines, several modifications have been made and are listed as follows: new routines have been added; most of the existing routines have been modified and converted from subroutines to functions; and a few minor mistakes in the existing routines have been found and corrected.

In the original interpolation package, there were 12 property routines and this number has increased to 17 in the current version. The routines, and a brief description, are listed below in the order they appear in the file REFPROP.f:

SaturationInt	various saturation properties as a function of pressure
PsatT	saturation pressure = f(temperature, quality)
TsatP	saturation temperature = f(pressure, quality)
hpt	enthalpy = f(pressure, temperature)
hpx	enthalpy = f(pressure, quality)
htx	enthalpy = f(temperature, quality)
vpt	volume = f(pressure, temperature)
vpx	volume = f(pressure, quality)
vtx	volume = f(temperature, quality)
vps2ph	2ph volume = f(pressure, entropy)

spt	entropy = f(pressure, temperature)
spx	entropy = f(pressure, quality)
stx	entropy = f(temperature, quality)
xhp	quality = f(enthalpy, pressure)
TPHInt	temperature = f(pressure, enthalpy) by inverse interpolation
TPHiter	temperature = f(pressure, enthalpy) by iteration
TPSiter	temperature = f(pressure, entropy) by iteration

Most of the modifications of the property routines have dealt with what happens when a thermodynamic property has been requested outside of its table range. In the original interpolation package, some of the property routines would return a value or 0.0 or 99.0 if the property request was outside of the table range. In the new package, all of the property routines, with the exception of the "TPHInt" routine, will return a value that is as close as possible to the requested value. For example, if the enthalpy at 100 psia and 310 °F is requested and the single-phase table only goes up to 300 °F, then a warning will print to the screen and the value of enthalpy at 100 psia and 300 °F will be returned.

The routine "TPHInt" is more complicated than the others because inverse interpolation is required to obtain the temperature values due to the of the functional form of the single-phase tables. It is more difficult to determine beforehand whether the given enthalpy is within the table range because this depends on the pressure. In other words, at one pressure the enthalpy may be in the table range, but at another pressure, the enthalpy may not be in the table range. The current 2-d inverse interpolation routine used in the interpolation package will return a value of 0.0 for the temperature if the pressure or enthalpy is out of range. However, the property routine will print a warning if this happens. Fortunately, this routine is not used in the RFSIM model and was kept in the interpolation package primarily to serve as an example of inverse interpolation. If the user needs to calculate temperature as a function of pressure and enthalpy, it would be better to use "TPHiter" since this routine will return a reasonable value if the arguments are out of the table range.

The few mistakes that were mentioned previously were found in the two iteration routines "TPHiter" and "TPSiter". There were identical errors in both routines in the calculation of the temperature in the subcooled liquid region. The original author probably did not catch these errors because he was modeling an evaporator, and thus, he did not call these routines in the subcooled liquid region.

D.4.2 Single-phase refrigerant property tables

D.4.2.1 General description

The single-phase table actually consists of six smaller tables placed together in the same file. The six smaller tables correspond to the refrigerant superheated enthalpy, subcooled enthalpy, superheated entropy, subcooled entropy, superheated volume, and subcooled volume. The property data in each of the tables are tabulated as functions of temperature and pressure. The temperature values are placed in the first column and the pressure values are placed in the first row of each single-phase table.

The data from these tables is read into storage arrays by the subroutine “refdata_read” located in REFPROP.f. These arrays are then accessible by all of the property routines through the use of the “common” statement in FORTRAN. The properties are determined from 2-D linear interpolation between the four nearest values.

Although the single-phase properties are unique for a given pressure and temperature, the property data is divided into superheated and subcooled tables because of the discontinuity in the property values that exist near the saturation line. Because of the 2-D linear interpolation, property calls near the saturation line would result in errors if all the data for a property were placed into a single table. This problem of obtaining problems near the saturation line will be discussed in detail below.

The data in the single-phase tables is only accessed if the property routine in use determines that the refrigerant is actually single-phase. For example, the routine “hpt” will typically only use the superheated enthalpy or subcooled enthalpy table, but it is possible, in the case of a zeotrope, that the routine would have to use the two-phase table to obtain the enthalpy. Regardless of the refrigerant type, the “hpt” routine will determine which table to use by comparing the given temperature to the bubble (saturated liquid) and dew (saturated vapor) temperature at that pressure.

D.4.2.2 Single-phase table format

The six single-phase tables mentioned above must be placed together in the single-phase data file whose name must agree with one of the file names listed in the subroutine “ref_init” in REFPROP.f. The general format for the file containing the single-phase tables is shown schematically below in Figure D.1.

```
# of rows - superheated enthalpy table
# of columns - superheated enthalpy table
99            p1            p2 (>p1)       ...
t1            Hsh(p1,t1)    Hsh(p2,t1)     ...
t2 (>t1)      Hsh(p1,t2)    Hsh(p2,t2)     ...
...           ...           ...            ...
# of rows - subcooled enthalpy table
# of columns - subcooled enthalpy table
99            p1            p2 (>p1)       ...
t1            Hsc(p1,t1)    Hsc(p2,t1)     ...
t2 (>t1)      Hsc(p1,t2)    Hsc(p2,t2)     ...
...           ...           ...            ...
# of rows - superheated entropy table
# of columns - superheated entropy table
99            p1            p2 (>p1)       ...
t1            Ssh(p1,t1)    Ssh(p2,t1)     ...
t2 (>t1)      Ssh(p1,t2)    Ssh(p2,t2)     ...
...           ...           ...            ...
# of rows - subcooled entropy table
# of columns - subcooled entropy table
99            p1            p2 (>p1)       ...
t1            Ssc(p1,t1)    Ssc(p2,t1)     ...
t2 (>t1)      Ssc(p1,t2)    Ssc(p2,t2)     ...
...           ...           ...            ...
# of rows - superheated volume table
# of columns - superheated volume table
99            p1            p2 (>p1)       ...
t1            Vsh(p1,t1)    Vsh(p2,t1)     ...
t2 (>t1)      Vsh(p1,t2)    Vsh(p2,t2)     ...
...           ...           ...            ...
# of rows - subcooled volume table
# of columns - subcooled volume table
99            p1            p2 (>p1)       ...
t1            Vsc(p1,t1)    Vsc(p2,t1)     ...
t2 (>t1)      Vsc(p1,t2)    Vsc(p2,t2)     ...
...           ...           ...            ...
```

Figure D.1 Single-phase tables file format

There should be no lines left in between the blocks of data for each table. The columns of the tables can be separated by spaces or commas. The number of rows and columns for each table can be different, but they must match the numbers stated before each table. Currently, the size limit for these single phase tables is 50 rows by 50 columns. This limit can be enlarged by editing REFPROP.INC and the subroutines “Terp2D” and “Invterpl2D” in REFPROP.f and changing the dimensions of the appropriate arrays. The row and columns numbers for each table have to include the row of pressure values and the column of temperature values. The number 99 in the (1,1) table position is arbitrary and is not used for anything.

If all the values of a particular property, like enthalpy, are calculated for given range of pressure and temperature and placed in a table, it would look something like the following.

Table D.1 Schematic of a single-phase property table

99	p1	p2	p3	p4	p5
t1	**Super**	*Subcool*	*Subcool*	*Subcool*	*Subcool*
t2	**Super**	**Super**	*Subcool*	*Subcool*	*Subcool*
t3	**Super**	**Super**	**Super**	*Subcool*	*Subcool*
t4	**Super**	**Super**	**Super**	**Super**	*Subcool*
t5	**Super**	**Super**	**Super**	**Super**	**Super**

The boldface "**Super**" positions represent the parts of the table that would be superheated and the italicized "*Subcool*" positions represents the parts of the table that would be subcooled. The dividing line between the two regions will be obvious because of the discontinuity in the table values. The first step in making separate superheated and subcooled tables is to obtain a table like this one and then "divide" it. This can be done quickly in a spreadsheet program by copying the table once and then deleting the subcooled part of the first copy and the superheated part of the second copy. However, this procedure leaves two tables that are only partially filled and reveals the major problem encountered during 2-D linear interpolation in the single phase tables; if the pressure and temperature values are near the dividing saturation line, a particular point may not have four neighboring point from which to interpolate. This problem can be best explained with a numerical example.

Table D.2 Portion of a single-phase enthalpy table

	P = 10.00	P = 20.00
T = -10.00	100.34	8.44
T = 0.00	102.25	101.71

This portion of a superheated table shows the enthalpy as function of pressure and temperature for R-134a. The three higher enthalpy values would be part of the superheated portion while the low value would be part of the subcooled portion. After the tables are separated, this part of the superheated table would only have three values. If an enthalpy value were to be calculated as a function of pressure and temperature, and it fell within the above pressure and temperature range, there would be a problem. The refrigerant would be superheated because the temperature is higher than the dew point (saturated vapor) temperature at the given pressure, but it would not have four surrounding points from which to interpolate. The location occupied by the 8.44 Btu/lbm value would be blank because superheated refrigerant does not exist at a pressure of 20 psia and temperature of -10 ^{o}F.

There are different ways that this problem could be addressed. One method would be to modify the interpolation procedure and use only three points to interpolate rather than four.

However, the method chosen for the this interpolation package was to expand the tables by linearly extrapolating the single-phase values on one side of the saturation line into the blank cell on the other side. For the present case, this would mean putting a "superheated" enthalpy value in the subcooled location. The simplest way to do this extrapolation is to neglect the dependency of the enthalpy on one of the state variables, pressure or temperature. Then the nearest neighboring enthalpy value, at the same value of the stronger dependency state variable, would be copied to the empty table location. In other words, since superheated enthalpy is a stronger function of temperature than pressure, the empty spot (T = -10 °F and P = 20 psia) in Table D.2 would be filled by the enthalpy value at the same temperature (T = -10 °F and P = 10 psia). The new superheated table is shown below in Table D.3.

Table D.3 Portion of a superheated enthalpy table

	P = 10.00	P = 20.00
T = -10.00	100.34	**100.34**
T = 0.00	102.25	101.71

Therefore, for the superheated enthalpy table, all the empty spots near the separation line that might be needed in an interpolation would be filled with the values to their left. To make things simple and to avoid mistakes, it is recommended that right-most real superheated value (P = 10.00 psia, T = -10.00 °F) be copied to all the locations to its right. Similar procedures should be followed for all six of single phase tables according to the property dependency guidelines shown in Table D.4.

Table D.4 Refrigerant property dependency on state variables

Property	**Temperature Dependency**	**Pressure Dependency**
Superheated enthalpy	stronger	weaker
Subcooled enthalpy	stronger	weaker
Superheated entropy	weaker	stronger
Subcooled entropy	stronger	weaker
Superheated volume	weaker	stronger
Subcooled volume	stronger	weaker

Basically, the information in Table D.4 means that for all the properties with a stronger temperature dependency, the extrapolation will be done in the horizontal direction. The extrapolation in superheated enthalpy table will be done to the right, and in the three subcooled tables the extrapolation will done to the left. The extrapolation for the superheated entropy and volume tables will be done vertically upward. It should be pointed out that these approximations only affect the points close to the saturation line and will become more accurate as the number of temperature rows and pressure columns increase. These extrapolations will become more clear when the guidelines for making the tables are discussed in the next section.

D.4.2.3 Guidelines for single-phase property table preparation

Although the general format for making the single-phase tables has already been discussed, some suggestions for the table development and examples will be presented in this section. These guidelines are not strict rules, but rather, they are an attempt to pass along some of the knowledge acquired by the author during the preparation of detailed single-phase tables for R-12 and R-134a.

D.4.2.3.1 Size and spacing of the tables

The first step in making the tables is to decide on the number of columns and rows to have in the table. The range covered by the pressure columns and temperature rows and the spacing between the columns and rows should be determined by the requirements of the thermal model that will be calling the property routines. Examples of these requirements have already been presented in D.2.2 and will not be repeated here. As a general rule, the pressure and temperature ranges covered by the tables should be made larger than range actually expected in the model solutions. This is sometimes necessary because it is possible, during iterations, for a temperature or pressure to reach an extreme, but temporary, value and then return to a more reasonable value by the final iteration. It should be noted that the spacing between the pressure columns and the temperature rows can be varied within the table. In fact, it seems that the pressure spacing should be fairly small at low pressures (10 to 30 psia for R-12 and R-134a) because all of the properties tend to change more rapidly with changes in pressure than in the higher pressure regions. Another good idea is to make two "special" temperature rows corresponding to the saturation temperature at the lowest pressure in the table and at the saturation temperature at the highest pressure in the table. The reason for this is somewhat difficult to explain and will further discussed with an example. Finally, it should reinforced that the each of the six tables may have a different pressure and temperature range and different spacings. The recommendations for each table will be covered in the examples to be covered.

D.4.2.3.2 Data generation

While the size and spacing of the tables are being decided, the user should keep in mind how he/she will actually generate the data for the tables. Depending on how much time can be invested in the table-making process, the method chosen for the property generation will probably affect the size and spacing of the tables.

The most efficient method of generating the table data is to use a property package that will calculate the all the properties (subcooled and superheated) in a "matrix" form identical to the eventual table format. In other words, a program that will calculate all of the properties, such as enthalpy, over the entire pressure and temperature range and display the results in a "table" similar to Table D.1. This method will obviously require the least manual processing since the user has only to "divide" the table and then extrapolate as previously described. The only program known by the author that will do this is EES. The "SetLookup" Function can be used to write values to a Lookup table using the temperatures in the left-most column and the pressures across the top row as the inputs. This method would allow the user to customize the six different tables because of the relative ease of making a new table once the Function format was developed. As an example, the superheated volume table could have more pressure columns and temperature rows than the subcooled volume table since the superheated values are more sensitive to pressure.

Another way to generate the properties in EES, NIST's REFPROP, and probably other packages is to use a "Parametric" table. Although the exact procedure will probably vary from one program to another, the general idea is to calculate the enthalpy, entropy, and volume for a range of temperatures and a fixed pressure. With this output, the user would copy and paste the three property columns into three separate tables under the corresponding pressure. This whole procedure would be repeated at different pressures until the desired range of pressures was covered. This method makes it more likely that all the tables will have the same number of pressure columns and temperature rows because the method becomes inefficient if the user customizes each table.

A third method of obtaining the property data is to manually copy it from a printed table from an ASHRAE Handbook, thermodynamic textbook, or any other reliable source of property data. With this method, the user has total flexibility with the table size and spacing since he/she will be typing in every number by hand. It is recommended that every opportunity be taken to reduce the amount of data required for the tables. These opportunities will be found primarily in the subcooled region since the properties don't change as much with pressure as they do in the superheated region. The user should examine the data in the printed tables before making the six tables to see where the pressure or temperature spacings can be enlarged without sacrificing too much accuracy.

D.4.2.3.3 Example single-phase tables

Several examples will be presented that illustrate the table making process for each of the six tables. The entire process of single-phase table preparation will be presented using the superheated and subcooled enthalpy tables as examples. Since the all three subcooled properties have a stronger dependency on temperature, only the subcooled enthalpy table will be presented. However, the making of the superheated entropy and volume tables will be presented since they have a stronger dependency on pressure.

Regardless of how the property data was obtained, the first step in the table making process is to construct a table in a spreadsheet program in the form shown in Table D.1 and in Table D.5 below. This table shows enthalpy (Btu/lbm) of R-12 as a function of pressure (psia) and temperature (°F). As in Table D.1, the superheated values have been put in boldface and the subcooled values have been italicized.

Table D.5 Single-phase enthalpy data

	14	30	50	75	100	125	150	200
-40	*0.007*	*0.038*	*0.077*	*0.126*	*0.175*	*0.224*	*0.273*	*0.371*
-30	*2.114*	*2.145*	*2.185*	*2.234*	*2.284*	*2.333*	*2.383*	*2.481*
-23.638	**74.711**	*3.491*	*3.531*	*3.581*	*3.631*	*3.680*	*3.730*	*3.829*
-20	**75.210**	*4.263*	*4.303*	*4.353*	*4.403*	*4.453*	*4.503*	*4.602*
0	**77.971**	*8.531*	*8.572*	*8.623*	*8.674*	*8.725*	*8.776*	*8.878*
20	**80.771**	**79.764**	*12.890*	*12.943*	*12.995*	*13.047*	*13.099*	*13.204*
40	**83.613**	**82.729**	**81.539**	*17.320*	*17.374*	*17.428*	*17.481*	*17.588*
60	**86.497**	**85.715**	**84.675**	*21.769*	*21.825*	*21.880*	*21.935*	*22.045*
80	**89.425**	**88.727**	**87.809**	**86.575**	*26.365*	*26.422*	*26.479*	*26.593*
100	**92.394**	**91.769**	**90.951**	**89.866**	**88.692**	**87.405**	*31.140*	*31.258*
120	**95.406**	**94.841**	**94.108**	**93.143**	**92.114**	**91.006**	**89.799**	*36.078*
131.741	**97.193**	**96.660**	**95.970**	**95.066**	**94.108**	**93.085**	**91.982**	*39.000*
140	**98.459**	**97.946**	**97.284**	**96.419**	**95.506**	**94.535**	**93.496**	**91.135**
160	**101.553**	**101.085**	**100.483**	**99.701**	**98.883**	**98.021**	**97.110**	**95.098**
180	**104.686**	**104.257**	**103.706**	**102.995**	**102.255**	**101.483**	**100.673**	**98.919**
200	**107.858**	**107.462**	**106.956**	**106.305**	**105.631**	**104.932**	**104.205**	**102.650**

The temperature range is from -40 °F to 200 °F, and the pressure range is from 14 psia to 200 psia. The ranges chosen for this example are totally arbitrary and were chosen primarily so that the table would fit conveniently on the page. As mentioned earlier, there are two “special” temperature rows corresponding to the saturation temperature at the lowest and highest pressures. In this table, these temperatures are -23.638 °F and 131.741 °F. These rows are placed in tables to mark the limits of the superheated and subcooled tables once the table is divided. The row at -23.638 °F marks the low-temperature limit of the superheated enthalpy table, and the row at 131.741 °F marks the high-temperature limit of the subcooled table. If these rows are not placed in the table, interpolation errors could occur. For example, if the row at -23.638 °F was not in the table, then the lowest temperature at 14 psia would be -20 °F. If the interpolation routines were used to calculate the enthalpy at 14 psia and -22 °F, the superheated tables would be used because the given temperature (-22 °F) is higher than the saturation temperature (-23.638 °F). Unfortunately, the temperature would be out of range of the superheated table, and therefore, the enthalpy value at 14 psia and -20 °F would be returned instead. The same errors could occur at the high temperature end of the subcooled table as well.

After Table D.5 has been constructed, the next step is to make a second copy of the table in the spreadsheet program. As mentioned earlier, one of the copies will become the superheated table, and the other one will become the subcooled table. Starting with the superheated table, all of the subcooled values should be erased. In the present example, the rows at -40 ^{o}F and -30 ^{o}F should also be erased since they don't contain any superheated values. Then, all of the right-most enthalpy values at each temperature should be copied to the right. Strictly speaking, only locations that are adjacent (horizontally, vertically, and diagonally) to the real values must be filled in with extrapolated values. However, it is just as easy, and safer, to copy them over to the last pressure column. Once the enthalpy values have been extrapolated, the superheated enthalpy table is complete. Table D.6 shows the superheated enthalpy table for this case.

Table D.6 Superheated enthalpy table

99	14	30	50	75	100	125	150	200
-23.638	**74.711**	74.711	74.711	74.711	74.711	74.711	74.711	74.711
-20	**75.210**	75.210	75.210	75.210	75.210	75.210	75.210	75.210
0	**77.971**	77.971	77.971	77.971	77.971	77.971	77.971	77.971
20	**80.771**	**79.764**	79.764	79.764	79.764	79.764	79.764	79.764
40	**83.613**	**82.729**	**81.539**	81.539	81.539	81.539	81.539	81.539
60	**86.497**	**85.715**	**84.675**	84.675	84.675	84.675	84.675	84.675
80	**89.425**	**88.727**	**87.809**	**86.575**	86.575	86.575	86.575	86.575
100	**92.394**	**91.769**	**90.951**	**89.866**	**88.692**	**87.405**	87.405	87.405
120	**95.406**	**94.841**	**94.108**	**93.143**	**92.114**	**91.006**	**89.799**	89.799
131.741	**97.193**	**96.660**	**95.970**	**95.066**	**94.108**	**93.085**	**91.982**	91.982
140	**98.459**	**97.946**	**97.284**	**96.419**	**95.506**	**94.535**	**93.496**	**91.135**
160	**101.553**	**101.085**	**100.483**	**99.701**	**98.883**	**98.021**	**97.110**	**95.098**
180	**104.686**	**104.257**	**103.706**	**102.995**	**102.255**	**101.483**	**100.673**	**98.919**
200	**107.858**	**107.462**	**106.956**	**106.305**	**105.631**	**104.932**	**104.205**	**102.650**

The original superheated values in the table have been left in boldface and the extrapolated values have been reduced in font size to contrast the two regions of the table. As explained before, the enthalpy values were extrapolated horizontally to the right because superheated enthalpy is a stronger function of temperature than it is of pressure. If the user wants to extrapolate only a few spaces to the right, the remaining locations should be filled with some number such as 0 or -1.

The subcooled enthalpy table is made in much the same way as the superheated table. In the second copy of the original enthalpy table shown in Table D.5, all of the superheated values should be erased. As before, the rows not containing any subcooled values (140 ^{o}F, 160 ^{o}F, 180 ^{o}F, & 200 ^{o}F) should be erased as well. Then, all of the left-most enthalpy values at each temperature should be copied to the left. Again, it is recommended that the real subcooled values be extrapolated to the first pressure column in the table. When the extrapolations are complete, the subcooled enthalpy table is finished and is shown in Table D.7 below.

Table D.7 Subcooled enthalpy table

99	14	30	50	75	100	125	150	200
-40	**0.007**	**0.038**	**0.077**	**0.126**	**0.175**	**0.224**	**0.273**	**0.371**
-30	**2.114**	**2.145**	**2.185**	**2.234**	**2.284**	**2.333**	**2.383**	**2.481**
-23.638	3.491	**3.491**	**3.531**	**3.581**	**3.631**	**3.680**	**3.730**	**3.829**
-20	4.263	**4.263**	**4.303**	**4.353**	**4.403**	**4.453**	**4.503**	**4.602**
0	8.531	**8.531**	**8.572**	**8.623**	**8.674**	**8.725**	**8.776**	**8.878**
20	12.890	12.890	**12.890**	**12.943**	**12.995**	**13.047**	**13.099**	**13.204**
40	17.320	17.320	17.320	**17.320**	**17.374**	**17.428**	**17.481**	**17.588**
60	21.769	21.769	21.769	**21.769**	**21.825**	**21.880**	**21.935**	**22.045**
80	26.365	26.365	26.365	26.365	**26.365**	**26.422**	**26.479**	**26.593**
100	31.140	31.140	31.140	31.140	31.140	31.140	**31.140**	**31.258**
120	36.078	36.078	36.078	36.078	36.078	36.078	36.078	**36.078**
131.741	39.000	39.000	39.000	39.000	39.000	39.000	39.000	**39.000**

As with the superheated table, the real subcooled values are shown in boldface and the extrapolated values are shown in a smaller font. The subcooled enthalpy values were extrapolated horizontally because they are a stronger function of temperature than of pressure. Since the subcooled entropy and volume are also stronger functions of temperature, their tables will be made in exactly the same way as the subcooled enthalpy table shown above.

Superheated entropy and volume, on the other hand, are stronger functions of pressure than of temperature. Therefore their tables differ slightly from the superheated enthalpy tables. Initially, however, the procedure for making these tables is the same as for the superheated enthalpy table. Once a complete table of data like Table D.5 is generated for entropy or volume, one of the two copies will become the superheated table. All of the subcooled data points, including the rows containing no superheated values, should be erased. The difference between the superheated enthalpy table and the other two superheated tables is in the direction of the extrapolation. The entropy and volume values should be extrapolated vertically upwards. Although the only requirement is that all the locations adjacent to the real values be filled with extrapolated values, it is better if they are extrapolated to the first row. Using volume as an example, the form of the superheated table for vertical extrapolation is shown in Table D.8 below.

Table D.8 Superheated volume table

99	14	30	50	75	100	125	150	200
-23.638	**2.6488**	1.3277	0.8025	0.5684	0.4314	0.3294	0.2801	0.2058
-20	**2.6746**	1.3277	0.8025	0.5684	0.4314	0.3294	0.2801	0.2058
0	**2.8145**	1.3277	0.8025	0.5684	0.4314	0.3294	0.2801	0.2058
20	**2.9522**	**1.3277**	0.8025	0.5684	0.4314	0.3294	0.2801	0.2058
40	**3.0882**	**1.3968**	**0.8025**	0.5684	0.4314	0.3294	0.2801	0.2058
60	**3.2227**	**1.4643**	**0.8471**	0.5684	0.4314	0.3294	0.2801	0.2058
80	**3.3562**	**1.5305**	**0.8902**	**0.5684**	0.4314	0.3294	0.2801	0.2058
100	**3.4886**	**1.5957**	**0.9321**	**0.5991**	**0.4314**	**0.3294**	0.2801	0.2058
120	**3.6204**	**1.6600**	**0.9731**	**0.6287**	**0.4556**	**0.3509**	**0.2801**	0.2058
131.741	**3.6974**	**1.6975**	**0.9968**	**0.6457**	**0.4693**	**0.3628**	**0.2910**	0.2058
140	**3.7514**	**1.7237**	**1.0133**	**0.6574**	**0.4788**	**0.3710**	**0.2984**	**0.2058**
160	**3.8820**	**1.7867**	**1.0529**	**0.6854**	**0.5012**	**0.3901**	**0.3157**	**0.2212**
180	**4.0121**	**1.8493**	**1.0920**	**0.7129**	**0.5229**	**0.4086**	**0.3320**	**0.2354**
200	**4.1419**	**1.9116**	**1.1306**	**0.7398**	**0.5441**	**0.4264**	**0.3477**	**0.2486**

Although superheated volume and entropy both have a stronger dependence on pressure than on temperature, it seems that volume has a much stronger dependence on pressure. This can be at least partially illustrated by the first few columns of Table D.8. From 14 psia to 30 psia, at any temperature, the specific volume of R-12 is reduced by approximately 50 %. There will definitely be some errors in linear interpolation between these volumes that change this much due to the relatively small pressure change. This is the reason for the previous recommendation that the pressure steps in the lower pressure regions be fairly small. Although the changes in volume with respect to pressure decrease as the pressure increases, the changes are still enough to cause errors in the linear interpolation. Therefore, the superheated volume table will probably need more pressure columns than any other table. The changes in superheated entropy with respect to pressure are not near as dramatic as the changes in volume, and therefore, the superheated entropy table may not require any more pressure columns than the superheated enthalpy table.

D.4.3 Two-phase refrigerant property table

D.4.3.1 General description

The two-phase table is easier to describe than the single-phase table because it consists of only one table. The table contains two-phase properties at the bubble point (saturated liquid) and dew point (saturated vapor) that are tabulated as functions of the refrigerant pressure. These two-phase properties include the following: bubble point pressure, bubble point density, bubble point enthalpy, bubble point entropy, dew point pressure, dew point density, dew point enthalpy, and dew point entropy. For pure refrigerants, the bubble and dew pressures are simply the saturation pressure, and the other bubble and dew properties correspond to the saturated liquid and vapor properties, respectively.

The data from these tables is read into storage arrays by the subroutine "refdata_read" located in REFPROP.f. These arrays are then accessible by all of the property routines through the use of the "common" statement in FORTRAN. These two-phase properties are determined in the various property routines from the table through the use of 1-D linear interpolation between any two columns.

The data in the two-phase table is only accessed if the property routine in use determines that the refrigerant is actually two-phase. As a previously example, the routine "hpt" will typically only use the superheated enthalpy or subcooled enthalpy table, but it is possible, in the case of a zeotrope, that the routine would have to use the two-phase table to obtain the enthalpy. The "hpt" routine will determine which table to use by comparing the given temperature to the bubble (saturated liquid) and dew (saturated vapor) temperature at that pressure. Of course, any routine that has quality (x) as an argument, such as "hpx" or "htx", will use the two-phase table to determine the property.

D.4.3.2 Two-phase table format

The format of the two-phase property table is similar to the saturation property tables found in the back of thermodynamic textbooks. The table consist of nine columns of data corresponding to the refrigerant temperature and the eight properties listed above. The general format for the file containing the two-phase table is shown schematically in Figure D.2 below.

```
# of rows in the two-phase table
# of columns in the two-phase table (currently = 9)
Temp   P_bub   D_bub   H_bub   S_bub   P_dew   D_dew   H_dew   S_dew
...    ...     ...     ...     ...     ...     ...     ...     ...
...    ...     ...     ...     ...     ...     ...     ...     ...
...    ...     ...     ...     ...     ...     ...     ...     ...
...    ...     ...     ...     ...     ...     ...     ...     ...
...    ...     ...     ...     ...     ...     ...     ...     ...
```

Figure D.2 Two-phase table file format

There should be no blank lines anywhere in the data file. The columns of data can be separated by spaces or commas. Although the number of rows in the two-phase table must match the number stated on the first line, the spacing between the temperature values can vary. Currently, the size limit for this two-phase table is 350 rows. This limit can be enlarged by editing REFPROP.INC and the subroutines "Terpl" and "Terpld" in REFPROP.f and changing the dimensions of the appropriate arrays.

There is nothing particularly difficult to understand about the two-phase tables if a pure refrigerant being used. However, when a two-phase table is being made with a zeotrope, the user should be aware of a few things. It was already mentioned that this interpolation routine assumes that the temperature "glide" is linear. This is a better approximation for some zeotropes than for others. The linear temperature "glide" can be better explained with a numerical example. Table D.9 shows part of two-phase table made for the zeotropic mixture of R134a/R32/R125 (52/23/25%). The headings have been added to this example but would not appear in the actual table.

Table D.9 Zeotropic two-phase table

Temp	P_bub	D_bub	H_bub	S_bub	P_dew	D_dew	H_dew	S_dew
0	**44.36**	82.03	12.97	0.0537	34.15	0.6341	111.07	0.2690
2	46.24	81.80	13.61	0.0551	35.72	0.6617	111.35	0.2686
4	48.18	81.57	14.25	0.0565	37.34	0.6903	111.63	0.2682
6	50.18	81.34	14.89	0.0578	39.02	0.7199	111.91	0.2679
8	52.25	81.10	15.53	0.0592	40.76	0.7505	112.19	0.2675
10	54.38	80.87	16.18	0.0606	42.56	0.7821	112.46	0.2672
12	56.58	80.63	16.83	0.0619	**44.42**	0.8147	112.73	0.2668
14	58.85	80.39	17.48	0.0633	46.35	0.8484	113.00	0.2665
16	61.18	80.15	18.13	0.0646	48.33	0.8832	113.27	0.2662
18	63.58	79.91	18.78	0.0660	50.39	0.9192	113.54	0.2659
20	66.06	79.67	19.44	0.0673	52.50	0.9563	113.80	0.2655
22	68.61	79.42	20.10	0.0687	54.69	0.9946	114.07	0.2652
24	71.23	79.18	20.75	0.0700	56.95	1.0340	114.33	0.2649

The temperature "glide" would be easier to explain if the pressure was the independent variable and there were a bubble and dew temperature, but unfortunately, this is not the case. The first bubble pressure in the table is equal to 44.36 psia and is equivalent to a temperature of 0 °F. The closest dew pressure in the table is 44.42 psia and is equivalent to a temperature of 12 °F. Therefore, if this refrigerant was evaporating at approximately 44.4 psia, the temperature would "glide" from 0 °F to 12 °F higher. The assumption with the linear temperature "glide" comes in when the temperature is desired at some point within the phase change region. If the temperature is requested at a pressure of 44.4 psia and a quality of 0.5, the interpolation routines will return a temperature value of 6 °F. Obviously, if the temperature change, at fixed pressure, is not linear, there will be some errors in the interpolations.

The property data in Table D.9 also illustrates another potential problem with the zeotropic two-phase tables. Since the bubble and dew pressures are different for a given temperature, a question arises as to what is the low pressure limit of the two-phase table. The lowest pressure in the table is the dew pressure of 34.15 psia at a temperature of 0 °F. However, the bubble pressure at the same temperature is 44.36 psia. Because of the difference in pressures, there are no bubble-point properties in between 44.36 psia and 34.15 psia. Therefore, any two-phase property routine involving pressure, such as "hpx", will not be able to interpolate since half of the necessary properties aren't in the table. As a result, the low pressure limit in the two-phase table is always taken to be the bubble, and not the dew, pressure.

The same problem exist at the high pressure end of the table, but the situation is reversed in that the dew pressure is taken to be the upper limit. This can be seen by looking a the last dew pressure in Table D.9 and the bubble pressure at the same numerical value. There is a pressure range (56.95 psia to 71.23 psia) for which there are bubble properties but no dew properties. These problems do not exist for the temperature range since there is a complete set of bubble and dew properties at each temperature in the table. The user should be aware of the possibility that the pressure range of the two-phase table may be smaller than anticipated. The table should be made large enough so that the range covered from the bubble pressure at the low temperature to the dew pressure at the high temperature is large enough for the desired application.

D.4.3.3 Guidelines for two-phase property table preparation

Although much less numerous than for the single-phase tables, guidelines for the preparation of the two-phase table will be discussed. As before, these guidelines are not strict rules, but rather, they are an attempt to pass along some of the knowledge acquired by the author during the preparation of detailed two-phase tables for R-12 and R-134a.

D.4.3.3.1 Size and spacing of the table

The decision about the size of the two-phase table is not difficult as those concerning the single-phase tables since there is only one table and the pressure and temperature are not independent of each other. Obviously, the user will want the temperature, and thus pressure, range of the two-phase tables to enclose any possible two-phase property calls. The only additional recommendation is that the pressure range of the two-phase table be larger than the pressure range of any of the single-phase tables. This is because the two-phase table will be used to determine if a correlation like "hpt" will be used in the subcooled, two-phase, or superheated region. For example, if the single-phase tables go down to 5 psia, but the two-phase table only goes as low as 10 psia, a problem could occur. If the actual property call was "hpt" at 8 psia and -10 °F, the routine would not correctly determine which table to use because it could not get the correct saturation temperature at 8 psia from the two-phase table. If the refrigerant is a zeotrope, then the discussion about the pressure range in D.4.3.2 should be kept in mind as well.

As mentioned, the spacing between the temperature values is variable and totally up to the user. With a current limit of 350 rows, it is quite easy to take steps of 0.5 °F or 1 °F and still cover a wide range of temperatures and pressures. If the user wishes to have a smaller table, or if more accuracy is needed in a particular range of temperatures and pressures, then the spacing could be made larger at the extreme ends of the table and smaller in the middle of the table. This variable spacing was used in the R-12 and R-134a tables because of the numerical partial derivative described in D.2.4. Since the resulting mass flow rate through the capillary tube is a very important system variable, the temperature spacing in this region of the table was made smaller than elsewhere in the two-phase table. The bottom line is that the user shouldn't have any problem obtaining a two-phase table with sufficient accuracy and range with 350 rows of data.

D.4.3.3.2 Data generation

The data for the two-phase table can, in most cases, be made very easily. If a property package is used to generate the data, the data can usually be written to the screen or a file in the exact form needed for the two-phase table. Once the data is in a file, the only remaining task is to add the first two lines that indicate how many rows and columns are present in the table. The data for the R-12 and R-134a tables were made with EES simply by using a nine column parametric table with the temperature being varied over the desired range.

D.4.3.3.3 Example two-phase table

Although part of a zeotropic two-phase table was shown before, another small two-phase table will be shown for completeness. Table D.9 shows a small R-134a two-phase table as it would appear in the data file, with the exception of the column titles.

Table D.10 Two-phase refrigerant table for a pure refrigerant

Temp	P_bub	D_bub	H_bub	S_bub	P_dew	D_dew	H_dew	S_dew
11								
9								
-10	16.675	85.435	8.436	0.0194	16.675	0.3715	100.148	0.2233
-9	17.090	85.330	8.726	0.0200	17.090	0.3802	100.292	0.2232
-8	17.514	85.225	9.018	0.0206	17.514	0.3891	100.436	0.2230
-7	17.945	85.120	9.309	0.0213	17.945	0.3982	100.580	0.2229
-6	18.385	85.015	9.602	0.0219	18.385	0.4074	100.723	0.2228
-5	18.834	84.910	9.895	0.0226	18.834	0.4168	100.866	0.2227
-4	19.291	84.805	10.188	0.0232	19.291	0.4264	101.010	0.2225
-3	19.757	84.699	10.483	0.0239	19.757	0.4362	101.153	0.2224
-2	20.232	84.593	10.777	0.0245	20.232	0.4461	101.295	0.2223
-1	20.716	84.487	11.073	0.0251	20.716	0.4562	101.438	0.2222
0	21.209	84.380	11.369	0.0258	21.209	0.4665	101.580	0.2220

The only noticeable difference between this table and the one shown earlier for the zeotrope is that the bubble and dew pressures are the same at every temperature. In this case, the bubble and due temperatures are simply the saturation temperature.

D.5 Conclusion

The purpose of this document was threefold: to cover the benefits of interpolation; to show how the interpolation package is used within the RFSIM model; and to give enough information about the property routines and tables so that someone with access to the FORTRAN code could make new refrigerant tables and use them. The interpolation package was inserted into the RFSIM model primarily for the speed and the flexibility associated with the interpolation method. Tables for three pure refrigerants and one zeotropic mixture currently exist in the interpolation package at the ACRC. It is a fairly simple procedure to switch between refrigerants and add new refrigerant capability to a thermal model such as the RFSIM model. The most difficult and time consuming part of adding a new refrigerant is the generation of the single- and two-phase property tables. However, if the recommendations put forth in D.4.2 and D.4.3 are followed, the time spent making the tables can be greatly reduced. In the author's opinion, any time spent implementing the interpolation package or adding refrigerants will be well worth the effort.

Appendix E

System Model Validation

E.1 Introduction

Before a simulation model can be used for the purposes for which it was developed, it must be validated, or proved to be accurate. For most models, the validation process is simply a matter of taking experimental data and comparing it with the model predictions. If the predictions of the model are within some allowable tolerance of the experimental quantities, then the model is validated for at least the range of operation tested. Once validated, that model or correlation can usually be used with relative ease in other situations.

The validation process for the RFSIM model is also a matter of taking experimental data and comparing it with the model predictions. However, the difference between the RFSIM model and other more physical models is that the accuracy of RFSIM depends on several input parameters that are somewhat difficult to obtain. Therefore, even if the RFSIM model has been "validated", its accuracy when used in other situations will still depend on the user-supplied input parameters. In view of this fact, the validation process for a model like RFSIM may be best described as a validation of the modeling procedure itself and not as statement that the model will always predict all the variables within some tolerance.

The modeling procedure that has to be validated includes all the assumptions, physical relations, and correlations contained in the model: two- and three-zone heat exchanger equations; finite-difference capillary tube-suction line heat exchanger equations; refrigerant-side pressure drop and heat transfer correlations; etc. In other words, the validation of RFSIM involves the evaluation of the model's ability to predict key variables assuming that the user-specified parameter values are accurate. Therefore, it was necessary to make every effort to obtain accurate parameter values.

Currently, there are eight parameters that have to be supplied by the user that are estimated from data. The parameter names, along with their definitions, are shown below:

UAf	overall heat transfer conductance of the fresh-food section
UAz	overall heat transfer conductance of the freezer section
haircond	air-side convection coefficient of the condenser at the nominal fan speed
hairevap	air-side convection coefficient of the evaporator at the nominal fan speed
hAcomp	overall heat transfer conductance (compressor shell to the air stream)
vdotcond	volumetric air flow rate over the condenser
vdotevap	volumetric air flow rate over the evaporator
frecirc	air recirculation fraction from the condenser exit to the condenser inlet

The process of estimating these parameters for the Whirlpool refrigerator is described by Krause and Bullard (1996). Similar techniques were used to obtain the parameters for the Amana refrigerator.

The fact that these eight parameters were estimated from data does not mean that the model will not work for other refrigerators without a similar estimation process. It just means that accurate values were needed in order to evaluate the validity of the modeling methods within the RFSIM model. Of course, the accuracy of the RFSIM model when used for other refrigerators will be dependent on the above parameters. Therefore, the purpose of the modeling done with the RFSIM model should dictate the amount of effort spent obtaining the input parameters. If the model is just being used to explore trends in refrigerator performance, then rough estimations of the parameters may be adequate. However, if the model is being used to design a refrigerator or to replace laboratory testing, then more careful approximations of the parameters would be justified.

E.2 Modes of model operation

Before the results of the model validation are presented, the different modes of operation for RFSIM will be discussed. A more comprehensive discussion of the operational modes of the model is also presented in Appendix A. Goodson and Bullard (1994) described RFSIM as a model that would operate in two modes: design or simulation. In the design mode of the old model, the user was required to specify one refrigerant property at each of the heat exchanger exits. These exit properties could be superheat or quality for the evaporator and subcooling or quality for the condenser. The current design mode can operate in this way, but it was made more flexible so that any two refrigerant properties in the system can be specified.

The rationale behind the design mode was twofold: to be a stepping stone for the development of the more complicated simulation model, and to provide researchers and designers with a reliable tool for the "rough" design of the major system components (other than the capillary tube). The two major differences between the design and simulation modes of the model are the capillary tube-suction line heat exchanger (ct-slhx) model and the charge conservation equations. In the design mode, the capillary tube model is not used to predict the mass flow rate in the system and the heat transfer in the suction line heat exchanger. In addition, the total refrigerant charge is not specified by the user, but it is calculated by the charge equations. Since the calculated total charge is not used by any other equation in the model, any error in the charge equations do not affect the overall accuracy of the model. In other words, the ability of the design model to predict accurately is not dependent on the accuracy of the ct-slhx model or the charge inventory equations.

One obvious drawback of the design mode, however, is its inability to predict system performance at off-design conditions. Since the refrigerant states at the heat exchanger exits are fixed, any change to the model parameters will result in an "artificial" solution because in an actual refrigerator, the exits would change. The simulation mode was created to address this problem and allow the model to predict what would really happen in a refrigerator when parameters are changed. As indicated, the simulation model uses a ct-slhx model, and the total charge is specified by the user in conjunction with the refrigerant charge inventory equations. The addition of these two sets of equations allows the simulation model to operate without any refrigerant state information (i.e. superheat and subcooling). Therefore, the simulation model will predict the exits of the heat exchangers as well as important variables such as COP, evaporation capacity, and power consumption. Thus, the simulation model can show the effect of changing a parameter like the capillary tube diameter or evaporator area on all the operating variables.

In addition to the "design" and "simulation" modes discussed so far, the current model will operate in between these two extremes. For example, if the user would like to design a ct-slhx that would yield a certain superheat at a specified condition, he/she can specify the superheat and include the ct-slhx equations in the system model. This ability to mix features of the design and simulation modes was useful in the evaluation of the ct-slhx model and the charge inventory equations. The model could be run with superheat specified and either the capillary tube model or the charge equations and the relative error caused by each could be evaluated. It was through such an exercise that the author determined that much of the error in the simulation model reported by Goodson and Bullard (1994) was due to the void-fraction correlation used in the charge inventory equations. Subsequently, other void-fraction correlations were tried, and one was found that significantly improved the accuracy of the simulation mode, as will be discussed below.

E.3 Amana Model Validation

E.3.1 Experimental Results

Goodson and Bullard (1994) previously reported on the model validation that was conducted with the data from the Amana refrigerator. The refrigerator used in the analysis was a 18 cu. ft., top-mount Amana (model TC18MBL) that was charged with the manufacturer's recommended 8 oz. of R-12. The data used for the analysis was taken in the spring of 1994 and consisted of 16 points at the four ambient temperatures of 60 °F, 75 °F, 90 °F, and 100 °F. The experimental techniques used to obtain this data are described by Rubas and Bullard (1993). For each ambient temperature, the temperature of the air entering the evaporator was varied over a range representative of normal cycling operation. Of the 16 points, all the points were superheated, however only seven were sufficiently subcooled to used in refrigerant-side energy balances on the evaporator and condenser.

E.3.2 Design Mode Validation

As Goodson and Bullard (1994) reported, the model operated fairly well when in design mode. Only the subcooled points were used in this analysis since design mode requires that the exits of the heat exchangers be specified. The model was run in design mode for the seven subcooled points by specifying the following parameters: power dissipated by the fresh-food and freezer heaters, atmospheric pressure, evaporator and condenser fan powers, average fresh-food and freezer compartment temperatures, ambient temperature, and the subcooling and superheat. Figure E.1 below is a "box" plot that shows the error in the model predictions relative to the experimental data for the model in design mode. The "measured" mass flow rate values were determined from a refrigerant-side energy balance about the capillary tube-suction line heat exchanger and the evaporator. The technique used to calculate the "measured" mass flow rate from other experimental measurements is described by Krause and Bullard (1996).

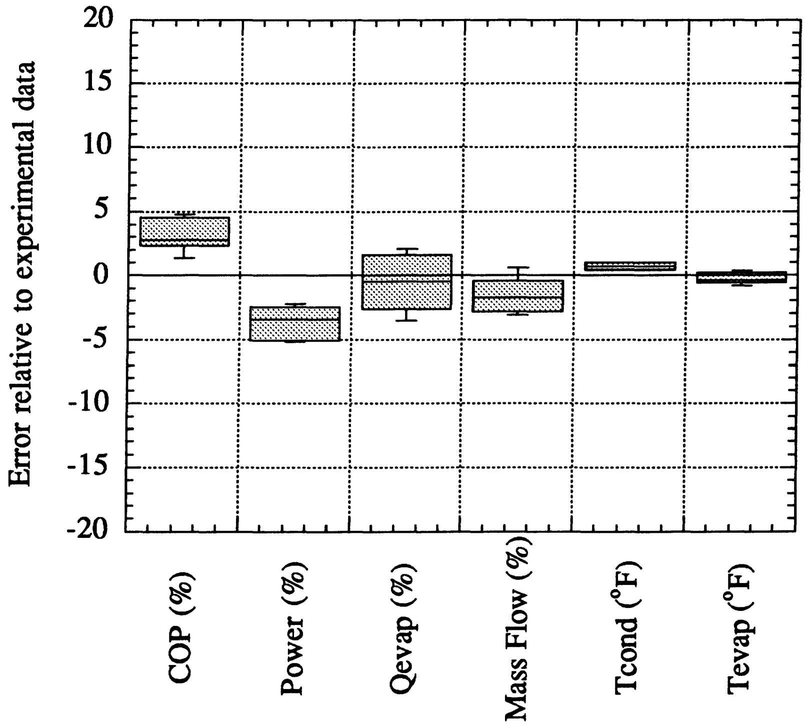


Figure E.1 Model error in design mode (Amana)

The line in the center of the box for each variable shows the median of that variable, and the upper and lower edges of the box show the limits of +/- 25% of the variable population. The lines extending from the top and bottom of the box show the maximum and minimum values for each variable.

As Figure E.1 shows, the model predicts the evaporation and condensation temperatures within 1 °F of the actual values. There is a slight underprediction in the mass flow rate (~ 2.5%)

and fairly even distribution (± 3%) of the errors in the evaporation capacity. The main story, however, with the model in design mode is the underprediction of the system power of about 4% and the subsequent overprediction of the COP of about 3.5%. It was determined by comparison with the measured power that the compressor map, when supplied with the correct evaporation and condensing temperatures, underpredicts the power by about 3.5%. This underprediction of power is reflected in the model results as shown above.

Although the overall results of the design model were very good, it was desired to eliminate the effect of the errors originating in the compressor map. To do this, new compressor maps were made for both the mass flow rate and the power variables. The new maps were made with the generic map points obtained from the compressor manufacturer, augmented by data points experimentally obtained from the Amana refrigerator. Although the mass flow predictions hardly changed, the mentioned 3.5% error in compressor power was effectively eliminated. These new compressor maps were inserted into the model and it was run for the same seven subcooled points previously mentioned . The errors between the model predictions and the experimental data were again calculated and are shown in Figure E.2.

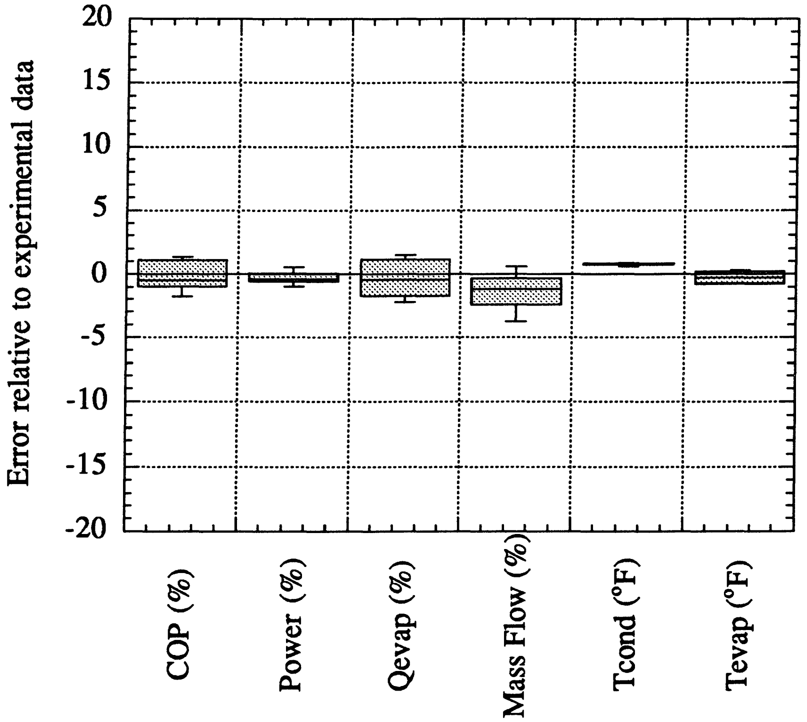


Figure E.2 Error in design mode with new compressor maps (Amana)

The new compressor maps greatly reduced the errors in the COP (± 1.5%) and the system power predictions (± 1%). Although the mass flow rate errors were virtually unchanged, the distribution of the evaporation capacity errors was slightly tightened to ± 2%. Figure E.2 shows that the model, at least in design mode, has the potential to predict the key variables very accurately if the compressor maps are accurate for the conditions to be modeled. Although the data points used in the validation were at "near-normal" conditions, the values of the key input parameters (e.g. heat exchanger volumetric flow rates and air-side heat transfer coefficients) were originally estimated from a much richer set of data spanning a range of off-design conditions.

E.3.3 Simulation Mode Validation

As reported by Goodson and Bullard (1994), there was significant error in the simulation mode of the RFSIM model. It was originally thought that the majority of the error was a result of the capillary tube model. However, as mentioned in E.2, it was discovered by running the model in various modes that most of the error actually originated with the charge inventory equations.

When run in design mode with the superheat and subcooling specified, the charge equations consistently overpredicted the amount of charge in the system. Subsequently, when the model was run in simulation mode with the charge fixed at the correct amount, the model incorrectly predicted the exits of the heat exchangers. For example, the model would predict that the condenser exit had a quality of 5% when it actually had 7 °F of subcooling. To be sure, there was and still is error in the capillary tube model, but the charge equations were the primary source of error. It was very difficult to distinguish between these two sources of error because of the capillary tube model's strong dependence on the refrigerant state at the condenser exit. In other words, any error in the charge equations would be transmitted or even magnified by the capillary tube model.

The problems in the simulation mode were solved in part by trying other void-fraction correlations in the charge equations. It was reported by Goodson and Bullard (1994) as well as other authors (Rice, 1987 and Farzad, 1993) that the Hughmark correlation was the best overall for the prediction of charge inventory in the two-phase regions of heat exchangers. Unfortunately, Hughmark's correlation did not predict well when used in RFSIM. One paper was found (Kuijpers, 1987) that concluded that the Premoli correlation was the best void-fraction correlation for small refrigeration systems such as domestic refrigerators. When this correlation was tried in the design mode of the model, the errors between the model predictions and the actual charge ranged from -5% to +2%. When the new correlation was tried in the simulation model, the results were equally encouraging.

As mentioned, the simulation mode of the model does not require any refrigerant state information. Therefore, all 16 data points should have been available for the analysis. Unfortunately, the model was only able to solve two of the two-phase condenser exit points due to a logic limitation in the ct-slhx model. Currently, the ct-slhx model will only handle a two-phase inlet to the suction line (evaporator exit) if the refrigerant on the capillary tube side of the suction line heat exchanger is also a two-phase mixture. Although all the data points actually had superheated evaporator exits, the model solution of the points "wanted" to make the evaporator exits two-phase. More information about this logic limitation in the ct-slhx model is presented in Appendix C.

After closer examination, it was discovered that one of these two points that were solved had very questionable data values. The heat exchanger exits of this point were quite different than the other points at the same ambient temperature (60 °F). From the trend in the data, this point should have had superheat and subcooling values of about 15 °F and 5 °F, respectively. However, it actually had 5 °F of superheat and a two-phase condenser exit. This discrepancy indicated that something was wrong with this data point, and therefore, it was not used in the analysis.

In the end, the model was run in simulation mode for the same seven points mention in E.3.2 and one additional two-phase condenser exit point (90 °F ambient). Again, the errors between some of the calculated variables and the experimental data were calculated. These errors in the model predictions are shown below in Figure E.3.

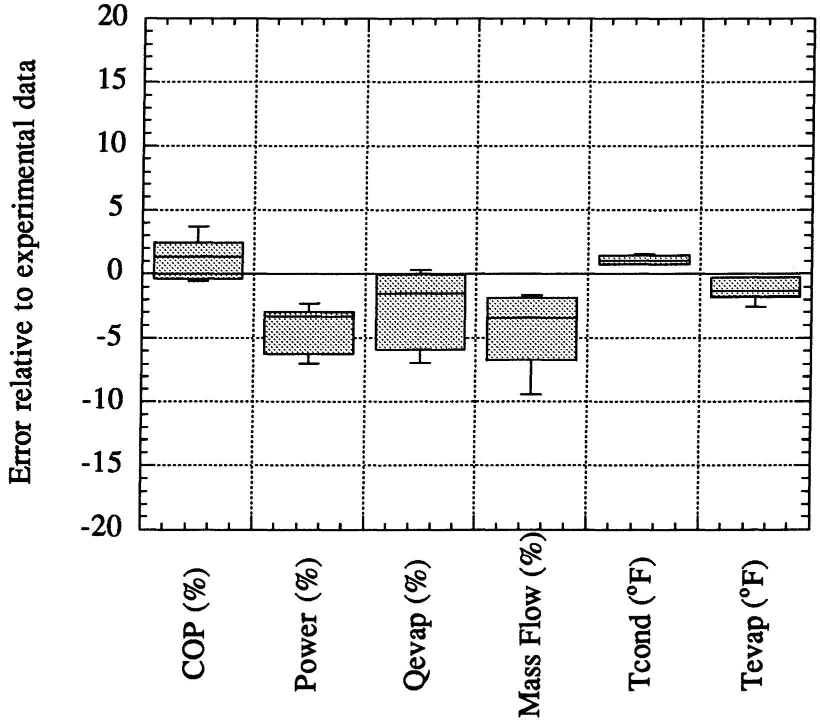


Figure E.3 Model error in simulation mode (Amana)

Although these errors are certainly worse than the design mode results shown in Figure E.1, they are much better than those reported by Goodson and Bullard (1994) for the simulation mode. Before, the <u>average</u> errors in the mass flow rate and evaporation capacity were -22% and -17%, respectively. Now, both of these average errors are within 4% of the experimental data. The fact that there is still some significant underprediction of the mass flow and evaporator capacity is evidence that the capillary tube is still underpredicting mass flow as Goodson and Bullard (1994) asserted. This underprediction may be a result of the actual capillary tube diameter being larger than the nominal value used in the simulations. The change in the void-fraction correlation has also improved significantly the predictions of evaporation temperature (within 2 °F), system power (~ 5%), and COP (2 to 3%).

The results in Figure E.3 were generated with the generic compressor maps. As with the design mode of the RFSIM model, the predictions from the simulation mode will also improve if the new compressor maps are used. The errors in the model predictions when the RFSIM model is run in simulation mode with the new maps are shown below in Figure E.4.

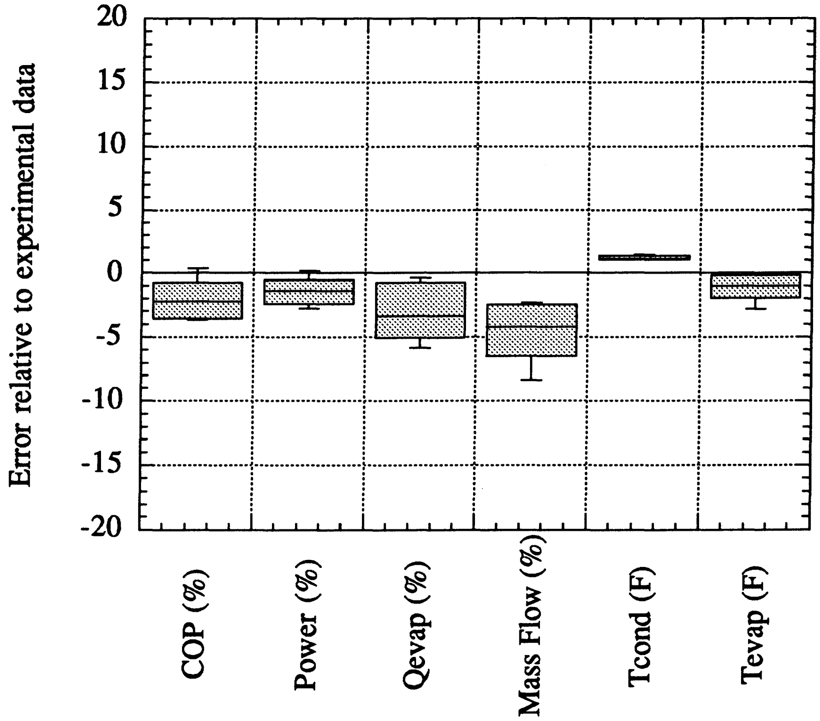


Figure E.4 Model error in simulation mode with the "new" maps (Amana)

The new maps cause the system power predictions to become higher in magnitude, and thus, more accurate (~ 2%) than before. However, the magnitude of the COP errors remain virtually unchanged at 2 to 3 %, but they are underpredictions instead of overpredictions. Regardless of which compressor maps are used, the simulation mode of the model predicts the key variables, for the majority of the data, within acceptable ranges (less than 5% error for COP, power, evaporator capacity, and mass flow and within 2 °F for the condensing and evaporating temperatures). Although it would have been better to use all of the available Amana data points for the analysis, it was not deemed absolutely necessary since the validation process was also being carried out with the Whirlpool refrigerator as will be discussed in the following sections.

E.4 Whirlpool Model Validation

E.4.1 Experimental Results

The refrigerator used in the analysis was a 20 cu. ft., top-mount Whirlpool refrigerator (model ET20PK) that was charged with the manufacturer's recommended 8.25 oz. of R-12. The data used for the analysis was taken in the fall of 1995 and consisted of 26 points at the four ambient temperatures of 60 °F, 75 °F, 90 °F, and 100 °F. The experimental techniques used to obtain this data are described by Krause and Bullard (1996). For each ambient temperature, the temperature of the air entering the evaporator was varied over a wide range of operating conditions. These operating conditions consisted both of compartment temperatures normally seen in a cycle as well as higher compartment temperatures necessary to obtain subcooling at the condenser exit. Of the 26 data points, all the points were superheated while only 12 points were subcooled. There were five subcooled points each at 60 °F and 75 °F and two at 90 °F.

E.4.2 Design Mode Validation

As with the analysis of the Amana data, only the subcooled points were used since design mode requires that the exits of the heat exchangers be specified. The model was run in design mode for the 12 subcooled points by specifying the parameters listed in E.3.2. The model predictions were compared to the experimental data and the errors for the same six variables were calculated. The "measured" mass flow rates, as before, were determined by a refrigerant-side energy balance. These errors for the design mode are shown below in Figure E.5.

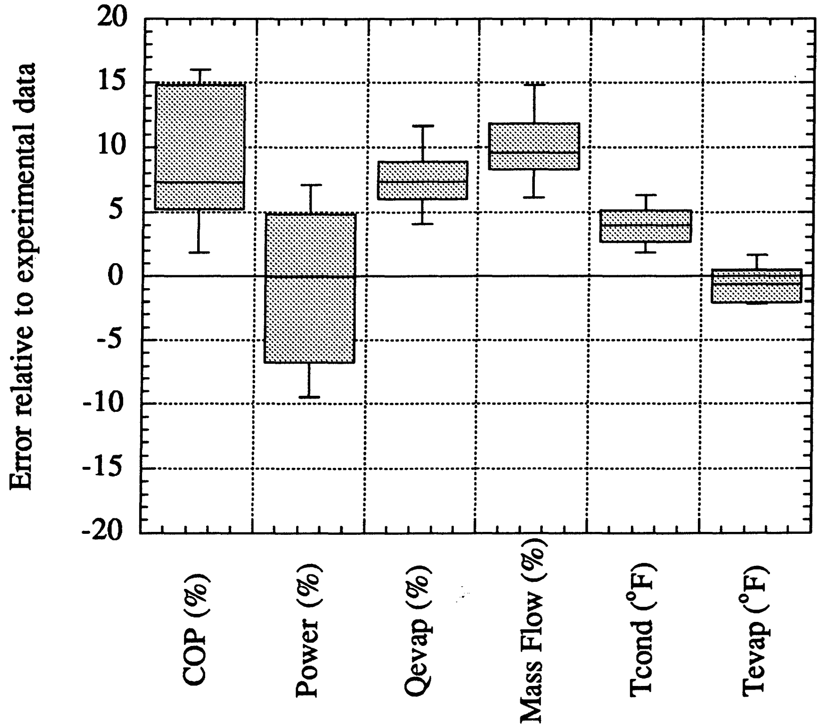


Figure E.5 Model error in design mode (Whirlpool)

Unfortunately, the model in design mode did not predict as well as it did for the Amana test unit. From Figure E.5, it can be seen that the mass flow rate is significantly overpredicted (~10%). This error, in turn, caused the evaporation capacity and the COP to also be overpredicted. To understand why the mass flow rate is overpredicted, one must look at the compressor map. It was noted by the authors of the current work and previous researchers (Krause and Bullard, 1996) that the compressor map for the rotary compressor in the Whirlpool tends to overpredict mass flow (at least for available subcooled data).

As mentioned in E.1, the goal of the model validation was to see if the modeling procedure that had been employed in RFSIM was correct. If there was already considerable error in the design mode due to the compressor maps, it is likely that the simulation mode results would be even worse since more assumptions are made. In that case, it would be hard to evaluate the accuracy of the capillary tube model and charge inventory equations. Therefore, new compressor maps were again made.

Originally, these new maps were made by putting all the data together (ACRC experimental data plus the manufacturer's data) and using the least-squares technique to determine the nine parameters of the new bi-quadratic curve fit. Although this new map predicted the experimental and manufacturer's data fairly well, the "shape" of the curve fit was incorrect at condensing and evaporating temperatures for which there was no data (such as -20 °F evaporating temperature). Consequently, it was decided to correct the generic map instead of making a completely new one. This was done by using least-squares to obtain an intercept (A) and slope (B) parameter that caused the predictions of the mass flow and power map to better match the experimental data (e.g. y = A + B * Generic Map).

It is not known for sure why the generic compressor maps for the reciprocating compressor in the Amana and the rotary compressor in the Whirlpool behave so differently. One possible reason is the differences in the path of the suction gas after it enters the compressor. Since the suction gas entering the rotary compressor goes straight into the cylinder, it is much more sensitive to the differences between the standard compressor testing condition and the conditions actually seen in a refrigerator compressor. In the reciprocating compressor, the suction gas enters the low-side sump where it is heated before it enters the cylinder. Therefore, whether the suction gas was 70° F or 90 °F does not matter as much in a reciprocating compressor since it will be heated to a much higher temperature before compression in either case. In view of the preceding argument, an attempt was made to introduce a physical correction to improve the accuracy of the generic mass flow map.

If it assumed that the compressor speed, displacement, and volumetric efficiency are constant, the mass flow rate can be scaled by the following relation:

$$MassFlow_{actual} = \left(\frac{\rho_{actual}}{\rho_{map}}\right) \times MassFlow_{map} \qquad \text{(E.1)}$$

However, when the preceding relation was used to scale the values given by the generic compressor map, the agreement with the data did not improve significantly. Perhaps the mass flow rate is affected by other variables (e.g. shell temperature) which differ between the standard condition and the actual refrigerator operation conditions. In any case, it was decided to proceed with the model validation with the new compressor maps.

The model, in design mode with the new compressor maps, was run again for the 12 subcooled points. As before, the model predictions were compared to the experimental data and the errors for the same six variables were calculated. These errors for the design mode are shown below in Figure E.6.

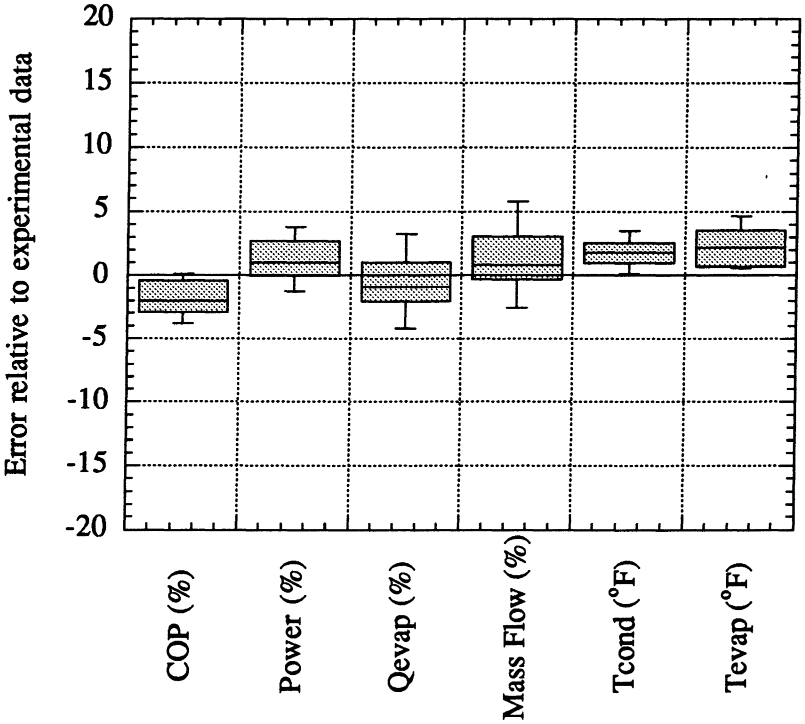


Figure E.6 Model error in design mode with the new maps (Whirlpool)

The more accurate mass flow and power maps significantly improved the design mode predictions of the Whirlpool data. Although there are some mass flow errors as high as 6%, the vast majority of the errors are in the 2 to 3% range. The reduction in the mass flow rate errors translate into smaller errors in the evaporation capacity (most within ±2%). The scatter in the power error was reduced, but there is still a consistent overprediction of 2 to 3 %. This overprediction in the power leads to the observed negative error in the COP (2% average). The corrected compressor maps decreased the average error in the condensing temperature from 4 °F to 2 °F, but it caused the average evaporating temperature error to become slightly worse at 2 °F. Although the overall agreement was slightly worse than with the Amana data, all the variables are predicted within acceptable limits.

E.4.3 Simulation Mode Validation

Before the simulation mode of the RFSIM model could evaluated, the void-fraction correlation of the charge inventory equations had to be chosen. Although not mentioned in the earlier discussion, both the Whirlpool data set and the Amana data was used to evaluate the void-fraction correlations. The design mode of the model was used to predict the amount of charge in the system for all 12 subcooled data points. When the Premoli correlation was used, the errors between the model predictions and the actual charge ranged from -2% to +3.5%. This information, along with the results from the Amana data set, pointed to Premoli as the best void-fraction correlation. Therefore, the Premoli correlation, along with the new compressor maps discussed in the previous section, was used in all of the following analysis.

Fortunately, there were no limitations placed on the Whirlpool data set by the capillary tube-suction line heat exchanger model as was the case with the Amana data. All 26 of the Whirlpool data points (subcooled and two-phase condenser exits) were used in the evaluation of the simulation mode of the model. The errors between the model predictions and the experimental data were calculated and are displayed in Figure E.7 below.

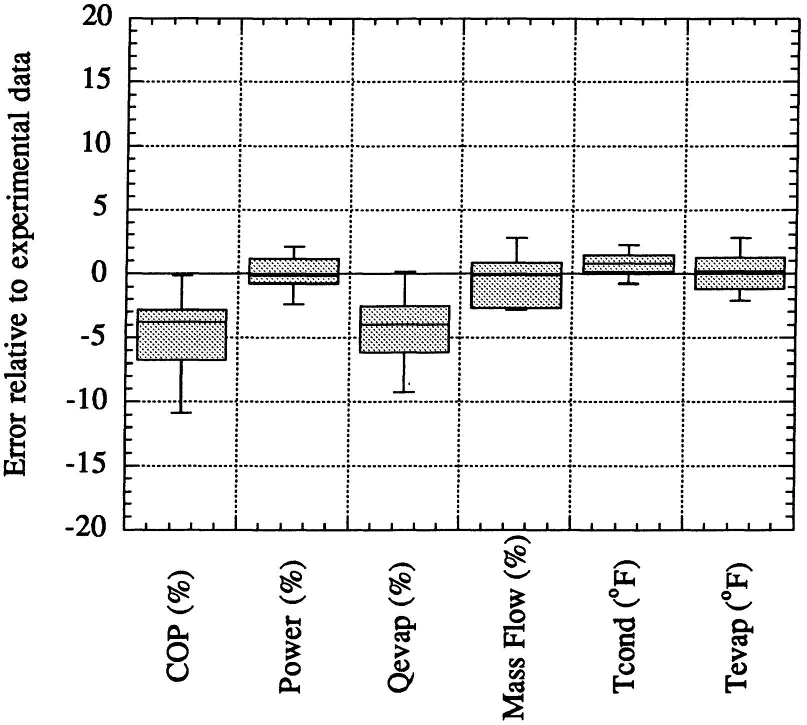


Figure E.7 Model error in simulation mode with the new maps (Whirlpool)

The simulation mode predictions of power, evaporating temperature, and condensing temperature shown in Figure E.7 are actually better than the design mode predictions shown in Figure E.6. However, there is a very noticeable underprediction in evaporation capacity that causes an underprediction in the COP.

To understand why the evaporation capacity is underpredicted, one must look at the mass flow rate predictions. Although not mentioned in the Amana discussion, the mass flow rate errors in the simulation mode results consist only of subcooled condenser exit points since they are only ones with experimental values of mass flow. In other words, the results in Figure E.7 are slightly misleading and would be probably be worse if experimental values of mass flow were available for the two-phase condenser exit points. To illustrate the difference between these two groups of data, the model errors were calculated just for the subcooled condenser exit points and are shown below in Figure E.8.

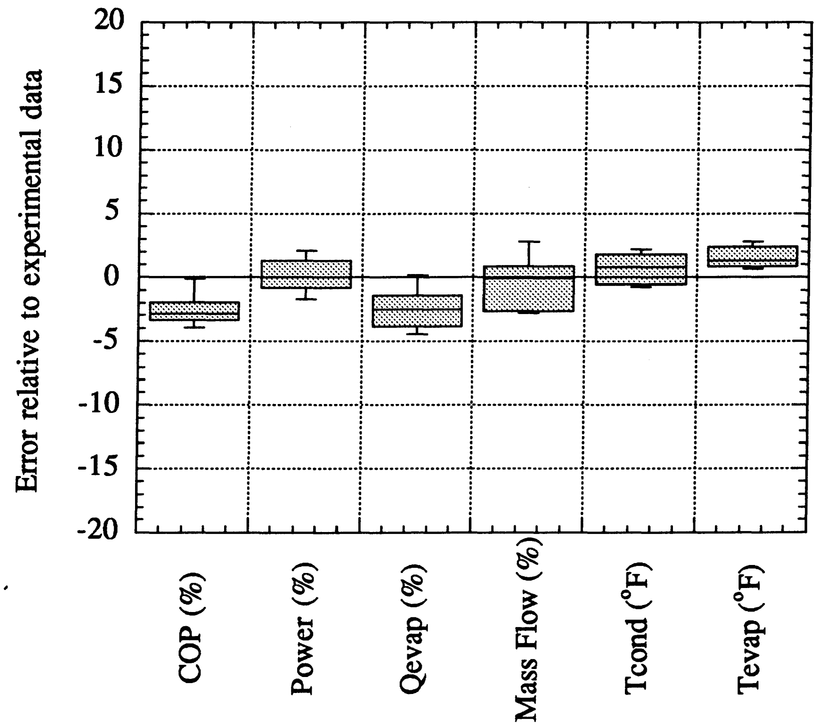


Figure E.8 Model error in simulation mode with the new maps for the subcooled data

Thus, if just the subcooled condenser points are considered, the model predictions in evaporation capacity and COP are very good (within 3% of data). If the errors in the evaporating temperatures in Figures E.7 and E.8 are compared, a basic difference in the subcooled and two-phase condenser exit points can be seen. There is a slight overprediction in evaporating temperature for the subcooled data points, but there is a slight underprediction for the two-phase points. The lower predictions of evaporating temperature for the two-phase points cause the mass flow rate to be underpredicted (from the compressor map). These low mass flow predictions (although not shown on Figure E.7) cause underpredictions in evaporation capacity and COP as seen in Figure E.7.

Although it is not known for sure, the evaporating temperatures of the two-phase points may be underpredicted because of the capillary tube-suction line heat exchanger model. It has been noted by the current author and other researchers (Liu and Bullard, 1996) that the ct-slhx model has a tendency to underpredict the mass flow rate for most of the available data. Liu has also found that the predictions of the ct-slhx model become better when a larger portion of the capillary tube contains subcooled liquid. In other words, the predictions for data points with high subcooling at the capillary tube inlet will usually be better than those for data points with low subcooling or two-phase inlets. Obviously, there is no subcooling at the inlet for the two-phase condenser exit points. Therefore, the ct-slhx model underpredicts the mass flow rate for the two-phase points. In order for the compressor map to predict an equal value of mass flow, the evaporating temperature must be lower.

It is difficult to identify the cause of the errors because all of the equations (e.g. compressor maps, ct-slhx model, charge conservation equations) in the RFSIM model depend on and affect the other equations in the model. This is especially true when dealing with the relatively small errors present in Figure E.7 and E.8. Some of the errors present in the compressor maps, ct-slhx model, and charge conservation equations are reduced and some are magnified when they are used together in the system model. However, it is important to realize that even if all the data points are used as in Figure E.7, the average predictions of the simulation mode of the model are still within 4% for COP and evaporation capacity.

E.5 Conclusion

In order to "validate" RFSIM, accurate parameter estimations were obtained for both refrigerators in the study. With these experimentally obtained parameters, the accuracy of RFSIM has been evaluated for the design and simulation mode for two different refrigerators over a range of operating conditions.

From the results of both refrigerators, it appears that the accuracy of the compressor maps is the limiting factor in the accuracy of the design mode of the model. This observation was not as pronounced in the reciprocating compressor of the Amana as it was in the rotary compressor of the Whirlpool. Even with the generic compressor maps, the average errors of the design model when run for the Amana data were all less than 4% for the COP, system power, evaporation capacity, and mass flow and were less than 1 °F for the evaporating and condensing temperatures. For the Whirlpool, however, it was necessary to use the new compressor maps obtained from in-situ calorimetry of the specific compressor. The majority of the errors of the design model when run for the Whirlpool data were less than 3% for the four non-temperature variables and were less than 3 °F for the evaporating and condensing temperatures.

In addition to the compressor maps, the accuracy of the simulation mode is very dependent on the choice of the void-fraction correlation in the charge inventory equations and on the capillary tube model. At this point, it seems that Premoli's correlation is the best void-fraction correlation for predicting two-phase charge inventory in domestic refrigerators. Although the equations and correlations used in the capillary tube model are not perfect, most of the errors in the capillary tube predictions are a result of the uncertainty in the capillary tube diameter.

It appears that an uncertainty in the diameter of the Amana capillary tube or the underprediction in power by the generic compressor map is responsible for some portion of the error in the simulation mode when run for the Amana data. Even so, the majority of the errors were less than 4% for the four non-temperature variables and were less than 2 °F for the evaporating and condensing temperatures. For the simulation mode of the model run with the Whirlpool data, the agreement with experimental data is not as good for the two-phase condenser exit points as it is for the subcooled exit points. However, the average errors of the COP and evaporation capacity are still within 4% for all the data. The majority of the errors of the other four variables are distributed around the zero error line in a fairly narrow range (1% for power, 3% for mass flow, and 1.5 °F for evaporating and condensing temperatures).

Since the majority of the errors in the model predictions are less than 5% for the four non-temperature variables and less than 2 °F for the evaporating and condensing temperatures, it can be said that the modeling procedure in RFSIM has been validated. RFSIM is sufficiently accurate, given good parameter values, to be used as a design or simulation tool by researchers or refrigerator designers. Even if exact parameter values are not available, RFSIM will still be useful for exploring trends and design trade-offs. It is reasonable to assume that the general modeling procedure employed in RFSIM could be extended to model refrigerator types other than the ones used in this study.

References

Farzad, M., and D.L. O'Neal, "The Effect of Void Fraction Model on Estimation of Air Conditioner System Performance Variables under a Range of Refrigerant Charging Conditions," Rev. Int. Froid. Vol. 16, No. 5, 1993.

Goodson, M.P., and Bullard, C.W., "Modeling a Refrigerator/Freezer System," ACRC TR-61, University of Illinois at Urbana-Champaign, 1994.

Krause, P. E., and C. W. Bullard, "Cycling and Quasi-Steady Behavior of a Refrigerator," Transactions of the American Society of Heating, Refrigeration, and Air Conditioning Engineers, 102:1, in press, 1996.

Kuijpers, L., and M. Janssen, and J. de Wit,"Experimental Verification of Liquid Hold-Up Predictions in Small Refrigeration Heat Exchangers," 17th International Congress of Refrigeration, Wien 1987, B, pp. 307-315.

Liu, L., and C.W. Bullard, forthcoming M.S. Thesis, University of Illinois at Urbana-Champaign, 1996.

Rice, C.K., "The Effect of Void Fraction Correlation and Heat Flux Assumption on Refrigerant Charge Inventory Predictions," ASHRAE Transactions, Vol. 93, Part 1, 1987, pp. 341-367.

Appendix F

Capillary tube model parametric analysis

F.1 Introduction

In an effort to validate the capillary tube-suction line heat exchanger simulation (CTSLHXSIM) model, three of the parameters that affect the solution of the model have been examined: the capillary tube diameter, the capillary tube roughness, and the two-phase friction factor correlation. The first two parameters are inputs to the model, and their uncertainty directly affects the model results. The third parameter is a correlation that is used within the model that also affects the model results. Although there are more parameters that affect the model output, these three seem to be the most significant as far as mass flow rate predictions are concerned.

F.2 Diameter uncertainty

Of all the parameters in the CTSLHXSIM model that affect the results, the diameter of the capillary tube is the most significant. Sweedyk (1981) reports that the commercially available tolerance on capillary tubes is +/- 0.001 in. For the 0.033 in. diameter capillary tube in the Amana refrigerator, this translates into a diameter uncertainty of +/- 3.03%. Sweedyk (1981) also reports that the mass flow uncertainty on a 0.030 in. adiabatic capillary tube due to the diameter uncertainty can be as high as +/- 9 %. Since the diameter of the capillary tube is usually an input to the CTSLHXSIM model, there is an uncertainty in the mass flow rate predicted by the model.

To obtain an estimate of the magnitude of this error, the model was run for a 7-point data set with three different diameters: one tolerance below the nominal, the nominal, and one tolerance above the nominal. The data used in the comparison came from a 0.031in. diameter ct-sl hx tested in the test stand at the ACRC (Greenfield, 1994). The results of the model runs are shown on the following page in Figure F.1

As Sweedyk reported, there is a large range in the mass flow rate predictions of the model when the diameter is varied by +/- 0.001 in. The average percentage difference from the nominal case mass flow is +9.4 % and -7.9 %. Although these numbers were obtained from just one set of data, there are consistent with the flow deviations reported by Sweedyk. The percentage deviations from the nominal would be greater if the capillary tube had a smaller nominal diameter and lower if the capillary tube had a larger nominal diameter.

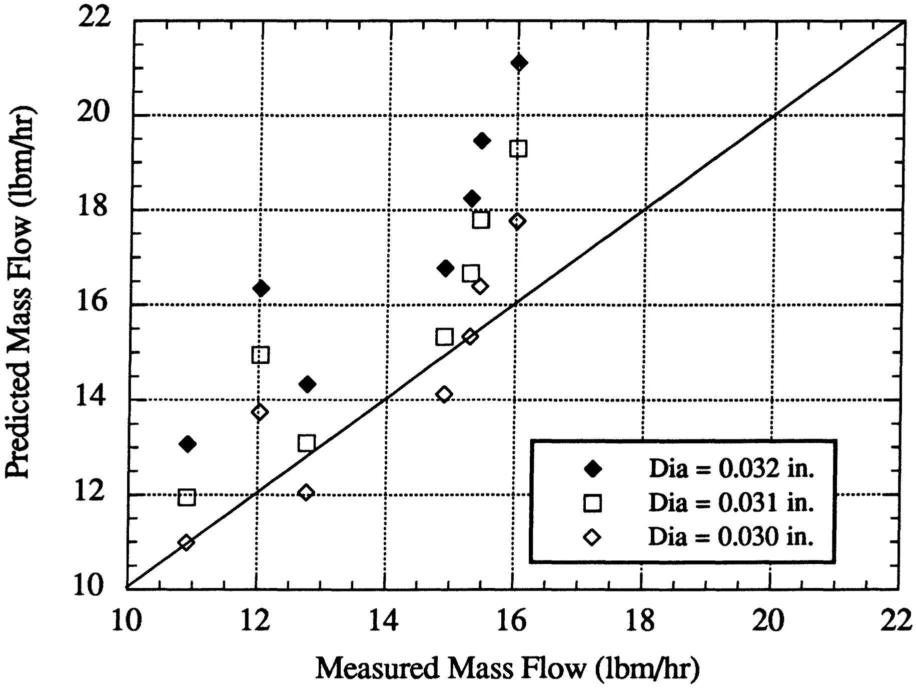


Figure F.1 The effect of diameter uncertainty on mass flow

F.3 Capillary tube roughness uncertainty

Sweedyk (1981) also measured, with a profilometer, the internal surface roughness in several capillary tubes and found that the roughness values varied from $3x10^{-6}$ in. to $73x10^{-6}$ in. A profilometer is a device that measures the arithmetical average of variations above and below a normal line over a 0.030 in. length of a tube or plate surface. Roughness affects the CTSLHXSIM model results because the friction factor calculation includes the relative roughness ratio (ε / D) as a variable. Because Sweedyk's sample size was limited (39 tubes), he stated that the large roughness deviation may not be totally representative of reality. Once again, the above 7-point data set was used in the model to estimate the error in the mass flow rates caused by the roughness uncertainty. In order to do this, it was necessary to select a low, average, and a high value of the absolute roughness for the tube. An average value of roughness was chosen from the arithmetic average of all the measured values to be approximately $12x10^{-6}$ in. The low and high values were chosen to be $5x10^{-6}$ in. and $40x10^{-6}$ in., respectively. The rationale behind choosing these values was to see the effect of a "smooth" and "rough" tube on the mass flow uncertainty without including the extreme values. The effects of the roughness uncertainty alone on the mass flow can be seen below in Figure F.2.

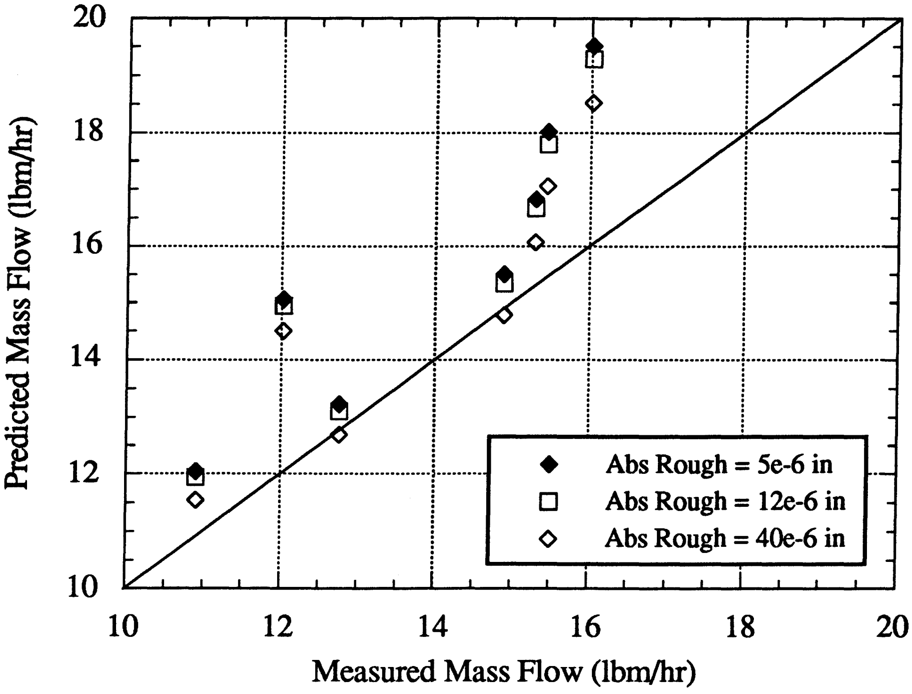


Figure F.2 The effect of internal roughness on mass flow

As the Figure F.2 shows, there is a greater uncertainty associated with the "rough" tube than with the "smooth" tube. The average percentage uncertainty in the mass flow due to the roughness uncertainty is + 1.0 % and - 3.5 %. Of course, the magnitude of the negative uncertainty would be much higher if Sweedyk's extreme value of $73x10^{-6}$ was used as the value for the "rough" tube. However, the internal roughness values that were used are probably more representative of the actual manufacturing uncertainty.

F.4 Two-Phase Friction Factor Correlations

Within the CTSLHXSIM model, the capillary tube is broken up into several segments, and the governing equations are solved simultaneously . There is one set of equations for the two adiabatic regions, and there is another group of equations for the heat exchanger region. Every segment has a momentum equation which contains a friction factor term. Collectively, these friction factors have a significant impact on the mass flow rate that is calculated by the model.

For modeling purposes, each of the capillary tube segments is either single-phase or two-phase. In general, the friction factor for single-phase turbulent flow is determined from experimental data as a function of Reynolds number and the relative roughness. Although there are different single-phase friction factor correlations, there is little difference between them. Equation F.1 shows the correlation that is used in the model for the single-phase regions of the capillary tube. It is a curve fit that approximates the transcendental Colebrook friction factor equation for single-phase turbulent flow (Swamee and Jain, 1976).

$$f = 0.25\left[\log\left(\frac{\varepsilon / D}{3.7} + \frac{5.74}{\mathrm{Re}^{0.9}}\right)\right]^{-2} \tag{F.1}$$

Unfortunately, it is more difficult to model the friction factor within the two-phase regions. Each of the phases, liquid and vapor, have a different set of transport properties like viscosity and specific heat. Unlike thermodynamic properties, transport properties cannot be "combined" based upon the quality of the two-phase mixture. The viscous dissipation that occurs within the two-phase flow is due to viscous dissipation within the liquid and vapor phases, individually, plus the interfacial effects. However, it is unclear how to combine these frictional effects to obtain an overall equivalent friction factor. In general, the method of combination that is used is dependent on the flow regime within the tube. Therefore, most two-phase friction factor correlations are associated with a particular flow-regime assumption.

Three different two-phase friction factors were examined within the CTSLHXSIM model. The first friction factor is based on the assumption of intermittent flow similar to that which can occur in a soda straw. Assuming that both phases are moving at the same velocity, the frictional forces acting on the tube wall can be expressed as the sum of the shear forces acting on the portions of the wall exposed to liquid and vapor.

$$\Delta P = \frac{1}{2D} f \frac{V^2}{\upsilon} \Delta Z = \Delta P_{\mathrm{vapor}} + \Delta P_{\mathrm{liquid}} = \frac{V^2}{2D}\left[\frac{f_g}{\upsilon_g}\Delta Z_g + \frac{f_f}{\upsilon_f}\Delta Z_f\right] \tag{F.2}$$

For slug/plug flow in a constant diameter tube, the lengths (ΔZ_i) in the above equation can be related to the void fraction. For $\Delta Z = 1$, this relation can be shown schematically in Figure F.3.

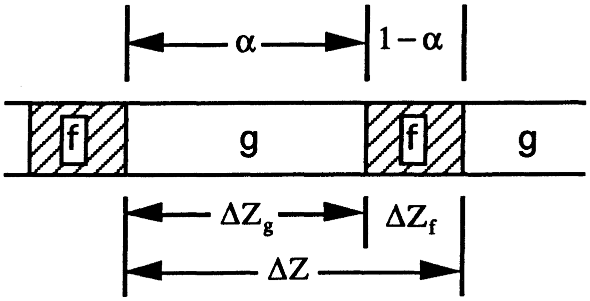


Figure F.3 Schematic of intermittent flow

From the above figure and the definition of void fraction, the lengths (ΔZ_i) can be expressed in terms of quality, total specific volume, vapor specific volume, and liquid specific volume.

$$\Delta Z_g = \alpha = \frac{x\upsilon_g}{\upsilon} \tag{F.3}$$

$$\Delta Z_f = 1-\alpha = 1-\frac{x\upsilon_g}{\upsilon} = \frac{\upsilon - x\upsilon_g}{\upsilon} = \frac{(1-x)\upsilon_f}{\upsilon} \tag{F.4}$$

When these expressions for the gas and liquid lengths are substituted back into Equation (F.2), the total pressure drop per unit length can be written in the following form:

$$\frac{\Delta P_{total}}{\Delta Z} = \frac{V^2}{2D}\left[f_g\frac{x}{\upsilon} + f_f\frac{(1-x)}{\upsilon}\right] = \frac{V^2}{2D\upsilon}\left[f_g x + f_f(1-x)\right] \tag{F.5}$$

If the last expression in Equation F.5 is compared to the first expression for the total pressure drop in Equation F.2, it can be seen that the two-phase viscosity is equivalent, for this flow regime assumption, to the quality-weighted average of the two-single phase friction factors. Therefore, the average two-phase friction factor can be written as

$$f_{avg} = (1-x)f_f + x\cdot f_g \tag{F.6}$$

Where f_f and f_g are determined from the single-phase correlation shown in Equation F.1 by using the vapor and fluid viscosities, respectively, in the Reynolds number calculations. The above derivation also holds for other homogeneous flows (e.g. bubble, fog) where the void fraction is assumed to define the fraction of the wall surface exposed to vapor and liquid shear.

The second friction factor is obtained by defining a "two-phase viscosity" which is used in the Reynolds number calculation. Although there are different ways to formulate this viscosity, Pate (1982) tried three different formulations and found that Dukler's (1964) void-fraction weighting of liquid and vapor viscosities was the best. In his paper, Dukler showed that the void-fraction weighting procedure was the only method compatible with proper scaling of homogeneous two-phase flow. The void-fraction weighting procedure can be seen in Equation F.3.

$$\mu = \frac{x\upsilon_v\mu_v + (1-x)\upsilon_l\mu_l}{\upsilon} \tag{F.7}$$

This "two-phase" viscosity is used to determine the Reynolds number, and then the friction factor is determined from Equation F.1.

The third friction factor is an empirical two-phase correlation that was developed by Souza (1992). The correlation, which uses the Lockhart-Martinelli parameter (X_{tt}) and the Froude number, was design to predict the two-phase pressure drop in heat-exchanger tubes. It calculates the friction factor as if it were a single-phase liquid flow, because of the liquid film where the shear force acts on the tube wall, and then modifies it to account for the fact that it is a two-phase flow. It was developed from extensive testing of several refrigerants in tubes greater than or equal to 0.118 inches in diameter.

These three different friction factors were used to calculate the mass flow rate through the capillary-tube suction-line heat exchanger under a variety of conditions. Seven data points from the Amana refrigerator were used for the comparison of the friction factors as shown in Figure F.4 below.

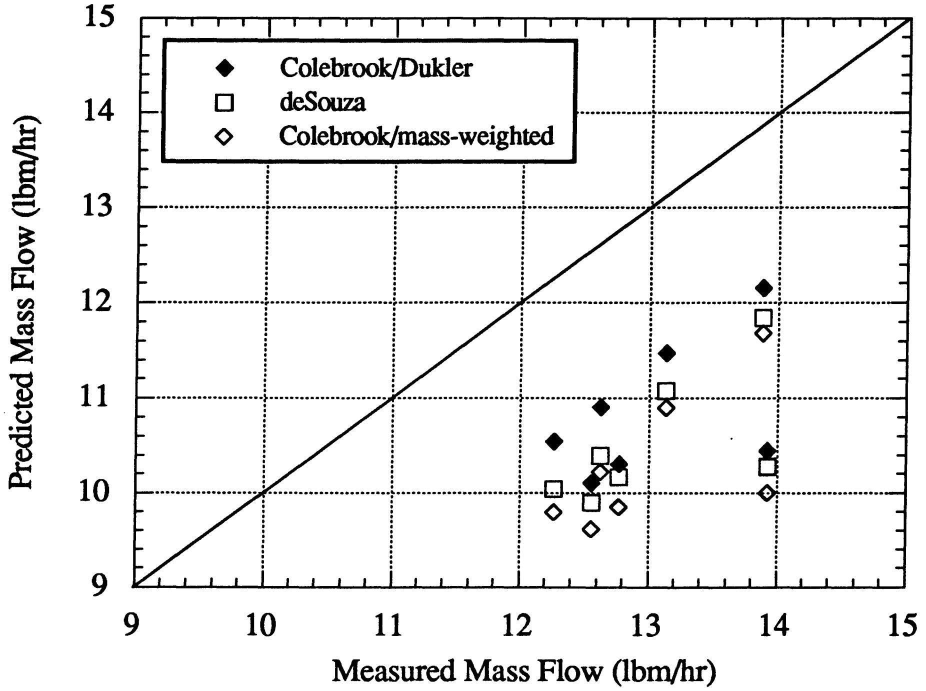


Figure F.4 The effect of two-phase friction factor on mass flow

As the figure shows, the mass flow rates predicted when the Colebrook/Dukler correlation is used are consistently higher than the other two correlations. The Souza correlation yields the second highest mass flow rates, and the Colebrook with the mass weighted friction factor correlation yields the lowest mass flow rates. The above graph indicates that the Colebrook/Dukler two-phase friction factor correlation predicts mass flow rates that are closest to this experimental data. If the model is underpredicting mass flow, as in the data from the Amana, it is clear that Colebrook/Dukler correlation is the best choice. However, when the model is run for other data sets, such as the one from Greenfield, it tends to slightly overpredict the mass flow rates. This difference in trends makes it hard to choose a two-phase friction factor because it is not clear which one causes the best agreement with the experimental data. This especially clear when looking at Figure F.5, which shows all of the currently available model runs.

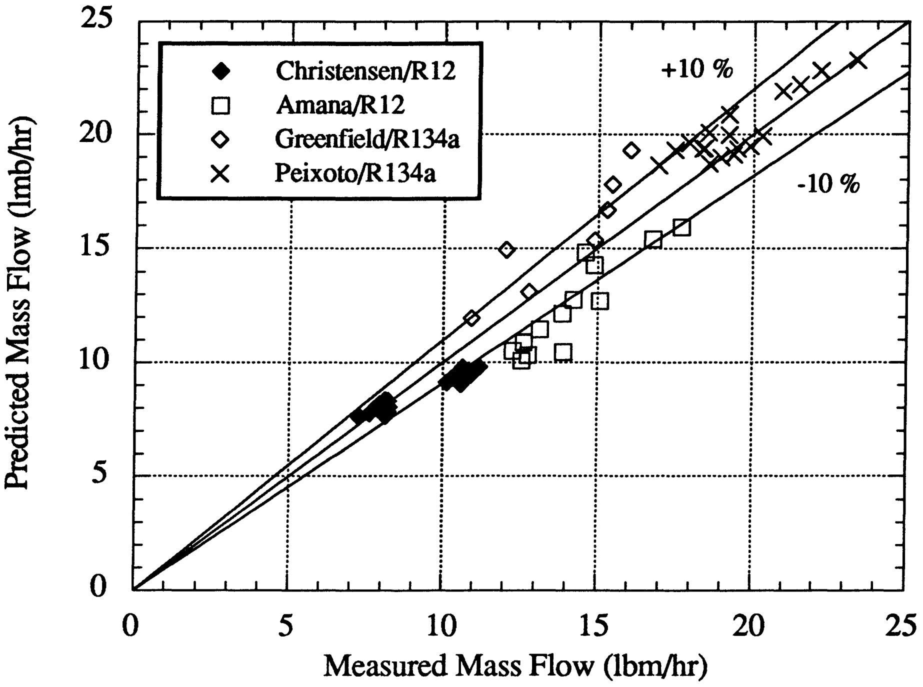


Figure F.5 Predicted verses measured mass flow

The error lines of +/- 10% represent the approximate error caused by the uncertainties in the diameter and roughness in the capillary tube. The predicted values in Figure F.5 were produced with the Colebrook/Dukler two-phase friction factor correlation. From Figure F.4 and the previous discussion, it is clear that the predicted values would decrease by about 0.3 or 0.5 lbm/hr for every point if the Colebrook/Dukler correlation were replaced by the Souza or the Colebrook/mass-weighted correlation, respectively. Since more of the predicted points lie below the negative uncertainty line than lie above the positive uncertainty line, the agreement with the experimental data would become worse if either of the other two correlations were used. Therefore, the Colebrook/Dukler correlation is probably the best friction factor of the three considered.

The reasons for the differences in the three friction factor correlations can be better understood by looking at Figure F.6. It shows the friction factor that is calculated for a particular experimental point at the beginning and end of every segment of the discretized solution for each of the three correlations.

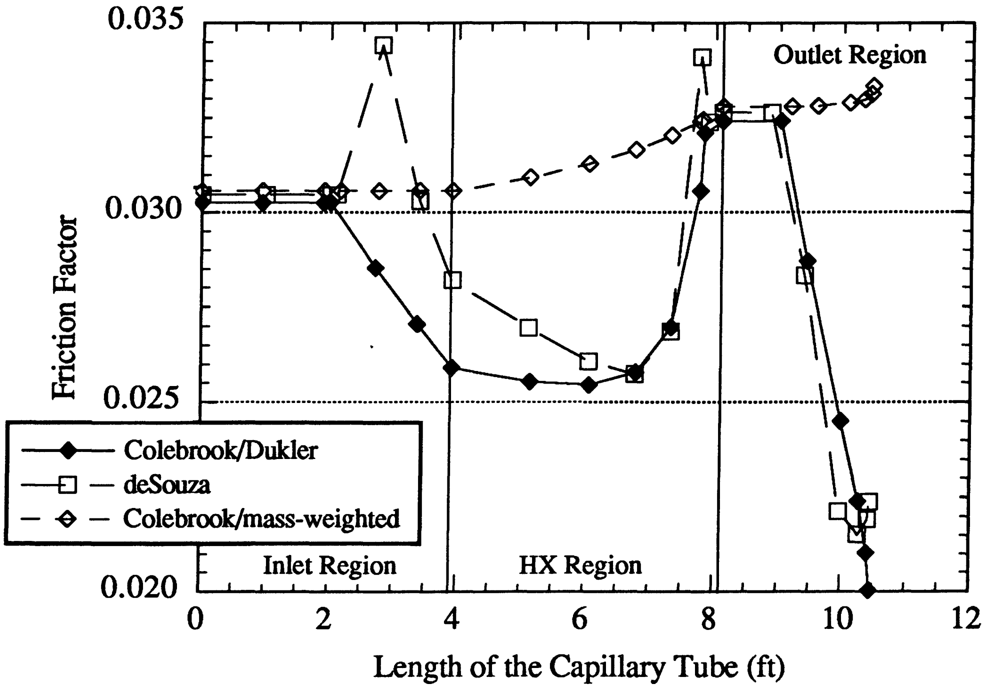


Figure F.6 Friction factor vs. capillary-tube length

The Colebrook/Dukler correlation predicts higher mass flow rates than the other two correlations because its friction factors are generally lower. The Souza correlation was developed from experimental data that had a quality range of 10 to 90%. When the correlation is used at qualities below 10%, as in the capillary tube model, the calculated friction factors are extrapolations of the correlation. This may explain why there is such a difference between the Colebrook/Dukler and the Souza friction factors just after the flash point in the inlet region. After flashing, the Colebrook/Dukler friction factor drops because of the presence of vapor, but the Souza correlation predicts a sharp increase in the friction factor. The flashing and recondensation points can be seen by looking at the calculated quality profiles in Figure F.7. The location of the initial flash point and the recondensation point are approximately the same for all three friction factors, but the location of the second flash point is slightly different for the three correlations. Some of the points on the graph that are labeled as zero quality are actually subcooled.

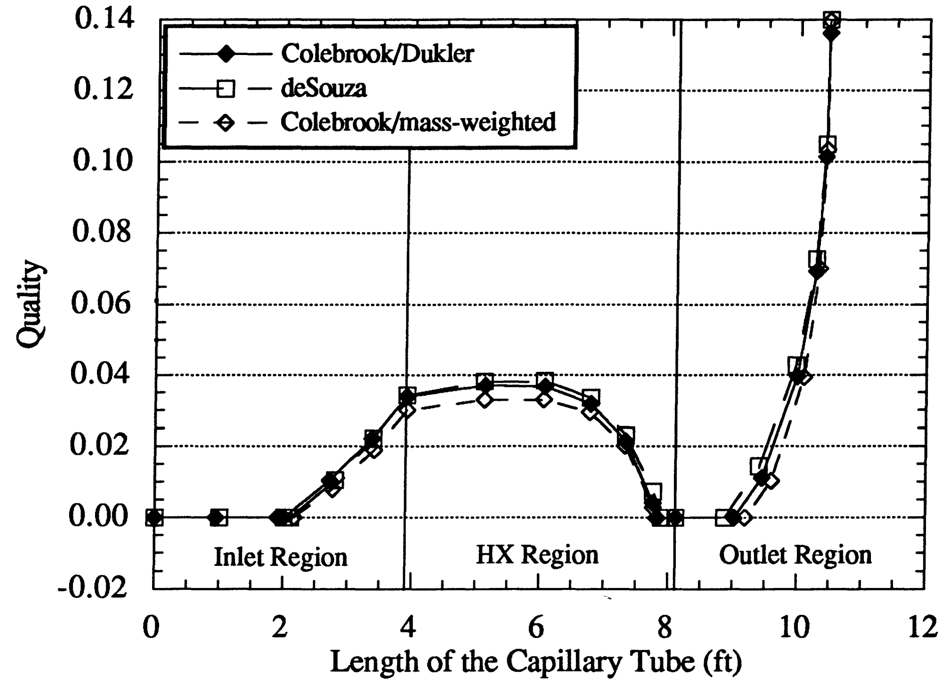


Figure F.7 Quality vs. capillary-tube length

As the quality increases, the Souza friction factors rapidly approach the Colebrook/Dukler values. There is another overprediction by the Souza correlation at the recondensation point near the end of the heat exchanger region. The fact that the Souza, an experimentally determined correlation, and the Colebrook/Dukler correlations follow the same trend and are approximately equal at the higher qualities is encouraging. However, the Colebrook/mass-weighted correlation does not follow the same trend as the other two because it is a quality-weighted friction factor. At the low qualities seen in the capillary tube, this correlation is essentially a single-phase liquid friction factor. The gradual increase in the friction factor along the capillary tube is due to the fact that the refrigerant viscosity increases as the temperature drops, faster than the quality increases. Although it appears that the Colebrook/Dukler two-phase friction factor correlation is the best of the three examined for the capillary tube model, more experimental data is needed to verify that this is the case.

F.5 Conclusion

The capillary tube diameter and roughness uncertainties and the choice of two-phase friction factors have been examined in an effort to reduce or at least understand the error in the model predictions. From the presented information, it is clear that if a particular capillary tube is to be used in model validation, it is important to know its diameter and, if possible, its roughness. It is also clear from Figure F.5 that if the capillary tube diameter and roughness are not known with certainty, the evaluation of two-phase friction factors will be more difficult. From the data that are available, it appears that the Colebrook/Dukler friction factor correlation is the best of the three examined.

It should be noted that there are other factors that may affect the model validation process that are much harder to quantify. The current CTSLHXSIM model assumes that vaporization will begin to occur as soon as the refrigerant's pressure in the capillary tube reaches the saturation pressure. Some researchers, however, have observed a delay in the vaporization that is characterized by an "underpressure of vaporization" and a longer liquid length in the capillary tube. Li, et al. (1990) have studied this metastable phenomenon in adiabatic capillary tubes and suggest that it occurs in all small diameter (~1mm) capillary tubes. If this is true, it means that the CTSLHXSIM model will not accurately predict mass flow even if there is no uncertainty in the diameter or roughness and the Colebrook/Dukler correlation is a perfect representation of the two-phase friction factor. Therefore, it would be beneficial to the development of the capillary-tube model to examine how a delay in vaporization might be modeled.

References

Dukler, A.E., et al., "Pressure Drop and Hold Up in Two-Phase Flow," A. I. Ch. Journal, Vol. 10, pp. 38-51, 1964.

Greenfield, K.P., and W.E. Dunn,"Experimental Study of Non-Adiabatic Flow of Refrigerant R-134a in a Capillary Tube," ACRC TR-67, University of Illinois at Urbana-Champaign, 1994.

Li, R., et al.,"Metastable Flow of R12 Through Capillary Tubes," Rev. Int. Froid, Vol. 13, pp. 181-186, 1990.

Pate, M.B., "A Theoretical and Experimental Analysis of Capillary Tube-Suction Line Heat Exchangers," Ph.D., Thesis, Purdue University, 1982.

Souza, A.L. and J.C. Chato,"Pressure Drop During Two-Phase Flow of Refrigerants in Horizontal Smooth Tubes," ACRC TR-25, University of Illinois at Urbana-Champaign, 1992.

Swamee, P.K., and Jain, A.K., "Explicit Equations for Pipe-Flow Problems," Proceedings of the ASCE, Journal of the Hydraulics Division, 102, HY5, p.657-664, May 1976.

Sweedyk, J.M., "Capillary Tubes - Their Standardization and Use," ASHRAE Transactions, Part 1, Vol. 87, pp. 1069-1076, 1981.

Appendix G
List of Variables, Parameters, and Calculated Values

G.1 Variables

G.1.1 Residual variables (Xs)

All of the residual variables used within the RFSIM model are shown below in Table G.1. They are listed in the order that they appear in the XK file for the Whirlpool refrigerator. As described in Appendix A, some of these variables could become parameters through the X-K switching. In addition to the variable name, a description, a typical value, and the units of the variable are also given in Table G.1.

Table G.1 Residual variables (Xs)

Variable Name	Description	Typical value	Units
Acond	Air-side area of the condenser	9.108	[ft^2]
Aevap	Air-side area of the evaporator	23.197	[ft^2]
CaCond	Heat capacity of the condenser air stream	105.887	[Btu/hr-F]
CaEvap	Heat capacity of the evaporator air stream	63.171	[Btu/hr-F]
COP	Coefficient of performance	0.96547	[ ]
DpDischarge	Pressure drop in the discharge line	0	[psia]
DpSupCond	Pressure drop in the superheated zone of condenser	0.3675	[psia]
Dp2phCond	Pressure drop in the two-phase zone condenser	3.2178	[psia]
DpSubCond	Pressure drop in the subcooled zone condenser	0.0019	[psia]
DpLiquid	Pressure drop in the liquid line	0	[psia]
Dp2phEvap	Pressure drop in the two-phase zone evaporator	0.1337	[psia]
DpSupEvap	Pressure drop in the superheated zone evaporator	0.0027	[psia]
DpSuction	Pressure drop in the suction line	0.266	[psia]
DPin	Pressure steps in inlet capillary tube section	8.6627	[psia]
DTsl	Temperature steps in suction line heat exchanger	14.8315	[F]
DPout	Pressure steps in outlet capillary tube secion	4.3856	[psia]
Energy	Steady-state energy usage for the entire year	775.04	[kW-hr/yr]
fsubcond	Fraction of subcooled portion of the condenser	0.006	[ ]
f2phcond	Fraction of two-phase portion of the condenser	0.9036	[ ]
fsupcond	Fraction of superheated portion of the condenser	0.0903	[ ]
f2phevap	Fraction of two-phase portion of the evaporator	0.9873	[ ]
fsupevap	Fraction of superheated portion of the evaporator	0.0127	[ ]
fz	Fraction of evaporator air flow to freezer	0.9137	[ ]
h0	Specific enthalpy at the compressor outlet	99.427	[Btu/lbm]
h1	Specific enthalpy at the condenser inlet	99.427	[Btu/lbm]
h21	Specific enthalpy at x=1.0 of condenser	87.553	[Btu/lbm]
h20	Specific enthalpy at x=0.0 of condenser	32.238	[Btu/lbm]

h3	Specific enthalpy at the condenser exit	31.995	[Btu/lbm]
h4	Specific enthalpy at the captube inlet	31.995	[Btu/lbm]
hcrit	Specific enthalpy at the exit of capillary tube	18.969	[Btu/lbm]
h7	Specific enthalpy at the inlet to the evaporator	19.407	[Btu/lbm]
h71	Specific enthalpy at x=1.0 in the evaporator	75.681	[Btu/lbm]
h9	Specific enthalpy at the exit of the evaporator	75.994	[Btu/lbm]
h11	Specific enthalpy at the compressor inlet	88.595	[Btu/lbm]
MDisLine	Mass of refrigerant in the discharge line	0	[lbm]
Mcond	Mass of refrigerant in the condenser	0.140957	[lbm]
MLiqLine	Mass of refrigerant in the liquid line	0.033429	[lbm]
MCapTube	Mass of refrigerant in capillary tube	0.000454	[lbm]
Mevap	Mass of refrigerant in the evaporator	0.198141	[lbm]
MSuctLine	Mass of refrigerant in suction line	0.001328	[lbm]
Maccum	Mass of refrigerant in accumulator	0.000587	[lbm]
Mcompvap	Mass of refrigerant refrigerant vapor in compressor	0.078366	[lbm]
Mcompoil	Mass of refrigerant dissolved in the compressor oil	0.087856	[lbm]
Mtotal	Mass of refrigerant in the entire system	0.541117	[lbm]
p0	Refrigerant pressure at the compressor outlet	144.269	[psia]
p1	Refrigerant pressure at the condenser inlet	144.269	[psia]
p21	Refrigerant pressure at x=1.0 in the condenser	143.902	[psia]
p20	Refrigerant pressure at x=0.0 in the condenser	140.684	[psia]
p3	Refrigerant pressure at the condenser outlet	140.682	[psia]
p4	Refrigerant pressure at the capillary tube inlet	140.682	[psia]
pcrit	Refrigerant pressure at the capillary tube outlet	19.102	[psia]
p7	Refrigerant pressure at the evaporator inlet	17.371	[psia]
p71	Refrigerant pressure at x=1.0 in the evaporator	17.238	[psia]
p9	Refrigerant pressure at the evaporator outlet	17.235	[psia]
p11	Refrigerant pressure at the compressor inlet	16.969	[psia]
powercomp	Power consumed by the compressor	102.767	[Watts]
qcond	Total heat transfer from the condenser	542.366	[Btu/hr]
qsupcond	Heat transfer from the superheated condenser zone	95.499	[Btu/hr]
q2phcond	Heat transfer from the two-phase condenser zone	444.915	[Btu/hr]
qsubcond	Heat transfer from the subcooled condenser zone	1.952	[Btu/hr]
qevap	Total heat transfer to the evaporator	455.143	[Btu/hr]
q2phevap	Heat transfer from the two-phase evaporator zone	452.623	[Btu/hr]
qsupevap	Heat transfer from the superheated evaporator zone	2.52	[Btu/hr]
qcomp	Heat transfer from the condenser	263.626	[Btu/hr]
Qfrez	Total heat input to the freezer compartment	145.633	[Btu/hr]
Qfrig	Total heat input to the fresh food compartment	145.906	[Btu/hr]
RunTime	Ratio of the total system load to the system capacity	0.6925	[ ]
t0	Refrigerant temperature at the compressor outlet	171.876	[F]
t1	Refrigerant temperature at the condenser inlet	171.876	[F]
t21	Refrigerant temperature at x=1.0 in the condenser	106.38	[F]
t20	Refrigerant temperature at x=0.0 in the condenser	104.716	[F]

t3	Refrigerant temperature at the condenser outlet	103.716	[F]
t4	Refrigerant temperature at the capillary tube inlet	103.716	[F]
tcrit	Refrigerant temperature at the capillary tube outlet	-10.194	[F]
t7	Refrigerant temperature at the evaporator inlet	-14.406	[F]
t71	Refrigerant temperature at x=1.0 in the evaporator	-14.745	[F]
t9	Refrigerant temperature at the evaporator outlet	-13.745	[F]
t11	Refrigerant temperature at the compressor inlet	75.244	[F]
Tsat0	Saturation temperature at p0 (for compressor map)	106.569	[F]
Tsat11	Saturation temperature at p11 (for compressor map)	-15.431	[F]
tacondin	Air temperature at condenser inlet	92.297	[F]
tacondout	Air temperature at condenser outlet	97.419	[F]
tacondfanin	Air temperature at the condenser fan inlet	99.908	[F]
tacondfanout	Air temperature at the condenser fan outlet	100.392	[F]
taevapin	Air temperature at the evaporator inlet	8.021	[F]
taevapmid	Air temperature in between the parallel-flow two-phase zone and the superheated zone in the evaporator	0.905	[F]
taevapout	Air temperature at the evaporator outlet/fan inlet	0.816	[F]
taevapfanout	Air temperature at the evaporator fan outlet	1.356	[F]
usupcond	Overall heat transfer coefficient for superheated zone of the condenser	3.77	[Btu/hr-ft^2-F]
u2phcond	Overall heat transfer coefficient for two-phase zone of the condenser	4.659	[Btu/hr-ft^2-F]
usubcond	Overall heat transfer coefficient for subcooled zone of the condenser	3.452	[Btu/hr-ft^2-F]
u2phevap	Overall heat transfer coefficient for two-phase zone of the evaporator	1.057	[Btu/hr-ft^2-F]
usupevap	Overall heat transfer coefficient for superheated zone of the evaporator	0.565	[Btu/hr-ft^2-F]
v0	Refrigerant volume at the compressor outlet	0.34213	[ft^3/lbm]
v1	Refrigerant volume at the condenser inlet	0.34213	[ft^3/lbm]
v21	Refrigerant volume at x=1.0 in the condenser	0.28156	[ft^3/lbm]
v20	Refrigerant volume at x=0.0 in the condenser	0.0128	[ft^3/lbm]
v3	Refrigerant volume at the condenser outlet	0.01278	[ft^3/lbm]
v4	Refrigerant volume at the capillary tube inlet	0.01278	[ft^3/lbm]
vcrit	Refrigerant volume at the capillary tube outlet	0.36746	[ft^3/lbm]
v70	Refrigerant volume at x=0.0 in the evaporator	0.01085	[ft^3/lbm]
v7	Refrigerant volume at the evaporator inlet	0.43933	[ft^3/lbm]
v71	Refrigerant volume at x=1.0 in the evaporator	2.18093	[ft^3/lbm]
v9	Refrigerant volume at the evaporator outlet	2.21158	[ft^3/lbm]
v11	Refrigerant volume at the compressor inlet	2.74221	[ft^3/lbm]
w	Refrigerant mass flow rate	8.043	[lbm/hr]
Woil	Mass fraction of refrigerant in oil/refrig. mixture	0.2269	[ ]
xcrit	Refrigerant quality at the capillary tube outlet	0.181	[ ]
xoc	Refrigerant quality at the condenser outlet	0	[ ]
xie	Refrigerant quality at the evaporator inlet	0.1989	[ ]
xoe	Refrigerant quality at the evaporator outlet	1	[ ]

G.1.2 Non-residual variables

The non-residual variables are those varibles which are not listed in the XK file. Their values are calculated explicitly from residual variables (Xs), parameters (Ks), calculated values (Cs), or other non-residual variables. They are used in the governing equations primarily to make the writing of the equations easier. Since they are intermediate values that do not require initial guesses, they are not documented here but are documented in the file containing the governing equations (EQNS.f).

G.2 Parameters (Ks) and calculated values (Cs)

The names of the parameters (Ks) and the calculated values (Cs), along with a description, typical value, and units, are displayed in Table G.2. The parameters (Ks) and calculated values (Cs) are placed together since many of them are interchangable depending on how the model is being used. For example, the effectiveness of the capillary tube-suction line heat exchanger (ectslhx) will be a "K" if the capillary tube model is not used and it will be a "C" if the capillary tube model is used. Therefore, all of the following quantities in Table G.2 that may be a C will be marked by an asterisk(*). As with the residual variables (Xs) described above, the parameters and calculated values are listed in the order in which they appear in the XK file.

Table G.2 Parameters and calculated values (Ks and Cs)

Parameter Name	Description	Typical Values	Units
AAFC	Frontal air-flow area in the condenser	0.426	[ft^2]
AAFE	Frontal air-flow area in the condenser	0.4155	[ft^2]
alphacond	Ratio of external to internal area in the condenser	3.268	[ ]
alphaevap	Ratio of external to internal area in the evaporator	6.976	[ ]
beta_Pmap	Multiplier to scale the the compressor power map	0.7192	[ ]
beta_wmap	Multiplier to scale the the compressor mass flow map	0.7305	[ ]
beta_condfan	Multiplier to scale the the condenser fan speed	1	[ ]
beta_evapfan	Multiplier to scale the the evaporator fan speed	1	[ ]
CaptubeModel	Indicates whether the captube model is used 'K' = the capillary tube model will be used 'X' = the capillary tube model will not be used	1	
CaptubeOutput	Indicates the amount of captube model output 0 = No output 1 = Captube variable profile at the final solution 2 = Captube variable profile every iteration	0	[ ]

CompNum	Compressor map to be used by the model 1 = Generic map for top-mount Amana 2 = Modified map for top-mount Amana 3 = Generic map for top-mount Whirlpool 4 = Modified map for top-mount Whirlpool 5 = Generic map for S-by-S Amana	4	[ ]
crtmult	Multiplier for the refrigerant-side heat transfer correlation in the condenser to simulate microfinned tubing	1	[ ]
Ddisline	Inside diameter of the discharge line	0.01583	[ft]
Dcond	Inside diameter of the condenser tubing	0.01096	[ft]
Dliqline	Inside diameter of the liquid line	0.01063	[ft]
Dct	Inside diameter of the capillary tube	0.002657	[ft]
Devap	Inside diameter of the evaporator tubing	0.02725	[ft]
Dsuctline	Inside diameter of the suction line	0.02104	[ft]
DZC	Total straight tube length in the condenser	71.385	[ft]
DZE	Total straight tube length in the evaporator	37.467	[ft]
ertmult	Multiplier for the refrigerant-side heat transfer correlation in the evaporator to simulate microfinned tubing	1	[ ]
ectslhx *	Effectiveness of the captube-suction line heat heat exchanger (calculated when the captube model is used, and specified when the captube model is not)	0.748	[ ]
frecirc	Fraction of the condenser exit air that recirculates back into the condenser inlet	0.221	[ ]
FrezHeater	Power input to the freezer compartment by the heaters(or any other source)	0	[Watts]
FrigHeater	Power input to the fresh food compartment by the heaters (or any other source)	0	[Watts]
hAcomp	Heat transfer coefficient of the compressor shell	5.169	[Btu/hr-F]
haircond *	Air-side heat transfer coefficient of the condenser	4.989	[Btu/hr-F]
hcondNum	Indicates whether haircond is specified by the user or obtained from a correlation 0 = user specified 1 & higher = correlations (See haircnd in EQNSUBS.f)	0	
hairevap *	Air-side heat transfer coefficient of the evaporator	1.333	[Btu/hr-F]
hevapNum	Indicates whether hairevap is specified by the user or obtained from a correlation 0 = user specified 1 & higher = correlations (See hairevp in EQNSUBS.f)	0	[ ]
K1oil	Coefficient for refrigerant-oil mixture solubility	-0.005992765	[ ]
K2oil	Coefficient for refrigerant-oil mixture solubility	0.04166151	[ ]
K3oil	Coefficient for refrigerant-oil mixture solubility	0.00200466	[ ]
K4oil	Coefficient for refrigerant-oil mixture solubility	-0.003268285	[ ]
K5oil	Coefficient for refrigerant-oil mixture solubility	0.001736844	[ ]
K6oil	Coefficient for refrigerant-oil mixture solubility	-0.000285522	[ ]
K7oil	Coefficient for refrigerant-oil mixture solubility	1.60929E-05	[ ]
Ldisline	Length of discharge line	0	[ft]

Lcond	Total length of condenser tubing (w/ return bends)	80.941	[ft]
Lliqline	Length of liquid line	0.542	[ft
Lin	Length of adiabatic inlet section of capillary tube	7.8963	[ft]
Lhx	Length of the captube-suction line heat exchanger	7.0833	[ft]
Lout	Length of the adiabatic outlet region of the capillary tube	0.5833	[ft]
Levap	Total length of evaporator tubing (w/ return bends)	38.845	[ft]
Lsuctline	Length of the suction line	9.458	[ft]
Moil	Mass of oil in the compressor sump	0.2993	[lbm]
NSECTC	Number of equivalent circuits in the condenser	1	[ ]
NSECTE	Number of equivalent circuits in the evaporator	1	[ ]
numDPin	Number of pressure steps in the captube inlet section used in the solution process	4	[ ]
numDPout	Number of pressure steps in the captube outlet section used in the solution process	5	[ ]
numDTsl	Number of temperature steps in the suction line heat exchanger section used in the solution process	6	[ ]
patm	Atmospheric pressure	14.5	[psia]
pcondfan	Power used by condenser fan	15	[Watts]
pevapfan	Power used by evaporator fan	10	[Watts]
rough	Average surface roughness inside system tubing	0.000005	[ft]
RTBCND	Number of return bends in the condenser	73	[ ]
RTBEVP	Number of return bends in the evaporator	17	[ ]
subcool	Degrees of subcooling at the condenser exit	1	[F]
superheat	Degrees of superheating at the evaporator exit	1	[F]
STC	Spacing between tubes in the condenser (return bend diameter)	0.08333	[ft]
STE	Spacing between tubes in the evaporator (return bend diameter)	0.05158	[ft]
tamb	Ambient room temperature	90	[F]
tafrig	Average air temperature in the fresh food compartment	40	[F]
tafrez	Average air temperature in the freezer compartment	5	[F]
UAf	Overall heat transfer coefficient for the freezer section of the refrigerator "box"	0.855	[Btu/hr-F]
UAz	Overall heat transfer coefficient for the fresh food section of the refrigerator "box"	0.502	[Btu/hr-F]
vdotcond	Volumetric air flow rate through the condenser	105.134	[ft^3/min]
vdotevap	Volumetric air flow rate through the evaporator	51.615	[ft^3/min]
volfilter	Internal free volume of the filter drier	0.000379	[ft^3]
volaccum	Internal volume of the accumulator (if any)	0.001609	[ft^3]
volcomp	Internal free vapor volume of the compressor	0.026811	[ft^3]
Cooling *	Availability, or exergy, delivered to the cold space by the refrigeration system	55.2	[Btu/hr]
Itot *	Total irreversibility, or destroyed exergy, in the system	246.8	[Btu/hr]

Icomp *	Total irreversibility, or destroyed exergy, in the compressor	138.3	[Btu/hr]
Icond *	Total irreversibility, or destroyed exergy, in the condenser	49.6	[Btu/hr]
Ievap *	Total irreversibility, or destroyed exergy, in the evaporator	48.1	[Btu/hr]
Ipipes *	Total irreversibility, or destroyed exergy, in the rest of the tubing	10.8	[Btu/hr]
pcondfan_calc*	Calculated condenser fan power based on the fan laws and beta_condfan	15	[Watts]
pevapfan_calc*	Calculated condenser fan power based on the fan laws and beta_evapfan	10	[Watts]
TcondAvg *	Average two-phase condensing temperature	105.548	[F]
TevapAvg *	Average two-phase evaporating temperature	-14.576	[F]
vdotcond_calc*	Calculated condenser volumetric air flow rate based on the fan laws and beta_condfan	105.134	[ft^3/min]
vdotevap_calc*	Calculated evaporator volumetric air flow rate based on the fan laws and beta_evapfan	51.615	[ft^3/min]
voldisline *	Internal volume of the discharge line	0	[ft^3]
Volcond *	Internal volume of the condenser	0.007636	[ft^3]
volliqline *	Internal volume of the liquid line	0.000048	[ft^3]
volcaptube *	Internal volume of the capillary tube	0.000086	[ft^3]
Volevap *	Internal volume of the evaporator	0.022655	[ft^3]
volsuctline *	Internal volume of the suction line	0.003288	[ft^3]

Appendix H

Results of variable-speed fan simulations

Some of the more important results of the variable-speed fan simulations for both compressor speeds are presented in Table H.1 and Table H.2.

H.1 High compressor speed

Table H.1 Energy use and evaporator capacity at the high compressor speed

High compressor speed						
Nominal fan speed			75% evaporator fan		75% condenser fan	
Tamb	Energy	Qevap	Energy	Qevap	Energy	Qevap
[F]	[kWh/yr]	[Btu/hr]	[kWh/yr]	[Btu/hr]	[kWh/yr]	[Btu/hr]
120	1387.5	494.0	1431.9	459.0	1416.9	482.0
117.5	1319.1	500.6	1361.1	465.2	1346.1	489.0
115	1253.8	507.0	1293.6	471.1	1278.4	495.7
112.5	1191.3	513.2	1228.9	476.8	1213.7	502.2
110	1132.2	518.8	1166.9	482.2	1151.9	508.5
107.5	1075.4	524.2	1107.4	487.4	1092.8	514.4
105	1020.8	529.4	1050.4	492.3	1036.8	519.9
102.5	968.5	534.1	996.4	496.7	982.9	525.2
100	918.1	538.6	944.2	500.9	930.9	530.3
97.5	869.6	542.7	893.9	504.9	880.8	535.2
95	822.9	546.5	845.2	508.8	833.0	539.4
92.5	778.1	549.8	798.1	512.4	786.7	543.6
90	735.1	552.4	752.7	515.8	742.2	547.4
87.5	693.8	554.5	708.9	519.0	699.3	550.7
85	654.1	556.0	666.6	522.0	658.0	553.6
82.5	616.0	556.8	626.1	524.2	618.3	556.1
80	579.6	556.5	587.6	525.4	580.0	558.1
77.5	544.7	555.2	550.6	526.0	543.1	559.5
75	511.0	553.1	515.1	525.8	507.7	560.0
72.5	478.8	549.9	481.2	524.5	473.6	559.8
70	447.7	545.6	448.8	522.0	440.8	559.0
Nominal fan speed			125% evaporator fan		125% condenser fan	
Tamb	Energy	Qevap	Energy	Qevap	Energy	Qevap
[F]	[kWh/yr]	[Btu/hr]	[kWh/yr]	[Btu/hr]	[kWh/yr]	[Btu/hr]
120	1387.5	494.0	1389.9	517.4	1397.4	501.2
117.5	1319.1	500.6	1322.7	523.8	1329.4	507.6
115	1253.8	507.0	1258.2	530.1	1264.4	513.8
112.5	1191.3	513.2	1196.2	536.4	1203.0	519.4
110	1132.2	518.8	1136.9	542.3	1143.9	524.9
107.5	1075.4	524.2	1080.1	547.9	1087.6	529.8
105	1020.8	529.4	1025.5	553.2	1033.2	534.6
102.5	968.5	534.1	973.2	558.1	981.0	538.9
100	918.1	538.6	923.0	562.5	930.8	542.8
97.5	869.6	542.7	875.0	566.3	882.6	546.4
95	822.9	546.5	828.8	569.8	836.5	549.3
92.5	778.1	549.8	784.5	572.6	792.3	551.4
90	735.1	552.4	742.1	574.7	750.2	552.7
87.5	693.8	554.5	701.4	576.0	709.7	553.4
85	654.1	556.0	662.2	576.9	671.1	553.0
82.5	616.0	556.8	624.6	576.7	634.3	551.3
80	579.6	556.5	588.7	575.5	599.1	548.5
77.5	544.7	555.2	554.3	573.1	565.2	544.8
75	511.0	553.1	521.1	569.9	532.7	540.0
72.5	478.8	549.9	488.8	566.1	501.5	534.0
70	447.7	545.6	457.6	561.2	471.2	526.9

H.2 Low compressor speed

Table H.2 Energy use and evaporator capacity at the low compressor speed

Low compressor speed								
Nominal fan speed				75% evaporator fan			75% condenser fan	
Tamb	Energy	Qevap		Energy	Qevap		Energy	Qevap
[F]	[kWh/yr]	[Btu/hr]		[kWh/yr]	[Btu/hr]		[kWh/yr]	[Btu/hr]
100	860.1	428.9		877.9	401.5		862.4	422.5
97.5	811.9	433.2		828.6	405.6		813.6	427.1
95	765.6	437.4		781.2	409.4		766.8	431.5
92.5	722.5	440.3		735.8	412.9		721.8	435.6
90	680.5	443.5		692.2	416.2		679.7	438.7
87.5	640.2	446.4		650.3	419.2		639.0	442.0
85	601.5	448.8		610.4	421.8		599.8	444.9
82.5	564.2	450.9		572.0	424.1		562.1	447.5
80	528.4	452.7		534.9	426.3		525.9	449.8
77.5	493.9	453.9		499.4	427.9		490.9	451.7
75	460.9	454.5		465.4	428.9		457.3	453.3
72.5	429.3	454.3		432.7	429.5		425.0	454.4
70	399.0	453.3		401.2	429.6		394.1	454.8
67.5	369.9	451.7		371.1	429.0		364.3	454.5
65	342.0	449.1		342.3	427.5		335.7	453.8
62.5	315.2	445.5		314.7	424.9		308.1	452.4
60	289.5	440.7		288.3	421.4		281.7	450.0
Nominal fan speed				125% evaporator fan			125% condenser fan	
Tamb	Energy	Qevap		Energy	Qevap		Energy	Qevap
[F]	[kWh/yr]	[Btu/hr]		[kWh/yr]	[Btu/hr]		[kWh/yr]	[Btu/hr]
100	860.1	428.9		872.8	447.1		883.6	432.7
97.5	811.9	433.2		825.1	451.0		834.6	436.9
95	765.6	437.4		779.3	454.6		788.9	440.0
92.5	722.5	440.3		735.2	458.1		744.6	443.3
90	680.5	443.5		692.7	461.5		702.0	446.2
87.5	640.2	446.4		651.9	464.4		661.1	448.7
85	601.5	448.8		612.9	466.8		621.9	450.7
82.5	564.2	450.9		575.5	468.6		584.1	452.4
80	528.4	452.7		539.4	470.2		547.9	453.5
77.5	493.9	453.9		504.9	471.1		513.2	453.9
75	460.9	454.5		471.8	471.1		480.2	453.3
72.5	429.3	454.3		440.2	470.3		448.6	451.8
70	399.0	453.3		409.9	468.7		418.4	449.4
67.5	369.9	451.7		380.6	466.5		389.4	446.1
65	342.0	449.1		352.5	463.3		361.5	441.6
62.5	315.2	445.5		325.6	458.9		334.9	435.8
60	289.5	440.7		299.7	453.2		309.2	429.0