



**SURVIVABILITY • SUSTAINABILITY • MOBILITY
SCIENCE AND TECHNOLOGY
SOLDIER SYSTEM INTEGRATION**



**TECHNICAL REPORT
NATICK/TR-97/005**

AD _____

MULTIPHASE HEAT AND MASS TRANSFER THROUGH HYGROSCOPIC POROUS MEDIA WITH APPLICATIONS TO CLOTHING MATERIALS

**By
Phillip Gibson**

December 1996

DTIC QUALITY INSPECTED 2

**FINAL REPORT
January 1994 - March 1996**

19961216 047

Approved for Public Release; Distribution Unlimited

**U.S. ARMY SOLDIER SYSTEMS COMMAND
NATICK RESEARCH, DEVELOPMENT AND ENGINEERING CENTER
NATICK, MASSACHUSETTS 01760-5019**

SURVIVABILITY DIRECTORATE

DISCLAIMERS

The findings contained in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of trade names in this report does not constitute an official endorsement or approval of the use of such items.

DESTRUCTION NOTICE

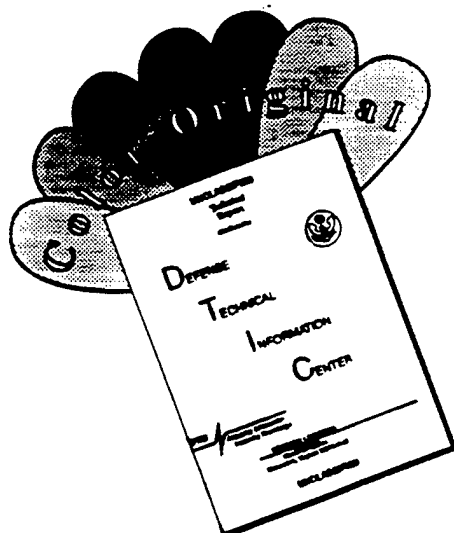
For Classified Documents:

Follow the procedures in DoD 5200.22-M, Industrial Security Manual, Section II-19 or DoD 5200.1-R, Information Security Program Regulation, Chapter IX.

For Unclassified/Limited Distribution Documents:

Destroy by any method that prevents disclosure of contents or reconstruction of the document.

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF COLOR PAGES WHICH DO NOT REPRODUCE LEGIBLY ON BLACK AND WHITE MICROFICHE.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.					
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE December 1996		3. REPORT TYPE AND DATES COVERED Final January 1994 / March 1996	
4. TITLE AND SUBTITLE Multiphase Heat and Mass Transfer Through Hygroscopic Porous Media With Applications to Clothing Materials				5. FUNDING NUMBERS Cost Code: 601A9A 3050 611101	
6. AUTHOR(S) Phillip Gibson					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Soldier Systems Command U.S. Army Natick RD&E Center Kansas Street, ATTN: SSCNC-II Natick, MA 01760-5019				8. PERFORMING ORGANIZATION REPORT NUMBER NATICK/TR-97/005	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES					
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.				12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A set of partial differential equations describing time-dependent heat and mass transfer through porous hygroscopic materials was developed which includes factors such as the swelling of the solid due to water imbibition, and the heat of sorption evolved when the water is absorbed by the polymeric matrix. A numerical code to solve the set of nonlinear coupled equations was developed, and applied to an experimental apparatus designed to simulate transient and steady-state convection/diffusion conditions for textile materials. Results are shown for hygroscopic porous textiles under conditions of pure diffusion, combined diffusion and convection, and pure forced convective flow. The numerical model was integrated with an existing human thermal physiology model to provide appropriate boundary conditions for the clothing model. The human thermal control model provides skin temperature, core temperature, skin heat flux, and water vapor flux, along with liquid water accumulation at the skin surface. The integrated model couples the dynamic behavior of the clothing system to the human physiology of heat regulation. This provided the opportunity to systematically examine a number of clothing parameters which are traditionally not included in steady-state thermal physiology studies, and to determine their potential importance under various conditions of human work rates and environmental conditions.					
14. SUBJECT TERMS HEAT TRANSFER DIFFUSION MASS TRANSFER		THERMAL INSULATION THERMAL CONDUCTIVITY DRYING CLOTHING		15. NUMBER OF PAGES 325	
		SORPTION WATER VAPOR EQUATIONS HYGROSCOPICITY		16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED		18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED		19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	
				20. LIMITATION OF ABSTRACT UL	

TABLE OF CONTENTS

	page
LIST OF FIGURES	v
LIST OF TABLES	ix
PREFACE	xiii
NOMENCLATURE	xv
1. INTRODUCTION	1
2. GOVERNING EQUATIONS	7
3. EXPERIMENTAL METHODS	49
4. NUMERICAL METHODS	89
5. ONE-DIMENSIONAL MODELING OF COUPLED DIFFUSION OF ENERGY AND MASS	99
6. TWO-DIMENSIONAL MODELING OF DIFFUSION AND DIFFUSION/CONVECTION PROCESSES IN TEXTILES	119
7. INTEGRATION OF HUMAN THERMAL PHYSIOLOGY CONTROL MODEL WITH NUMERICAL MODEL FOR COUPLED HEAT AND MASS TRANSFER THROUGH HYGROSCOPIC POROUS TEXTILES	135
8. CONCLUSIONS / RECOMMENDATIONS	157
9. REFERENCES	161
 <u>APPENDICES</u>	
APPENDIX A - Fabric Physical Properties	173
APPENDIX B - One-Dimensional Numerical Code to Solve Transient Coupled Diffusion Problem	179
APPENDIX C - Two-Dimensional Numerical Code to Solve Transient Coupled Diffusion/Convection Problem	203
APPENDIX D - Numerical Code for Coupled Heat and Mass Transfer Through Hygroscopic Porous Textiles Integrated with a Human Thermal Physiology Control Model	265

LIST OF FIGURES

Figure	page
1. Three Phases Present in Hygroscopic Porous Media	7
2. Two Methods of Accounting for Shrinkage/Swelling Due to Water Uptake by a Porous Solid	11
3. Material Volume Containing a Phase Interface, with Velocities and Unit Normals Indicated	15
4. Generic Differential Heat of Sorption for Textile Fibers (sorption hysteresis neglected)	35
5. Schematic of DMPC Test Arrangement	50
6. Schematic and Dimensions of the Dynamic Moisture Permeation Cell.....	52
7. Results for Samples Made of Combined Layers of Microporous PTFE Membranes	56
8. Variation in Intrinsic Log Mean Diffusion Resistance as a Function of the Mean Relative Humidity on the Two Sides of the Test Sample	59
9. Correlation Between DMPC and ISO 11092 for Several Fabrics and Microporous Membrane Laminates	62
10. Correlation of DMPC Results with Modified ASTM E 96 BW Inverted Cup Tests, for Three Air Flow Conditions	64
11. Effect of Parallel or Countercurrent Flow Direction on Concentration Gradient Across Test Sample	65
12. Testing of Two Materials in Both Parallel and Counterflow Arrangements in the DMPC	66
13. Instrumented Test Fabric in DMPC to Record Temperature Changes of Hygroscopic Fabrics	68
14. Temperature Changes of Two Layers of Cotton Fabric Subjected to Changes in Relative Humidity, at a Constant Gas Flow Temperature of 20°C	70

LIST OF FIGURES (continued)

Figure	page
15. Temperature Changes due to Water Vapor Sorption for Seven Fabrics During Step Change in Relative Humidity from 0.0 to 1.0	71
16. Temperature Changes for the Three Thermocouples, for the Cotton Fabric, due to Water Vapor Sorption During Step Change in Relative Humidity from 0.0 to 1.0	71
17. Variability in Measured Temperature Changes for the Cotton Fabric, due to Water Vapor Sorption During Step Change in Relative Humidity from 0.0 to 1.0	72
18. Relative Humidity Normalized by Final Equilibrium Value, During Step Change from 0.0 to 0.6, at Constant Temperature of 20°C	73
19. Convection/Diffusion Experiment in the DMPC	74
20. Experimental Curve of Measured Relative Humidity at Outlet of DMPC, as Pressure Drop Across the Fabric is Varied	74
21. Automated Gas Permeability Test	77
22. Pressure Drop Versus Volumetric Flow Rate for Two Fabrics as a Function of Relative Humidity	79
23. Illustration of the Additivity of Darcy Flow Resistance for One and Three Layers of Nonhygroscopic Polyester Fabric	81
24. Apparent Flow Resistance of Seven Fabrics as a Function of Relative Humidity	82
25. Reversal of Air Permeability Ranking due to Relative Humidity Test Conditions for Two Fabrics	83
26. Decrease in Permeability of Five Nylon Fabrics as a Function of Relative Humidity	84
27. Volume Fraction of Water Absorbed by the Fiber for the 100% Cotton Fabric Compared to the Darcy Flow Resistance as a Function of Relative Humidity	85

LIST OF FIGURES (continued)

Figure	page
28. DMPC Configuration for Measuring Humidity-Dependent Air Permeability of Textiles	87
29. Apparent Flow Resistance of Seven Fabrics as a Function of Relative Humidity as Determined in the Air Permeability Flow Cell and the DMPC	87
30. Control Volumes and Grid Points for Finite-Difference Method	90
31. Locations of Staggered Control Volume for Velocity Components and Momentum Equations	91
32. Discretization of Grid for x -Direction Only	92
33. Nondimensional Unsteady Heat Conduction Boundary and Initial Conditions	96
34. Uniform Grid Used in SIMPLEC for Comparison with Results of Patankar and Baliga	97
35. Predicted Value of the Dimensionless Temperature Gradient at the Surface for Time Step Parameter $\lambda = 16$	98
36. Schematic for Definition of "Bound Water" Volume Fraction	102
37. Comparison of Numerical Predictions to Experimental Results of Centerline Temperature of Wool, Cotton, and Silk fabrics Subjected to Step Change in Relative Humidity	106
38. Comparison of Numerical Predictions to Experimental Results of Centerline Temperature of Wool/Polyester, Nylon/Cotton, and Polyester fabrics Subjected to Step Change in Relative Humidity	107
39. Fractional Approach to Equilibrium at 65% Relative Humidity, 20°C, for Seven Fabrics	108
40. Numerical Experiment to Simulate Sudden Temperature and Relative Humidity Change on One Side of a Fabric	109

LIST OF FIGURES (continued)

Figure	page
41. Change in Centerline Temperature of a Hygroscopic and a Nonhygroscopic Fabric as the Ambient Conditions are Changed on One Side	110
42. Calculated Temperature Profiles Through the Nonhygroscopic and the Hygroscopic Fabric at Various Times After Conditions on One Side are Changed	111
43. Calculated Water Vapor Concentration Profiles Through the Nonhygroscopic and the Hygroscopic Fabric at Various Times After Conditions on One Side are Changed	112
44. Effect of Number of Grid Points on Temperature Transients During Water Vapor Sorption of a Hygroscopic Fabric	113
45. Effect of Time Step on Calculation of Temperature Transients During Water Vapor Sorption of a Hygroscopic Fabric	114
46. Effect of Different Boundary Heat Transfer Coefficients on Calculation of Temperature Transients During Water Vapor Sorption for a Cotton Fabric	116
47. Effect of Different Boundary Heat Transfer Coefficients on Water Vapor Sorption for a Cotton Fabric	116
48. Model Geometry to Simulate Convection/Diffusion Processes in the DMPC	126
49. Comparison of Experimental Versus Numerical Results for Convection/Diffusion in the DMPC for the Cotton Fabric	127
50. Comparison of Experimental Versus Numerical Results for Convection/Diffusion in the DMPC for the Wool Fabric	128
51. Comparison of Experimental Versus Numerical Results for Convection/Diffusion in the DMPC for the Silk and Polyester Fabric	129

LIST OF FIGURES (continued)

Figure	page
52. Comparison of Experimental Versus Numerical Results for Convection/Diffusion in the DMPC for the Wool/Polyester, Nylon/Cotton, and Nylon Fabrics	129
53. Flow Field Simulation Obtained Using Numerical Solution for Cotton Fabric in DMPC at Two Different Pressure Drop Conditions. Gas Phase Velocity Vectors, and Contours of Relative Humidity Shown on Plot.....	131
54. (a) Numerical and (b) Experimentally Measured Temperature Transients for a Hygroscopic Cotton Fabric Subjected to a Sudden Change in Relative Humidity	132
55. Numerical Simulation of Flow Field and Temperature for Cotton Fabric Undergoing Water Vapor Sorption Under Relative Humidity Step Change from 0.0 to 1.0 at 20°C (293 K)	133
56. One-Dimensional Thermoregulatory Model of the Human Body	136
57. Human Thermal Control System Model Combined with Clothing Material Model	140
58. Calculated Temperatures for Nonhygroscopic and Hygroscopic Fabrics Covering a Sweating Human, When Subjected to Large Changes in Environmental Relative Humidity	143
59. Skin Surface Relative Humidity, Liquid Sweat Accumulation at Skin Surface, and Vaporization Rate at Skin Surface, for Hygroscopic Fabric Case	144
60. Skin Surface Relative Humidity, Liquid Sweat Accumulation at Skin Surface, and Vaporization Rate at Skin Surface, for Hygroscopic Fabric Case	144
61. Calculated Fabric Surface Temperatures, for Four Layering Arrangements	145
62. Calculated Skin Surface Temperatures, for Four Layering Arrangements	146

LIST OF FIGURES (continued)

Figure	page
63. Differences in Calculated Temperatures Between Hygroscopic and Nonhygroscopic Fabrics, for a Change in Work Rate from 20 to 100 W/m ²	147
64. Differences in Vaporization Rate and Skin Relative Humidity for Hygroscopic and Nonhygroscopic Fabrics, for a Change in Work Rate from 20 to 100 W/m ²	148
65. Typical Appearance of Capillary Pressure Curves as a Function of Liquid Saturation for Porous Materials	149
66. Comparison of a Wicking Versus a Nonwicking Fabric (other properties identical) During Changes in Human Work Rate	154
A-1. Schematic of Fabric Illustrating Intrayarn and Interyarn Volume Fraction	174

LIST OF TABLES

Table	page
3-1 Test Fabrics	58
3-2 Woven Test Fabrics	67
3-3 Nine Setpoints for Transient Diffusion Tests	69
6-1 Coefficients for General Transport Equations	125
7-1 Assumed Fabric Layer Properties	141
A-1 Fabric Physical Properties	175
A-2 Diffusion Properties of Two Fabric Layers	176
A-3 Textile Fiber Thermal Properties	177
A-4 Darcy Flow Resistance Properties	178
B-1 Sorption Rate Factors	180

PREFACE

The work described in this report was partly funded by the In-House Laboratory Independent Research (ILIR) project "Unsteady Heat and Mass Transfer Through Clothing Materials." This same ILIR project also supported the author's doctoral thesis, on the same subject matter, while the author was a student at the University of Massachusetts Lowell. Professor Majid Charmchi, the author's academic advisor in the Department of Mechanical Engineering, provided guidance, suggestions, and encouragement during all aspects of this work. Don Rivin, Cy Kendrick, Ron Segars, Tom Tassinari, and Tom Pease of the U.S. Army Soldier Systems Command, Natick Research, Development and Engineering Center, also contributed to the work contained in this report either through their advice on technical matters, or through their managerial support.

NOMENCLATURE

Roman Letters

A	area [m ²]
$a_{\sigma\beta}$	$A_{\sigma\beta}/V$, surface of the σ - β interface per unit volume [m ⁻¹]
$\mathcal{A}_m(t)$	material surface [m ²]
c_a	water vapor concentration of ambient atmosphere [kg/m ³]
c_p	constant pressure heat capacity [J/kg-°K]
$(c_p)_{ds}$	constant pressure heat capacity of the dry solid [J/(kg·K)]
$(c_p)_w$	constant pressure heat capacity of liquid water [4182 J/(kg·K) @290 K]
$(c_p)_v$	constant pressure heat capacity of water vapor [1862 J/(kg·K) @290 K]
$(c_p)_a$	constant pressure heat capacity of dry air [1003 J/(kg·K) @290 K]
C_p	mass fraction weighted average constant pressure heat capacity [J/kg-°K]
c_{so}	water vapor concentration at skin surface [kg/m ³]
ΔC_a	concentration difference between the two gas streams at one end of the flow cell [kg/m ³]
ΔC_b	concentration difference between the two gas streams at other end of the flow cell [kg/m ³]
d_f	effective fiber diameter [m]
$\mathcal{D}$	gas phase molecular diffusivity [m ² /sec]
D_a	diffusion coefficient of water vapor in air [m ² /sec]
D_{eff}	effective gas phase diffusivity [m ² /sec]
D_{solid}	effective solid phase diffusion coefficient [m ² /s]
E	metabolic energy generated in muscle layer due to exercise [W/m ³ -s]
$\bar{g}$	gravity vector [m/sec ²]
h	enthalpy per unit mass [J/kg]
h_c	convective heat transfer coefficient [W/m ² -s]
h_m	convective mass transfer coefficient [m/s]
h°	reference enthalpy [J/kg]
$\bar{h}_i$	partial mass enthalpy for the i th species [J/kg]
$h_{\sigma\beta}$	heat transfer coefficient for the σ - β interface [J/sec-m ² -K]
Δh_{vap}	enthalpy of vaporization per unit mass [J/kg]
k	thermal conductivity [J/sec-m-°K]
k_B	thermal conductivity of body tissue [J/(s·m·K)]
k_{so}	reference body tissue thermal conductivity [0.498 J/(s·m·K)]
k_{ds}	thermal conductivity of the dry solid [J/(s·m·K)]

NOMENCLATURE (continued)

k_w	thermal conductivity of liquid water [0.600 J/(s·m·K) @290 K]
k_v	thermal conductivity of saturated water vapor [0.0246 J/(s·m·K) @380K]
k_a	thermal conductivity of dry air [0.02563 J/(s·m·K) @290 K]
k_ε	$\partial\langle P_c \rangle / \partial \varepsilon_\beta$ [N/m ²]
$k_{\langle T \rangle}$	$\partial\langle P_c \rangle / \partial \langle T \rangle$ [N/m ² ·°K]
K	permeability coefficient [m ²]
K_β	Darcy permeability for liquid phase [m ²]
K_β	liquid phase permeability tensor [m ² /sec]
L	total half-thickness of body model system [0.056 m]
$\dot{m}$	mass flux of water vapor across the sample [kg/s]
$\langle \dot{m}_{sl} \rangle$	mass rate of desorption from solid phase to liquid phase per unit volume [kg/sec-m ³] $\langle \dot{m}_{sl} \rangle = \frac{1}{V} \int_{A_{\sigma\beta}} \rho_\sigma (\vec{v}_\sigma - \vec{w}_2) \cdot \vec{n}_{\sigma\beta} dA$
$\langle \dot{m}_{sv} \rangle$	mass rate of desorption from solid phase to vapor phase per unit volume [kg/sec-m ³]
$\langle \dot{m}_{lv} \rangle$	mass rate of evaporation per unit volume [kg/sec-m ³]
M_a	molecular weight of air [28.97 kg/kgmole]
M_w	molecular weight of water vapor [18.015 kg/kgmole]
M_{avl}	liquid water available on skin surface for evaporation [kg/m ²]
$\vec{n}$	outwardly directed unit normal
p	pressure [N/m ²]
p_γ	total gas pressure [N/m ²]
p_a	partial pressure of air [N/m ²]
p_v	partial pressure of water vapor [N/m ²]
p_s	saturation vapor pressure (function of T only) [N/m ²]
P_c	$p_\gamma - p_\beta$, capillary pressure [N/m ²]
p_0	reference pressure [N/m ²]
p_1°	reference vapor pressure for component 1 [N/m ²]
q_m	rate of heat generation due to metabolism [W/m ³ ·s]
Q	volumetric flow rate [m ³ /sec]
Q_{sl}, Q_l	enthalpy of desorption from solid phase per unit mass [J/kg]
$\vec{q}$	heat flux vector [J/sec-m ²]
ΔQ_{sh}	metabolic energy generated due to shivering [W/m ³ ·s]
$\vec{r}$	position vector [m]
r	characteristic length of a porous media [m]

NOMENCLATURE (continued)

R_i	gas constant for the i th species [N-m/kg-°K]
R_i	intrinsic diffusion resistance of sample [s/m]
R	universal gas constant [8314.5 N-m/(kg-K)]
R_f	textile measurement (@ $\phi=0.65$), grams of water absorbed per 100 grams of fiber [fraction]
R_{skin}	equilibrium regain at fiber surface [fraction]
R_{total}	total fiber regain from last time step [fraction]
R_{bl}	diffusion resistance of boundary air layers [s/m]
s	sweating rate [kg/m ² -s]
s_0	reference basal sweating rate [2.80 x 10 ⁶ kg/m ² -s]
S	saturation, fraction of void space occupied by liquid [fraction]
s_{ir}	irreducible saturation; saturation level at which liquid phase is discontinuous
T	temperature [°K]
T_a	ambient air temperature [K]
T_B	average body temperature [K]
T_0	reference temperature [°K]
T°	reference temperature [°K]
T_s	reference temperature at standard conditions of 0°C in degrees K (273.15 K)
T_{sk}	skin temperature [K]
ΔT_B	deviation from the body's setpoint [K]
$\mathbf{T}$	total stress tensor [N/m ²]
t	time [sec]
$\bar{u}_i$	diffusion velocity of the i th species [m/s]
$\bar{v}$	mass average velocity [m/s]
$\bar{v}_i$	velocity of the i th species [m/s]
$\langle \bar{v}_\beta \rangle$	volume average liquid velocity [m/s]
$V_\sigma(t)$	volume of the solid phase contained within the averaging volume [m ³]
$V_\beta(t)$	volume of the liquid phase contained within the averaging volume [m ³]
$V_\gamma(t)$	volume of the gas phase contained within the averaging volume [m ³]
$\mathcal{V}$	averaging volume [m ³]
$\mathcal{V}_m(t)$	material volume [m ³]
$\bar{w}$	velocity of the β - γ interface [m/sec]
$\bar{w}_1$	velocity of the σ - γ interface [m/sec]
$\bar{w}_2$	velocity of the σ - β interface [m/sec]

NOMENCLATURE (continued)

Greek Letters

α_{k1}	thermal proportional control coefficient [1/K]
α_{k2}	thermal proportional control coefficient [1/K]
α_{sh1}	shivering proportional control coefficient [W/m ³ -K]
α_{sh2}	shivering proportional control coefficient [W/m ³ -K ²]
α_{sh3}	shivering proportional control coefficient [W/m ³ -K]
α_{s1}	sweating proportional control coefficient [kg/m ² -s-K]
α_{s2}	sweating proportional control coefficient [kg/m ² -s-K ⁴]
δC	$= C_2 - C_1$, water vapor concentration difference between incoming stream (C_1) and outgoing stream (C_2) in top or bottom portion of the moisture permeation cell [kg/m ³]
$\delta \phi$	$= \phi_2 - \phi_1$, relative humidity difference between incoming stream (ϕ_1) and outgoing stream (ϕ_2) in top or bottom portion of the moisture permeation cell
$\Delta \bar{C}$	log mean concentration difference between top and bottom nitrogen streams [kg/m ³]
$\varepsilon_o(t)$	V_o/V , volume fraction of the solid phase
$\varepsilon_\beta(t)$	V_β/V , volume fraction of the liquid phase
$\varepsilon_\gamma(t)$	V_γ/V , volume fraction of the gas phase
ε_{ds}	V_{ds}/V , volume fraction of the dry solid (constant)
$\varepsilon_{bw}(t)$	V_{bw}/V , volume fraction of the water dissolved in the solid phase
Φ	rate of heat generation [J/sec-m ³]
ϕ	p/p_s , relative humidity
$\bar{\lambda}$	unit tangent vector
γ	thermal rate control coefficient [s/K]
μ	shear coefficient of viscosity [N-sec/m ²]
μ_β	viscosity of the liquid phase [for water, 9.8×10^{-4} kg/m-sec at 20°C]
ρ	density [kg/m ³]
ρ_β	density of liquid phase [kg/m ³]
ρ_i	density of the i th species [kg/m ³]
ρ_{ds}	density of dry solid [for polymers typically 900 to 1300 kg/m ³]
ρ_w	density of liquid water [approximately 1000 kg/m ³]
ρ_γ	density of gas phase (mixture of air and water vapor) [kg/m ³]
ρ_v	density of water vapor in the gas volume (equivalent to mass concentration) [kg/m ³]

NOMENCLATURE (continued)

ρ_a	density of the inert air component in the gas volume (equivalent to mass of air/total gas volume) [kg/m ³]
τ	viscous stress tensor [N/m ²]
τ	tortuosity factor
$\vec{\xi}$	thermal dispersion vector [J/sec-m ³]
ξ	dummy integration variable
ξ	a function of the topology of the liquid phase

Subscripts

i	designates the i th species in the gas phase
l	liquid
L	liquid
s	solid
S	solid
σ	designates a property of the solid phase
β	designates a property of the liquid phase
γ	designates a property of the gas phase
$\sigma\beta$	designates a property of the σ - β interface
$\sigma\gamma$	designates a property of the σ - γ interface
$\beta\gamma$	designates a property of the β - γ interface

Mathematical Symbols

d/dt	total time derivative
D/Dt	material time derivative
$\partial/\partial t$	partial time derivative
$\langle \psi \rangle$	spatial average of a function ψ which is defined everywhere in space
$\langle \psi_\beta \rangle$	phase average of a function ψ_β which represents a property of the β phase
$\langle \psi_\beta \rangle^\beta$	intrinsic phase average of a function ψ_β which represents a property of the β phase

MULTIPHASE HEAT AND MASS TRANSFER THROUGH HYGROSCOPIC POROUS MEDIA WITH APPLICATIONS TO CLOTHING MATERIALS

CHAPTER 1

INTRODUCTION

The transfer of heat and mass from a clothed human being to the environment is a complex process. In rare instances, the flows of mass and energy may be treated as a steady-state problem. In most cases, however, the steady-state situation is not achieved, and the dynamics of the transport of energy and mass must be understood in order to predict how much energy is being exchanged between the human body and the environment.

The dissipation of heat from the human body is often a particularly critical problem for protective clothing systems. For example, chemical protective clothing systems must protect people from vapors, liquids, and aerosols, but they must also maximize the heat dissipated to the environment to minimize the heat stress imposed upon the wearer. As a practical matter, most chemical protective clothing systems are quite thin, and the heat transfer due to conduction, convection, and radiation is comparable to many other types of clothing. Since the principal mechanism by which humans dissipate heat in hot environments is the evaporation of sweat from the skin surface, with the water vapor carrying away the enthalpy of vaporization, the barrier properties of the chemical protective clothing will usually also impede the movement of water vapor. The way in which the clothing layers mediate the flow of liquid and vapor become the dominant factor controlling the dissipation of heat from the human body.

The variety of materials which are contained in a clothing system present a very wide spectrum of properties. Woven and nonwoven textiles may include both hygroscopic and non-hygroscopic fibers. Transport through the clothing system involves diffusion of heat and moisture, convective air flows, and liquid water capillary wicking. Hygroscopic fibers may absorb water in vapor or liquid form and release the heat of sorption, which serves as a energy source within the clothing. Depending on the ambient environment, water vapor may condense in outer layers of clothing, which liberates the heat of condensation and serves as another heat source within the clothing. Many modern protective clothing systems also include polymeric membranes, which may be either a microporous hydrophobic polymer (e.g. polytetrafluoroethylene), or a very thin solid layer of a hydrophilic polymer (e.g. polyurethane). The various steps involved in sorption of liquid water or vapor into the membrane, diffusion through the structure, and desorption from the other side, are often complicated by the concentration-dependent permeation properties of many of the polymers in common use.

All of these processes are time-dependent. Particularly for the hygroscopic materials, equilibrium does not take place within a matter of seconds, but may require time scales of minutes to hours. Since humans are rarely working at a sustained constant work level for hours on end, steady-state approximations to determine quantities such as total moisture accumulation within the clothing, or total heat and mass transferred through the clothing, are often inaccurate since the steady-state heat and mass transfer properties are inapplicable.

The intent of this study was to develop a useful model of the coupled transport of energy and mass through porous hygroscopic clothing materials, and to verify that model by experimental measurements. Such a model should account for the complicating factors of sorption, condensation, evolution of heat, liquid water capillary transport, and coupled diffusion of heat and mass. The model is also integrated with existing human thermal model, which provides time-dependent boundary conditions as a function of human work rate, core temperature, skin temperature, and evaporative flux. This made it possible to assess the transient behavior of various clothing systems and to optimize properties both on the basis of an individual layer, and the arrangement of multiple layers.

BACKGROUND

Traditionally, studies of the coupled transfer of heat and moisture through textile materials focused on the diffusion of heat and water vapor. Liquid water accumulation and transport within a textile layer are usually presented as complicating factors which are neglected.

Fundamental studies by Henry [1] on how the sorptive properties of wool fibers produce a buffering effect between the human body and the ambient environment were incorporated by Crank [2] in his classic work on the mathematics of diffusion. The heat released as water vapor diffuses through, and is absorbed by, a bed of hygroscopic fibers, is significant for processing operations, such as when bales of wool or cotton must be dried to a specified moisture content. Similarly, for woollen clothing, the heat released or absorbed when the relative humidity or temperature within a woollen textile layer changes is often physiologically significant. Various workers have extended Henry's approach over the years, most notably Nordon and David [3], Farnworth [4], Wehner [5], and Li and Holcombe [6]. While continuing to concentrate on the coupled diffusion processes for energy and water vapor through porous hygroscopic layers, each group made different assumptions concerning the nature of the sorption process within the hygroscopic fiber.

The groups mentioned above all focused on woven and nonwoven textiles as the material of interest. A great many more studies have been undertaken for the coupled heat and mass transfer problem through porous media in general. Much of the literature on drying of capillary porous bodies or hygroscopic porous materials is directly applicable to the problem of mass and energy transport through textile materials. Many studies concentrate on a limited set of materials, derive a small number of partial differential equations based on either diffusion or convection, and develop solutions based on these particular situations. Well-known examples of this approach include studies by Berger and Pei [7], Eckert and Faghri [8], and Stanish et. al. [9]. More comprehensive systems of equations which account for diffusion of heat and water vapor, along with convective gas flows, and capillary liquid transport, are available in works by Luikov [10] and Whitaker [11]. The approach taken by Whitaker, in particular, has been adopted by many groups investigating transfer of heat and moisture through food materials (drying) and thermal insulation.

Simonson, Tao, and Besant [12] have successfully applied Whitaker's approach to a fibrous insulation which releases heat due to sorption of water vapor by the fiber, and also by condensation occurring within the porous structure. They showed that the transient nature of the heat flow process in fibrous materials can produce large errors if the fiber heat of sorption is ignored in an analysis.

Most approaches to studying heat and mass transfer through porous media utilize constant temperature or constant flux boundary conditions. For heat and mass transfer from the human body, however, the interaction between the human thermal control system and the behavior of the clothing system can become as important as the behavior of the clothing system itself. Efforts to incorporate human thermal models as part of a complete simulation of the dynamic behavior of the human-clothing-environment system have been undertaken. To date, all of them only incorporate diffusion of heat and diffusion of water vapor within the clothing system, and ignore convection of air and liquid wicking.

Ma, et. al. [13], developed a detailed thermal model of the interactions between a hygroscopic clothing layer, the environment, and the human body. This one-dimensional model divided the human body into a core region which generated the majority of the basal metabolic heat, a muscle region which also generated some heat through exercise and shivering, and a skin layer. The appropriate thermal properties such as heat capacity and thermal conductivity for each tissue layer are used to write the time-dependent partial differential equations for the various layers, which were then solved numerically. This human model was coupled to a clothing model very similar to that developed by Henry [1], and Norden and David [3], and was fairly successful at assessing the influence of various clothing parameters upon such variables such as human core and skin temperature. A less sophisticated, but equally valid, one-

dimensional approach was used by Li, et. al. [14], to determine the change in surface temperature of a hygroscopic fabric close to the skin, and the perceived tactile sensation as a result of the buffering effect of a hygroscopic fabric as opposed to a non-hygroscopic fabric. Perhaps the most sophisticated approach taken to date is that of Jones [15], who used a two-node model of the human thermal control system, coupled to a segmented whole-body clothing model, to determine the interactions between the human thermal control system, the clothing system, and the external environment, and compared the transient results with traditional steady-state model results. Even in this study, liquid water accumulation and transport was neglected, even though it is known that such factors are very important when humans are sweating heavily inside multi-layer clothing systems.

It is clear that there are a variety of basic questions concerning the optimum material properties desired in protective clothing systems. Even under steady-state conditions, there is uncertainty about the importance of wicking, air permeability, diffusion, and the proper arrangement of the various layers of a clothing system. When we try to consider the unsteady-state behavior of these materials, the problem becomes even more difficult, particularly since there are often several different materials even within a single textile layer.

To date, research on heat and mass transfer through textile materials has concentrated on simplified models which usually neglect convection and capillary transport. There is a need for comprehensive models of clothing material behavior which can include more than just vapor diffusion. It is apparent that the very large body of work available in the general technical literature on coupled heat and mass transfer through porous media has been largely ignored by researchers in the textile field. The application of these existing sophisticated and experimentally verified analysis techniques to unsteady heat and mass transfer through textile materials is a necessary step in advancing our understanding of the interactions between human, clothing, and environment.

Since the performance of clothing systems is intimately connected to the thermal properties of the human wearing the system, it is desirable to couple the dynamic behavior of the clothing system to the human physiology of heat regulation. A versatile model of unsteady heat and mass transfer through textile materials, well-verified with experimental data, would be a valuable tool both for basic research, and for the evaluation and design of new protective clothing concepts.

The main objective of this investigation is to develop a numerical model capable of including all the major modes of energy and mass transport which occur in clothing. The model will include diffusion of heat and mass, condensation and evaporation of water within the clothing system, sorption and diffusion of water within the solid matrix, liquid water capillary transport, and forced convection of air and water vapor through the porous structure. Such a detailed model will be a valuable tool during the design and evaluation process for new protective clothing concepts based on modern multi-component textile technology.

The approach relied on both modeling and experiment. A set of partial differential equations describing time-dependent heat and mass transfer through porous hygroscopic materials were developed. The theory developed by Whitaker [11], based on the control volume approach, was the starting point for the system of equations. Since Whitaker developed his equations for a rigid inert porous matrix, which only participates in the transport process through its thermal properties, the derivations had to be modified to account for the absorption and diffusion of water within the solid matrix. Water in a hygroscopic porous solid may exist in vapor or liquid form in the pore spaces or in bound form when it has been absorbed by the solid, which is typically some kind of hydrophilic polymer. Factors such as the swelling of the solid due to water imbibition, and the heat of sorption evolved when the water is absorbed by the polymeric matrix, were incorporated into the appropriate conservation and transport equations.

The numerical methods and experimental techniques treated the transport problem as either one-dimensional and two-dimensional cases. This simplified the numerical analysis considerably, both in terms of computational expense, and in the ability to use a variety of well-known finite-difference methods for the solution of the system of equations. The primary method used was Patankar's control volume approach [17]. This implicit method has been used for similar problems concerning the coupled transfer of heat and mass in insulation materials [18, 19].

The numerical experiments for the finite-difference code were compared to results from laboratory experiments performed in support of the numerical modeling effort. The transport coefficients and parameters required in the partial differential equations, such as thermal conductivity, diffusion coefficient, heat capacity, sorption isotherms and heats of sorption, etc., were determined using equipment already available, equipment built especially for this purpose, or values taken from the literature.

Following the verification of the numerical model for coupled heat and mass transfer through hygroscopic porous materials, an existing human thermal control model was selected to provide time-dependent boundary conditions for the numerical model. A variety of models exist and are available for use. The segmented whole-body models of Stolwijk [21] and Wissler [22], which are among the most well-known and accurate, were all candidates which could be adapted to provide a generic simplified model which would provide appropriate boundary conditions for a one-dimensional material model. The simplest model to use is probably the two-node model of Gagge et. al. [23, 24], which is in wide use, and requires little modification.

The combination of a valid material model with a well-known human thermal physiology model provided an opportunity to systematically examine a number of clothing parameters, and determine their importance under various conditions of human work rates and environmental conditions. However, for this portion of the study, it was difficult to verify the results by experiment, since most human thermal tolerance studies usually assume constant clothing heat and mass transfer coefficients.

This research effort is aimed at providing a useful analysis tool for studies of the various transport phenomena affecting the thermal balance of clothed humans under different environmental conditions. It addresses the current need for more realistic models of coupled heat and mass transfer through hygroscopic porous textile materials which include phenomena such as liquid water wicking, condensation/evaporation within textile layers, concentration-dependent permeation behavior of semipermeable membrane laminates, and the sorption behavior of hygroscopic textile fibers.

The present design and evaluation process for a new protective clothing system is very expensive and time-consuming. Textile materials, which are the basic building blocks of any garment, are usually selected based on a few steady-state properties determined from laboratory tests. A long process of design iteration using heated manikins, and human wear trials in environmentally-controlled chambers, is then begun in order to evaluate the dynamic interactions between the human, clothing, and the environment. The numerical model described here will make a significant impact on all aspects of the design process by incorporating many of these dynamic interactions into the basic material selection process, hopefully obviating some of the need for such extensive human chamber evaluations.

CHAPTER 2

GOVERNING EQUATIONS

2.1 Introduction

The basis for the set of governing equations is Whitaker's comprehensive theory for mass and energy transport through porous media [25]. These equations are also applicable to mass and energy transport through textile materials if some modifications are made. Whitaker modeled the solid portion of the solid matrix as a rigid inert material which only participates in the transport process through its thermal properties. In hygroscopic textile materials the diffusion of water into the solid is a significant part of the total transport process. The inclusion of the extra transport terms into and out of the solid matrix necessitate extensive modifications of Whitaker's original derivations.

The structure of this chapter follows Whitaker's derivations as closely as possible. Many references are made to his original derivation. Where possible, the nomenclature and symbols are identical to Whitaker's original derivation to facilitate cross-referencing between this modified set of equations and Whitaker's original set of equations.

2.2 Mass and Energy Transport Equations

The hygroscopic porous media is modeled as shown below.

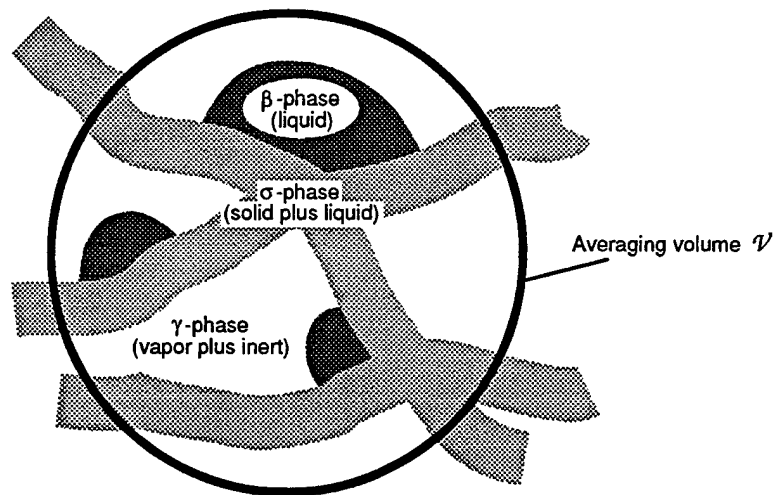


Figure 1. Three phases present in hygroscopic porous media.

A typical porous hygroscopic textile material may be described as a mixture of a solid phase, a liquid phase, and a gaseous phase. The solid σ phase consists of the solid material (usually a polymer e.g. wool or cotton) plus any bound water absorbed in the solid polymer matrix. The solid phase is thus a mixture of the solid and the liquid. This definition of the solid phase means that the density is dependent on the amount of water contained in the solid phase.

The liquid β phase consists of the free liquid water which may be present within the structure of the porous solid. This would also include water which is contained within the pore spaces of the solid but is not sorbed into the polymeric matrix. This liquid β phase is a pure component, and we will be able to assume a constant density for it.

The gaseous γ phase consists of the vapor component of the liquid (water vapor) plus the inert air component. Since it is a mixture of water vapor and air, its density will not be constant, but will be a function of temperature, concentration, etc.

The appropriate general transport and conservation equations to be used are:

continuity equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (2.2.1)$$

linear momentum:

$$\rho \frac{D\vec{v}}{Dt} = \rho \vec{g} + \nabla \cdot \vec{T} \quad (2.2.2)$$

thermal energy equation:

$$\rho \frac{Dh}{Dt} = -\nabla \cdot \vec{q} + \frac{Dp}{Dt} + \nabla \vec{v} : \tau + \Phi \quad (2.2.3)$$

In keeping with Whitaker's derivation, we will neglect the viscous stress tensor.

Point Equations

σ -Phase -- Solid

The solid σ phase is made up of the true dry solid (polymer) plus any of the liquid phase or the vapor component of the gas phase which has dissolved into it or adsorbed onto its surface. This may also result in a volume change for the solid σ phase (swelling). The solid is now a mixture of the true dry solid plus the liquid, so we now must account for the two components.

Since swelling is possible, which results in a small velocity of the solid portion due to its displacement, we must account for the velocity of the solid by using the continuity equation:

$$\frac{\partial \rho_\sigma}{\partial t} + \nabla \cdot (\rho_\sigma \bar{v}_\sigma) = 0 \quad (2.2.4)$$

and for the two components of liquid (1) + solid component (2), the species continuity equation is:

$$\frac{\partial \rho_j}{\partial t} + \nabla \cdot (\rho_j \bar{v}_j) = 0, \quad i = 1, 2, \dots \quad (2.2.5)$$

The σ phase density is not constant, since it includes the density of the solid plus the density of the liquid contained within the solid. The species densities are likewise not constant, since the species density is calculated on the basis of the total phase volume.

For the two species:

$$\rho = \frac{m_1 + m_2}{V_\sigma} = \frac{m_1}{V_\sigma} + \frac{m_2}{V_\sigma} = \rho_1 + \rho_2 \quad (2.2.6)$$

We will assume that the dry density of the solid, and the density of the liquid are constant, and will call them ρ_s and ρ_L .

For the solid phase, we can divide the solid phase volume into the fraction taken up by the liquid, and the fraction taken up by the solid:

$$\epsilon_{\sigma L} = \frac{\text{Volume of Liquid}}{\text{Total } \sigma \text{ Phase Volume}} \quad (2.2.7)$$

The relation between the species densities and the solid and liquid densities is:

$$\rho = \epsilon_{\sigma L} \rho_L + (1 - \epsilon_{\sigma L}) \rho_s = \rho_1 + \rho_2$$

$$\rho_1 = \epsilon_{\sigma L} \rho_L \quad (2.2.8 - 2.2.10)$$

$$\rho_2 = (1 - \epsilon_{\sigma L}) \rho_s$$

The density and velocity of the mixture, in terms of the species densities, is given as:

$$\rho_\sigma = \rho_1 + \rho_2 \quad (2.2.11 - 2.2.12)$$

$$\rho_\sigma \bar{v}_\sigma = \rho_1 \bar{v}_1 + \rho_2 \bar{v}_2$$

or

$$\rho_\sigma = \epsilon_{\sigma L} \rho_L + (1 - \epsilon_{\sigma L}) \rho_s \quad (2.2.13 - 2.2.14)$$

$$\rho_\sigma \bar{v}_\sigma = \epsilon_{\sigma L} \rho_L \bar{v}_1 + (1 - \epsilon_{\sigma L}) \rho_s \bar{v}_2$$

The species velocity is written in terms of the mass average velocity and the diffusion velocity as:

$$\bar{v}_i = \bar{v}_\sigma + \bar{u}_i \quad (2.2.15)$$

and the continuity equation becomes:

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \bar{v}_\sigma) = -\nabla \cdot (\rho_i \bar{u}_i), \quad i = 1, 2, 3, \dots \quad (2.2.16)$$

The diffusion flux may be written in terms of a diffusion coefficient as:

$$\rho_i \bar{u}_i = -\rho_\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_i}{\rho_\sigma} \right) \quad (2.2.17)$$

and the continuity equation may be written as:

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \bar{v}_\sigma) = \nabla \cdot \left\{ \rho_\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_i}{\rho_\sigma} \right) \right\}, \quad i = 1, 2, 3, \dots \quad (2.2.18)$$

For the purposes of comparison to other models of heat and mass transfer through porous materials, it will be convenient later on to write these equations in terms of concentrations of water (component 1) in the solid (component 2).

We define the concentration of water in the solid (C_s) as:

$$C_s = \frac{\text{Mass of water}}{\text{Mass of the solid phase}} = \frac{m_1}{m_1 + m_2} = \frac{\rho_1}{\rho_\sigma} \quad (2.2.19)$$

If we only want to consider the continuity equation for the liquid phase, since it's really the only material moving into or out of the solid phase, we can just use the continuity equation for the liquid, which is:

$$\frac{\partial \rho_1}{\partial t} + \nabla \cdot (\rho_1 \bar{v}_\sigma) = \nabla \cdot \left\{ \rho_\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_1}{\rho_\sigma} \right) \right\} \quad (2.2.20)$$

Now depending on how we want to treat the solid velocity, we can rewrite this a couple of ways. If we say that the solid velocity is included, then in terms of the true liquid density, where the species density is given by:

$$\rho_1 = \varepsilon_{\sigma L} \rho_L \quad \text{and} \quad \rho_\sigma = \varepsilon_{\sigma L} \rho_L + (1 - \varepsilon_{\sigma L}) \rho_S, \quad (2.2.21)$$

the continuity equation can be rewritten as:

$$\rho_L \left[\frac{\partial \varepsilon_{\sigma L}}{\partial t} + \nabla \cdot (\varepsilon_{\sigma L} \bar{v}_\sigma) \right] = \nabla \cdot \{ \rho_\sigma \mathcal{D}_{L\sigma} \nabla (C_s) \} \quad (2.2.22)$$

or

$$\frac{\partial \varepsilon_{\sigma L}}{\partial t} + \nabla \cdot (\varepsilon_{\sigma L} \bar{v}_\sigma) = \left(1 - \frac{\rho_S}{\rho_L} \right) \nabla \cdot [\varepsilon_{\sigma L} \mathcal{D}_{L\sigma} \nabla (C_s)] + \frac{\rho_S}{\rho_L} \{ \nabla \cdot [\mathcal{D}_{L\sigma} \nabla (C_s)] \}. \quad (2.2.23)$$

If we can neglect the solid velocity, the continuity equation becomes:

$$\frac{\partial \varepsilon_{\sigma L}}{\partial t} = \left(1 - \frac{\rho_S}{\rho_L} \right) \nabla \cdot [\varepsilon_{\sigma L} \mathcal{D}_{L\sigma} \nabla (C_s)] + \frac{\rho_S}{\rho_L} \{ \nabla \cdot [\mathcal{D}_{L\sigma} \nabla (C_s)] \} \quad (2.2.24)$$

We must also include the momentum balance:

$$\rho_{\sigma} \frac{D\bar{v}_{\sigma}}{Dt} = \rho_{\sigma} \bar{g} + \nabla \cdot \bar{T}_{\sigma} \Rightarrow \rho_{\sigma} \left\{ \frac{\partial \bar{v}_{\sigma}}{\partial t} + (\bar{v}_{\sigma} \cdot \nabla) \bar{v}_{\sigma} \right\} = \rho_{\sigma} \bar{g} + \nabla \cdot \bar{T}_{\sigma} \quad (2.2.25)$$

According to Jomaa and Puigali [26] we may also write the linear momentum equation as:

$$\rho_{\sigma} \frac{\partial \bar{v}_{\sigma}}{\partial t} = \rho_{\sigma} \bar{g} + \nabla \cdot \bar{T}_{\sigma} \quad (2.2.26)$$

There are a couple of ways to address the mass average solid phase velocity. If we assume that the total thickness of the materials we are trying to model does not change, then total volume remains constant, and the change in volume of the solid is directly related either to the change in volume of the liquid phase, or the change in volume of the gas phase. Another approach is to let the total volume of the material change with time. As the material dries out, and the total mass changes, the thickness of the material will decrease with time proportional to the water loss which takes place. Allowing the thickness of the material to change with time would result in a solid phase velocity, which we could relate to the total material shrinkage. The two situations are illustrated in Figure 2 for a matrix of solid fibers undergoing shrinkage due to water loss.

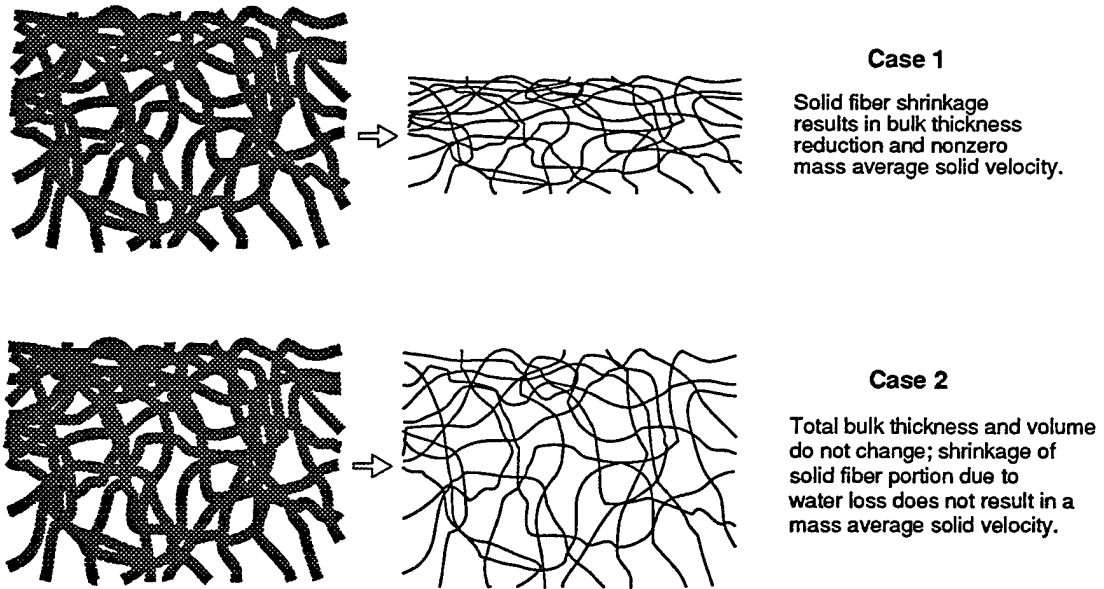


Figure 2. Two methods of accounting for shrinkage/swelling due to water uptake by a porous solid.

We will initially assume that the shrinkage behavior is like the first case shown. This means that we must include the mass average velocity in the derivations, and that the total material volume (or thickness in one dimension) will no longer remain constant.

Jomaa and Puiggali also give an equation for the solid velocity, in terms of the intrinsic phase average (discussed later) as:

$$\langle v_\sigma \rangle^\sigma = \frac{1}{\langle \rho \rangle^\sigma \xi^{n-1}} \int_0^\xi \frac{\partial}{\partial t} \langle \rho_\sigma \rangle d\xi \quad (2.2.27)$$

where ξ is the generalized space coordinate, with the origin at the center of symmetry, and n depends on the geometry ($n=1$ --plane, $n=2$ --cylinder, $n=3$ --sphere) according to the paper by Crapiste et. al. [27]

The thermal energy equation is:

$$\rho_\sigma \frac{Dh_\sigma}{Dt} = -\nabla \cdot \bar{q}_\sigma + \frac{Dp}{Dt} + \nabla \bar{v}_\sigma : \tau + \Phi_\sigma \quad (2.2.28)$$

Some simplifying assumptions can be made at this point by neglecting several effects. We'll start by dropping the reversible and irreversible work terms in the thermal energy equation, along with the source term, and expand the material derivative:

$$\rho_\sigma \frac{Dh_\sigma}{Dt} = \rho_\sigma \left(\frac{\partial h_\sigma}{\partial t} + \bar{v}_\sigma \cdot \nabla h_\sigma \right) = -\nabla \cdot \bar{q}_\sigma \quad (2.2.29)$$

It will be assumed that enthalpy is independent of pressure, and is only a function of temperature, and that heat capacity is constant for all the phases.

We can replace the enthalpy by:

$h = c_p T + \text{constant}$, in the σ , β , and γ phases.

We can now rewrite the thermal energy equation as:

$$\rho_\sigma \frac{\partial (c_p)_\sigma T_\sigma}{\partial t} + \rho_\sigma \left[\bar{v}_\sigma \cdot \nabla (c_p)_\sigma T_\sigma + \text{constant} \right] = -\nabla \cdot \bar{q}_\sigma \quad (2.2.30)$$

$$\rho_\sigma (c_p)_\sigma \frac{\partial T_\sigma}{\partial t} + \rho_\sigma (c_p)_\sigma \bar{v}_\sigma \cdot \nabla T_\sigma = \rho_\sigma (c_p)_\sigma \left\{ \frac{\partial T_\sigma}{\partial t} + \bar{v}_\sigma \cdot \nabla T_\sigma \right\} = -\nabla \cdot \bar{q}_\sigma \quad (2.2.31)$$

We may apply Fourier's law to obtain:

$$\rho_\sigma (c_p)_\sigma \left\{ \frac{\partial T_\sigma}{\partial t} + \bar{v}_\sigma \cdot \nabla T_\sigma \right\} = k_\sigma \nabla^2 T_\sigma \quad (2.2.32)$$

or, for a multi-component mixture:

$$\rho_{\sigma}(c_p)_{\sigma} \left(\frac{\partial T_{\sigma}}{\partial t} + \bar{v}_{\sigma} \cdot \nabla T_{\sigma} \right) = k_{\sigma} \nabla^2 T - \nabla \cdot \left(\sum_{j=1}^{j=N} \rho_j \bar{u}_j \bar{h}_j \right) \quad (2.2.33)$$

where $(c_p)_{\sigma} = \sum_{j=1}^{j=N} \frac{\rho_j}{\rho_{\sigma}} (\bar{c}_p)_j$

and the partial mass heat capacity and enthalpies $(\bar{c}_p)_j$, $\bar{h}_j$ are given by the partial molar enthalpy and the partial molar heat capacity divided by the molecular weight of that component.

β Phase -- Liquid

The continuity equation for the liquid phase is:

$$\frac{\partial \rho_{\beta}}{\partial t} + \nabla \cdot (\rho_{\beta} \bar{v}_{\beta}) = 0 \quad (2.2.34)$$

For the thermal energy equation, as we did before, we neglect compressional work and viscous dissipation:

$$\frac{Dp}{Dt} = \nabla \bar{v}_{\beta} : \tau_{\beta} = \Phi_{\beta} = 0 \quad (2.2.35)$$

which reduces the thermal energy equation to:

$$\rho_{\beta} \left(\frac{\partial h_{\beta}}{\partial t} + \bar{v}_{\beta} \cdot \nabla h_{\beta} \right) = -\nabla \cdot \bar{q}_{\beta} \quad (2.2.36)$$

If we assume enthalpy only depends on temperature and specific heat, as we did for the solid, we may write the thermal energy equation for the liquid phase as:

$$\rho_{\beta}(c_p)_{\beta} \left(\frac{\partial T_{\beta}}{\partial t} + \bar{v}_{\beta} \cdot \nabla T_{\beta} \right) = k_{\beta} \nabla^2 T_{\beta} \quad (2.2.37)$$

The liquid momentum equation will be discussed later in terms of a permeability coefficient which depends on the level of liquid saturation in the porous solid.

γ Phase -- Gas

The gas phase is made up of the vapor form of the liquid β phase, and an inert component (air). We do not need to modify any of the assumptions made by Whitaker for this phase, so we may simply write down the equations given by Whitaker [25].

continuity equation:

$$\frac{\partial \rho_\gamma}{\partial t} + \nabla \cdot (\rho_\gamma \bar{v}_\gamma) = 0 \quad (2.2.38)$$

and for the two components of vapor (1) + inert component (2) , the species continuity equation is:

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \bar{v}_i) = 0 \quad , \quad i = 1, 2, \dots \quad (2.2.39)$$

The density and velocity of the mixture is given as:

$$\begin{aligned} \rho_\gamma &= \rho_1 + \rho_2 \\ \rho_\gamma \bar{v}_\gamma &= \rho_1 \bar{v}_1 + \rho_2 \bar{v}_2 \end{aligned} \quad (2.2.40 - 2.2.41)$$

The species velocity is written in terms of the mass average velocity and the diffusion velocity as:

$$\bar{v}_i = \bar{v}_\gamma + \bar{u}_i \quad (2.2.42)$$

and the continuity equation becomes:

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \bar{v}_\gamma) = -\nabla \cdot (\rho_i \bar{u}_i) \quad , \quad i = 1, 2, 3, \dots \quad (2.2.43)$$

The diffusion flux may be written in terms of a diffusion coefficient as:

$$\rho_i \bar{u}_i = -\rho_\gamma \mathcal{D} \nabla \left(\frac{\rho_i}{\rho_\gamma} \right) \quad (2.2.44)$$

and the continuity equation may be written as:

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \bar{v}_\gamma) = \nabla \cdot \left\{ \rho_\gamma \mathcal{D} \nabla \left(\frac{\rho_i}{\rho_\gamma} \right) \right\} \quad , \quad i = 1, 2, 3, \dots \quad (2.2.45)$$

It is possible that we can neglect the change in gas density with time, or at least the change in the density of the inert component, and only consider the continuity equation for the vapor component of the gas phase (component 1):

$$\frac{\partial \rho_1}{\partial t} + \nabla \cdot (\rho_1 \bar{v}_\gamma) = \nabla \cdot \left\{ \rho_\gamma \mathcal{D} \nabla \left(\frac{\rho_1}{\rho_\gamma} \right) \right\} \quad (2.2.46)$$

and if we have no gas phase convection, with the gas phase stagnant in the pore spaces, the continuity equation becomes:

$$\frac{\partial \rho_1}{\partial t} = \nabla \cdot \left\{ \rho_\gamma \mathcal{D} \nabla \left(\frac{\rho_1}{\rho_\gamma} \right) \right\} \quad (2.2.47)$$

The thermal energy equation is given as:

$$\rho_\gamma (c_p)_\gamma \left(\frac{\partial T_\gamma}{\partial t} + \bar{v}_\gamma \cdot \nabla T_\gamma \right) = k_\gamma \nabla^2 T - \nabla \cdot \left(\sum_{i=1}^{i=N} \rho_i \bar{u}_i \bar{h}_i \right) \quad (2.2.48)$$

where $(c_p)_\gamma = \sum_{i=1}^{i=N} \frac{\rho_i}{\rho_\gamma} (\bar{c}_p)_i$,

and the partial mass heat capacities and enthalpies $(\bar{c}_p)_i$, $\bar{h}_i$ are again given by the partial molar enthalpy and the partial molar heat capacity divided by the molecular weight of that component.

Boundary Conditions

Whitaker next derives the boundary conditions for each phase interface. This section of the original derivation must be extensively modified since we no longer have a rigid solid phase with zero velocity. We will no longer have a simple set of boundary conditions for the solid-liquid and solid-vapor interface. The conventions and nomenclature for the phase interface boundary conditions are given below in Figure 3, which follows Whitaker's approach as closely as possible.

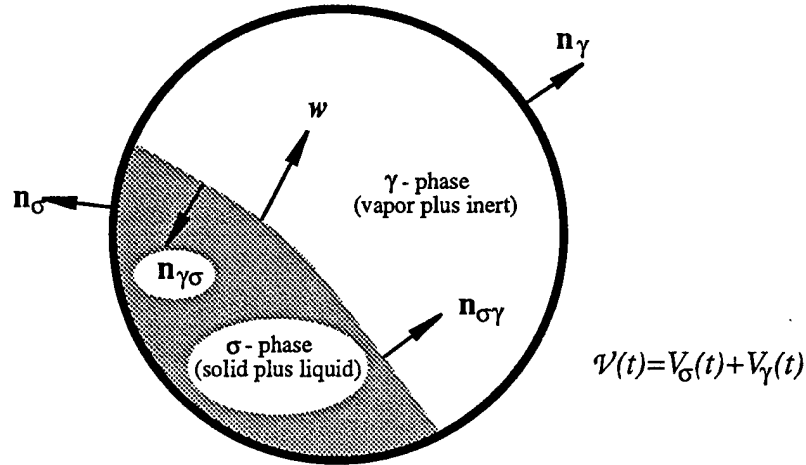


Figure 3. Material volume containing a phase interface, with velocities and unit normals indicated. Only two phases (solid and gas) shown.

Liquid-Gas Boundary Conditions

Whitaker gives the appropriate boundary conditions for the liquid-gas interface as:

$$\rho_\beta h_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \rho_\gamma h_\gamma (\bar{v}_\gamma - \bar{w}) \cdot \bar{n}_{\gamma\beta} = - \left\{ \bar{q}_\beta \cdot \bar{n}_{\beta\gamma} + \left[\bar{q}_\gamma + \sum_{i=1}^{i=N} \rho_i \bar{u}_i \bar{h}_i \right] \cdot \bar{n}_{\gamma\beta} \right\} \quad (2.2.49)$$

$$\rho_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \rho_\gamma (\bar{v}_\gamma - \bar{w}) \cdot \bar{n}_{\gamma\beta} = 0 \quad (2.2.50)$$

continuous tangent components to the phase interface λ :

$$\bar{v}_\beta \cdot \bar{\lambda}_{\beta\gamma} = \bar{v}_\gamma \cdot \bar{\lambda}_{\gamma\beta} \quad (2.2.51)$$

species jump condition given by:

$$\begin{aligned} \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} + \rho_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} &= 0, \quad i=1 \\ \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} &= 0, \quad i=2, 3, \dots \end{aligned} \quad (2.2.52 - 2.2.53)$$

Solid-Liquid Boundary Conditions

The boundary conditions for the solid-liquid interface are identical except that the phase interface velocity is given by w_2 .

$$\rho_\sigma h_\sigma (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} = - \left\{ \bar{q}_\beta \cdot \bar{n}_{\beta\sigma} + \left[\bar{q}_\sigma + \sum_{j=1}^{j=N} \rho_j \bar{u}_j \bar{h}_j \right] \cdot \bar{n}_{\sigma\beta} \right\} \quad (2.2.54)$$

$$\rho_\sigma (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} = 0 \quad (2.2.55)$$

continuous tangent components to the phase interface λ :

$$\bar{v}_\sigma \cdot \bar{\lambda}_{\sigma\beta} = \bar{v}_\beta \cdot \bar{\lambda}_{\beta\sigma} \quad (2.2.56)$$

species jump condition given by:

$$\begin{aligned} \rho_j (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} + \rho_\sigma (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} &= 0, \quad j=1 \\ \rho_j (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} &= 0, \quad j=2, 3, \dots \end{aligned} \quad (2.2.58)$$

Solid-Gas Boundary Conditions

The boundary conditions for the solid-liquid interface are modified even more because we have a phase interface between two multi-component phases. The phase interface velocity is given by w_1 .

$$\begin{aligned} &\rho_\sigma h_\sigma (\bar{v}_\sigma - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \rho_\gamma h_\gamma (\bar{v}_\gamma - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} \\ &= - \left\{ \left[\bar{q}_\sigma + \sum_{j=1}^{j=N} \rho_j \bar{u}_j \bar{h}_j \right] \cdot \bar{n}_{\sigma\gamma} + \left[\bar{q}_\gamma + \sum_{i=1}^{i=N} \rho_i \bar{u}_i \bar{h}_i \right] \cdot \bar{n}_{\gamma\sigma} \right\} \end{aligned} \quad (2.2.59)$$

$$\rho_{\sigma}(\bar{v}_{\sigma} - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \rho_{\gamma}(\bar{v}_{\gamma} - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} = 0 \quad (2.2.60)$$

continuous tangent components to the phase interface λ :

$$\bar{v}_{\sigma} \cdot \bar{\lambda}_{\sigma\gamma} = \bar{v}_{\gamma} \cdot \bar{\lambda}_{\gamma\sigma} \quad (2.2.61)$$

species jump condition given by:

$$\rho_j(\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \rho_i(\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} = 0, \quad i=1, j=1$$

$$\rho_j(\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} = 0, \quad j=2,3, \dots$$

$$\rho_i(\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} = 0, \quad i=2,3, \dots$$

Volume Averaged Equations

Whitaker uses the volume-averaging approach outlined by Slattery [28] so that many of the complicated phenomena going on due to the geometry of the porous material are simplified. He defines three averages:

Spatial Average: Average of some function everywhere in the volume.

$$\langle \psi \rangle = \frac{1}{\mathcal{V}} \int_{\mathcal{V}} \psi dV \quad (2.65)$$

Phase Average: Average of some quantity associated solely with each phase.

$$\langle T_{\sigma} \rangle = \frac{1}{\mathcal{V}} \int_{\mathcal{V}} T_{\sigma} dV = \frac{1}{\mathcal{V}} \int_{\mathcal{V}_{\sigma}} T_{\sigma} dV \quad (2.2.66)$$

Intrinsic Phase Average:

$$\langle T_{\sigma} \rangle^{\sigma} = \frac{1}{V_{\sigma}} \int_{\mathcal{V}} T_{\sigma} dV = \frac{1}{V_{\sigma}} \int_{\mathcal{V}_{\sigma}} T_{\sigma} dV \quad (2.2.67)$$

We can also define volume fractions for the three phases as:

$$\varepsilon_{\sigma}(t) = \frac{V_{\sigma}(t)}{\mathcal{V}}, \quad \varepsilon_{\beta}(t) = \frac{V_{\beta}(t)}{\mathcal{V}}, \quad \varepsilon_{\gamma}(t) = \frac{V_{\gamma}(t)}{\mathcal{V}} \quad (2.2.68)$$

We now have the volume and volume fraction of the solid changing with time, which is not the case with Whitaker's original derivation.

We are going to say that the total volume is conserved, or that:

$$\mathcal{V} = V_{\sigma}(t) + V_{\beta}(t) + V_{\gamma}(t) \quad (2.2.69)$$

The volume fractions for the three phases are related by:

$$\varepsilon_{\sigma}(t) + \varepsilon_{\beta}(t) + \varepsilon_{\gamma}(t) = 1 \quad (2.2.70)$$

and the phase average and the intrinsic phase averages are related as:

$$\varepsilon_\sigma \langle T_\sigma \rangle^\sigma = \langle T_\sigma \rangle \quad (2.2.71)$$

Volume Average for Liquid β Phase

We will first look at the volume average for the β phase. It will be complicated because of the three different phase interface velocities which we must now include in the analysis, whereas Whitaker only had to account for the single liquid-gas interface velocity.

The continuity equation for the liquid phase is:

$$\frac{\partial \rho_\beta}{\partial t} + \nabla \cdot (\rho_\beta \bar{v}_\beta) = 0 \quad (2.2.72)$$

We will integrate over the time-dependent liquid volume within the averaging volume, and divide by the averaging volume to obtain:

$$\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \left(\frac{\partial \rho_\beta}{\partial t} \right) dV + \frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \nabla \cdot (\rho_\beta \bar{v}_\beta) dV = 0 \quad (2.2.73)$$

We may take the first term:

$$\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \left(\frac{\partial \rho_\beta}{\partial t} \right) dV \quad (2.2.74)$$

and apply the general transport theorem [29]:

$$\frac{d}{dt} \int_{V(s)} \psi dV = \int_{V(s)} \frac{\partial \psi}{\partial t} dV + \int_{S(s)} \psi \bar{v}_{(s)} \cdot \bar{n} dS \quad (2.2.75)$$

$$\text{We note that } \Psi = \frac{\partial \rho_\beta}{\partial t}, \quad (2.2.76)$$

and using the modified general transport theorem we obtain:

$$\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \left(\frac{\partial \rho_\beta}{\partial t} \right) dV = \frac{d}{dt} \left[\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \rho_\beta dV \right] - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\beta \bar{w} \cdot \bar{n}_{\beta\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_\beta \bar{w}_2 \cdot \bar{n}_{\beta\sigma} dA \quad (2.2.77)$$

For the other term:

$$\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \nabla \cdot (\rho_\beta \bar{v}_\beta) dV, \quad (2.2.78)$$

we may use the volume averaging theorem:

$$\langle \nabla \psi_\beta \rangle = \nabla \langle \psi_\beta \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \psi_\beta \bar{n}_{\beta\sigma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \psi_\beta \bar{n}_{\beta\gamma} dA \quad (2.2.79)$$

to rewrite the term as:

$$\begin{aligned} \frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \nabla \cdot (\rho_{\beta} \bar{v}_{\beta}) dV &= \langle \nabla \cdot (\rho_{\beta} \bar{v}_{\beta}) \rangle = \\ \nabla \cdot \langle \rho_{\beta} \bar{v}_{\beta} \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma(t)}} \rho_{\beta} \bar{v}_{\beta} \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma(t)}} \rho_{\beta} \bar{v}_{\beta} \cdot \bar{n}_{\beta\sigma} dA \end{aligned} \quad (2.2.80)$$

Whitaker says that we may also rewrite the time derivative term as:

$$\frac{d}{dt} \left[\frac{1}{\mathcal{V}} \int_{V_{\beta(t)}} \rho_{\beta} dV \right] = \frac{d}{dt} \langle \rho_{\beta} \rangle = \frac{\partial}{\partial t} \langle \rho_{\beta} \rangle \quad (2.2.81)$$

This allows us to rewrite the continuity equation for the liquid phase as:

$$\frac{\partial}{\partial t} \langle \rho_{\beta} \rangle + \nabla \cdot \langle \rho_{\beta} \bar{v}_{\beta} \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA = 0 \quad (2.2.82)$$

We may assume that the density in the liquid is constant, so that:

$$\begin{aligned} \langle \rho_{\beta} \bar{v}_{\beta} \rangle &= \rho_{\beta} \langle \bar{v}_{\beta} \rangle \\ \langle \rho_{\beta} \rangle &= \varepsilon_{\beta} \rho_{\beta} \end{aligned} \quad (2.2.83 - 2.2.84)$$

The liquid velocity vector may be used to calculate volumetric flow rates. Whitaker gives the example of the flow rate of the liquid phase past a surface area as:

$$Q_{\beta} = \int_A \langle \bar{v}_{\beta} \rangle \cdot \bar{n} dA \quad (2.2.85)$$

The two constant-density liquid relations given above allow the liquid phase continuity equation to be rewritten as:

$$\frac{\partial \varepsilon_{\beta}}{\partial t} + \nabla \cdot \langle \bar{v}_{\beta} \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA = 0 \quad (2.2.86)$$

The thermal energy equation for the liquid phase was given previously as:

$$\rho_{\beta} \left(\frac{\partial h_{\beta}}{\partial t} + \bar{v}_{\beta} \cdot \nabla h_{\beta} \right) = -\nabla \cdot \bar{q}_{\beta} \quad (2.2.87)$$

Whitaker notes that this may be rewritten by adding the term $h_{\beta} \left[\frac{\partial \rho_{\beta}}{\partial t} + \nabla \cdot (\rho_{\beta} \bar{v}_{\beta}) \right]$ to the left-hand side to get:

$$\frac{\partial}{\partial t} (\rho_{\beta} h_{\beta}) + \nabla \cdot (\rho_{\beta} h_{\beta} \bar{v}_{\beta}) = -\nabla \cdot \bar{q}_{\beta} \quad (2.2.88)$$

We may follow the same procedure used previously to obtain a volume averaged form of the thermal energy equation where we use the general transport theorem for the first term and the averaging theorem for the second and third terms to get:

$$\begin{aligned} \frac{\partial}{\partial t} \langle \rho_\beta h_\beta \rangle + \nabla \cdot \langle \rho_\beta h_\beta \bar{v}_\beta \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA \\ = -\nabla \cdot \langle \bar{q}_\beta \rangle + \langle \Phi_\beta \rangle - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \bar{q}_\beta \cdot \bar{n}_{\beta\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \bar{q}_\beta \cdot \bar{n}_{\beta\sigma} dA \end{aligned} \quad (2.2.89)$$

All we did here was add an additional term to Whitaker's equations due to the solid-liquid interface velocity.

Whitaker uses the relation for the enthalpy of the liquid phase:

$$h_\beta = h_\beta^\circ + (c_p)_\beta (T_\beta - T_\beta^\circ) \quad (2.2.90)$$

and goes through several steps, accounting for the deviation and dispersion terms from the average properties (marked with a tilde), to write an expression for the two terms:

$$\begin{aligned} \frac{\partial}{\partial t} \langle \rho_\beta h_\beta \rangle + \nabla \cdot \langle \rho_\beta h_\beta \bar{v}_\beta \rangle = \varepsilon_\beta \rho_\beta (c_p)_\beta \frac{\partial \langle T_\beta \rangle^\beta}{\partial t} \\ + \rho_\beta \left[h_\beta^\circ + (c_p)_\beta \left(\langle T_\beta \rangle^\beta - T_\beta^\circ \right) \right] \left(\frac{\partial \varepsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle \right) \\ + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle \cdot \nabla \langle T_\beta \rangle^\beta + \rho_\beta (c_p)_\beta \nabla \cdot \langle \tilde{T}_\beta \tilde{\bar{v}}_\beta \rangle \end{aligned} \quad (2.2.91)$$

We can recognize that the term: $\left(\frac{\partial \varepsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle \right)$

is contained in the liquid phase continuity equation:

$$\left(\frac{\partial \varepsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA = 0 \quad (2.2.92)$$

so that:

$$\frac{\partial \varepsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle = - \left\{ \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA \right\} \quad (2.93)$$

The expression for the two terms $\frac{\partial}{\partial t} \langle \rho_\beta h_\beta \rangle + \nabla \cdot \langle \rho_\beta h_\beta \bar{v}_\beta \rangle$ may be written as:

$$\begin{aligned} \frac{\partial}{\partial t} \langle \rho_\beta h_\beta \rangle + \nabla \cdot \langle \rho_\beta h_\beta \bar{v}_\beta \rangle = \varepsilon_\beta \rho_\beta (c_p)_\beta \frac{\partial \langle T_\beta \rangle^\beta}{\partial t} + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle \cdot \nabla \langle T_\beta \rangle^\beta + \rho_\beta (c_p)_\beta \nabla \cdot \langle \tilde{T}_\beta \tilde{\bar{v}}_\beta \rangle \\ - \left[h_\beta^\circ + (c_p)_\beta \left(\langle T_\beta \rangle^\beta - T_\beta^\circ \right) \right] \left\{ \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA \right\} \end{aligned} \quad (2.2.94)$$

We may now substitute back into the thermal energy equation for the liquid phase:

$$\begin{aligned}
& \epsilon_{\beta} \rho_{\beta} (c_p)_{\beta} \frac{\partial \langle T_{\beta} \rangle^{\beta}}{\partial t} + \rho_{\beta} (c_p)_{\beta} \langle \tilde{v}_{\beta} \rangle \cdot \nabla \langle T_{\beta} \rangle^{\beta} + \rho_{\beta} (c_p)_{\beta} \nabla \cdot \langle \tilde{T}_{\beta} \tilde{v}_{\beta} \rangle \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (c_p)_{\beta} (T_{\beta} - T_{\beta}^{\circ}) (\tilde{v}_{\beta} - \tilde{w}) \cdot \tilde{n}_{\beta\gamma} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_{\beta} (c_p)_{\beta} (T_{\beta} - T_{\beta}^{\circ}) (\tilde{v}_{\beta} - \tilde{w}_2) \cdot \tilde{n}_{\beta\sigma} dA \\
& - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (c_p)_{\beta} \left(\langle T_{\beta} \rangle^{\beta} - T_{\beta}^{\circ} \right) (\tilde{v}_{\beta} - \tilde{w}) \cdot \tilde{n}_{\beta\gamma} dA \\
& - \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_{\beta} (c_p)_{\beta} \left(\langle T_{\beta} \rangle^{\beta} - T_{\beta}^{\circ} \right) (\tilde{v}_{\beta} - \tilde{w}_2) \cdot \tilde{n}_{\beta\sigma} dA \\
& = -\nabla \cdot \langle \tilde{q}_{\beta} \rangle + \langle \Phi_{\beta} \rangle - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \tilde{q}_{\beta} \cdot \tilde{n}_{\beta\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \tilde{q}_{\beta} \cdot \tilde{n}_{\beta\sigma} dA
\end{aligned} \tag{2.2.95}$$

Whitaker now uses Gray's definitions of the point functions for the phase properties [30] as:

$$T_{\beta} = \langle T_{\beta} \rangle^{\beta} + \tilde{T}_{\beta} \tag{2.2.96}$$

This allows the liquid phase thermal energy equation to be written as:

$$\begin{aligned}
& \epsilon_{\beta} \rho_{\beta} (c_p)_{\beta} \frac{\partial \langle T_{\beta} \rangle^{\beta}}{\partial t} + \rho_{\beta} (c_p)_{\beta} \langle \tilde{v}_{\beta} \rangle \cdot \nabla \langle T_{\beta} \rangle^{\beta} + \rho_{\beta} (c_p)_{\beta} \nabla \cdot \langle \tilde{T}_{\beta} \tilde{v}_{\beta} \rangle \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (c_p)_{\beta} \tilde{T}_{\beta} (\tilde{v}_{\beta} - \tilde{w}) \cdot \tilde{n}_{\beta\gamma} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_{\beta} (c_p)_{\beta} \tilde{T}_{\beta} (\tilde{v}_{\beta} - \tilde{w}_2) \cdot \tilde{n}_{\beta\sigma} dA \\
& = -\nabla \cdot \langle \tilde{q}_{\beta} \rangle + \langle \Phi_{\beta} \rangle - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \tilde{q}_{\beta} \cdot \tilde{n}_{\beta\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \tilde{q}_{\beta} \cdot \tilde{n}_{\beta\sigma} dA
\end{aligned} \tag{2.2.97}$$

Whitaker now rewrites the heat flux term $-\nabla \cdot \langle \tilde{q}_{\beta} \rangle$ using Fourier's law

($\tilde{q}_{\beta} = -k_{\beta} \nabla T_{\beta}$) and the averaging theorem to write:

$$\langle \tilde{q}_{\beta} \rangle = -k_{\beta} \langle \nabla T_{\beta} \rangle = -k_{\beta} \left[\nabla \langle T_{\beta} \rangle + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} T_{\beta} \tilde{n}_{\beta\sigma} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} T_{\beta} \tilde{n}_{\beta\gamma} dA \right] \tag{2.2.98}$$

We may also substitute the intrinsic phase average temperature $\varepsilon_\beta \langle T_\beta \rangle^\beta$ for the phase average temperature $\langle T_\beta \rangle$ to obtain an expression for the heat flux vector:

$$\langle \bar{q}_\beta \rangle = -k_\beta \langle \nabla T_\beta \rangle = -k_\beta \left[\nabla \left(\varepsilon_\beta \langle T_\beta \rangle^\beta \right) + \frac{1}{V} \int_{A_{\beta\sigma}} T_\beta \bar{n}_{\beta\sigma} dA + \frac{1}{V} \int_{A_{\beta\gamma}} T_\beta \bar{n}_{\beta\gamma} dA \right] \quad (2.2.99)$$

The thermal energy equation for the liquid phase may now be written as:

$$\begin{aligned} & \varepsilon_\beta \rho_\beta (c_p)_\beta \frac{\partial \langle T_\beta \rangle^\beta}{\partial t} + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle \cdot \nabla \langle T_\beta \rangle^\beta + \rho_\beta (c_p)_\beta \nabla \cdot \langle \tilde{T}_\beta \tilde{v}_\beta \rangle \\ & + \frac{1}{V} \int_{A_{\beta\gamma}} \rho_\beta (c_p)_\beta \tilde{T}_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{V} \int_{A_{\beta\sigma}} \rho_\beta (c_p)_\beta \tilde{T}_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA \\ & = \nabla \cdot \left\{ k_\beta \left[\nabla \left(\varepsilon_\beta \langle T_\beta \rangle^\beta \right) + \frac{1}{V} \int_{A_{\beta\sigma}} T_\beta \bar{n}_{\beta\sigma} dA + \frac{1}{V} \int_{A_{\beta\gamma}} T_\beta \bar{n}_{\beta\gamma} dA \right] \right\} \\ & - \frac{1}{V} \int_{A_{\beta\gamma}} \bar{q}_\beta \cdot \bar{n}_{\beta\gamma} dA - \frac{1}{V} \int_{A_{\beta\sigma}} \bar{q}_\beta \cdot \bar{n}_{\beta\sigma} dA \end{aligned} \quad (2.2.100)$$

Volume Average for Gas γ Phase

The gas phase continuity equation is identical, up to a point, to the continuity equations for the solid and liquid phases:

$$\frac{\partial}{\partial t} \langle \rho_\gamma \rangle + \nabla \cdot \langle \rho_\gamma \bar{v}_\gamma \rangle + \frac{1}{V} \int_{A_{\gamma\beta}} \rho_\gamma (\bar{v}_\gamma - \bar{w}) \cdot \bar{n}_{\gamma\beta} dA + \frac{1}{V} \int_{A_{\gamma\sigma}} \rho_\gamma (\bar{v}_\gamma - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} dA = 0 \quad (2.2.101)$$

For the liquid and solid phases we could assume constant density, and simplify the equation further; we can't do this for the gas phase since the density depends on the temperature and the pressure.

Whitaker uses Gray's expressions for the point functions again, along with the definition of the intrinsic phase average to rewrite the gas phase continuity equation as:

$$\begin{aligned} & \frac{\partial}{\partial t} \left(\varepsilon_\gamma \langle \rho_\gamma \rangle^\gamma \right) + \nabla \cdot \left(\langle \rho_\gamma \rangle^\gamma \langle \bar{v}_\gamma \rangle \right) + \nabla \cdot \langle \tilde{\rho}_\gamma \tilde{v}_\gamma \rangle \\ & + \frac{1}{V} \int_{A_{\gamma\beta}} \rho_\gamma (\bar{v}_\gamma - \bar{w}) \cdot \bar{n}_{\gamma\beta} dA + \frac{1}{V} \int_{A_{\gamma\sigma}} \rho_\gamma (\bar{v}_\gamma - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} dA = 0 \end{aligned} \quad (2.2.102)$$

Whitaker then assumes we can neglect terms with products of the dispersion or deviations, so we can drop that term to write the gas phase continuity equation as:

$$\begin{aligned} \frac{\partial}{\partial t} \left(\epsilon_Y \langle \rho_Y \rangle^Y \right) + \nabla \cdot \left(\langle \rho_Y \rangle^Y \langle \bar{v}_Y \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{Y\beta}} \rho_Y (\bar{v}_Y - \bar{w}) \cdot \bar{n}_{Y\beta} dA \\ + \frac{1}{\mathcal{V}} \int_{A_{Y\sigma}} \rho_Y (\bar{v}_Y - \bar{w}_1) \cdot \bar{n}_{Y\sigma} dA = 0 \end{aligned} \quad (2.2.103)$$

Since the gas is a multi-component mixture we must also go through the species continuity equation:

$$\frac{\partial}{\partial t} \langle \rho_i \rangle + \nabla \cdot \langle \rho_i \bar{v}_i \rangle + \frac{1}{\mathcal{V}} \int_{A_{Y\beta}} \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{Y\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{Y\sigma}} \rho_i (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{Y\sigma} dA = 0 \quad i = 1, 2, \dots \quad (2.2.104)$$

Whitaker's derivations may be used directly, and we can write the final form of the gas phase species continuity equation as:

$$\frac{\partial}{\partial t} \left(\epsilon_Y \langle \rho_i \rangle^Y \right) + \nabla \cdot \left(\langle \rho_i \rangle^Y \langle \bar{v}_i \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{Y\beta}} \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{Y\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{Y\sigma}} \rho_i (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{Y\sigma} dA \quad (2.2.105)$$

If we neglect the deviation terms, and if we also consider only the species continuity equation for the vapor component (component 1), we can rewrite the gas phase continuity equation as:

$$\frac{\partial}{\partial t} \left(\epsilon_Y \langle \rho_1 \rangle^Y \right) + \nabla \cdot \left(\langle \rho_1 \rangle^Y \langle \bar{v}_Y \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{Y\beta}} \rho_1 (\bar{v}_1 - \bar{w}) \cdot \bar{n}_{Y\beta} dA = \nabla \cdot \left\{ \langle \rho_Y \rangle^Y \mathcal{D} \nabla \left(\frac{\rho_1}{\langle \rho_Y \rangle^Y} \right) \right\} \quad (2.106)$$

The corresponding thermal energy equation for the gas phase may also be written as:

$$\begin{aligned}
& \left\{ \sum_{i=1}^{i=N} \langle \rho_i \rangle (c_p)_i \right\} \frac{\partial \langle T_\gamma \rangle^\gamma}{\partial t} + \left\{ \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right\} \cdot \nabla \langle T_\gamma \rangle^\gamma \\
& + \frac{1}{\mathcal{V}} \int_{A_{\gamma\beta}} \sum_{i=1}^{i=N} \rho_i (c_p)_i \tilde{T}_\gamma (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} \sum_{i=1}^{i=N} \rho_i (c_p)_i \tilde{T}_\gamma (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} dA \\
& + \frac{\partial}{\partial t} \sum_{i=1}^{i=N} (c_p)_i \langle \tilde{\rho}_i \tilde{T}_\gamma \rangle + \nabla \cdot \sum_{i=1}^{i=N} (c_p)_i \langle \tilde{\rho}_i \bar{v}_i \tilde{T}_\gamma \rangle \\
& = \nabla \cdot \left\{ k_\gamma \left[\nabla \left(\epsilon_\gamma \langle T_\gamma \rangle^\gamma \right) + \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} T_\gamma \bar{n}_{\gamma\sigma} dA + \frac{1}{\mathcal{V}} \int_{A_{\gamma\beta}} T_\gamma \bar{n}_{\gamma\beta} dA \right] \right\} \\
& - \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} \bar{q}_\gamma \cdot \bar{n}_{\gamma\sigma} dA - \frac{1}{\mathcal{V}} \int_{A_{\gamma\beta}} \bar{q}_\gamma \cdot \bar{n}_{\gamma\beta} dA
\end{aligned} \tag{2.2.107}$$

Volume Average for Solid σ Phase

The volume averaging procedure for the liquid phase was made general enough so that the same equations also apply for the solid phase. The only differences are that now the phase interface velocities are w_2 for the solid-liquid interface, and w_1 for the solid-gas interface. We also need to account for the species continuity equations. Since the two components (liquid and the solid) are assumed to have a constant density, we will not run into the same complications we did with the gas phase continuity equation. The appropriate subscripts for the solid phase also need to be added to the equations.

We cannot assume that the solid phase density is constant, since it is a mixture of the solid and the liquid component. However, it will be less complicated than the gas phase density since we can assume that each component's density is constant.

The solid phase continuity equation is:

$$\frac{\partial}{\partial t} \langle \rho_\sigma \rangle + \nabla \cdot \langle \rho_\sigma \bar{v}_\sigma \rangle + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} (\bar{v}_\sigma - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA = 0 \tag{2.2.108}$$

and the species continuity equation is:

$$\frac{\partial}{\partial t} \langle \rho_j \rangle + \nabla \cdot \langle \rho_j \bar{v}_j \rangle + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA = 0 \quad j=1, 2, \dots \tag{2.2.109}$$

We can follow the same derivation used for the gas phase to write the gas phase continuity equation as:

$$\begin{aligned} \frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_\sigma \rangle^\sigma) + \nabla \cdot (\langle \rho_\sigma \rangle^\sigma \langle \bar{v}_\sigma \rangle) + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_\sigma (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA \\ + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \rho_\sigma (\bar{v}_\sigma - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA = 0 \end{aligned} \quad (2.2.110)$$

and the final form of the solid phase species continuity equation is:

$$\begin{aligned} \frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_j \rangle^\sigma) + \nabla \cdot (\langle \rho_j \rangle^\sigma \langle \bar{v}_j \rangle) + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_j (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \rho_j (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA \\ = \nabla \cdot \left\{ \langle \rho_\sigma \rangle^\sigma \mathcal{D}_\sigma \left[\nabla \left(\frac{\rho_j}{\langle \rho_\sigma \rangle^\sigma} \right) \right] - \langle \tilde{\rho}_j \tilde{v}_j \rangle \right\} \quad j = 1, 2, \dots \end{aligned} \quad (2.2.111)$$

If we want to just follow the single liquid component (component 1) and write the continuity equation for that species, we may write:

$$\begin{aligned} \frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_1 \rangle^\sigma) + \nabla \cdot (\langle \rho_1 \rangle^\sigma \langle \bar{v}_1 \rangle) + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \rho_1 (\bar{v}_1 - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA \\ = \nabla \cdot \left\{ \langle \rho_\sigma \rangle^\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_1}{\langle \rho_\sigma \rangle^\sigma} \right) \right\} \end{aligned} \quad (2.2.112)$$

Later on, we may also want to assume that the solid velocity is zero, so we could rewrite the solid phase continuity equation as:

$$\begin{aligned} \frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_1 \rangle^\sigma) + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \rho_1 (\bar{v}_1 - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA \\ = \nabla \cdot \left\{ \langle \rho_\sigma \rangle^\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_1}{\langle \rho_\sigma \rangle^\sigma} \right) \right\} \end{aligned} \quad (2.2.113)$$

The corresponding thermal energy equation for the solid phase may also be written as:

$$\begin{aligned}
& \left\{ \sum_{j=1}^{j=N} \langle \rho_j \rangle (c_p)_j \right\} \frac{\partial \langle T_\sigma \rangle^\sigma}{\partial t} + \left\{ \sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \tilde{v}_j \rangle \right\} \cdot \nabla \langle T_\sigma \rangle^\sigma \\
& + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \sum_{j=1}^{j=N} \rho_j (c_p)_j \tilde{T}_\sigma (\tilde{v}_j - \tilde{w}_2) \cdot \tilde{n}_{\sigma\beta} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \sum_{j=1}^{j=N} \rho_j (c_p)_j \tilde{T}_\sigma (\tilde{v}_j - \tilde{w}_1) \cdot \tilde{n}_{\sigma\gamma} dA \\
& + \frac{\partial}{\partial t} \sum_{j=1}^{j=N} (c_p)_j \langle \tilde{\rho}_j \tilde{T}_\sigma \rangle + \nabla \cdot \sum_{j=1}^{j=N} (c_p)_j \langle \tilde{\rho}_j \tilde{v}_j \tilde{T}_\sigma \rangle \\
& = \nabla \cdot \left\{ k_\sigma \left[\nabla (\epsilon_\sigma \langle T_\sigma \rangle^\sigma) + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} T_\sigma \tilde{n}_{\sigma\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} T_\sigma \tilde{n}_{\sigma\beta} dA \right] \right\} \\
& - \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \tilde{q}_\sigma \cdot \tilde{n}_{\sigma\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \tilde{q}_\sigma \cdot \tilde{n}_{\sigma\beta} dA
\end{aligned} \tag{2.2.114}$$

This completes the continuity and thermal energy volume averaged equations for all three phases. The various continuity equations are given in several forms, depending on whether we want to include the solid velocity, and whether we just want to use the continuity equation for the liquid component only, since it is the only species which is transferring between the three phases.

2.3 Total Thermal Energy Equation

The three phases are assumed to be in local thermal equilibrium so that:

$$\langle T_\sigma \rangle^\sigma = \langle T_\beta \rangle^\beta = \langle T_\gamma \rangle^\gamma \tag{2.3.1}$$

$$\langle T \rangle \equiv \epsilon_\sigma \langle T_\sigma \rangle^\sigma + \epsilon_\beta \langle T_\beta \rangle^\beta + \epsilon_\gamma \langle T_\gamma \rangle^\gamma = \langle T_\sigma \rangle^\sigma = \langle T_\beta \rangle^\beta = \langle T_\gamma \rangle^\gamma \tag{2.3.2}$$

We can now write the total thermal energy equation by adding together the thermal energy equations for each phase, and using the local thermal equilibrium relations given above. This equation is identical to Whitaker's, except for the addition of extra terms due to the solid-gas and solid-liquid phase interface velocities, which are no longer zero. The equation is also written so positive flux terms imply liquid evaporating into the gas phase, rather than condensing.

$$\begin{aligned}
& \left[\varepsilon_{\sigma} \left\{ \sum_{j=1}^{j=N} \langle \rho_j \rangle (c_p)_j \right\} + \varepsilon_{\beta} \rho_{\beta} (c_p)_{\beta} + \varepsilon_{\gamma} \left\{ \sum_{i=1}^{i=N} \langle \rho_i \rangle (c_p)_i \right\} \right] \frac{\partial \langle T \rangle}{\partial t} \\
& + \left[\sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \bar{v}_j \rangle + \rho_{\beta} (c_p)_{\beta} \langle \bar{v}_{\beta} \rangle + \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right] \cdot \nabla \langle T \rangle \\
& + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \sum_{j=1}^{j=N} \rho_j (c_p)_j \tilde{T}_{\sigma} (\bar{v}_{\sigma} - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} \sum_{i=1}^{i=N} \rho_i (c_p)_i \tilde{T}_{\gamma} (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \sum_{j=1}^{j=N} \rho_j (c_p)_j \tilde{T}_{\sigma} (\bar{v}_{\sigma} - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA + \frac{1}{\mathcal{V}} \int_{A_{\beta\sigma}} \rho_{\beta} (c_p)_{\beta} \tilde{T}_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (c_p)_{\beta} \tilde{T}_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\gamma\beta}} \sum_{i=1}^{i=N} \rho_i (c_p)_i \tilde{T}_{\gamma} (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} dA \\
& = \nabla \cdot \left\{ \begin{aligned} & \nabla \left[(k_{\sigma} \varepsilon_{\sigma} + k_{\beta} \varepsilon_{\beta} + k_{\gamma} \varepsilon_{\gamma}) \langle T \rangle \right] \\ & + (k_{\sigma} - k_{\beta}) \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} T_{\sigma} \bar{n}_{\sigma\beta} dA \\ & + (k_{\beta} - k_{\gamma}) \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} T_{\beta} \bar{n}_{\beta\gamma} dA \\ & + (k_{\sigma} - k_{\gamma}) \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} T_{\gamma} \bar{n}_{\sigma\gamma} dA \end{aligned} \right\} \\
& - \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \bar{q}_{\sigma} \cdot \bar{n}_{\sigma\beta} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \bar{q}_{\sigma} \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} \bar{q}_{\gamma} \cdot \bar{n}_{\sigma\gamma} dA \tag{2.3.3}
\end{aligned}$$

Whitaker defines a spatial average density:

$$\langle \rho \rangle = \varepsilon_{\sigma} \sum_{j=1}^{j=N} \langle \rho_j \rangle^{\sigma} + \varepsilon_{\beta} \langle \rho_{\beta} \rangle^{\beta} + \varepsilon_{\gamma} \sum_{i=1}^{i=N} \langle \rho_i \rangle^{\gamma} \tag{2.3.4}$$

and a mass fraction weighted average heat capacity by:

$$C_p = \frac{\varepsilon_{\sigma} \sum_{j=1}^{j=N} \langle \rho_j \rangle^{\sigma} (c_p)_j + \varepsilon_{\beta} \rho_{\beta} (c_p)_{\beta} + \varepsilon_{\gamma} \sum_{i=1}^{i=N} \langle \rho_i \rangle^{\gamma} (c_p)_i}{\langle \rho \rangle} \tag{2.3.5}$$

This allows the first term in the thermal energy equation to be written as:

$$\left[\varepsilon_{\sigma} \left\{ \sum_{j=1}^{j=N} \langle \rho_j \rangle (c_p)_j \right\} + \varepsilon_{\beta} \rho_{\beta} (c_p)_{\beta} + \varepsilon_{\gamma} \left\{ \sum_{i=1}^{i=N} \langle \rho_i \rangle (c_p)_i \right\} \right] \frac{\partial \langle T \rangle}{\partial t} = \langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} \tag{2.3.6}$$

We must now consider the interphase flux terms in the total thermal energy equation. In Whitaker's derivation, he only had one interphase flux term to consider, that of the exchange of mass between the liquid and the gas. We now have two more interphase fluxes to consider: that between the liquid and the solid, and that between the gas and the solid.

We first follow his derivation for the liquid-gas interface, and then apply it to the other two interfaces.

The jump boundary condition for the liquid-gas interface was shown previously to be:

$$\rho_\beta h_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \rho_\gamma h_\gamma (\bar{v}_\gamma - \bar{w}) \cdot \bar{n}_{\gamma\beta} = - \left\{ \bar{q}_\beta \cdot \bar{n}_{\beta\gamma} + \left[\bar{q}_\gamma + \sum_{i=1}^{i=N} \rho_i \bar{u}_i \bar{h}_i \right] \cdot \bar{n}_{\gamma\beta} \right\} \quad (2.3.7)$$

and this may be rewritten as:

$$\rho_\beta h_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \sum_{i=1}^{i=N} \rho_i \bar{h}_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} = -(\bar{q}_\beta - \bar{q}_\gamma) \cdot \bar{n}_{\beta\gamma} \quad (2.3.8)$$

The jump boundary condition for the solid-gas interface was shown previously to be:

$$\begin{aligned} & \rho_\sigma h_\sigma (\bar{v}_\sigma - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \rho_\gamma h_\gamma (\bar{v}_\gamma - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} \\ &= - \left\{ \left[\bar{q}_\sigma + \sum_{j=1}^{j=N} \rho_j \bar{u}_j \bar{h}_j \right] \cdot \bar{n}_{\sigma\gamma} + \left[\bar{q}_\gamma + \sum_{i=1}^{i=N} \rho_i \bar{u}_i \bar{h}_i \right] \cdot \bar{n}_{\gamma\sigma} \right\} \end{aligned} \quad (2.3.9)$$

and this may be rewritten as:

$$\sum_{j=1}^{j=N} \rho_j \bar{h}_j (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \sum_{i=1}^{i=N} \rho_i \bar{h}_i (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} = -(\bar{q}_\sigma - \bar{q}_\gamma) \cdot \bar{n}_{\sigma\gamma} \quad (2.3.10)$$

The jump boundary condition for the solid-liquid interface was shown previously to be:

$$\rho_\sigma h_\sigma (\bar{v}_\sigma - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} = - \left\{ \bar{q}_\beta \cdot \bar{n}_{\beta\sigma} + \left[\bar{q}_\sigma + \sum_{j=1}^{j=N} \rho_j \bar{u}_j \bar{h}_j \right] \cdot \bar{n}_{\sigma\beta} \right\} \quad (2.3.11)$$

This may be rewritten as:

$$\sum_{j=1}^{j=N} \rho_j \bar{h}_j (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} = -(\bar{q}_\sigma - \bar{q}_\beta) \cdot \bar{n}_{\sigma\beta} \quad (2.3.12)$$

Using these results, we may write the interphase flux terms in the total thermal energy equation as:

$$\begin{aligned}
& -\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} (\bar{q}_\sigma - \bar{q}_\beta) \cdot \bar{n}_{\sigma\beta} dA - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} (\bar{q}_\beta - \bar{q}_\gamma) \cdot \bar{n}_{\beta\gamma} dA - \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} (\bar{q}_\sigma - \bar{q}_\gamma) \cdot \bar{n}_{\sigma\gamma} dA \\
& = +\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left[\sum_{j=1}^{j=N} \rho_j \bar{h}_j (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_\beta h_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} \right] dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \left[\rho_\beta h_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \sum_{i=1}^{i=N} \rho_i \bar{h}_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} \right] dA \\
& + \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \left[\sum_{j=1}^{j=N} \rho_j \bar{h}_j (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \sum_{i=1}^{i=N} \rho_i \bar{h}_i (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} \right] dA \tag{2.3.13}
\end{aligned}$$

The total thermal energy equation is now written as:

$$\begin{aligned}
& \langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + \left[\sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \bar{v}_j \rangle + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle + \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right] \cdot \nabla \langle T \rangle \\
& - \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left\{ \sum_{j=1}^{j=N} \rho_j \left[\bar{h}_j - (c_p)_j \tilde{T}_\sigma \right] (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} \right. \\
& \quad \left. + \rho_\beta \left[h_\beta - (c_p)_\beta \tilde{T}_\beta \right] (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} \right\} dA \\
& - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \left\{ \rho_\beta \left[h_\beta - (c_p)_\beta \tilde{T}_\beta \right] (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} \right. \\
& \quad \left. + \sum_{i=1}^{i=N} \rho_i \left[\bar{h}_i - (c_p)_i \tilde{T}_\gamma \right] (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} \right\} dA \\
& - \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \left\{ \sum_{j=1}^{j=N} \rho_j \left[\bar{h}_j - (c_p)_j \tilde{T}_\sigma \right] (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} \right. \\
& \quad \left. + \sum_{i=1}^{i=N} \rho_i \left[\bar{h}_i - (c_p)_i \tilde{T}_\gamma \right] (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} \right\} dA \\
& = \nabla \cdot \left\{ \begin{aligned} & \nabla \left[(k_\sigma \varepsilon_\sigma + k_\beta \varepsilon_\beta + k_\gamma \varepsilon_\gamma) \langle T \rangle \right] \\ & + (k_\sigma - k_\beta) \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} T_\sigma \bar{n}_{\sigma\beta} dA \\ & + (k_\beta - k_\gamma) \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} T_\beta \bar{n}_{\beta\gamma} dA \\ & + (k_\sigma - k_\gamma) \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} T_\gamma \bar{n}_{\sigma\gamma} dA \end{aligned} \right\} \quad (2.3.14)
\end{aligned}$$

Next, we can begin to express the phase interface velocities in terms of enthalpies of vaporization, sorption, and desorption.

The enthalpies for each phase were defined previously as:

$$\begin{aligned}\bar{h}_j &= h_j^\circ + (c_p)_j (T_\sigma - T_\sigma^\circ) \\ h_\beta &= h_\beta^\circ + (c_p)_\beta (T_\beta - T_\beta^\circ) \\ \bar{h}_i &= h_i^\circ + (c_p)_i (T_\gamma - T_\gamma^\circ)\end{aligned}\quad (2.3.15 - 2.3.17)$$

We also know that the intrinsic phase average temperatures, temperature dispersion, and overall average temperatures are related by:

$$\begin{aligned}\tilde{T}_\sigma &= \langle T_\sigma \rangle^\sigma - T_\sigma \\ \tilde{T}_\beta &= \langle T_\beta \rangle^\beta - T_\beta \\ \tilde{T}_\gamma &= \langle T_\gamma \rangle^\gamma - T_\gamma \\ \langle T_\sigma \rangle^\sigma &= \langle T_\beta \rangle^\beta = \langle T_\gamma \rangle^\gamma = \langle T \rangle\end{aligned}\quad (2.3.18 - 2.3.21)$$

We may use these relations to rewrite the integrands inside the volume integrals on the left hand side of the total thermal energy equation. Whitaker gives the result for the liquid-gas interface as:

$$\begin{aligned}& -\frac{1}{\eta'} \int_{A_{\beta\gamma}} \left\{ \rho_\beta \left[h_\beta - (c_p)_\beta \tilde{T}_\beta \right] (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \sum_{i=1}^{i=N} \rho_i \left[\bar{h}_i - (c_p)_i \tilde{T}_\gamma \right] (\bar{v}_i - \bar{w}) \cdot \bar{n}_{i\beta} \right\} dA \\ &= -\frac{1}{\eta'} \int_{A_{\beta\gamma}} \left\{ \left[h_\beta^\circ - (c_p)_\beta \left(\langle T_\beta \rangle^\beta - T_\beta^\circ \right) \right] \rho_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} \right. \\ & \quad \left. + \sum_{i=1}^{i=N} \left[h_i^\circ - (c_p)_i \left(\langle T_\gamma \rangle^\gamma - T_\gamma^\circ \right) \right] \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{i\beta} \right\} dA\end{aligned}\quad (2.3.22)$$

From the species jump conditions:

$$\begin{aligned}\rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{i\beta} + \rho_\beta (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} &= 0, \quad i=1 \\ \rho_i (\bar{v}_i - \bar{w}) \cdot \bar{n}_{i\beta} &= 0, \quad i=2, 3, \dots\end{aligned}\quad (2.3.23 - 2.3.24)$$

where the subscript 1 refers to the component (water) which is actually crossing the phase boundary as it goes from a liquid to a vapor.

From the species jump conditions we may also write:

$$\rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} = -\rho_\beta (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma}\quad (2.3.25)$$

We may rewrite the integral as:

$$\begin{aligned}
& -\frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \left\{ \left[h_{\beta}^{\circ} - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \rho_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} \right. \\
& \quad \left. + \left[h_{g1}^{\circ} - (c_p)_1 (\langle T \rangle - T_{\gamma}^{\circ}) \right] \rho_1 (\bar{v}_1 - \bar{w}) \cdot \bar{n}_{\gamma\beta} \right\} dA \\
& = -\frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \left\{ \left[h_{\beta}^{\circ} - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \rho_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} \right. \\
& \quad \left. - \left[h_{g1}^{\circ} - (c_p)_1 (\langle T \rangle - T_{\gamma}^{\circ}) \right] \rho_1 (\bar{v}_1 - \bar{w}) \cdot \bar{n}_{\beta\gamma} \right\} dA \\
& = \left\{ \left[h_{g1}^{\circ} - h_{\beta}^{\circ} + (c_p)_1 (\langle T \rangle - T_{\gamma}^{\circ}) \right] \right. \\
& \quad \left. - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right\} \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA \tag{2.3.26}
\end{aligned}$$

We may use the following definitions:

$$\Delta h_{vap} \text{ (at temperature } \langle T \rangle) = \left\{ \left[h_{g1}^{\circ} - h_{\beta}^{\circ} + (c_p)_1 (\langle T \rangle - T_{\gamma}^{\circ}) - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \right\} \tag{2.3.27}$$

$$\langle \dot{m}_{lv} \rangle = \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_{\beta} (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA \tag{2.3.28}$$

to rewrite the integral as:

$$-\frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \left\{ \rho_{\beta} \left[h_{\beta} - (c_p)_{\beta} \tilde{T}_{\beta} \right] (\bar{v}_{\beta} - \bar{w}) \cdot \bar{n}_{\beta\gamma} + \sum_{i=1}^N \rho_i \left[\bar{h}_i - (c_p)_i \tilde{T}_{\gamma} \right] (\bar{v}_i - \bar{w}) \cdot \bar{n}_{\gamma\beta} \right\} dA = \Delta h_{vap} \langle \dot{m}_{lv} \rangle \tag{2.3.29}$$

The corresponding terms for the phase interface between the solid and the liquid are identical, except that we no longer use the quantity Δh_{vap} , but instead use the differential enthalpy of sorption [31], which we will give the notation Q_i . The differential heat of sorption is the heat evolved when one gram of water is absorbed by an infinite mass of the solid, when that solid is at a particular equilibrated moisture content. This is very similar to the heat of solution or heat of mixing that occurs when two liquid components are mixed. For textile fibers there is a definite relationship between the equilibrium values of the differential heat of sorption and the water content of the fibers, and we can use those relationships in our thermodynamic equations which will be discussed in a later section.

The solid-liquid interface integral term is thus given as:

$$-\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left\{ \rho_{\beta} \left[h_{\beta} - (c_p)_{\beta} \tilde{T}_{\beta} \right] (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} + \sum_{j=1}^{j=N} \rho_j \left[\bar{h}_j - (c_p)_j \tilde{T}_{\sigma} \right] (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} \right\} dA \quad (2.3.30)$$

From the species jump conditions we may also write:

$$\rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} = -\rho_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} \quad (2.3.31)$$

We may rewrite the integral as:

$$\begin{aligned} & -\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left\{ \left[h_{\beta}^{\circ} - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \rho_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} + \left[h_{s1}^{\circ} - (c_p)_1 (\langle T \rangle - T_{\sigma}^{\circ}) \right] \rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} \right\} dA \\ &= -\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left\{ \left[h_{s1}^{\circ} - (c_p)_1 (\langle T \rangle - T_{\sigma}^{\circ}) \right] \rho_1 (\bar{v}_1 - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} - \left[h_{\beta}^{\circ} - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \rho_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} \right\} dA \\ &= \left\{ \left[h_{s1}^{\circ} - h_{\beta}^{\circ} + (c_p)_{s1} (\langle T \rangle - T_{\sigma}^{\circ}) - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \right\} \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_{\beta} (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA \quad (2.3.32) \end{aligned}$$

We may use the following definitions:

$$Q_1 \text{ (at temperature } \langle T \rangle) = \left[h_{s1}^{\circ} - h_{\beta}^{\circ} + (c_p)_{s1} (\langle T \rangle - T_{\sigma}^{\circ}) - (c_p)_{\beta} (\langle T \rangle - T_{\beta}^{\circ}) \right] \quad (2.3.33)$$

$$\langle \dot{m}_{sl} \rangle = \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \rho_{\sigma} (\bar{v}_{\sigma} - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} dA \quad (2.3.34)$$

to rewrite the original integral as:

$$\begin{aligned} & -\frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} \left\{ \sum_{j=1}^{j=N} \rho_j \left[\bar{h}_j - (c_p)_j \tilde{T}_{\sigma} \right] (\bar{v}_j - \bar{w}_2) \cdot \bar{n}_{\sigma\beta} + \rho_{\beta} \left[h_{\beta} - (c_p)_{\beta} \tilde{T}_{\beta} \right] (\bar{v}_{\beta} - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} \right\} dA \\ &= Q_1 \langle \dot{m}_{sl} \rangle \quad (2.3.35) \end{aligned}$$

For the gas-solid interface, the heat of desorption for the vapor is equal to the energy required to desorb the liquid plus the enthalpy of vaporization required to evaporate the liquid:

$$Q_v = Q_l + \Delta h_{vap} \quad (2.3.36)$$

The derivation is exactly the same as for the other two interfaces, where the only component crossing the phase interface is component 1 (water) and we may write the integral as:

$$\begin{aligned} & -\frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} \left\{ \sum_{j=1}^{j=N} \rho_j \left[\bar{h}_j - (c_p)_j \tilde{T}_\sigma \right] (\bar{v}_j - \bar{w}_1) \cdot \bar{n}_{\sigma\gamma} + \sum_{i=1}^{i=N} \rho_i \left[\bar{h}_i - (c_p)_i \tilde{T}_\gamma \right] (\bar{v}_i - \bar{w}_1) \cdot \bar{n}_{\gamma\sigma} \right\} dA \\ & = (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle \end{aligned} \quad (2.3.37)$$

For these equations $\langle \dot{m}_{sl} \rangle$ is the mass flux desorbing from the solid to the liquid phase, $\langle \dot{m}_{sv} \rangle$ is the mass flux desorbing from the solid into the gas phase, and $\langle \dot{m}_{lv} \rangle$ is the mass flux evaporating from the liquid phase to the gas phase.

The total thermal energy equation now becomes:

$$\begin{aligned} & \langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + \left[\sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \bar{v}_j \rangle + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle + \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right] \cdot \nabla \langle T \rangle \\ & + \Delta h_{vap} \langle \dot{m}_{lv} \rangle + Q_l \langle \dot{m}_{sl} \rangle + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle \\ & = \nabla \cdot \left[\begin{aligned} & \left[\nabla \left[(k_\sigma \epsilon_\sigma + k_\beta \epsilon_\beta + k_\gamma \epsilon_\gamma) \langle T \rangle \right] \right. \\ & + (k_\sigma - k_\beta) \frac{1}{\mathcal{V}} \int_{A_{\sigma\beta}} T_\sigma \bar{n}_{\sigma\beta} dA \\ & + (k_\beta - k_\gamma) \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} T_\beta \bar{n}_{\beta\gamma} dA \\ & \left. + (k_\sigma - k_\gamma) \frac{1}{\mathcal{V}} \int_{A_{\sigma\gamma}} T_\gamma \bar{n}_{\sigma\gamma} dA \right] \end{aligned} \right] \end{aligned} \quad (2.3.38)$$

We may simplify the total thermal energy equation based on an effective thermal conductivity, and write our total thermal energy equation in a much shorter form as:

$$\begin{aligned} & \langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + \left[\sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \bar{v}_j \rangle + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle + \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right] \cdot \nabla \langle T \rangle \\ & + \Delta h_{vap} \langle \dot{m}_{lv} \rangle + Q_l \langle \dot{m}_{sl} \rangle + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle \\ & = \nabla \cdot (K_{eff}^T \cdot \nabla \langle T \rangle) \end{aligned} \quad (2.3.39)$$

The effective thermal conductivity can be expressed in a variety of ways as described by Whitaker, depending on the assumptions you choose to make about the isotropy of the porous medium, the importance of the dispersion terms, etc. The effective thermal conductivity is also an appropriate place to include radiative heat transfer, and one could add an apparent radiative component of thermal conductivity to the effective thermal conductivity to account for radiation heat transfer.

2.4 Thermodynamic Relations

The gas phase is assumed to be ideal, which gives the intrinsic phase partial pressures of the gas phase as:

$$\langle p_i \rangle^Y = \langle \rho_i \rangle^Y R_i \langle T \rangle \quad i = 1, 2, \dots \quad (2.4.1)$$

We also have the relations for the gas phase, where for our case component 1 is water, and component 2 is air:

$$\begin{aligned} \langle \rho_Y \rangle^Y &= \langle \rho_1 \rangle^Y + \langle \rho_2 \rangle^Y \\ \langle p_Y \rangle^Y &= \langle p_1 \rangle^Y + \langle p_2 \rangle^Y \end{aligned} \quad (2.4.2 - 2.4.3)$$

We must also connect the differential heat of sorption, Q_i , with the concentration of water in the solid phase. An example of a general form for Q_i , (in J/kg) can be expressed as a function of the relative humidity ϕ [32].

$$Q_i \text{ (J/kg)} = 1.95 \times 10^5 (1 - \phi) \left(\frac{1}{(0.2 + \phi)} + \frac{1}{(1.05 - \phi)} \right), \quad \phi = \frac{p_v}{p_s} = \frac{\langle p_1 \rangle^Y}{p_s} \quad (2.4.4)$$

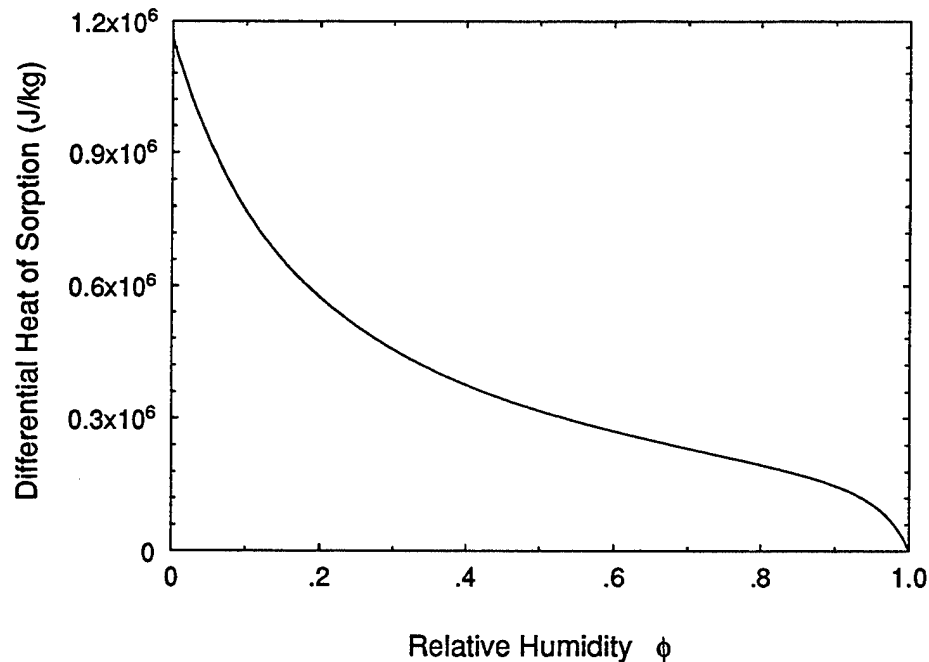


Figure 4. Generic differential heat of sorption for textile fibers (sorption hysteresis neglected).

We must connect the differential heat of sorption with the actual equilibrium water content in the solid phase. For the two component mixture of solid (component 2) plus bound water (component 1) in the solid phase, the density of the solid phase is given by:

$$\langle \rho_s \rangle^\sigma = \langle \rho_1 \rangle^\sigma + \langle \rho_2 \rangle^\sigma \quad (2.4.5)$$

We could make the assumption that mass transport in the textile fiber portion is so rapid that the fiber is always in equilibrium with the partial pressure of the gas phase, or is saturated if any liquid phase is present. This would eliminate the need to account for the transport through the solid phase at all. There are a variety of sorption isotherm relationships we could use, including the experimentally-determined relationships for a specific fiber type, but a convenient one is given by [32].

$$\text{Regain } (R) = R_f(0.55\phi) \left[\frac{1}{(0.25 + \phi)} + \frac{1}{(1.25 - \phi)} \right] \quad (2.4.6)$$

R_f is the standard textile measurement of grams of water absorbed per 100 grams of fiber, measured at 65% relative humidity. We may rewrite this in terms of the intrinsic phase averages for our two phases as:

$$R = \frac{\langle \rho_1 \rangle^\sigma}{100 \langle \rho_2 \rangle^\sigma} = R_f \left(55 \frac{\langle p_1 \rangle^\gamma}{p_s} \right) \left[\frac{1}{\left(0.25 + \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} + \frac{1}{\left(1.25 - \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} \right] \quad (2.4.7)$$

If we don't want to make the assumption that the solid phase is always in equilibrium, we may use relations available between the rate of change of concentration of the solid phase and the relative humidity of the gas phase, an example of which is given by Norden and David [33].

We may also write the vapor pressure-temperature relation for the vaporizing β phase, which Whitaker gives for porous media as:

$$\langle p_1 \rangle^\gamma = p_1^\circ \exp \left\{ - \left[\left(\frac{2\sigma_{\beta\gamma}}{r\rho_\beta R_1 \langle T \rangle} \right) + \frac{\Delta h_{vap}}{R_1} \left(\frac{1}{\langle T \rangle} - \frac{1}{T_o} \right) \right] \right\} \quad (2.4.8)$$

This relation gives the reduction or increase in vapor pressure from a curved liquid surface resulting from a liquid droplet influenced by the surface interaction between the solid and the liquid, usually in a very small capillary.

In many cases, the Clausis-Clapeyron equation will be sufficiently accurate for the vaporizing species, and the gas phase vapor pressure may be found from:

$$\langle p_1 \rangle^\gamma = p_1^\circ \exp \left\{ - \left[\frac{\Delta h_{vap}}{R_1} \left(\frac{1}{\langle T \rangle} - \frac{1}{T_o} \right) \right] \right\} \quad (2.4.9)$$

This vapor pressure-temperature relation is only good if we have the liquid phase present in the averaging volume. We may also have the situation where we only have the solid phase, containing adsorbed water, and the gas phase. To get the vapor pressure in the gas phase in this situation, we will use the sorption isotherm, and assume that the gas phase is in equilibrium with the sorbed water content of the solid phase.

We can use any isotherm relation where we have the solid water concentration as a function of relative humidity. The equation given previously is one example:

$$\frac{\langle \rho_1 \rangle^\sigma}{\langle \rho_2 \rangle^\sigma} = \frac{\epsilon_{sl} \rho_l}{(1 - \epsilon_{sl}) \rho_s} = R_f \left(0.55 \frac{\langle p_1 \rangle^\gamma}{p_s} \right) \left[\frac{1}{\left(0.25 + \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} + \frac{1}{\left(1.25 - \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} \right] \quad (2.4.10)$$

2.5 Mass Transport in the Gas Phase

The volume average form of the gas phase continuity equation was found to be:

$$\begin{aligned} \frac{\partial}{\partial t} \left(\epsilon_\gamma \langle \rho_\gamma \rangle^\gamma \right) + \nabla \cdot \left(\langle \rho_\gamma \rangle^\gamma \langle \vec{v}_\gamma \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\gamma (\vec{v}_\gamma - \vec{w}) \cdot \vec{n}_{\beta\gamma} dA \\ + \frac{1}{\mathcal{V}} \int_{A_{\gamma\sigma}} \rho_\gamma (\vec{v}_\gamma - \vec{w}_1) \cdot \vec{n}_{\gamma\sigma} dA = 0 \end{aligned} \quad (2.5.1)$$

and the species continuity equation was given as:

$$\frac{\partial}{\partial t} \left(\epsilon_\gamma \langle \rho_1 \rangle^\gamma \right) + \nabla \cdot \left(\langle \rho_1 \rangle^\gamma \langle \vec{v}_1 \rangle \right) + \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_1 (\vec{v}_1 - \vec{w}) \cdot \vec{n}_{\beta\gamma} dA = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D} \nabla \left(\frac{\langle \rho_i \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.2)$$

where the dispersion and source terms were dropped from the equation.

If we use the definition of the mass flux from one phase to another as:

$$\langle \dot{m}_{lv} \rangle = \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\beta (\vec{v}_\beta - \vec{w}) \cdot \vec{n}_{\beta\gamma} dA \quad (2.5.3)$$

or

$$\langle \dot{m}_{lv} \rangle = - \frac{1}{\mathcal{V}} \int_{A_{\beta\gamma}} \rho_\gamma (\vec{v}_\gamma - \vec{w}) \cdot \vec{n}_{\beta\gamma} dA, \quad (2.5.4)$$

with the same form for the mass flux from the solid to the gas phase, the gas phase continuity equation may be rewritten as:

$$\frac{\partial}{\partial t} \left(\epsilon_\gamma \langle \rho_\gamma \rangle^\gamma \right) + \nabla \cdot \left(\langle \rho_\gamma \rangle^\gamma \langle \vec{v}_\gamma \rangle \right) = \langle \dot{m}_{lv} \rangle + \langle \dot{m}_{sv} \rangle \quad (2.5.5)$$

For the two species (1--water, and 2--air), the species continuity equations are written (again dropping the source and dispersion terms) as:

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_1 \rangle^\gamma) + \nabla \cdot (\langle \rho_1 \rangle^\gamma \langle \bar{v}_\gamma \rangle) - \langle \dot{m}_{lv} \rangle - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D} \nabla \left(\frac{\langle \rho_1 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.6)$$

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_2 \rangle^\gamma) + \nabla \cdot (\langle \rho_2 \rangle^\gamma \langle \bar{v}_\gamma \rangle) = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D} \nabla \left(\frac{\langle \rho_2 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.7)$$

If we again ignore the effects of the dispersion terms in the diffusion equations derived by Whitaker, we may incorporate an effective diffusivity into the species continuity equations, which are now given as:

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_1 \rangle^\gamma) + \nabla \cdot (\langle \rho_1 \rangle^\gamma \langle \bar{v}_\gamma \rangle) - \langle \dot{m}_{lv} \rangle - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_1 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.8)$$

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_2 \rangle^\gamma) + \nabla \cdot (\langle \rho_2 \rangle^\gamma \langle \bar{v}_\gamma \rangle) = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_2 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.9)$$

The effective diffusivity will be some kind of function of the gas phase volume ϵ ; as the solid volume and the liquid volume fractions increase, there will be less space available in the gas phase for the diffusion to take place. We might try to define the effective diffusivity as:

$$D_{eff} = \frac{D_{12} \epsilon_\gamma}{\tau} = \frac{D_a \epsilon_\gamma}{\tau}, \quad (2.5.10)$$

where the effective diffusivity D_{eff} is related to the diffusion coefficient of water vapor in air (D_{12} or D_a) divided by the effective tortuosity factor τ .

An example of a good relation for the binary diffusion coefficient of water vapor in air is given by Stanish [34] as:

$$D_{12} = \left(\frac{2.23}{\langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma} \right) \left(\frac{T}{273.15} \right)^{1.75} \quad (\text{mks units}) \quad (2.5.11)$$

To simplify matters, one could assume the tortuosity factor to be constant, and let the variation in the gas phase volume take care of the change in the effective diffusion coefficient as the volume available for gas phase diffusion changes with solid swelling and/or liquid volume.

Another simplification is to only account for the water vapor movement, so the continuity equation would become:

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_1 \rangle^\gamma) + \nabla \cdot (\langle \rho_1 \rangle^\gamma \langle \bar{v}_\gamma \rangle) - \langle \dot{m}_{lv} \rangle - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \frac{D_{12}}{\tau} \nabla \left(\frac{\langle \rho_1 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.5.12)$$

2.6 Gas Phase Convective Transport

It is important to include forced convection through porous media since this can be an important part of the transport process of mass and energy through porous materials with high air permeability.

It is not necessary to modify any of Whitaker's derivations for the gas phase, and if we neglect gravity, we may write the gas phase velocity as:

$$\langle \bar{v}_\gamma \rangle = -\frac{1}{\mu_\gamma} K_\gamma \cdot \left\{ \epsilon_\gamma \left[\nabla \langle p_\gamma - p_0 \rangle^\gamma \right] \right\} \quad (2.6.1)$$

where the permeability tensor K_γ is a transport coefficient.

There are other methods to obtain an estimate of the convective velocity of a gas flow through a porous material. It may be desirable to use one of these other relations to obtain the volume average form of the gas velocity.

For example, we could start directly with Darcy's law:

$$\nabla P + \frac{\mu}{K} \bar{v}_\gamma = 0 \quad (2.6.2)$$

and assume that for the dry porous material we have available the experimental measurement of the specific permeability coefficient K , and then modify it to account for the decrease in gas phase volume as the solid phase swells and/or the liquid phase accumulates. We could make the variation in K a function of the gas phase volume, which has been an approach used by Stanish et. al. [34].

$$K_\gamma = K_{dry}^\gamma \left(\frac{\epsilon_\gamma}{\epsilon_{\gamma dry}} \right) \quad (2.6.3)$$

This is a very simple model, and may be improved upon. In the book by Dullien [35], there are a variety of relationships for how K varies with porosity; some of those relations may be more realistic for our purposes. We could also relate the change in the permeability to the effective tortuosity function τ , which also has the same factors related to the decrease in gas phase volume, and change in geometry, that we need to account for the Darcy's law relation for convective gas flow.

2.7 Liquid Phase Convective Transport

Whitaker's derivation for the convection transport of the liquid phase is the one of the most complicated parts of his general theory. He accounts for the capillary liquid transport, which is greatly influenced by the geometry of the solid phase, and the changeover from a continuous to a discontinuous liquid phase. His eventual transport equation, which gives an expression for the liquid phase average velocity is quite complicated, and depends on several hard-to-obtain transport coefficients. The final equation is given as:

$$\langle \bar{v}_\beta \rangle = - \left(\frac{\epsilon_\beta \xi K_\beta}{\mu_\beta} \right) \cdot [k_\epsilon \nabla \epsilon_\beta + k_{(T)} \nabla \langle T \rangle - (\rho_\beta - \rho_\gamma)] \quad (2.7.1)$$

(symbol definitions given in nomenclature table)

One advantage of Whitaker's derivation is that it is almost completely independent of the other transport equation derivations. This should mean that we may use another expression for the liquid phase velocity if we find one that is more amenable to experimental measurement and verification.

An example of an equation which is more empirical is again given by Stanish [34]. The velocity is assumed proportional to the gradient in pressure within the liquid. The pressure in the liquid phase is assumed to be the sum of the gas pressure within the averaging volume minus the capillary pressure (P_c):

$$\langle \bar{v}_\beta \rangle = - \left(\frac{k_\beta}{\mu_\beta} \right) \nabla (\langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma - P_c) \quad (2.7.2)$$

If we use a relation of this kind it is necessary to obtain an equation for the capillary pressure as a function of the fraction of non-solid volume occupied by the liquid phase, as well as a relation for the variation in the permeability coefficient as a function of liquid phase volume fraction. It is also necessary to determine when the liquid phase becomes discontinuous so that liquid flow ceases at that point. These types of relations can be determined experimentally for materials of interest, or they may be found in the literature for quite a wide variety of materials.

2.8 Summary of Modified Transport Equations

The set of modified equations which describe the coupled transfer of heat and mass through hygroscopic porous materials are summarized below.

Total thermal energy equation (2.8.1)

$$\begin{aligned} \langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + \left[\sum_{j=1}^{j=N} (c_p)_j \langle \rho_j \bar{v}_j \rangle + \rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle + \sum_{i=1}^{i=N} (c_p)_i \langle \rho_i \bar{v}_i \rangle \right] \cdot \nabla \langle T \rangle \\ + \Delta h_{vap} \langle \dot{m}_{lv} \rangle + Q_l \langle \dot{m}_{sl} \rangle + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = \nabla \cdot (K_{eff}^T \cdot \nabla \langle T \rangle) \end{aligned}$$

Liquid phase equation of motion

$$\langle \bar{v}_\beta \rangle = - \left(\frac{k_\beta}{\mu_\beta} \right) \nabla (\langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma - P_c) \quad (2.8.2)$$

Liquid phase continuity equation

$$\frac{\partial \epsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle + \frac{1}{\eta'} \int_{A_{\beta\gamma}} (\bar{v}_\beta - \bar{w}) \cdot \bar{n}_{\beta\gamma} dA + \frac{1}{\eta'} \int_{A_{\beta\sigma}} (\bar{v}_\beta - \bar{w}_2) \cdot \bar{n}_{\beta\sigma} dA = 0 \quad (2.8.3)$$

which may be rewritten as:

$$\frac{\partial \epsilon_\beta}{\partial t} + \nabla \cdot \langle \bar{v}_\beta \rangle + \frac{(\langle \dot{m}_{lv} \rangle - \langle \dot{m}_{sv} \rangle)}{\rho_\beta} = 0 \quad (2.8.40)$$

Gas phase equation of motion

$$\langle \bar{v}_\gamma \rangle = - \left(\frac{k_\gamma}{\mu_\gamma} \right) \nabla (\langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma) \quad (2.8.5)$$

Gas phase continuity equation

$$\frac{\partial}{\partial t} (\epsilon_\gamma \langle \rho_\gamma \rangle^\gamma) + \nabla \cdot (\langle \rho_\gamma \rangle^\gamma \langle \bar{v}_\gamma \rangle) = \langle \dot{m}_{lv} \rangle + \langle \dot{m}_{sv} \rangle \quad (2.8.6)$$

Gas phase diffusion equations

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_1 \rangle^\gamma) + \nabla \cdot (\langle \rho_1 \rangle^\gamma \langle \bar{v}_\gamma \rangle) - \langle \dot{m}_{lv} \rangle - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_1 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.8.7)$$

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_2 \rangle^\gamma) + \nabla \cdot (\langle \rho_2 \rangle^\gamma \langle \bar{v}_\gamma \rangle) = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_2 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.8.8)$$

Solid phase density relations

$$\langle \rho_\sigma \rangle^\sigma = \langle \rho_1 \rangle^\sigma + \langle \rho_2 \rangle^\sigma$$

$$\rho_1 = \epsilon_{\sigma L} \rho_L$$

$$\rho_2 = (1 - \epsilon_{\sigma L}) \rho_S \quad (2.8.9 - 2.8.12)$$

$$\epsilon_{\sigma S} + \epsilon_{\sigma L} = 1$$

Solid phase continuity equation

$$\frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_\sigma \rangle^\sigma) + \nabla \cdot (\langle \rho_\sigma \rangle^\sigma \langle \bar{v}_\sigma \rangle) + \langle \dot{m}_{sl} \rangle + \langle \dot{m}_{sv} \rangle = 0 \quad (2.8.13)$$

Solid phase equation of motion (for one dimensional geometry)

$$\langle v_\sigma \rangle^\sigma = \frac{1}{\langle \rho \rangle^\sigma \xi^{n-1}} \int_0^\xi \frac{\partial}{\partial t} \langle \rho_\sigma \rangle^\sigma d\xi \quad (2.8.14)$$

Solid phase diffusion equation (for vaporizing component)

$$\frac{\partial}{\partial t}(\epsilon_\sigma \langle \rho_1 \rangle^\sigma) + \nabla \cdot (\langle \rho_1 \rangle^\sigma \langle \bar{v}_1 \rangle) + \langle \dot{m}_{sl} \rangle + \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\sigma \rangle^\sigma \mathcal{D}_\sigma \nabla \left(\frac{\langle \rho_1 \rangle^\sigma}{\langle \rho_\sigma \rangle^\sigma} \right) \right\} \quad (2.8.15)$$

Volume constraint

$$\epsilon_\sigma(t) + \epsilon_\beta(t) + \epsilon_\gamma(t) = 1 \quad (2.8.16)$$

Thermodynamic relations

$$\langle p_1 \rangle^\gamma = \langle \rho_1 \rangle^\gamma R_1 \langle T \rangle$$

$$\langle p_2 \rangle^\gamma = \langle \rho_2 \rangle^\gamma R_2 \langle T \rangle$$

$$\langle \rho_\gamma \rangle^\gamma = \langle \rho_1 \rangle^\gamma + \langle \rho_2 \rangle^\gamma \quad (2.8.17 - 2.8.20)$$

$$\langle p_\gamma \rangle^\gamma = \langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma$$

If any liquid phase is present, vapor pressure is given by:

$$\langle p_1 \rangle^\gamma = p_1^\circ \exp \left\{ - \left[\left(\frac{2\sigma_{\beta\gamma}}{r\rho_\beta R_1 \langle T \rangle} \right) + \frac{\Delta h_{vap}}{R_1} \left(\frac{1}{\langle T \rangle} - \frac{1}{T_o} \right) \right] \right\} \text{ or}$$

$$\langle p_1 \rangle^\gamma = p_1^\circ \exp \left\{ - \left[\frac{\Delta h_{vap}}{R_1} \left(\frac{1}{\langle T \rangle} - \frac{1}{T_o} \right) \right] \right\} \quad (2.8.21)$$

If the liquid phase is not present, and the liquid component is desorbing from the solid, the reduced vapor pressure in equilibrium with the solid must be used. This relation may be determined directly from the sorption isotherm for the solid:

$$\langle p_1 \rangle^\gamma = f(p_s, \rho_l, \rho_s, \epsilon_{\sigma L}) \text{ at the temperature } \langle T \rangle, \text{ only } \epsilon_{\sigma L} \text{ is unknown.} \quad (2.8.22)$$

Sorption relations (volume average solid equilibrium)

$$Q_{sl} \text{ (J/kg)} = 0.195 \left(1 - \frac{\langle p_1 \rangle^\gamma}{p_s} \right) \left(\frac{1}{\left(0.2 + \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} + \frac{1}{\left(1.05 - \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} \right) \quad (2.8.23)$$

$$\frac{\langle \rho_1 \rangle^\sigma}{\langle \rho_2 \rangle^\sigma} = \frac{\epsilon_{\sigma l} \rho_l}{(1 - \epsilon_{\sigma l}) \rho_s} = R_f \left(0.55 \frac{\langle p_1 \rangle^\gamma}{p_s} \right) \left[\frac{1}{\left(0.25 + \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} + \frac{1}{\left(1.25 - \frac{\langle p_1 \rangle^\gamma}{p_s} \right)} \right] \quad (2.8.24)$$

This is a total of 20 main equations and 20 unknown variables, which should allow for the solution of the set of equations using numerical methods. The 20 unknown variables are:

$$\begin{aligned} & \epsilon_\sigma, \epsilon_\beta, \epsilon_\gamma, \langle \bar{v}_\sigma \rangle, \langle \bar{v}_\beta \rangle, \langle \bar{v}_\gamma \rangle, \langle T \rangle, \\ & \langle \dot{m}_{sl} \rangle, \langle \dot{m}_{sv} \rangle, \langle \dot{m}_{lv} \rangle, Q_{sl} \\ & \langle p_\gamma \rangle^\gamma, \langle p_1 \rangle^\gamma, \langle p_2 \rangle^\gamma, \langle \rho_\gamma \rangle^\gamma, \langle \rho_1 \rangle^\gamma, \langle \rho_2 \rangle^\gamma \\ & \langle \rho_\gamma \rangle^\sigma, \langle \rho_1 \rangle^\sigma, \langle \rho_2 \rangle^\sigma \end{aligned}$$

2.9 Comparison with Previously-Derived Equations

The simplified systems of partial differential equations given in the previous chapter still contain many equations with a large number of unknown variables. Even for the simplified case of vapor diffusion, the system of equations is quite confusing, and it is difficult to verify their accuracy, other than by checking for dimensional consistency. One way of checking their validity is to see if they simplify down to more well-known diffusion equations for the transport of water vapor in air through a porous hygroscopic solid. Such a system of equations has been well documented by Henry [36], Norden and David [33], and Li and Holcombe [37], who have used them to describe the diffusion of water vapor through a hygroscopic porous material.

We will make the same assumptions used by these previous workers, and attempt to transform the system of equations for the case of vapor diffusion (no liquid or gas phase convection) to their system of equations. For completeness, we will also need to write the various equations in terms of the variables and units used their work.

The major simplifying assumptions are: 1) there is no liquid or gas phase convection, 2) there is no liquid phase present, 3) the heat capacity of the gas phase can be neglected, 4) the volume of the solid remains constant and does not swell, 5) the solid and gas phase volume fractions are both constant, 6) the thermal conductivity tensor may be expressed as a constant scalar thermal conductivity coefficient, 7) the gas phase diffusion coefficient is constant, 7) the transport is one-dimensional (x-direction).

The total thermal energy equation becomes:

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = \nabla \cdot (K_{eff}^T \cdot \nabla \langle T \rangle) \quad (2.9.1)$$

or

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.2)$$

The gas phase continuity equation becomes:

$$\epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_\gamma \rangle^\gamma) = \langle \dot{m}_{sv} \rangle \quad (2.9.3)$$

The gas phase diffusion equation (component 1--water vapor):

$$\epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_1 \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \quad (2.9.4)$$

or

$$\epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) - \langle \dot{m}_{sv} \rangle = \mathcal{D}_{eff} \frac{\partial^2 \langle \rho_1 \rangle^\gamma}{\partial x^2} \quad (2.9.5)$$

The solid phase continuity equation (component 1--water):

$$\epsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle = 0 \quad (2.9.6)$$

For the solid phase diffusion equation (component 1--water) we assume that the diffusional transport through the solid phase is insignificant compared to the diffusion through the gas phase, so the diffusion equation reduces to the continuity equation:

$$\epsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\sigma \rangle^\sigma \mathcal{D}_\sigma \nabla \left(\frac{\rho_1}{\langle \rho_\sigma \rangle^\sigma} \right) \right\} = 0 \quad (2.9.7)$$

Volume fraction constraint

$$\epsilon_\gamma + \epsilon_\sigma = 1 \quad ; \quad \epsilon_\sigma = 1 - \epsilon_\gamma \quad (2.9.8)$$

Thermodynamic relations

$$\begin{aligned} \langle p_1 \rangle^\gamma &= \langle \rho_1 \rangle^\gamma R_1 \langle T \rangle \\ \langle p_2 \rangle^\gamma &= \langle \rho_2 \rangle^\gamma R_2 \langle T \rangle \\ \langle \rho_\gamma \rangle^\gamma &= \langle \rho_1 \rangle^\gamma + \langle \rho_2 \rangle^\gamma \\ \langle p_\gamma \rangle^\gamma &= \langle p_1 \rangle^\gamma + \langle p_2 \rangle^\gamma \end{aligned} \quad (2.9.9 - 2.9.12)$$

We made the assumption that the mass transport through the solid phase is negligible compared to mass transport through the gas phase. This is reasonable since the diffusion coefficient for water in a solid is always much less than the diffusion coefficient of water vapor through air. We thus only have accumulation of water in the solid, and the solid acts as a source or sink for water vapor.

We can combine the continuity equations for water (component 1) for both phases by connecting the phase equations through the mass flux from the solid to the gas phase:

$$\begin{aligned} \epsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle &= 0 \\ \epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) - \langle \dot{m}_{sv} \rangle &= \mathcal{D}_{eff} \frac{\partial^2 \langle \rho_1 \rangle^\gamma}{\partial x^2} \end{aligned} \quad (2.9.13 - 2.9.14)$$

$$\left[\epsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle \right] + \left[\epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) - \langle \dot{m}_{sv} \rangle \right] = \mathcal{D}_{eff} \frac{\partial^2 \langle \rho_1 \rangle^\gamma}{\partial x^2} \quad (2.9.15)$$

which we may rewrite in terms of the gas phase volume fraction as:

$$(1 - \epsilon_\gamma) \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \epsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) = \mathcal{D}_{eff} \frac{\partial^2 \langle \rho_1 \rangle^\gamma}{\partial x^2} \quad (2.9.16)$$

Through these various assumptions, we have reduced our large equation set down to two main equations for the energy balance and the mass balance:

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.17)$$

$$(1 - \varepsilon_\gamma) \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \varepsilon_\gamma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\gamma) = \mathcal{D}_{eff} \frac{\partial^2 \langle \rho_1 \rangle^\gamma}{\partial x^2} \quad (2.9.18)$$

To make the comparison with the existing equations of Henry [36], Norden and David [33], and Li and Holcombe [37], easier, we can rewrite the intrinsic phase averages in terms of the concentration of water in the solid (C_F) and concentration of water in the gas phase (C):

$$C_F = \frac{\text{mass of water in solid phase}}{\text{solid phase volume}} = \frac{m_{1\sigma}}{V_\sigma} = \rho_{1\sigma} \quad (2.9.19 - 2.9.20)$$

$$C = \frac{\text{mass of water in gas phase}}{\text{gas phase volume}} = \frac{m_{1\gamma}}{V_\gamma} = \rho_{1\gamma}$$

Since the definition of intrinsic phase average gives the same quantity as the true point value, we may use the fact that

$$\begin{aligned} \langle \rho_1 \rangle^\sigma &= \langle C_F \rangle^\sigma = C_F \\ \langle \rho_1 \rangle^\gamma &= \langle C \rangle^\gamma = C \end{aligned} \quad (2.9.21 - 2.9.22)$$

to rewrite the mass balance equation as:

$$(1 - \varepsilon_\gamma) \frac{\partial C_F}{\partial t} + \varepsilon_\gamma \frac{\partial C}{\partial t} = \mathcal{D}_{eff} \frac{\partial^2 C}{\partial x^2} \quad (2.9.23)$$

We can rewrite the effective diffusion coefficient by using the diffusion coefficient for water vapor in air modified by the gas volume fraction and the tortuosity of the gas volume fraction:

$$(1 - \varepsilon_\gamma) \frac{\partial C_F}{\partial t} + \varepsilon_\gamma \frac{\partial C}{\partial t} = \frac{D_a \varepsilon_\gamma}{\tau} \frac{\partial^2 C}{\partial x^2} \quad (2.9.24)$$

The thermal energy equation

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.25)$$

may also be modified by recognizing that the mass flux term is contained in the solid phase continuity equation:

$$\varepsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle = 0 \Rightarrow \langle \dot{m}_{sv} \rangle = -\varepsilon_\sigma \frac{\partial C_F}{\partial t} \quad (2.9.26)$$

so that the thermal energy equation may be rewritten as:

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} - (Q_l + \Delta h_{vap}) \epsilon_\sigma \frac{\partial C_F}{\partial t} = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.27)$$

If we go back to our definitions for the mass fraction weighted average heat capacity,

$$C_p = \frac{\epsilon_\sigma \sum_{j=1}^{j=N} \langle \rho_j \rangle^\sigma (c_p)_j + \epsilon_\gamma \sum_{i=1}^{i=N} \langle \rho_i \rangle^\gamma (c_p)_i}{\langle \rho \rangle} \quad (2.9.28)$$

and spatial average density,

$$\langle \rho \rangle = \epsilon_\sigma \sum_{j=1}^{j=N} \langle \rho_j \rangle^\sigma + \epsilon_\gamma \sum_{i=1}^{i=N} \langle \rho_i \rangle^\gamma \quad (2.9.29)$$

the thermal energy equation may be rewritten as:

$$\epsilon_\sigma [\langle \rho_1 \rangle^\sigma (c_p)_1 + \langle \rho_2 \rangle^\sigma (c_p)_2] + \epsilon_\gamma [\langle \rho_1 \rangle^\gamma (c_p)_1 + \langle \rho_2 \rangle^\gamma (c_p)_2] \frac{\partial \langle T \rangle}{\partial t} \quad (2.9.30)$$

$$- (Q_l + \Delta h_{vap}) \epsilon_\sigma \frac{\partial C_F}{\partial t} = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.31)$$

If we make the assumption that the heat capacity of the gas phase is negligible, then the thermal energy equation becomes:

$$\epsilon_\sigma [\langle \rho_1 \rangle^\sigma (c_p)_1 + \langle \rho_2 \rangle^\sigma (c_p)_2] \frac{\partial \langle T \rangle}{\partial t} - (Q_l + \Delta h_{vap}) \epsilon_\sigma \frac{\partial C_F}{\partial t} = k_{eff} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.31)$$

or dividing through by the solid volume fraction:

$$[\langle \rho_1 \rangle^\sigma (c_p)_1 + \langle \rho_2 \rangle^\sigma (c_p)_2] \frac{\partial \langle T \rangle}{\partial t} - (Q_l + \Delta h_{vap}) \frac{\partial C_F}{\partial t} = \frac{k_{eff}}{\epsilon_\sigma} \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.32)$$

For consistent nomenclature with Li and Holcombe¹⁴ we will write the effective thermal conductivity k_{eff} as K .

We can also define a volumetric heat capacity C_v as:

$$C_v \left(\frac{\text{J}}{\text{m}^3 \cdot \text{K}} \right) = \langle \rho_1 \rangle^\sigma (c_p)_1 + \langle \rho_2 \rangle^\sigma (c_p)_2$$

Units for $\langle \rho_j \rangle^\sigma (c_p)_j$ are $\left(\frac{\text{kg}}{\text{m}^3} \right) \left(\frac{\text{J}}{\text{kg} \cdot \text{K}} \right) \Rightarrow \left(\frac{\text{J}}{\text{m}^3 \cdot \text{K}} \right)$ (2.9.33)

The final thermal energy equation reduces to:

$$C_v \frac{\partial \langle T \rangle}{\partial t} - (Q_l + \Delta h_{vap}) \frac{\partial C_F}{\partial t} = K \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.34)$$

The two simplified equations for the mass and energy balance are thus:

$$(1 - \varepsilon_\gamma) \frac{\partial C_F}{\partial t} + \varepsilon_\gamma \frac{\partial C}{\partial t} = \frac{D_a \varepsilon_\gamma}{\tau} \frac{\partial^2 C}{\partial x^2} \quad (2.9.35)$$

$$C_v \frac{\partial \langle T \rangle}{\partial t} - (Q_l + \Delta h_{vap}) \frac{\partial C_F}{\partial t} = K \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.36)$$

These two simplified equations are very encouraging, since they are exactly the same as previous equations derived by Henry [36], Norden and David [33], and Li and Holcombe [37], for describing the diffusion of water vapor through a hygroscopic porous material. In their equations they define the heat of sorption from the vapor phase into the solid (which is the opposite of the heat of desorption which we used) as:

$$\lambda = (Q_l + \Delta h_{vap}) \quad (2.9.37)$$

so that their equations are:

$$(1 - \varepsilon_\gamma) \frac{\partial C_F}{\partial t} + \varepsilon_\gamma \frac{\partial C}{\partial t} = \frac{D_a \varepsilon_\gamma}{\tau} \frac{\partial^2 C}{\partial x^2} \quad (2.9.38)$$

$$C_v \frac{\partial \langle T \rangle}{\partial t} - \lambda \frac{\partial C_F}{\partial t} = K \frac{\partial^2 \langle T \rangle}{\partial x^2} \quad (2.9.39)$$

2.10 Conclusions

Whitaker's theory of coupled heat and mass transfer through porous media was modified to include hygroscopic porous materials which can absorb liquid into the solid matrix. The system of equations described in this report should make it possible to evaluate the time-dependent transport properties of hygroscopic and non-hygroscopic clothing materials by including many important factors which are usually ignored in the analysis of heat and mass transfer through textile materials. The set of equations allows for the unsteady capillary wicking of sweat through fabric structure, condensation and evaporation of sweat within various layers of the clothing system, forced gas phase convection through the porous structure of a textile layer, and the swelling and shrinkage of fibers and yarns as they absorb/desorb liquid water and water vapor.

The simplified set of equations for heat and mass transport, where mass transport occurs due to diffusion within the air spaces of the porous solid, was shown to reduce to the well-known coupled heat and mass transfer models for hygroscopic fabrics, as exemplified by the work of Li and Holcombe [37].

CHAPTER 3

EXPERIMENTAL METHODS

3.1 Introduction

Experimental measurements of the transport properties of textile materials were conducted to provide experimental data for comparison with the predictions of a numerical model, and also to conduct research on the changes in transport properties evident in these materials due to factors such as fiber swelling. Experimental measurements focused on two modes of transport: 1) diffusion/sorption of water vapor, and 2) gas phase convection of water vapor and air through porous textiles. Two new test methods were developed which are capable of determining both steady-state and transient properties of porous materials under situations of pure diffusion (concentration gradients), combined gas phase convection/diffusion, and pure convection (gas phase pressure differences). A complete description of both of these test methods, and their use to determine the diffusion and convection transport properties of textiles, are given in this chapter, along with representative results for several different classes of materials.

3.2 Diffusion Test Method (Dynamic Moisture Permeation Cell)

Water vapor permeation properties of clothing materials are usually determined using the ASTM Test Method for Water Vapor Transmission of Materials (E 96) [38], and the ISO Test Method for Measurement of Thermal and Water Vapour Resistance under Steady-State Conditions (ISO 11092) [39]. Both methods are time-consuming, require large amounts of material, and are not capable of a very wide range of test conditions. These are also pseudoequilibrium test methods, and do not give information on any of the dynamic properties of the test materials which may be important under transient conditions. There was a need for a convenient water vapor diffusion test which allows one to test small quantities of fabrics, membranes, or membrane laminates, and also to conduct research into the behavior of these materials under nonstandard conditions.

There are a wide variety of test methods developed over the years to characterize the diffusion and permeation properties of polymer films, laminates, textiles, and paper products [40-47]. The method described here, which has been called the Dynamic Moisture Permeation Cell (DMPC), is similar to a method used by Wehner, et. al [48], to measure the dynamics of water vapor transmission through hygroscopic and nonhygroscopic fabrics.

A schematic of the DMPC test arrangement is shown in Figure 5.

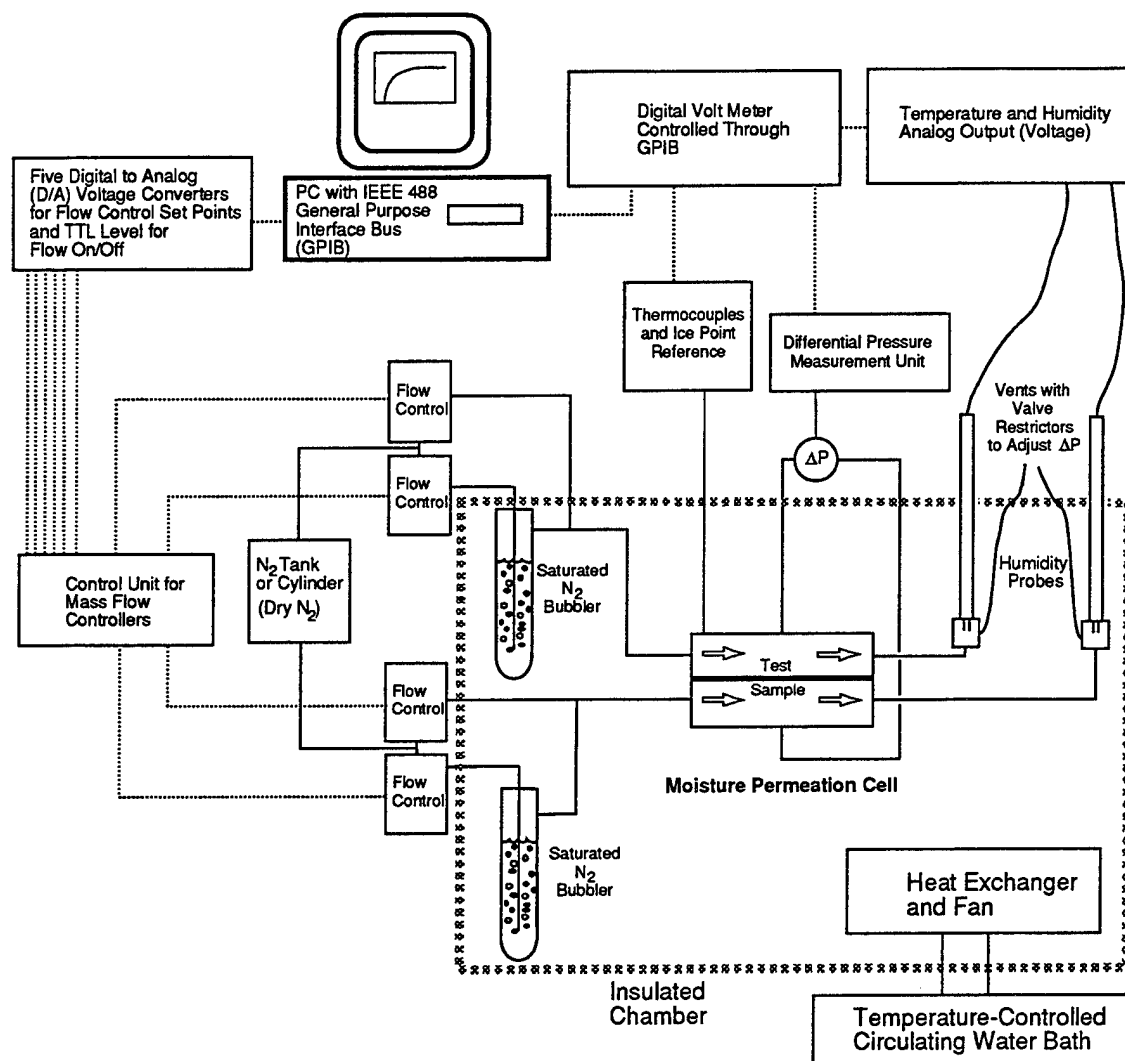


Figure 5. Schematic of DMPC test arrangement.

Nitrogen streams consisting of a mixture of dry nitrogen and water-saturated nitrogen are passed over the top and bottom surfaces of the sample. The relative humidity of these streams is varied by controlling the proportion of the saturated and the dry components. By knowing the temperature and water vapor concentration of the entering nitrogen flows, and by measuring the temperature and water vapor concentration of the nitrogen flows leaving the cell, one may measure the flux of water vapor diffusing through the test sample.

The following equations for calculating water vapor flux apply to either the top or bottom flows in the cell. Strictly speaking, only one measurement on one side of the cell is necessary; the use of two separate humidity transducer for the top and bottom flows allows two measurements of water vapor flux to be made at the same time, using the equations given below for either the top or bottom flow, as appropriate.

For this type of test, the mass flow rate of water vapor diffusing through the test sample from one side of the cell to the other is given by:

$$\frac{\dot{m}}{A} = \frac{Q(\delta C)}{A} = \frac{Q(C_2 - C_1)}{A} \quad (3.1)$$

- $\dot{m}$ mass flux of water vapor across the sample [kg/s]
- A area of test sample [m²]
- Q volumetric flow rate through top or bottom portion of the cell [m³/s]
- $\delta C = C_2 - C_1$, water vapor concentration difference between incoming stream (C_1) and outgoing stream (C_2) in top or bottom portion of the moisture permeation cell [kg/m³]

The incoming water vapor concentration is determined by the ratio of the mass flows of the saturated and the dry nitrogen streams. The mass flow rates are controlled by MKS model 1259C mass flow controllers, with a Model 247C 4-Channel Readout (MKS Instruments, Inc.). At constant mass flow, the true volumetric flow rate will vary with temperature; the flow rate set by the MKS controllers is indicated in terms of volumetric flow rates at standard conditions of 0°C and atmospheric pressure (1.01325 x 10⁵ Pa) The actual volumetric flow rate at different temperatures may be found from the mass flow rate, the temperature, and the pressure of the actual flow.

The critical measurement is the outgoing flow water vapor concentration C_2 , which we may measure in a variety of ways. At present, capacitance-type relative humidity probes (Vaisala HMI 32) with Type HMP 35 sensors are used (Vaisala Inc.). To obtain the water vapor concentration in the outgoing air stream, one must be able to convert from the known values of relative humidity and temperature to water vapor concentration. We may use an empirical formula (or tables) for the vapor pressure of saturated water vapor in air as a function of temperature, and then use the perfect gas law to convert vapor pressure to concentration.

We may express the water vapor transmission rate in terms of the indicated volumetric flow rate at standard conditions, the humidity difference, and the temperature:

$$\frac{\dot{m}}{A} = \frac{\delta \phi Q_s p_s M_w}{A R T_s} \quad (3.2)$$

M_w	molecular weight of water vapor [18.015 kg/kgmole]
Q_s	volume flow rate at standard conditions of 0°C and atmospheric pressure [m ³ /s]
R	universal gas constant [8314.5 N-m/kg-°K]
T_s	reference temperature at standard conditions of 0°C in degrees K (273.15 K)
p_s	saturation vapor pressure of water [Pa]
$\delta\phi$	$= \phi_2 - \phi_1$, relative humidity difference between incoming stream (ϕ_1) and outgoing stream (ϕ_2) in top or bottom portion of the moisture permeation cell
ϕ	$= p_v/p_s$, relative humidity
p_v	vapor pressure of water [Pa]

For the present test apparatus, various sample holders are available, which have different test sample measurement areas, and which have different downstream locations from the flow inlet. All test results given in this study used a sample measurement area of $1.0 \times 10^{-3} \text{ m}^2$, and the sample was located equidistant from the inlet and outlet ports of the cell. The typical volumetric flow rate used was $3.33 \times 10^{-5} \text{ m}^3/\text{s}$ (2000 cm³/min). The dimensions of the DMPC were chosen to assure flow velocities of at least 0.5 m/sec over the sample to minimize the contribution of boundary air layer resistances to the test measurements. Most of the tests were conducted with the flow parallel on the two sides of the test sample; for comparison, some countercurrent flow testing was also conducted and is described later. Details of the moisture permeation cell are shown in Figure 6.

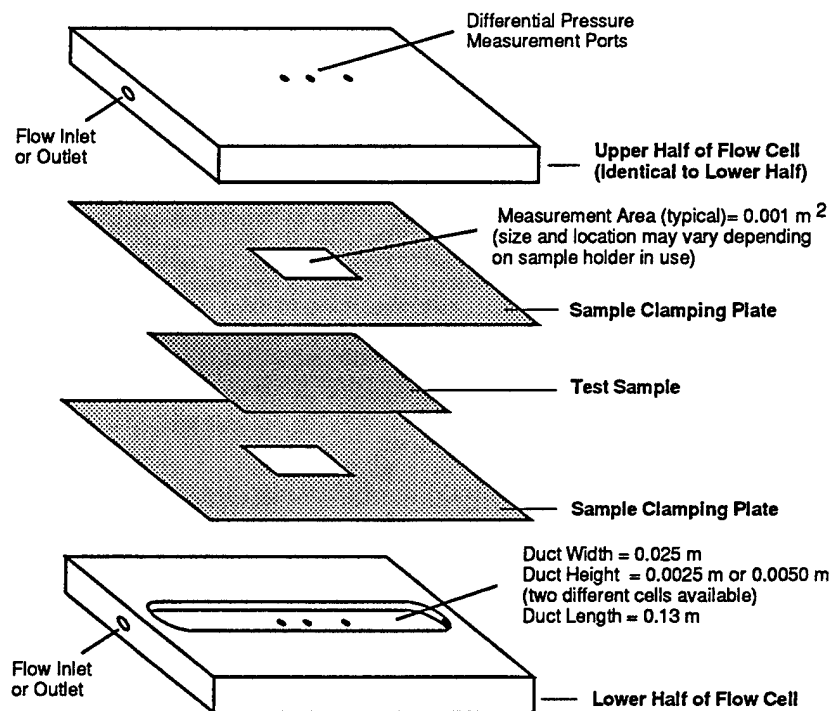


Figure 6. Schematic and Dimensions of the Dynamic Moisture Permeation Cell.

The flow path both above and below the sample is a long, wide duct, with a small clearance between the sample and the upper or lower duct surface. The flow in such a duct is approximated by the flow between parallel plates. Since the flow velocities are so low, and the flow geometry is quite narrow, it is clear that the flow is laminar. It is desirable to have the entrance length long enough for the flow to become fully developed by the time the flow reaches the sample. For laminar flow in a duct, the entrance length for the velocity profile to become fully-developed is given by:

$$\begin{aligned} X_{Lv} &= 0.05 \text{Re}_{Duct} H = 0.022 \text{ m} \\ X_{Lv} &= \text{entrance length for fully developed momentum boundary layer} \\ \text{Re}_{Duct} &= \frac{\rho V D_H}{\mu} = 175 \end{aligned} \quad (3.3)$$

$$\rho = 1.2 \text{ kg/m}^3 \text{ (Dry Nitrogen)}$$

$$V = 0.57 \text{ m/s (based on flow rate of } 2000 \text{ cm}^3/\text{min} \text{ and given flow area)}$$

$$D_H = \text{hydraulic diameter} = 4 \times (\text{flow area} / \text{wetted perimeter}) = 0.00454 \text{ m}$$

$$\mu = \text{absolute viscosity} = 1.78 \times 10^{-5} \text{ N-s/m}^2$$

$$H = \text{duct height (0.0025 m)}$$

The fluid properties and geometry of the duct give an entrance length for the fully-developed velocity profile of 0.02 meters, or 2 centimeters, which is a good distance away from the beginning of the sample location in the moisture permeation cell.

We can check this result by also assuming that we have flow between parallel plates, and base our Reynolds number solely on the plate separation distance. In the paper by Chen [49], we have a solution for the entrance length as a function of Reynolds number.

The Reynolds number is defined based on the half-height of the channel:

$$\begin{aligned} \text{Re} &= \frac{2a\rho V}{\mu} = \frac{H\rho V}{\mu} = 96 \\ a &= \text{duct half - height} = H/2 \end{aligned} \quad (3.4)$$

For parallel plate flow, the dimensionless entry length X_e (x_e is dimensional entry length) is given as a function of Reynolds number as:

$$X_e = \frac{x_e}{2a} = \frac{x_e}{H} = \frac{0.79}{0.04 \text{Re} + 1} + 0.053 \text{Re} = 6.09 \Rightarrow x_e = 0.015 \text{ m} \quad (3.5)$$

As we will see later from the flow analysis using a two-dimensional numerical fluid flow program, this shorter entrance length of 1.5 cm agrees with the calculated flow velocity profiles.

The test sample sizes are kept quite small to make it possible to evaluate novel membranes and laminates, which are often produced in quantities too small for testing by some of the standard water vapor diffusion test methods. Sample mounting methods vary according to the material being tested. Thin materials, such as laminated materials and woven cloth, were originally tested with rubber sealing gaskets to prevent leakage, but the sealing proved to be unnecessary for most materials; the clamping force provided by the mounting bolts has proven to be sufficient to prevent any leakage. Thick materials which are highly permeable require special sealing methods such as edge sealing by molten wax, or the use of a curable sealant. A new cell, which is capable of testing very thick materials such as battings used for cold weather clothing, is presently under development.

3.2.1 Test Procedure

The actual test is conducted under the control of a personal computer (PC) connected to the flow controllers and the relative humidity instruments through a General Purpose Interface Bus (GPIB) controller (see Figure 5). The operator inputs up to 20 desired humidity setpoints for the upper and lower nitrogen streams. The computer applies the proper setpoint voltage to each controller to produce the desired relative humidity in the upper and lower streams entering the moisture permeation cell. The analog voltage output of the relative humidity measurement instrument is read by the digital voltmeter and sent to the PC through the GPIB, and displayed on the screen. The computer plots the relative humidity, records the data to disk, and applies operator-determined equilibration criteria to determine when equilibration has been reached for that setpoint. Once equilibration is reached, the results (humidity, calculated flux, etc.) are output to a printer and to a data file on disk. The computer then proceeds to the next setpoint and repeats the process.

Calibration and Setup

Three calibration procedures must be observed before a series of tests begins. The flow meters must be calibrated either by an independent flow meter, or they may be calibrated directly by a special procedure of balancing using the humidity meters and switching of gas inputs to check for equality of flows. The zero reading and a full scale range check have proved sufficient so far. The particular flow controllers in use are quite stable from day to day if left on and warmed up.

After the calibration of the flow meter, the offset of the digital-to-analog (D/A) converters used to apply a setpoint voltage to the flow controllers must be checked. This is done by applying the nominal full scale voltage for each D/A converter to the controller, and checking the setpoint. The actual setpoint is then used by the software in the data acquisition and control program on the PC to determine a scale factor for each D/A converter.

The third calibration procedure is for the relative humidity instrument. A calibration curve for the relative humidity instrument may be determined *in situ* by placing an impermeable aluminum foil sample in the cell and varying the relative humidity of the gas flow in the top and bottom of the cell by means of the flow controllers. The resulting curves of measured relative humidity versus true relative humidity (set by the flow controllers) are used as calibration factors to correct the measured relative humidity for subsequent tests.

The pressure drop across the sample is monitored by means of an MKS Baratron Type 398 differential pressure transducer, with a Type 270B signal conditioner (MKS Instruments, Inc.). For measurement of pure diffusion, especially for materials such as fabrics, which may be quite permeable to convective flows, it is important to make sure that the pressure drop across the sample is zero, so that transport takes place only by pure diffusion. The pressure drop is continuously monitored and displayed, and is controlled by means of a valve restrictor on the outlet of one of the gas streams. For the permeable fabrics, this system also allows one to do testing under controlled conditions of a defined pressure drop across the sample, so that transport takes place by both diffusion and convection. This makes it possible to determine an air permeability value from the apparatus, in addition to the water vapor diffusion properties of the test sample.

Materials which have a constant mass transfer coefficient show a linear slope on plots of flux versus concentration difference across the sample. These kinds of materials do not change their transport properties as a function of water content or test conditions.

For materials which do not have a constant slope, the data points for a test series will not superimpose, but will form a set of curves for each test condition. We may still calculate a diffusion resistance for these materials, but now we have to evaluate the flux versus concentration difference curve at various points to derive our values for the material diffusion resistance, which will now be a function of the concentration of water in the material.

We define a total resistance to mass transfer as the simple addition of an intrinsic diffusion resistance due to the sample (R_f) and the diffusion resistance of the boundary air layers (R_{bl}):

$$\frac{\dot{m}}{A} = h_m(\Delta\bar{C}) = \frac{\Delta\bar{C}}{(R_f + R_{bl})} \quad (3.6)$$

$$R_f = \left[\frac{\Delta\bar{C}}{\left(\frac{\dot{m}}{A}\right)} \right] - R_{bl} \quad (3.7)$$

$\dot{m}$ = mass flux of water vapor across the sample (kg/s)

A = area of test sample (m^2)

$h_m = [1 / (R_f + R_{bl})]$ = mass transfer coefficient (m/s)

$\Delta\bar{C}$ = log mean concentration difference between top and bottom nitrogen streams (kg/m³)

R_f = intrinsic diffusion resistance of sample (s/m)

R_{bl} = diffusion resistance of boundary air layers (s/m)

The log mean concentration difference across the sample is appropriate since there is a significant change in the concentration of the gas stream both below and above the sample. In addition, the gas streams may not necessarily be in parallel flow, but may be run in counter flow to maintain a more constant concentration gradient across the sample. The log mean concentration difference [50] is defined as:

$$\Delta \bar{C} = \frac{\Delta C_a - \Delta C_b}{\ln(\Delta C_a / \Delta C_b)} \quad (3.8)$$

ΔC_a = concentration difference between the two gas streams at one end of the flow cell

ΔC_b = concentration difference between the two gas streams at the other end of the flow cell

For parallel flow, the concentration differences are between the top and bottom incoming flow at one end of the cell (ΔC_a), and the difference between the top and bottom outgoing flows at the other end of the cell (ΔC_b). For countercurrent flow, the concentration differences are between the incoming and outgoing flows at one end of the cell (ΔC_a), and the incoming and outgoing flows at the other end of the cell (ΔC_b).

It is useful to have a calibration or reference material to check the operation of the system. A microporous polytetrafluorethylene (PTFE) membrane has proven to be very useful for this purpose. A single layer of this membrane has a very small resistance to water vapor diffusion, but has very high resistance to convective flow, because of the very small pore sizes. The membrane may be layered to produce a material with a lower effective diffusivity. Since diffusion takes place only through the pore spaces of the membrane, the material has a very linear and reproducible plot of flux versus concentration difference. The plot of flux versus concentration difference may be used as a calibration curve for the apparatus, and may also be used to determine the boundary air layer resistance present in the test cell.

It is easiest to first look at the results for the PTFE membranes, shown in Figure 7.

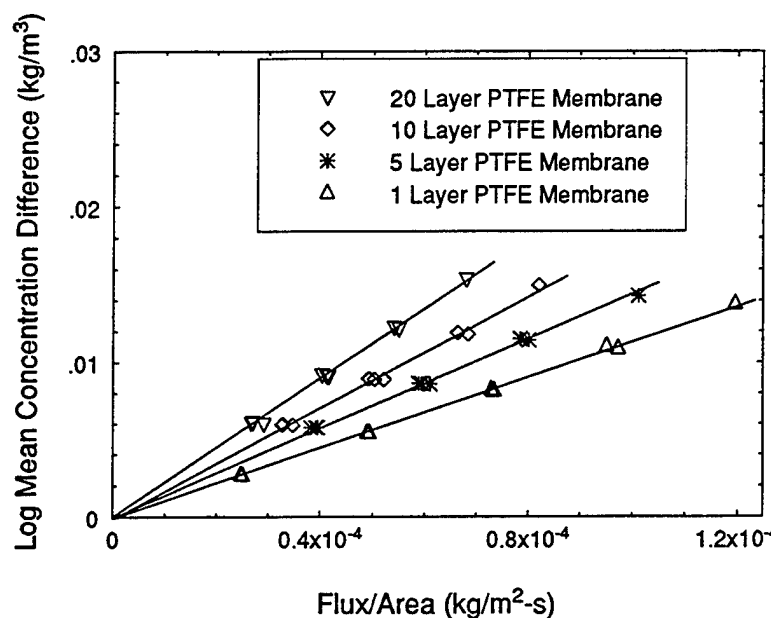


Figure 7. Results for Samples Made of Combined Layers of Microporous PTFE Membranes.

The results for the series of PTFE membranes show that the moisture permeation cell gives the expected type of plots. Since these materials are microporous, and transport takes places only through the interconnected air spaces of the membrane, we expect that the plot of the mass flux versus the concentration difference across the sample will be linear, which is readily apparent from Figure 7. We also note that these linear plots have all the test results from the nine different test conditions superimposed on the same constant slope line, which means that the diffusion resistance of each sample is constant.

We may use this test series of microporous PTFE membranes to derive an estimate of the boundary layer resistance on both the top and bottom of the sample. From our definition of resistance, we know that the resistance of the sample and the boundary air layers is equal to the slope of the line for each sample in Figure 7. We also know that for these types of materials, we can assume that the mass transfer resistance is additive; the resistance of 20 layers is twice the resistance of 10 layers. We may derive a value for the boundary air layer resistance from the relation:

$$R_{bl} = R_{total} - nR_{1-layer} \quad (3.9)$$

R_{bl} = boundary air layer resistance (s / m)

R_{total} = measured mass transfer resistance of sample (s / m)

n = number of teflon layers

$R_{1-layer}$ = calculated resistance of 1 PTFE layer (s / m)

From the relations given above, we find the boundary air layer resistance (R_{bl}) is in the range of 95 -100 s/m, and the resistance of a single layer of the PTFE membrane is in the range of 5-8 s/m. The boundary air layer resistance is fairly constant at a given set of flow conditions. We note that a single layer of the PTFE membrane will give practically the same value as the boundary air layer resistance, and thus serves as a convenient way to directly measure the boundary layer resistance present within the cell at other flow conditions (if we correct for the resistance of the single PTFE layer), and as a standard reference material to check the results generated by the cell.

3.2.2 DMPC Results for Various Clothing Materials

The test results for three different classes of clothing materials will illustrate various factors which must be taken into account when analyzing the results generated by this type of test. In the next section of this study, the results obtained by the DMPC are compared with the results obtained by two other tests in order to show the correlation between them.

The three classes of materials are 1) permeable fabrics, 2) microporous fabric-laminated membranes, 3) hydrophilic nonporous fabric-laminated membranes. The specific fabrics tested are described in Table 3-1.

Table 3-1. Test Fabrics

Material Type	Sample Identification	Materials and Reference
Air-Permeable Fabrics	HWBDU Fabric	Hot Weather Battle Dress Uniform (HWBDU) 100% cotton fabric [51]
	BDO Shell Fabric	Battle Dress Overgarment (BDO) Shell Fabric 50% cotton/ 50% nylon fabric [52]
	USMC System (Shell Fabric + Liner)	U.S. Marine Corps chemical protective garment system 100% cotton fabric over nonwoven laminated carbon-loaded polyester knit liner [53],[54],[55]
	BDO System (Shell Fabric + Liner)	Battle Dress Overgarment (BDO) chemical protective garment system 50% cotton/50% nylon fabric over nylon tricot laminated carbon-loaded polyurethane foam liner [52], [57], [58]
Microporous Laminated Membranes	PTFE Membrane (1 layer)	Microporous polytetrafluorethylene (PTFE) membrane
	Gore Tex III Membrane Laminate	Gore Tex III membrane laminated between Taslan nylon shell fabric and nylon tricot knit liner [59]
	Repel Membrane Laminate	Repel membrane laminated between Nomex shell fabric and knit liner
Hydrophilic Laminated Membranes	Azekura Membrane Laminate	Hydrophilic membrane laminated to fabric on one side
	Gore Tex II Membrane Laminate	Gore Tex II membrane laminated between Taslan nylon shell fabric and nylon tricot knit liner [59]

We would expect in general that under conditions of pure diffusion (i.e. no pressure drop or convective flow across the sample) the permeable fabrics and the microporous fabric-laminated membranes will have a linear slope on a plot of vapor mass flux versus vapor concentration difference across the sample. Also, that hydrophilic nonporous fabric-laminated membranes would not be linear but would show lower resistances at test conditions which produce a high water content in the membrane, and higher resistances at test conditions where less water is present in the membrane.

Using equation (3.7) we may calculate the diffusion resistance of each of these materials. If we subtract the boundary layer resistance (100 s/m) from the total resistance measured in this test, we obtain the intrinsic mass transfer resistance of each sample.

We can show the results of the testing for this set of materials in terms of an average relative humidity at the membrane, an approach that is often used for materials which exhibit concentration-dependent permeation behavior [60,61]. By doing this, we assume the average of the relative humidities of the two incoming gas streams could be related back to the water content of the sample estimated from a water vapor isotherm. This definition neglects the influence of the resistance of the boundary air layer, which will further decrease the concentrations at the surfaces of the hydrophilic materials, and it neglects the variation in vapor concentration along the sample. A log mean average concentration/water content in the sample would be a more appropriate factor to use, but the average relative humidity method will be sufficient to illustrate the general trend of material behavior. A plot of measured intrinsic resistance as a function of average relative humidity is shown in Figure 8.

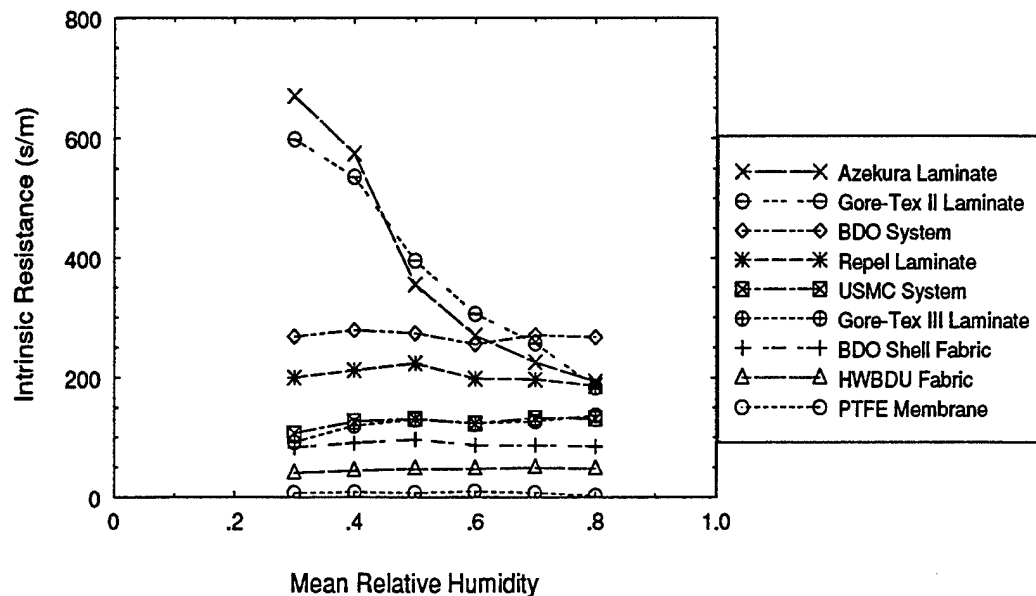


Figure 8. Variation in intrinsic log mean diffusion resistance as a function of the mean relative humidity on the two sides of the test sample.

As expected, the fabrics and microporous membrane laminates show a fairly constant mass transfer resistance, which is relatively independent of the test conditions. In contrast to the relatively simple behavior of the fabrics and the microporous membrane laminates, the two hydrophilic laminated membranes have a great variation in their measured mass transfer resistance depending on the test conditions used.

We can see that the properties of the hydrophilic membranes are much more complicated than the other materials tested. The primary barrier to water vapor diffusion in these materials is the solid polymer layer through which the water must diffuse. The permeability of water in these materials is highly concentration-dependent, and makes the analysis of the test results generated for these materials much more complicated.

We see that there is a clear trend of decreasing resistance as the average relative humidity (equivalent to average water concentration) in the hydrophilic membrane laminates increases. As the hydrophilic materials approach saturation, they become comparable to some of the low resistance materials such as the fabrics. There also seem to be clear differences in the degree of the dependence of material properties on the average relative humidity. We believe that presenting data in this way, as a function of average relative humidity, gives a clearer picture of the differences in materials which show up under actual use conditions.

3.2.3 Comparison of DMPC With Two Other Vapor Permeability Test Methods

Results generated by the dynamic moisture permeation cell may be compared with results generated by other methods used for measuring the water vapor diffusion properties of materials. Two well-accepted standard methods are the ASTM Test Method for Water Vapor Transmission of Materials (E 96) [38], and the International Standards Organization Test Method ISO 11092, Measurement of Thermal and Water-Vapour Resistance Under Steady-State Conditions (Sweating Guarded Hot Plate Test) [39]. The set of materials described in Table 3-1 were tested by all three methods, and a correlation between the test methods was determined. For the correlation plots only the fabrics and the microporous membrane laminates are shown, even though the hydrophilic membrane laminates were also tested. The concentration-dependent properties of the hydrophilic materials are affected much more by the different temperatures and humidity gradients present in the three tests, and unnecessarily complicate the direct test-to-test comparison.

ISO 11092 (Sweating Guarded Hot Plate)

The ISO 11092 test method is briefly described below, further details of the test method and some typical results may be found in Reference [62].

The porous guarded hot plate is saturated with water so that its surface is completely wet. A thin saturated cellophane film placed over the plate prevents liquid water from wicking into the fabric, yet allows water to freely permeate through the film and evaporate from the surface. The resistance of the film is so small that it behaves like the surface of liquid water, and the vapor pressure is equal to saturation vapor pressure. Since the ISO 11092 test conditions call for no temperature difference between the plate surface and the ambient air, the power required to maintain the plate surface at a given temperature is directly related to the rate at which water evaporates from the surface of the plate and diffuses through the test material.

The equation used for calculating the water vapor resistance is:

$$R_{etotal} = \frac{A(p_s - \phi p_a)}{E} \quad (3.10)$$

R_{etotal} = water vapor resistance of the material plus the boundary air layer resistance ($\text{m}^2 \cdot \text{Pa} / \text{Watt}$)

A = guarded hot plate measurement area (m^2)

p_s = saturated water vapor pressure at the plate surface (Pa) at the plate temperature

p_a = saturated water vapor pressure of ambient air (Pa) at ambient air temperature

E = power required to maintain a constant plate surface temperature (Watt)

ϕ = relative humidity of the ambient air (fractional)

The intrinsic water vapor resistance R_{et} of the fabric may be determined by subtracting out the value of the water vapor resistance measured for the bare plate, R_{eto} :

$$R_{et} = R_{etotal} - R_{eto} \quad (3.11)$$

The intrinsic water vapor resistance is equivalent to the intrinsic mass transfer resistance measured in the DMPC apparatus. We may also convert the R_{et} value obtained from the sweating guarded hot plate to the intrinsic mass transfer units of (s/m) used in the DMPC test.

$$R_f = R_{et} \left(\frac{M_w \Delta H_{vap}}{RT} \right) \quad (3.12)$$

R_f = intrinsic mass transfer resistance (s / m)

R_{et} = intrinsic water vapor resistance ($\text{m}^2 \cdot \text{Pa} / \text{Watt}$)

ΔH_{vap} = enthalpy of vaporization for water ($2.42 \times 10^6 \text{ J/kg @ } 35^\circ \text{C}$)

R = universal gas constant ($8314.5 \text{ N} \cdot \text{m} / \text{Kgmole} \cdot \text{K}$)

T = temperature (K)

We can make another small correction for the difference in test temperatures for the fabrics and microporous laminates by using the ratio of the diffusion coefficients of water vapor in air at the two test temperatures. This particular temperature correction is only valid for the materials in which water vapor transport occurs through the air spaces of the material. A similar temperature correction for the hydrophilic solid polymer membranes would be possible if information is available on the variation of their diffusion coefficients with temperature.

$$R_f(@T_1) = R_f(@T_2) \left(\frac{D_a(@T_1)}{D_a(@T_2)} \right) \quad (3.13)$$

$$D_a = 2.23 \times 10^{-5} \left[\frac{(T + 273.15)}{273.15} \right]^{1.75} = \text{diffusion coefficient of water vapor in air (m}^2 / \text{s)}$$

A correlation between the DMPC and ISO 11092 is shown in Figure 9 for the fabrics and microporous laminated membranes given in Table 3-1. A schematic of the ISO 11092 test apparatus and conditions is also shown in Figure 9. There is an excellent correlation between the tests, especially when one considers that the test methods are very different, in that the sweating hot plate is a calorimetric method, and the DMPC is a direct measurement of concentration differences.

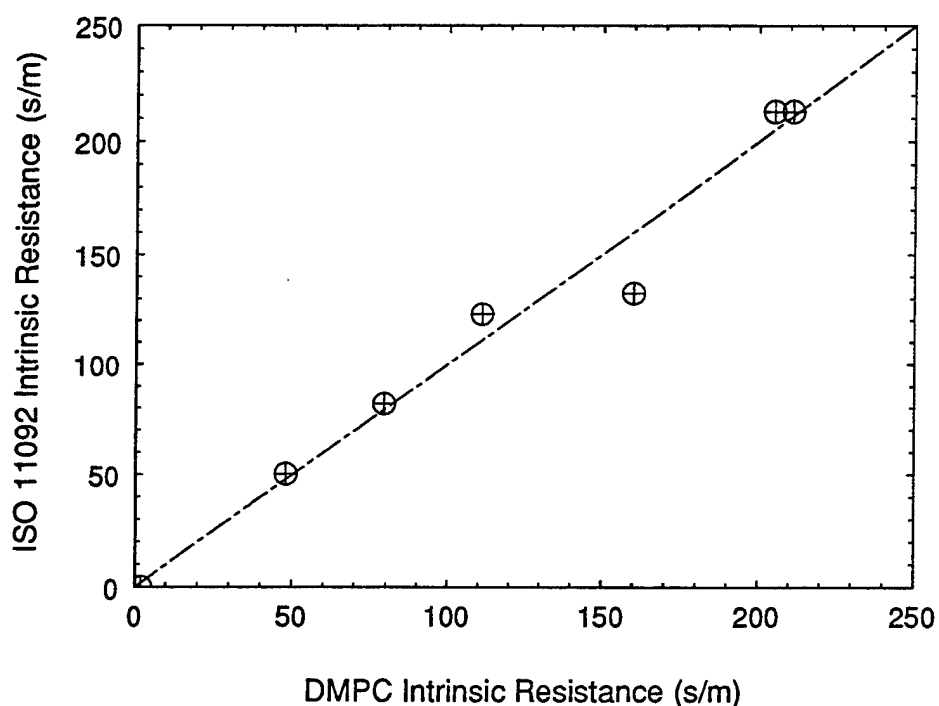
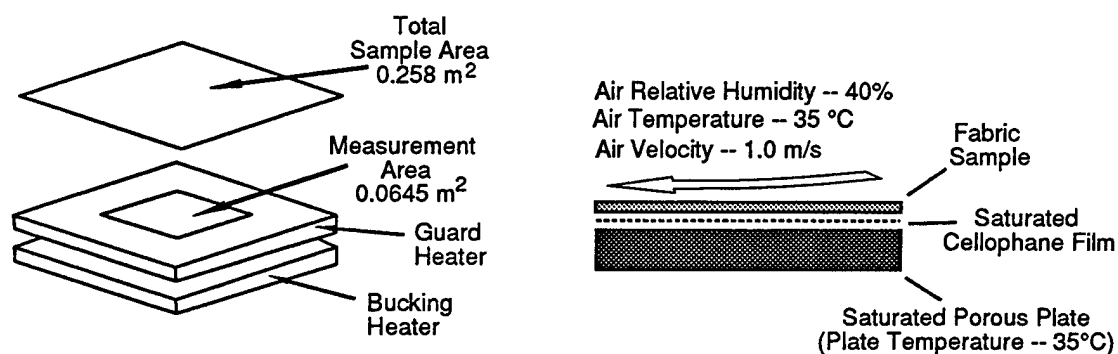


Figure 9. Correlation between DMPC and ISO 11092 for several fabrics and microporous membrane laminates.

ASTM E 96 (Cup Method)

ASTM Method E 96 "Standard Test Methods for Water Vapor Transmission of Materials" is another test technique which is widely used to evaluate the water vapor transmission properties of woven textiles, membranes, and membrane/fabric laminates. For this set of comparison tests, a modified Procedure BW of ASTM E 96 was used, which is an inverted cup test which eliminates the air space between the sample and the surface of the water. The modified E 96 BW test was conducted at standard test conditions of 23°C and 50% relative humidity (wet bulb temperature of 16.4°C).

Several air flow conditions were tried to see if they had any effect on the results. Two air flow conditions tangential to the fabric surface of 0.5 m/s and 3.1 m/s, and one air flow condition of 6.5 m/s perpendicular (face-on) to the fabric surface.

To prevent the liquid water from wicking into the fabric in these inverted tests, a hydrophobic microporous polytetrafluoroethylene (PTFE) membrane was sealed over the cup, and the test sample was placed over the membrane. This approach of using a membrane which has a minimal resistance to water vapor transfer, but which is a barrier to liquid penetration, is often used to produce a saturation condition for one side of fabrics and textiles [60].

The inverted cup test minimizes the possibility of air penetrating the fabric and skewing the results whereas in an upright cup test (e.g. ASTM E 96, Procedure B), the air flow over the cup can easily circulate through a highly air-permeable fabric and cause an increased evaporation rate due to the disruption of the still air layer underneath the fabric. Thus for fabric with high air permeability, upright cup results in ASTM E 96 B are highly dependent on the orientation of the external flow, and the geometry and size of the cup and associated air spaces between the water surface and the fabric. A major advantage of the DMPC over the ASTM cup tests, and the sweating guarded hot plate, is that fabrics can be tested under conditions of combined diffusion and convection in a way that lets us separate the two effects.

When the modified inverted cups are weighed periodically, the slope of the line of mass loss per area versus time will give the flux through the material in terms of mass/area-time ($\text{kg/m}^2\text{-sec}$). We may also define a water vapor concentration gradient across the sample as the difference between the saturation conditions on the water side of the sample, and the water vapor concentration of the environment (50% relative humidity at 23°C). A mass transfer resistance may be calculated for the material based on equation (3.7). We know from previous testing that the single layer of PTFE membrane has a very low water vapor resistance (about 5-8 s/m), so without too much error, we can take the calculated resistance for the PTFE membrane as the boundary layer resistance due to the air flow over the cup. For each of the three air flow test conditions (0.5 and 3.1 m/s tangential, 6.5 m/s face-on), we subtract the boundary layer resistance using the PTFE test from the total resistance calculated for each material, to obtain an intrinsic water vapor resistance value which should be independent of the air flow conditions, and which should be directly comparable to the resistance measured in the DMPC.

The test comparison was run for the same set of fabrics and microporous membrane laminates shown previously in Table 3-1. In this case, however, we were able to use the same pieces of material for both the DMPC and the modified ASTM E 96 Procedure BW tests, with the series of ASTM E 96 tests being run prior to the DMPC tests.

The calculated intrinsic resistance values for the three air flow conditions are plotted against the DMPC results in Figure 10. A schematic of the modified inverted cup test is also shown in Figure 10.

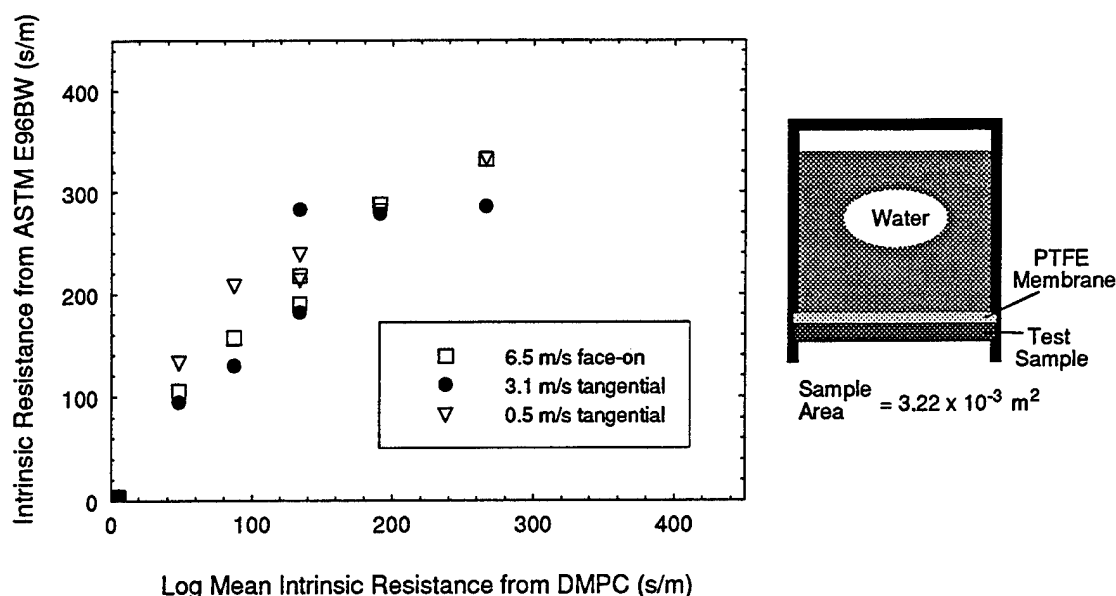


Figure 10. Correlation of DMPC results with modified ASTM E 96 BW inverted cup tests, for three air flow conditions.

We see again a fairly good correlation between the results and rankings generated between the DMPC and the modified ASTM E 96 BW inverted cup test, which increases our confidence about the reliability of the rankings of materials generated with the DMPC, especially considering that we have previously shown a very good correlation with the sweating guarded hot plate test method.

3.2.4 Effect of Parallel versus Countercurrent Flows in the DMPC

The fact that we use a log-mean concentration difference in analyzing the results for the DMPC means that we recognize the direction of flow has an influence on the concentration profile down the length of the test sample. For parallel flow, we would expect that the concentration difference between the two sides of the sample becomes smaller at the downstream end of the sample. To maintain a more constant concentration gradient down the length of the sample, we would expect that a counterflow arrangement would promote higher fluxes (and possibly lower boundary layer resistances). The two flow situations are illustrated in Figure 11.

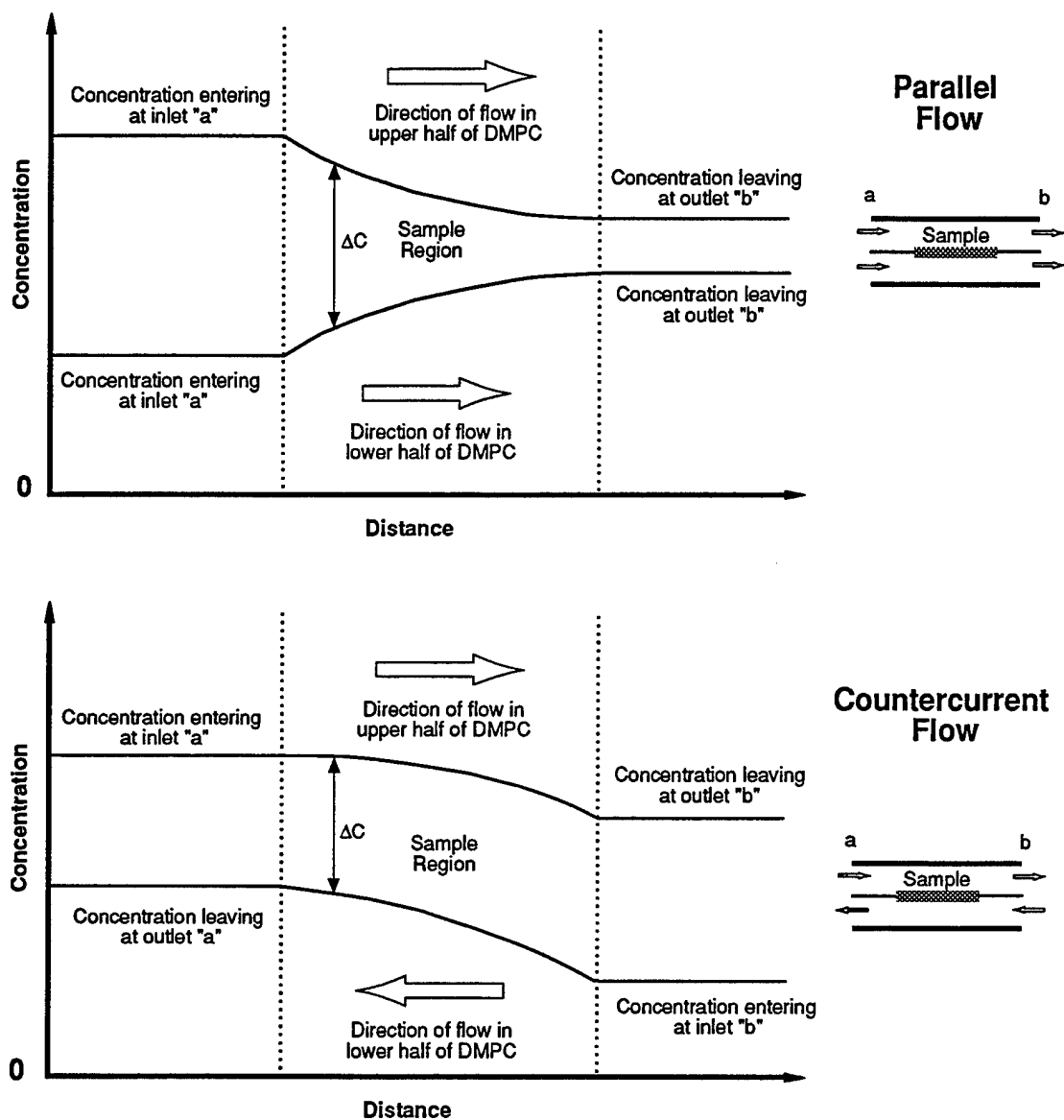


Figure 11. Effect of Parallel or Countercurrent Flow Direction on Concentration Gradient Across Test Sample.

We were curious to see if the direction of flow has a large effect on the results obtained with DMPC. We used two materials to look at this effect. The 1-layer PTFE membrane was tested in both parallel and counterflow to determine the relative effect on the boundary layers. We also tested the Gore Tex III sample in both the parallel and counterflow situations. The results are shown in Figure 12.

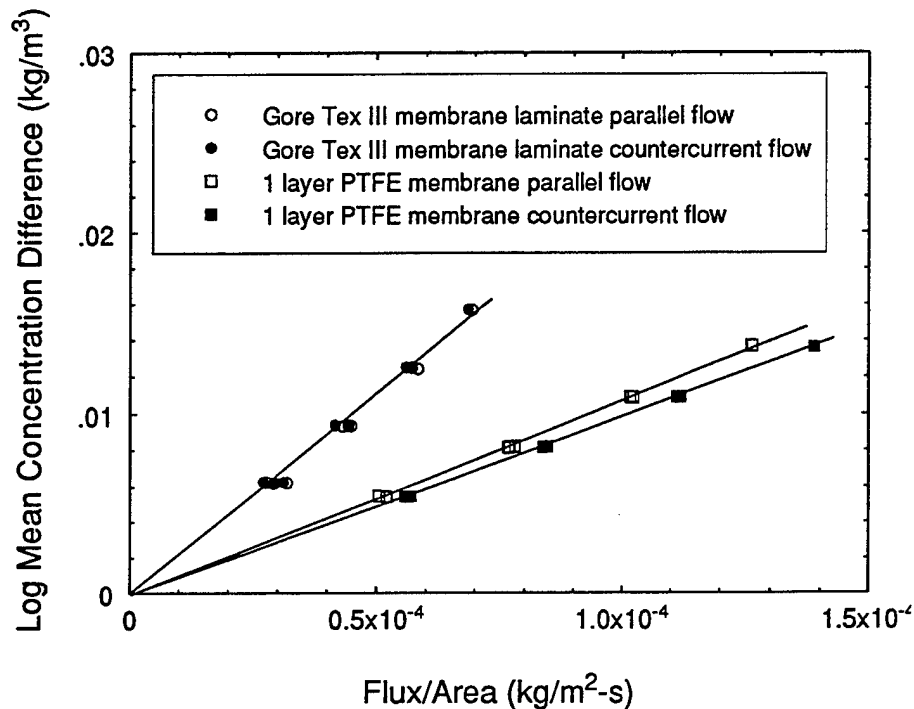


Figure 12. Testing of Two Materials in Both Parallel and Counterflow Arrangements in the DMPC.

We do not observe any difference in the measured properties of the Gore Tex III membrane laminate for the two flow situations. We do see some differences in the measurement for the PTFE membrane, which since it has a resistance of 6 to 8 s/m, is essentially a measurement of the boundary layer resistance. From the slope of the lines in Figure 12, the parallel flow resistance is about 105 s/m and the counterflow resistance is about 95 s/m. The countercurrent flow resistance is lower, which we would expect, but the difference between the two flow situations is not large enough to be discernible for materials with lower water vapor transmission rates. Although all the testing results given in this report are for parallel flow, it may be better practice to conduct testing using countercurrent flow, to maximize water vapor flux, and to maintain a more nearly constant concentration difference across the test sample. Both methods would be useful when studying the hydrophilic materials, since for the same test conditions of relative humidity in the upper and lower halves of the DMPC, the mean relative humidity (average of both sides) would be quite different down the length of the sample for the two different flow situations.

3.2.5 Use of DMPC for Transient Diffusion Studies

The correlations we have shown confirm the validity of using the DMPC to determine the steady-state water vapor diffusion properties of materials. However, in contrast to the other tests, the automated DMPC apparatus can be used to conduct testing of materials under non-steady-state conditions, such as a change in relative humidity, temperature, or pressure difference across the sample. The ability of the DMPC to perform an automated sequence of user-defined test conditions is useful for verifying the numerical solution of the governing equations used to describe the time-dependent transport of water vapor through hygroscopic materials. In these transient situations, the variable properties of the material become very important, along with factors such as the sorption rate at which a fiber takes up or releases water vapor to the atmosphere and the coupling of the differential equations describing the transport of energy and mass through the material.

To illustrate the use of the DMPC for characterizing transient diffusion properties, we again select a group of porous textile materials which have a range of properties. This group of materials includes only permeable woven fabrics, and is the set of materials which will be used to illustrate experimental and numerical results throughout the rest of this report. Table 3-2 lists the fabrics, along with pertinent references if available. Physical properties of these fabrics are listed in Appendix A.

Table 3-2. Woven Test Fabrics

Sample Identification	Materials and Reference
Wool	100% Wool Twill Fabric [25]
Cotton	Hot Weather Battle Dress Uniform (HWBDU) 100% cotton fabric [63]
Silk	Woven Silk Fabric, Plain Weave
Wool / Polyester	40% Wool / 60% Polyester Fabric [64]
Nylon / Cotton	Battle Dress Overgarment (BDO) Shell Fabric 50% cotton/ 50% nylon fabric [52]
Nylon	100% Nylon Fabric, Basket Weave
Polyester	100% Polyester Fabric

We first illustrate the use of the DMPC to obtain transient results for fabrics subjected to step changes in relative humidity. In these experiments, thermocouples are sandwiched between two layers of fabric, to record temperature changes as the fabric absorbs or desorbs water vapor from the gas stream flowing on the two sides of the DMPC. Three thermocouples are used as shown in Figure 13. Thermocouple diameter was 1.27×10^{-4} m (0.005 inch), with a response time listed by the manufacturer as 0.04 s in water and 1 s in still air. Smaller thermocouples with a diameter of 2.54×10^{-5} m (0.001 inch) were used initially, but proved to be quite fragile and easily damaged. The temperature changes recorded with both sizes of thermocouple were identical, so we believe that errors due to conduction and the heat capacity of the thermocouple wire are minimal.

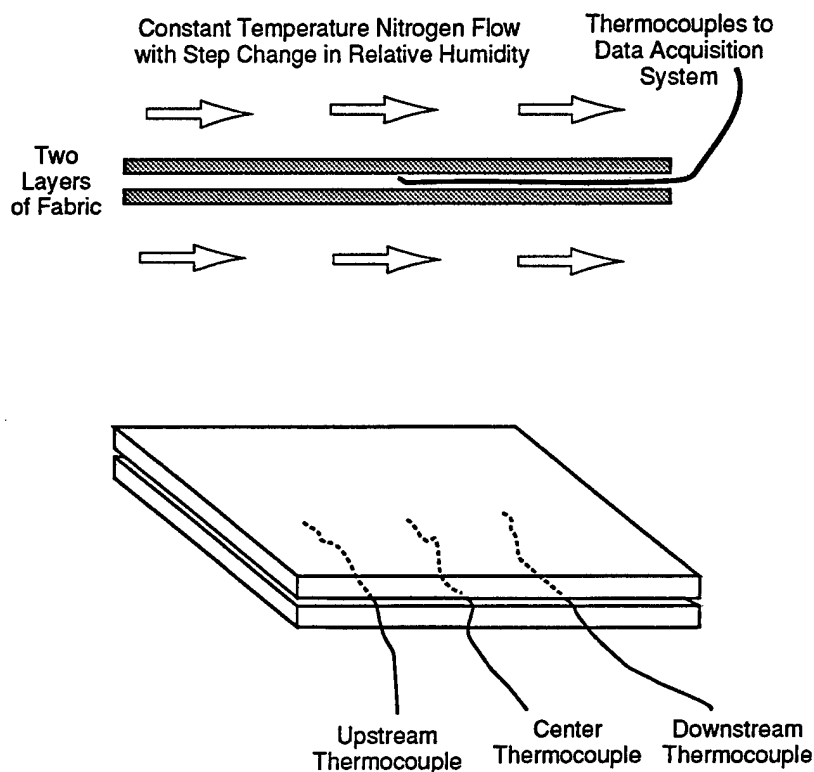


Figure 13. Instrumented test fabric in DMPC to record temperature changes of hygroscopic fabrics.

Nine setpoints were used to examine the coupled diffusion of heat and mass in these hygroscopic porous textile layers, as given in Table 3-3. In this series of setpoints, the pressure drop across the sample is set to zero, so that there is no convective flow across the sample, and transport takes place only by diffusion driven by concentration differences.

Table 3-3. Nine Setpoints for Transient Diffusion Tests.

Setpoint	Relative Humidity for Top Gas Flow	Relative Humidity for Bottom Gas Flow
1	0.0	0.0
2	1.0	1.0
3	0.0	0.0
4	0.6	0.0
5	0.8	0.0
6	1.0	0.0
7	1.0	0.2
8	1.0	0.4
9	1.0	0.6

The first three setpoints are used to look at the situation when a completely dry fabric, equilibrated at 0% relative humidity, is suddenly exposed to a relative humidity of 100% on the two sides of the fabric facing the gas flows on the top and bottom of the DMPC. For hygroscopic fabrics, the textile fibers will absorb water vapor from the gas flow, and release the heat of sorption, which results in a rise in temperature of the fabric, as recorded by the three thermocouples. When the relative humidity is suddenly changed back to 0%, the water is desorbed from the textile fibers, and the temperature drops, due to the change of phase of the water as it leaves its sorbed state in the textile fiber and vaporizes. The rate at which the temperature rises and falls is related to the mass transport and thermal transport properties of the textile material, and the gas flows, and serve as a convenient experimental verification of the numerical prediction of transient behavior, which is the subject of Chapters 5 and 6 of this report.

The next six setpoints are a sequence of humidity gradients across the sample, where there is a net flux of water vapor from one side of the cell to another. Setpoint #4, where the relative humidity is suddenly changed from 0% to 60%, provides another set of experimental results which are particularly convenient for verifying the numerical code, since both temperature measurements, and the measurement of relative humidity of the gas flow as a function of time are available.

An example of the full sequence of nine setpoints, for the cotton fabric, shown as a function of time, is given in Figure 14. For clarity, only one thermocouple record is shown on Figure 14.

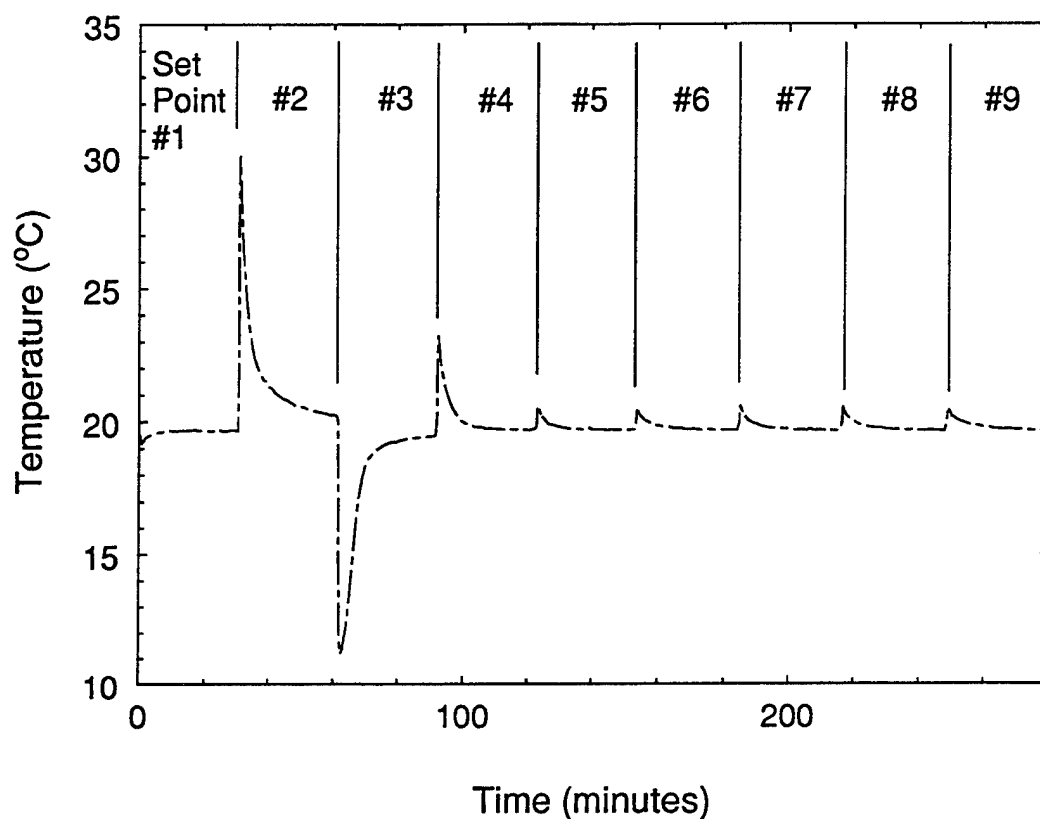


Figure 14. Temperature changes of two layers of cotton fabric subjected to step changes in relative humidity, at a constant gas flow temperature of 20°C.

The large temperature changes due to sorption/desorption are particularly evident for setpoints #2 and #3.

The shape of these transient temperature curves are a complex function of the velocity of the gas flows on the two sides of the fabric samples, which influences the external heat and mass transfer coefficients and the thermal and mass transport properties of the textile layers. An example of the temperature transients for all seven test fabrics, for setpoint #2 (step change for 0.0 to 1.0 relative humidity), is shown in Figure 15. Here only the first four minutes are shown, to make it easier to distinguish the response of the different materials shown. The peak temperatures for each of the materials is also shown on the plot.

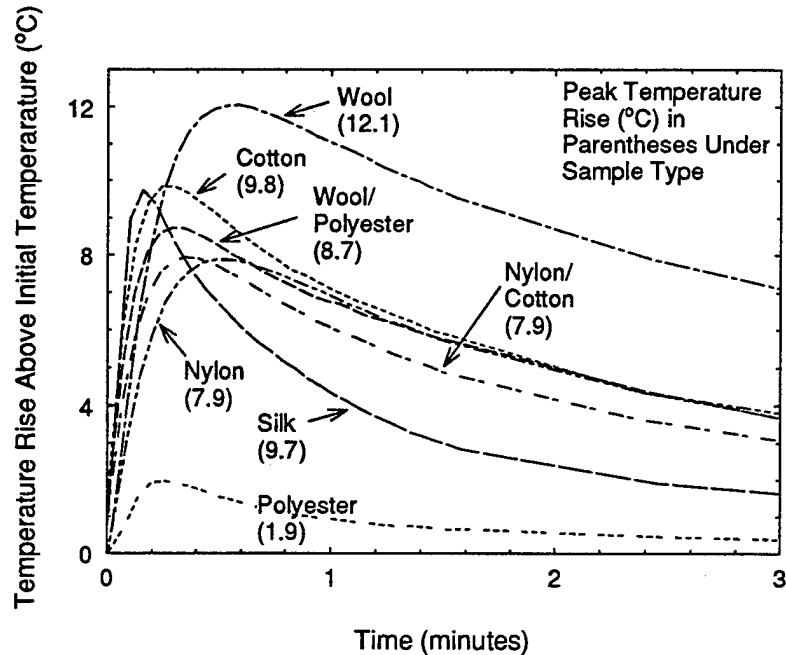


Figure 15. Temperature changes due to water vapor sorption for seven fabrics during step change in relative humidity from 0.0 to 1.0.

Figure 15 can be misleading, in that it shows a single well-defined experimental temperature transient for each material. Since there are concentration and temperature gradients down the length of the test sample, due to the influence of the developing thermal and concentration boundary layers, the temperatures of the upstream, center, and downstream thermocouples (refer to Figure 9) are slightly different, as shown in Figure 16.

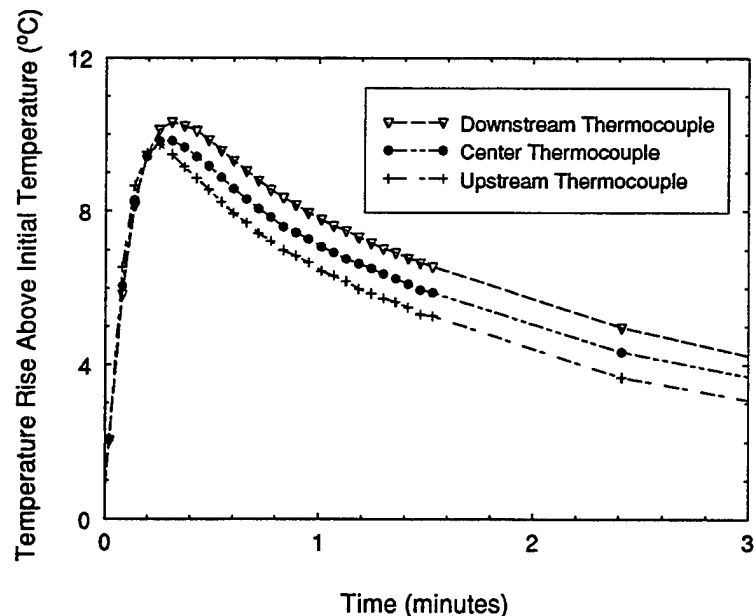


Figure 16. Temperature changes for the three thermocouples, for the cotton fabric, due to water vapor sorption during step change in relative humidity from 0.0 to 1.0.

There is also significant variability in the temperature traces obtained for different pieces of the same fabric, at different times. Figure 17 shows various results obtained for different samples of the cotton fabric, again for water vapor sorption, under a step change in relative humidity from 0.0 to 1.0, at 20°C.

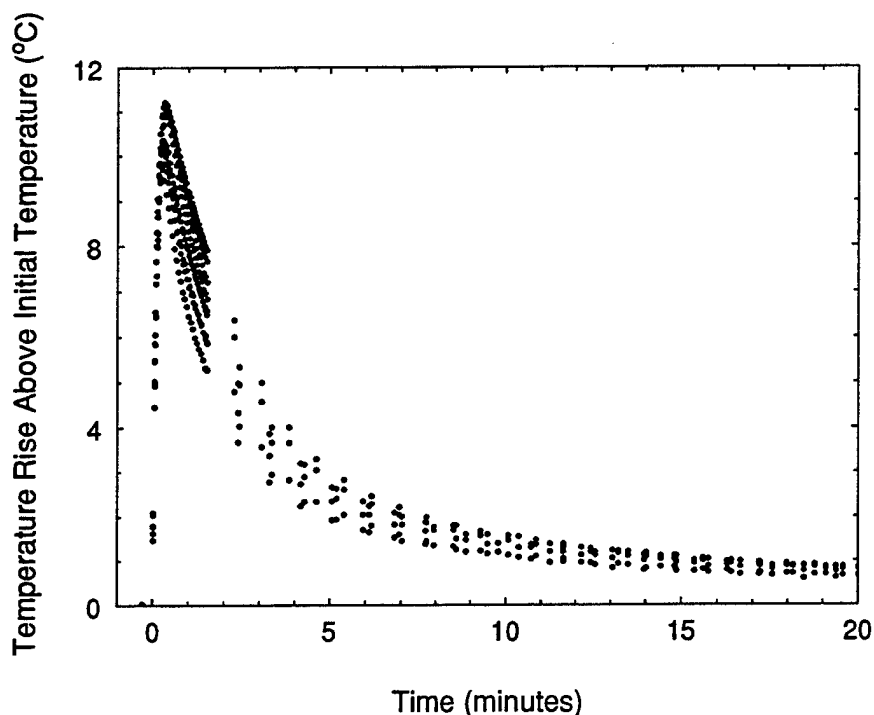


Figure 17. Variability in measured temperature changes for the cotton fabric, due to water vapor sorption during step change in relative humidity from 0.0 to 1.0.

Another transient measurement, which is useful for verifying numerical predictions of transient diffusion behavior, is the experimental measurement of the water vapor concentration change on one side of the fabric as the relative humidity on the other side of the fabric is changed. Setpoint #4 is useful for this purpose, since there is a step change from 0% relative humidity on both sides of the sample, to 60% relative humidity one side. Following this step change in relative humidity, the rate of change of the measured relative humidity on the "dry side" of the cell is greatly affected by the sorption kinetics of a hygroscopic material. Figure 18 shows the change in relative humidity on the bottom side of the DMPC when the humidity on the top side is suddenly change to 0.6. Figure 18 has had the relative humidity normalized by the final equilibrium relative humidity for setpoint #4. Also shown in this plot is the value for two layers of the PTFE membrane, which is completely nonhygroscopic, and serves to indicate the response time of the relative humidity sensors to a step change.

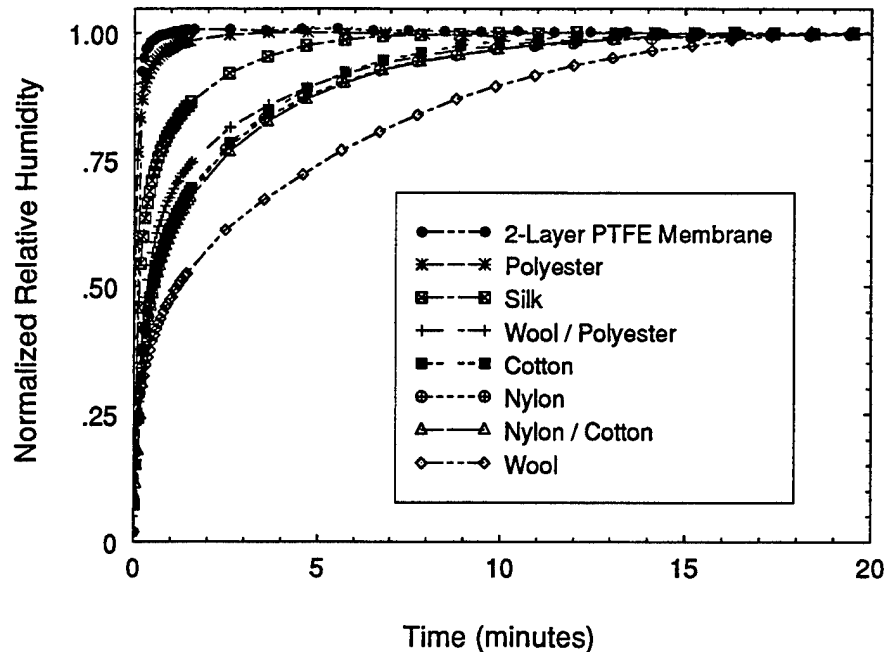


Figure 18. Relative humidity normalized by final equilibrium value, during step change from 0.0 to 0.6, at constant temperature of 20°C.

Further reference to these transient measurements will be made in Chapters 5 and 6, where these experimental results will be compared with the results of the numerical code which solves the partial differential equations describing the coupled diffusion of heat and mass through hygroscopic porous textiles.

3.2.6 Use of DMPC for Convection/Diffusion Studies

The DMPC may also be run with a specified pressure drop across the sample so that transport takes place across the sample by both diffusion (driven by concentration differences) and convection (driven by gas phase pressure differences). Experiments of this type are useful for verifying the numerical code presented in Chapter 6, which simulates all aspects of the transport processes taking place in the DMPC. The simplest experiment to run is shown in Figure 19. Gas enters the DMPC at a relative humidity of 1.0 on the top portion of the cell, and 0.0 on the bottom of the cell. A series of setpoints is run where the flow outlet one side of the cell is gradually restricted. This causes the pressure in one side of the cell to be higher than in the other, causing convective flow across the sample, in addition to the diffusion flux taking place due to the concentration gradients. In this particular experimental series, the tests were again conducted on both one and two layers of the fabrics given in Table 3-2. The two layer tests were also instrumented with the three thermocouples as shown in Figure 13.

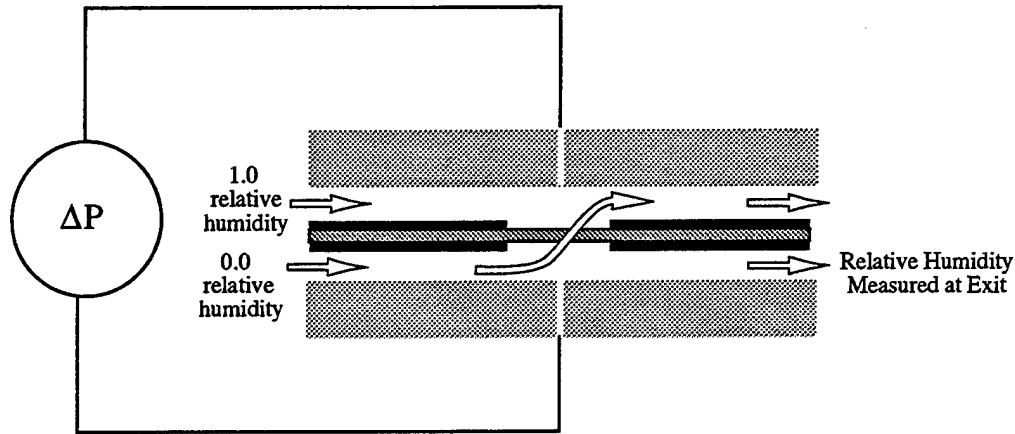


Figure 19. Convection/diffusion experiment in the DMPC: example shows bottom outlet flow restricted to force convective flow across sample, which opposes diffusive flux of vapor.

The measurement of relative humidity at the bottom outlet of the cell, as a function of pressure drop across the sample, is a experimental curve which is convenient to use for comparison to the numerical flow simulations presented in Chapter 6. An example of a typical experimental curve, for one layer of the cotton and polyester fabric, is shown below in Figure 20.

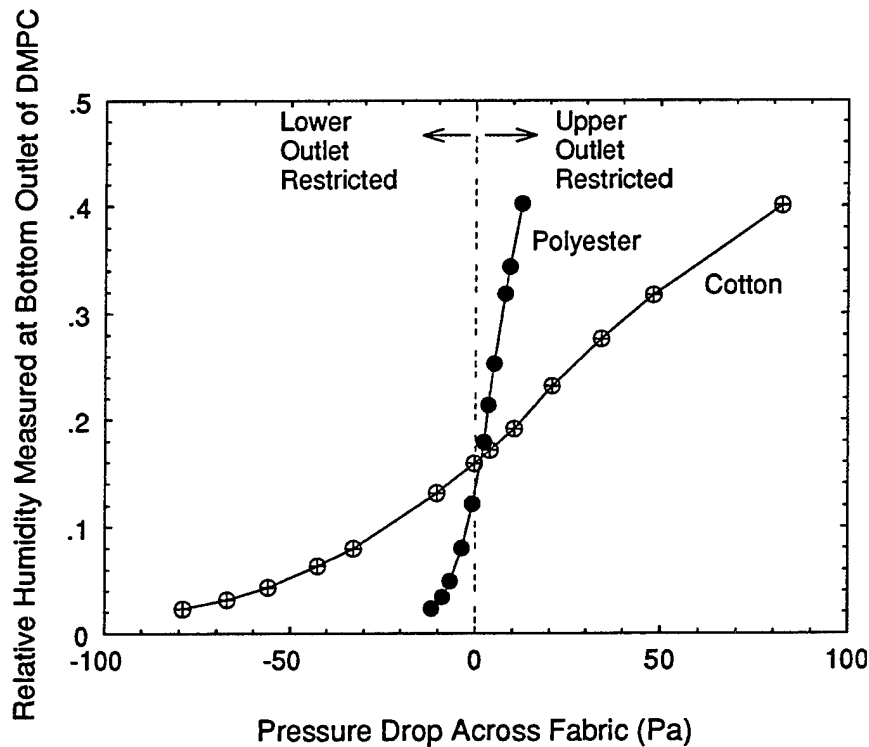


Figure 20. Experimental curve of measured relative humidity at outlet of DMPC, as pressure drop across the fabric is varied.

Figure 20 is particularly interesting because the cotton fabric and the polyester fabric have nearly identical diffusion resistances. They differ greatly in their air permeability properties (covered in the next section), so that when they are both tested under conditions of the same pressure drop, there is much more convective flow through the polyester fabric than through the cotton fabric.

3.3 Automated Convective Gas Flow Test Method

3.3.1 Introduction

Fabrics which absorb water vapor from the atmosphere (such as cotton or wool) experience fiber swelling which tends to close off the pores in the fabric and increase the resistance to convective flow through the material.

This effect is important for its influence on the ability of the fabric layer to pass heat and water vapor through its structure, since fiber swelling reduces the free air volume within the fabric and affects both the convective and diffusive transport of energy and mass.

Changes in fabric permeability as a function of relative humidity are also important since this will influence the transport of chemical agents in vapor or aerosol form. If testing is conducted at only one relative humidity, we may have quite different results than if we performed those tests at another relative humidity. For example, we will show later that it is easy to find fabrics which double their resistance to convective flow due simply to changes in relative humidity. A set of fabrics evaluated for aerosol filtration properties at one condition of relative humidity may have different rankings at another relative humidity test condition, which could possibly influence our decision on which fabric provides the best protection for the soldier.

Changes in fabric permeability have as a function of relative humidity have been studied by Wehner, Miller, and Rebenfeld [65], who found that there can be large changes in convective gas flow transport properties of woven and nonwoven textile materials due to changes in fabric structure caused by fiber swelling. The experimental methods we developed are similar to those used by Wehner, Miller, and Rebenfeld to characterize this humidity-dependent permeability.

Most studies of gas and liquid flow through porous materials have focused on correlation of structural properties, such as gas phase volume fraction, and fiber diameter, with pressure drop - flow rate data [66-70].

There are many definitions of the permeability or the flow resistance. Usually, the permeability is given as a definition from Darcy's Law such as [71]:

$$v = \frac{-k_D}{\mu} \frac{\Delta p}{\Delta x} \quad (3.14)$$

v = apparent gas flow velocity (m/s)

k_D = permeability constant (m^2)

μ = gas viscosity (17.85×10^{-6} kg/m-s for N_2)

Δp = pressure drop across sample (N/m^2 or Pa)

Δx = thickness (m)

For low velocity flows, where the apparent Reynolds number (based on nominal particle diameter or pore sizes) remains much less than about 10, a plot of pressure drop versus volumetric flow rate or velocity will give a constant value for the permeability constant k_p . At higher flow rates, where inertial effects begin to compete with viscous flow effects, pressure drop - flow rate plots will begin to deviate from linearity, and inertial effects need to be considered. Previous work on air penetration through clothing systems has shown that air pressure differences across textile layers, due to factors such as wind or body movement, are usually less than 100 Pa [72-75]. For the testing presented in this chapter, flow rates and pressure drops are low enough so that inertial effects are not readily apparent in the experimental results. Inertial effects would need to be considered for testing at higher flow rates.

3.3.2 Test Method

The automated gas permeability test we developed is shown in Figure 21. It is very similar to the DMPC test. The major differences are that only two flow controllers and one humidity meter are required.

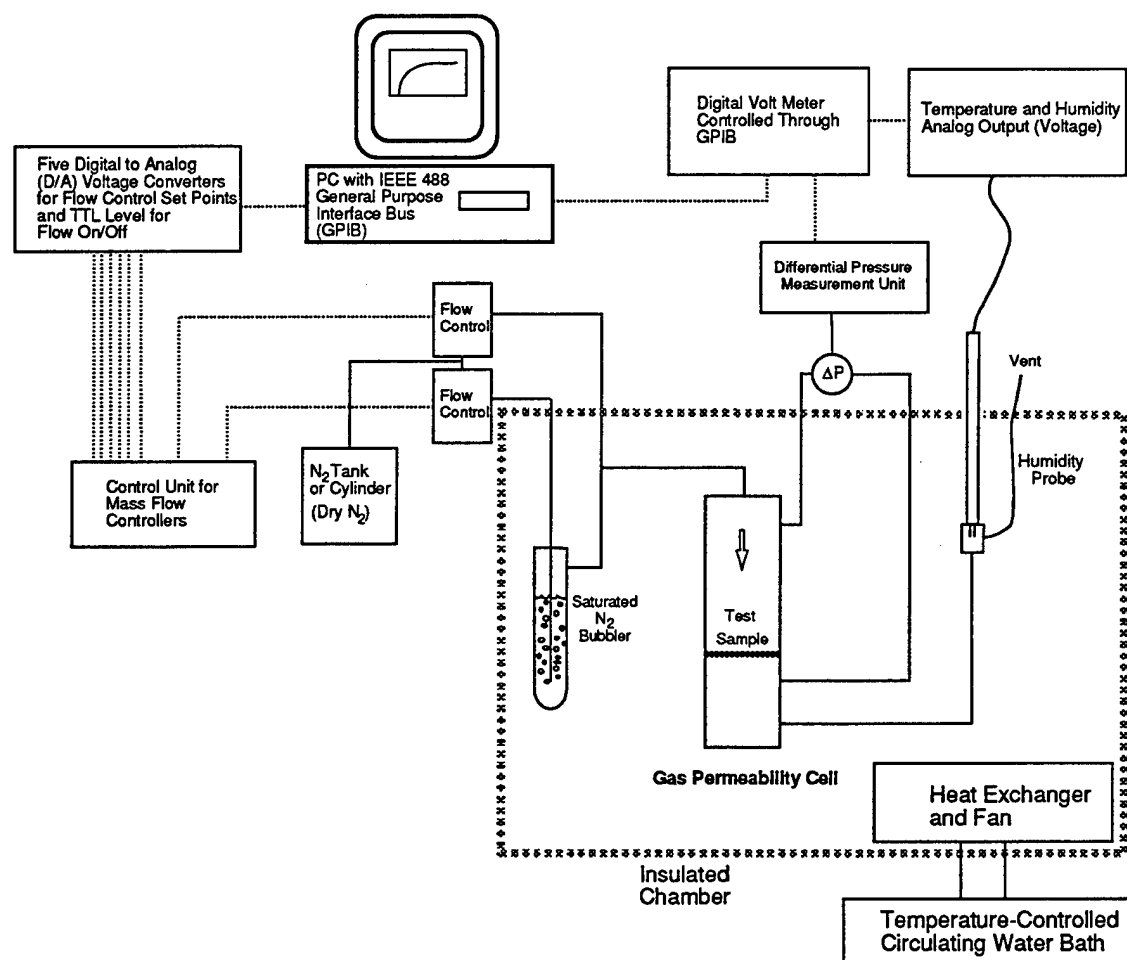


Figure 21. Automated Gas Permeability Test.

Nitrogen streams consisting of a mixture of dry nitrogen and water-saturated nitrogen flow into the gas permeability cell. The relative humidity of these streams is varied by controlling the proportion of the saturated and the dry components. By varying the flow rate, and measuring the pressure drop across the test sample, we may obtain a relationship for volumetric or mass flow rate through the sample as a function of pressure drop across the sample, which is related to the Darcy permeability k_D .

The mass flow rates are controlled by MKS model 1259C mass flow controllers, with a Model 247C 4-Channel Readout (MKS Instruments, Inc.). At constant mass flow, the true volumetric flow rate will vary with temperature; the flow rate set by the MKS controllers is indicated in terms of volumetric flow rates at standard conditions of 0°C and atmospheric pressure (1.01325×10^5 Pa). The actual volumetric flow rate at different temperatures may be found from the mass flow rate, the temperature, and the pressure of the actual flow.

The mass flow rate may be found from the volumetric flow rate given by the mass flow controllers:

$$\dot{m}_{N_2} = Q_s \left(\frac{p_a M_{N_2}}{RT_s} \right) \quad (3.15)$$

$$\dot{m}_v = \dot{m}_{N_2} \left(\frac{M_w}{M_{N_2}} \right) \left(\frac{\phi p_s}{p_a - \phi p_s} \right) \quad (3.16)$$

$$\dot{m}_{total} = \dot{m}_{N_2} + \dot{m}_v \quad (3.17)$$

$$Q = \frac{\dot{m}_{total} RT}{\left[\phi p_s (M_w - M_{N_2}) + p_a M_{N_2} \right]} \quad (3.18)$$

$\dot{m}_v$ = mass flow rate of water vapor (kg / s)

$\dot{m}_{N_2}$ = mass flow rate of dry nitrogen (kg / s)

$\dot{m}_{total}$ = total mass flow rate of gas (kg / s)

M_w = molecular weight of water (18.0 kg / kmole)

M_{N_2} = molecular weight of dry nitrogen (28.0 kg / kmole)

ϕ = relative humidity

p_a = atmospheric pressure (101325 Pa)

p_s = saturation vapor pressure (water vapor) (Pa)

Q_s = volumetric flow rate at standard conditions of 273 K (0°C) (m^3 / sec)

Q = total volumetric flow rate (m^3 / sec)

R = universal gas constant (8314.5 N-m / kg-K)

T = temperature (K)

T_s = temperature at standard conditions of 273 K

3.3.3 Test Procedure

The automated air permeability test is conducted under the control of a PC interfaced to the flow controllers and instrumentation through a digital voltmeter and a GPIB (see Figure 21). The test sequence is: at a given relative humidity, to vary the flow rate through the test sample and record the resulting pressure drop, for a variety of flow rates, and then to repeat the process for another relative humidity. In between each group of setpoints at a given relative humidity, a setpoint at no flow is included. This allows us to see if the zero point for the differential pressure transducer has drifted from true zero (which it does slowly over time). This allows us to shift the data back to zero in analyzing the data.

The usual test sequence of test points starts at 0% relative humidity (dry nitrogen) and flow rates of 2000, 1500, 1000, 750, and 500 cm^3/min flows through the sample as indicated on the MKS controllers. After this sequence, the flow is shut off, and data is taken to determine the zero shift of the transducer. This process is repeated for humidities of 20%, 40%, 60%, 80%, and 100% relative humidity.

For each relative humidity, we may plot the pressure drop versus the mass flow rate, volumetric flow rate, or apparent flow velocity, and from the slope of the line, determine the permeability constant.

Figure 22 shows plots of pressure drop versus volumetric flow rate for one layer of the cotton and the polyester fabrics, given previously in Table 3-2. The slope of each set of lines is related to the permeability constant for that test condition of relative humidity. We can see very clearly that the hygroscopic cotton fabric shows a consistent variation with relative humidity, while the relatively nonhygroscopic polyester fabric shows no variation with relative humidity.

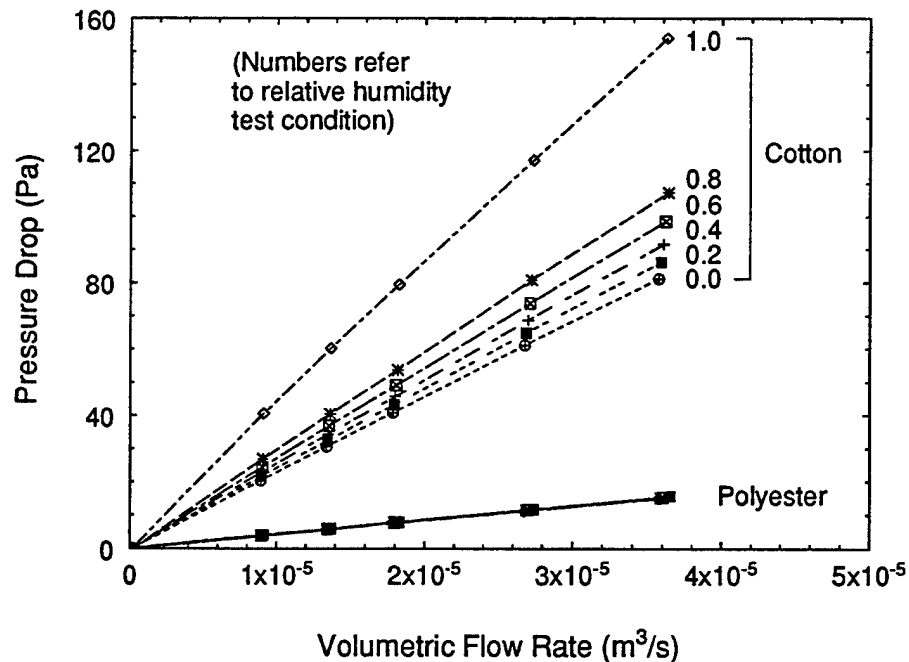


Figure 22. Pressure drop versus volumetric flow rate for two fabrics as a function of relative humidity

From the data shown in Figure 22, we may calculate the convective gas flow resistance from Darcy's Law for the two fabrics, over the range of relative humidity test conditions.

As we stated previously, we are ignoring inertial effects in our analysis of the flow rate/pressure-drop data. We should also note that another reason this method of simply drawing straight lines through the pressure-drop/flow rate data is not strictly correct for gas flows is that the density is dependent on the absolute gas pressure on the two sides of the test sample. A more correct expression for the permeability as a function of pressure drop in gas flows is given by Dullien [76]:

$$v = -\left(\frac{k_D}{\mu}\right) \frac{(p_2^2 - p_1^2)}{2p_2} = -\left(\frac{k_D}{\mu}\right) \left(\frac{p_m}{p_2}\right) \Delta p \quad (3.19)$$

$$p_m = \frac{(p_1 + p_2)}{2}, \text{ average pressure in sample}$$

p_1 = upstream pressure

p_2 = downstream pressure

This expression has not been used for the pressure drop/flow rate data, since at the low flow rates and pressures used in present apparatus, the assumption of incompressibility does not seem to have introduced any obvious errors.

The automated air permeability apparatus provides plots of pressure drop versus either mass flow rate or volumetric flow rate. Volumetric flow rate is the most convenient to use, so the permeability constant may be found from Equation (3.14) as:

$$k_D = \left(\frac{\mu Q}{A}\right) \left(\frac{\Delta x}{\Delta p}\right) \quad (3.20)$$

μ = gas viscosity (17.84×10^{-6} kg/m-s for N_2 at 20°C)

Q = total volumetric flow rate (m^3/s)

A = apparent sample flow area ($9.678 \times 10^{-4} \text{ m}^2$ for present sample holder)

Δx = thickness (m)

Δp = pressure drop across sample (N/m^2 or Pa)

For textiles, although thickness measurements seem simple, they are often problematic, and can be a large source of error if they are incorporated into reported measurements of Darcy permeability. It is preferable to present the pressure-drop/flow rate results in terms of an apparent flow resistance defined as:

$$R_D = \left(\frac{A \Delta p}{\mu Q_{total}}\right) \quad (3.21)$$

R_D = apparent Darcy flow resistance (m^{-1})

The Darcy permeability can thus be found from the apparent flow resistance as:

$$k_D = \frac{\Delta x}{R_D} \quad (3.22)$$

For the low flow rates and pressure drops used in this study, the definition of Darcy flow resistance R_d is a useful concept, since the flow resistances of materials measured individually can be added together to give a total flow resistance. An example of this is shown in Figure 23 for the polyester fabric from Table 3-2. The polyester fabric is used to illustrate the additivity of flow resistances since it is not affected by humidity-dependent porosity changes.

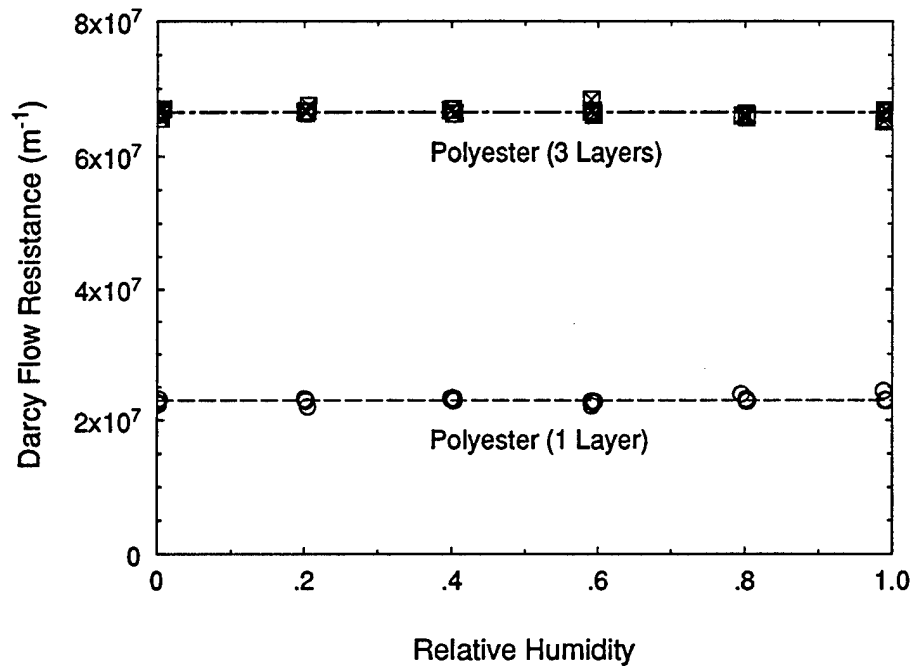


Figure 23. Illustration of the additivity of Darcy flow resistance for one and three layers of nonhygroscopic polyester fabric.

Figure 23 shows that the measured Darcy flow resistance for three layers of the polyester fabric is equal to three times the flow resistance of the single layer of fabric. Although the concept of the additivity of Darcy flow resistance is valid for low flow rates, it would not be safe to rely upon it for flow rates where inertial effects become important.

The measured Darcy flow resistance of the seven fabrics given in Table 3-2, as a function of relative humidity, is shown in Figure 24.

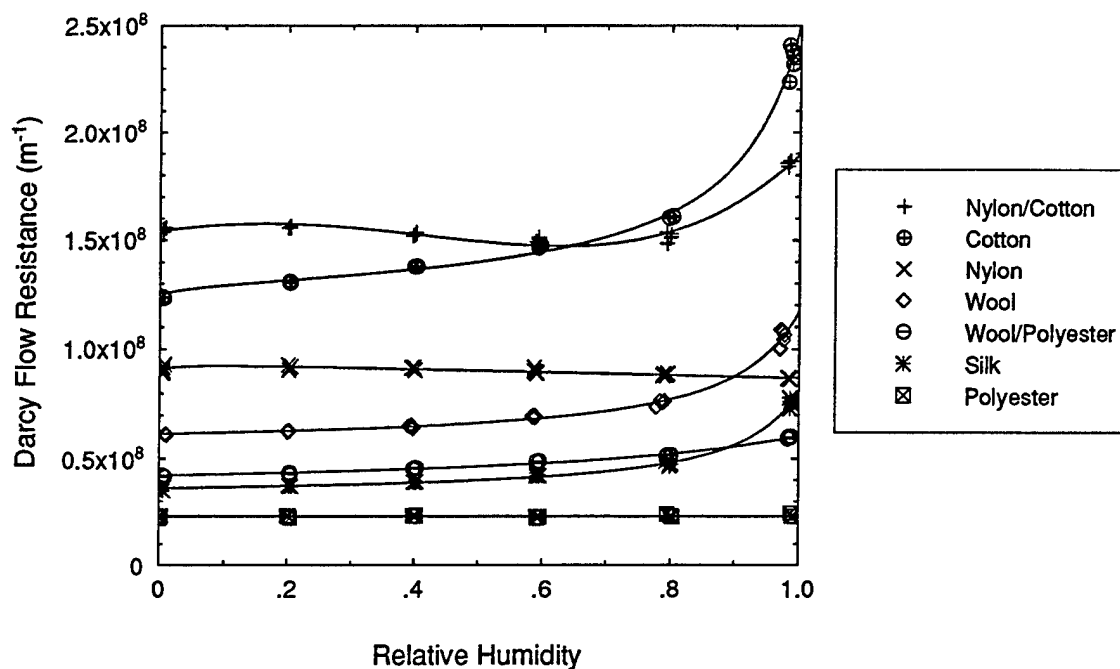


Figure 24. Apparent flow resistance of seven fabrics as a function of relative humidity.

As expected, the hygroscopic fabrics such as cotton, wool, and silk, show a large change in the resistance to convective flow at the higher humidities, due to fiber swelling, and a decrease in free air volume in the fabric, and possibly, some increase in fabric thickness. The fabrics which are much less hygroscopic, such as the polyester and nylon fabrics, show much less variation with relative humidity.

There are several examples of fabrics which change their relative air permeability rankings based on the relative humidity at which they are tested. Of special interest to us is the reversal of rankings shown by the Cotton and the Nylon/Cotton fabric (shown in Figure 25). These two fabrics are used as the outer shell fabrics in different versions of military chemical protective uniforms. Laboratory testing (aerosol and liquid/vapor) conducted to characterize chemical agent transport properties under convective flows are often carried out at constant flow rate or pressure drop conditions. The variation of the intrinsic permeability of these fabrics as a function of relative humidity would affect the evaluation of chemical agent testing conducted using laboratory tests, and is one possible source of the scattered results often seen in laboratory tests of chemical agent transport properties. The humidity-dependent transport properties are also of obvious importance in the actual design and use of clothing systems incorporating these two fabrics, particularly under use conditions where appreciable convective flows through the clothing system are expected.

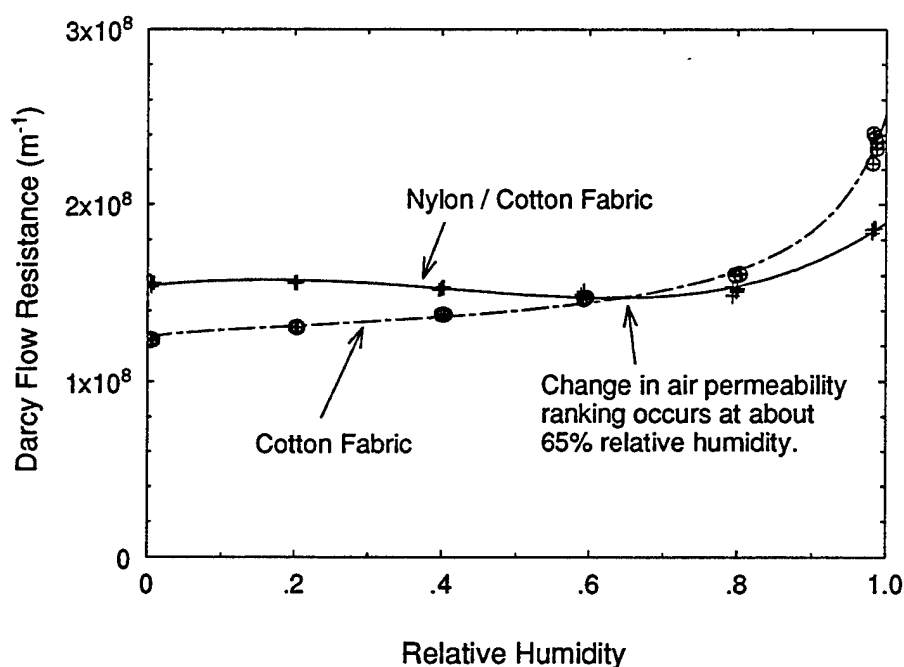


Figure 25. Reversal of air permeability ranking due to relative humidity test conditions for two fabrics.

A more complete study, with approximately 50 different types of fabrics, is presently being conducted to explore the relationship between fabric structural parameters, air permeability, and the changes in air permeability due to fiber swelling, using concepts available in the literature [65-71]. Although the study is ongoing, and is outside the scope of the work contained in this thesis, the behavior of nylon fabrics will be mentioned. Close examination of Figure 24 shows that the nylon fabric seems to show a decrease in flow resistance (increase in permeability) at high humidities, which is counter to the behavior of other types of fabrics. This type of behavior is postulated to be caused by the tendency of nylon fibers to swell axially rather than radially [77] as most other textile fibers do, thereby causing fabric pores to open up rather than close down. Our measurements on approximately 15 nylon fabrics, of which five are shown below in Figure 26, show that this is a consistent property of nylon fabrics which is common to many types of fabric weaves and thickness.

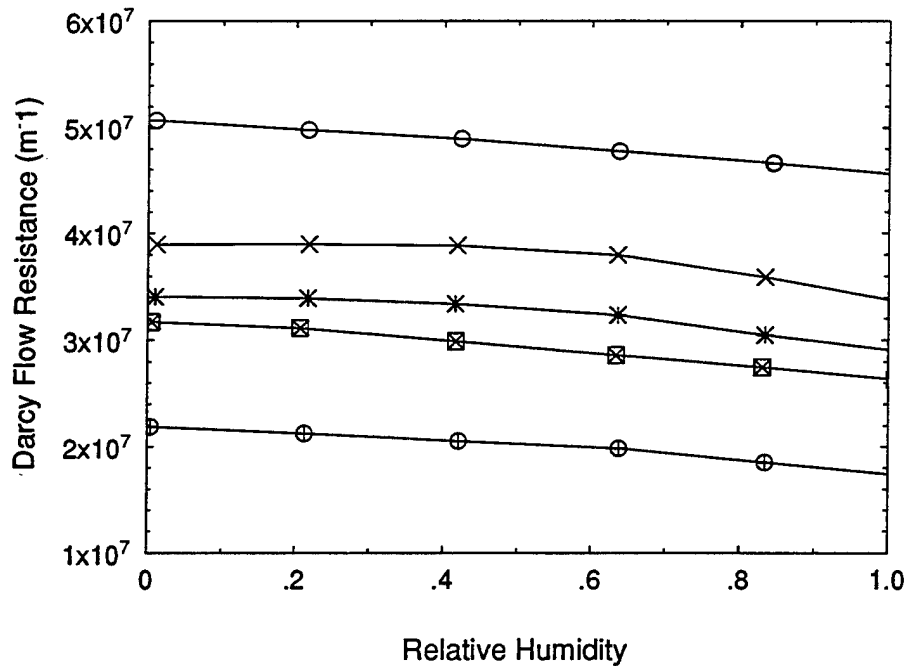


Figure 26. Decrease in permeability of five nylon fabrics as a function of relative humidity.

3.3.4 Sorption Scaling Approach to Provide Darcy Permeability Coefficient for Fabrics With Humidity-Dependent Porosity

Most hygroscopic fabrics we have looked at exhibit a general shape such as that shown by the cotton fabric, as seen in Figure 24. There is clear evidence of fiber swelling at high humidities, which has a large effect on the flow resistance.

The general shape of Darcy flow resistance versus relative humidity curve follows the shape of the curve for the sorption isotherm of cotton. We have previously defined a variable describing the volume fraction of water dissolved in the solid phase, which is directly related to the sorption isotherm:

$$\epsilon_{bw} = 0.578 R_f \left(\epsilon_{ds} \frac{\rho_{ds}}{\rho_w} \right) \left(\frac{p_v}{p_s} \right) \left[\frac{1}{\left(0.321 + \frac{p_v}{p_s} \right)} + \frac{1}{\left(1.262 - \frac{p_v}{p_s} \right)} \right] \quad (3.23)$$

- R_f textile measurement of fabric regain at 65% relative humidity
 $\epsilon_{bw}(t)$ V_{bw}/V , volume fraction of the water dissolved in the solid phase
 ϵ_{ds} V_{ds}/V , volume fraction of the dry solid (constant)
 ϵ_γ V_γ/V , volume fraction of the gas phase = $1 - \epsilon_{ds} - \epsilon_{bw}$
 ρ_{ds} density of dry solid [for polymers typically 900 to 1300 kg/m³]
 ρ_w density of liquid water [approximately 1000 kg/m³]
 ϕ = p_v/p_s , relative humidity

A plot of the volume fraction of water absorbed by the fiber as a function of relative humidity is shown in Figure 27, along with the measured Darcy flow resistance curve, using properties for the cotton fabric given in Appendix A:

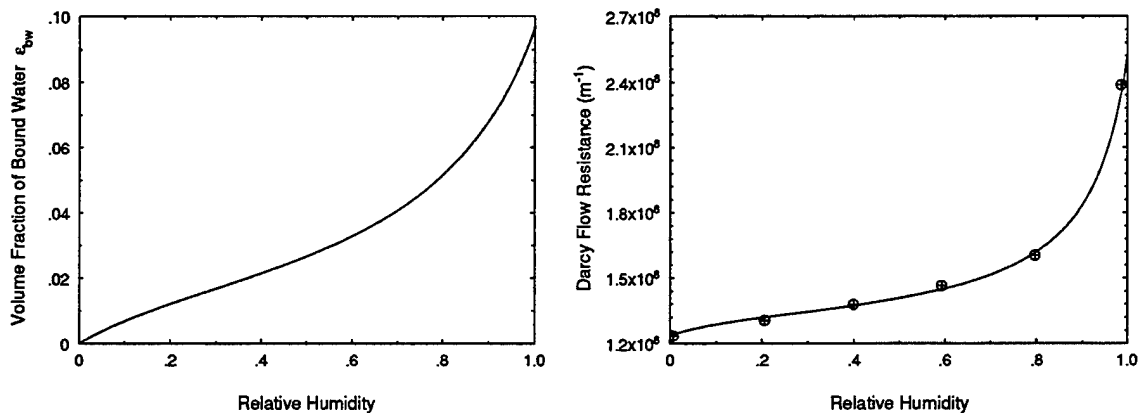


Figure 27. Volume fraction of water absorbed by the fiber for the 100% cotton fabric compared to the Darcy flow resistance as a function of relative humidity.

We can use the similarity in shape between the sorption isotherm and the flow resistance curve to construct an approximation for the change in apparent flow resistance as a function of relative humidity. This approach is useful because an experimental curve of permeability as a function of relative humidity is not always be available, and the convective transport properties are tied to the volume fraction of air space available in the porous material, which is the main factor affecting the convective flow properties.

Using the similarity of the flow resistance and sorption curves, a simple scaling approach to calculate the effective flow resistance of the fabric as a function of relative humidity (ϕ) can be written as:

$$R(\phi) = R_{dry} + \left(\frac{\varepsilon_{bw}(\phi)}{\varepsilon_{bwsat}} \right) (R_{sat} - R_{dry}) \quad (3.24)$$

R_{dry} Darcy permeability of fabric at $\phi=0.0$ [m^2]

R_{sat} Darcy permeability of fabric at $\phi=1.0$ [m^2]

ε_{bwsat} Volume fraction of bound water at $\phi=1.0$

This simple approach works well enough for the cotton, wool, and silk fabrics, but may not be appropriate for the nylon fabrics, which have a less straightforward relationship between the sorption isotherm and the curve of Darcy flow resistance versus relative humidity.

3.4 Use of DMPC to Measure Humidity-Dependent Air Permeability of Fabrics

The automated air permeability apparatus shown in Figure 21 uses the traditional flow cell for these types of studies: a circular pipe with an approach and exit section which are long compared to the thickness of the test material. We were curious to see how well we could reproduce the air permeability results by performing the same test in the DMPC. The reason for doing this is that one automated series of setpoints could determine all diffusion and convective flow properties without changing samples, or changing test cells, which would further increase the number of tests which could be done per day. Such a test would have great value for laboratories which conduct testing for quality control purposes, and the automated nature of the apparatus would also make the test device useful for nearly real-time monitoring of the transport properties of textile properties during the manufacturing process.

The DMPC may be converted to an air permeability apparatus by setting the flow for the lower inlet to zero, and closing the restriction valve for the upper outlet. The data acquisition and control program is identical to that used for the normal automated air permeability apparatus shown in Figure 21, except that the proper sample flow area must be used. The DMPC configuration for the convective flow property testing is shown in Figure 28.

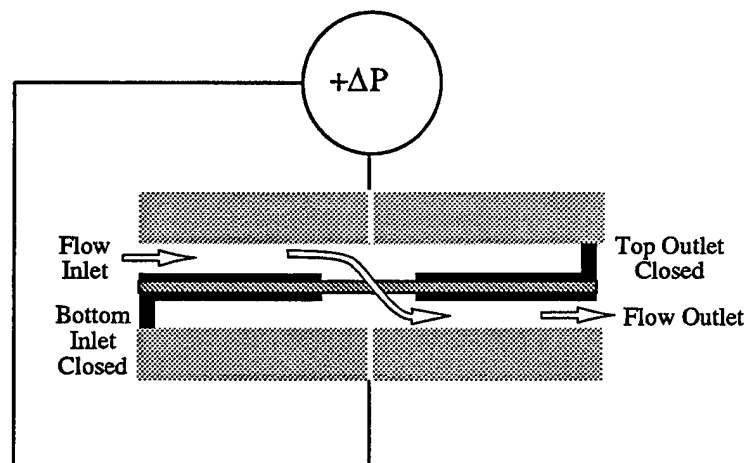


Figure 28. DMPC configuration for measuring humidity-dependent air permeability of textiles.

We were surprised to see that the DMPC in this configuration gives results which are comparable to those obtained using the more traditional type of permeability cell, as shown in Figure 29.

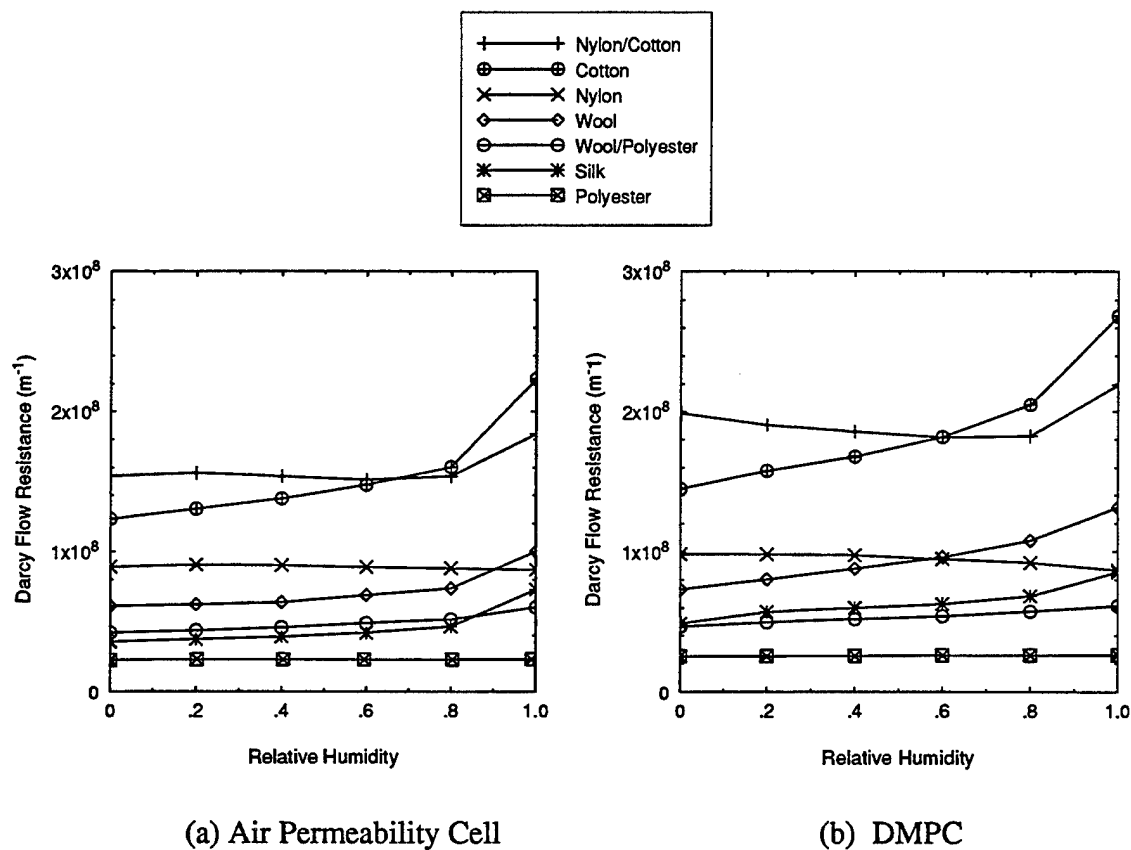


Figure 29. Apparent flow resistance of seven fabrics as a function of relative humidity as determined in the air permeability flow cell and the DMPC.

There is not a perfect agreement between the two test cells, but the rankings and relative trends are the same for each set of results. Since identical samples were not used, it's difficult to say if the small differences seen are due to sample variability, or to some real differences in the test conditions, e.g., some errors in the apparent flow area used for each test. We plan to continue to address the possibility of a single test cell which can determine both diffusion and convection transport properties in the context of an improved version of the DMPC.

3.5 Conclusions

The dynamic moisture permeation cell permits rapid testing of small quantities of permeable fabrics and semipermeable laminates under a wide variety of test conditions. This allows determination of a material's transport behavior under test conditions which are difficult or impossible to reach using existing standard laboratory test methods.

Water vapor permeation results obtained with the DMPC are in excellent agreement with those from the ISO 11092 sweating guarded hot plate test method, and from a modified inverted ASTM E 96 cup test.

The DMPC is particularly useful for characterizing transient diffusion of hygroscopic fabrics subjected to step changes in relative humidity. It provides excellent control over temperatures, pressures, and vapor concentrations, which is necessary for accurately determining the time-dependent and nonlinear transport properties of clothing materials. The DMPC also provides a controlled means of testing porous fabrics under conditions of combined diffusion and convective transport, in a way that provides experimental data to verify numerical models of the transport processes occurring in hygroscopic porous textiles.

The automated air permeability apparatus provides experimental data on convective gas flow transport properties, e.g. Darcy permeability or Darcy flow resistance. It provides a means of exploring and characterizing the way in which fabric structural changes, caused by fiber swelling at high humidities, can drastically change the air permeability of fabrics incorporated into clothing systems.

CHAPTER 4

NUMERICAL METHODS

4.1 Introduction

The governing equations given in Chapter 3 may be solved by using finite-difference techniques to discretize the equations and cast them into algebraic form, and then to solve the set of equations by an iterative process for each time step. For the equations which involve diffusion only, with no convection driven by pressure differences, the solution of the unsteady diffusion equation is straightforward, and there are a wide variety of finite-difference techniques which will all work quite well.

For the case of combined diffusion and convection, the need to satisfy the continuity of mass and conservation of momentum relations, with fluid velocity driven by pressures which are known only at discrete grid points, presents special problems which have been the subject of much research in the area of computational fluid dynamics. The method we use to solve the system of governing equations is described below.

4.2 SIMPLEC Program

The SIMPLEC program is a modification of the SIMPLER (Semi-Implicit Method for Pressure-Linked Equations-Revised) algorithm described by Patankar [78]. The SIMPLEC program has been modified to make it possible to solve unsteady partial differential equations of the general form:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla \cdot (\rho \bar{u}\phi) = \nabla \cdot (\Gamma \nabla \phi) + S \quad (4.1)$$

ρ = density

ϕ = specific dependent variable

$\bar{u}$ = velocity field

Γ = diffusion coefficient for ϕ

S = volumetric source term for ϕ

One special feature of the SIMPLEC code is that it creates finite-difference approximations of the partial differential equations by using control volumes, rather than just equations for a set of grid points. This ensures continuity within a control volume, and lessens the need for interpolation between grid points, since continuity and conservations relations for mass and energy are satisfied automatically by the discretization equations. Another special feature is the special treatment of the equations of momentum and continuity. The momentum equations also have the form of equation 4.1, but the source term for the momentum equation includes the pressure gradient,

which is the driving force for fluid motion. The essence of the SIMPLE, SIMPLER, and SIMPLEC algorithms is the relationships which can be developed between the velocity field, the momentum equations, and the continuity equation. By an iterative method, the pressure field is adjusted so that the velocity field produced by the pressure gradients also satisfies the continuity equation.

The discretization equations contain the unknown values of the variable f at discrete locations, located at the main grid points. Control volumes are defined at each grid point, and the main discretization equations are formed by integrating equation 4.1 over the control volume. Control volume boundaries are usually placed so that they coincide with actual material boundaries, so the control volume interface may also be an actual material interface. The form of the discretization equations, based as they are on a control volume rather than on grid points, make it very easy to satisfy conservation and continuity across material interfaces, which often present problems for finite-difference schemes based on grid points only. A schematic of the arrangement of control volumes and grid points for a Cartesian coordinate system is shown in Figure 30.

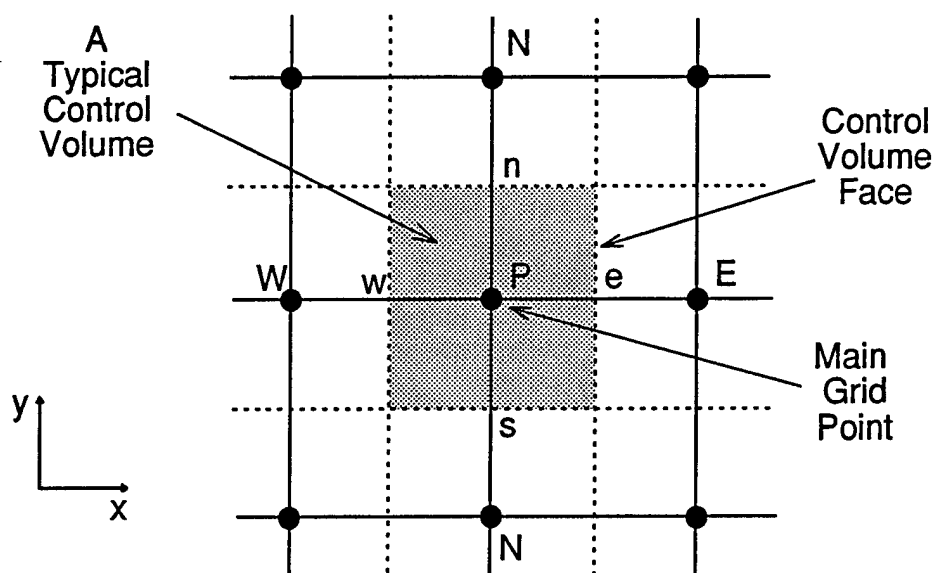


Figure 30. Control volumes and grid points for finite-difference method.

Figure 30 shows the grid numbering scheme as used by Patankar [78] in deriving the discretization scheme. An example of the discretization equation, for the situation of the diffusion equation which contains no convection terms, is presented later in this chapter in Section 4.3 and 4.4. The letter P refers to the main grid point about which the discretization equation is being written and integrated, the letters W, N, E , and S refer to the West, North, East, and South main grid points, and the lower-case letters, w, n, e , and s , refer to the west, north, east, and south locations for the control volume faces. The grid numbering scheme becomes very important because there are really two sets of control volumes and grid points used in the SIMPLE family of finite-difference methods.

The special treatment used for the momentum and continuity equation involves staggering the location of the calculated velocity components so that they are known only at the control volume interfaces, and not at the main grid points. This is a natural consequence of the control volume approach, because the convection term which brings material or energy into a control volume is most easily thought of as entering through the boundary of the control volume at the control volume face. The use of a staggered control volume for the velocity components also allows the pressures calculated at the main grid points to more easily be incorporated as the driving force for fluid flow, since the pressures and velocities are being calculated at different locations. An example of a typical staggered control volume for the velocity components is shown in Figure 31.

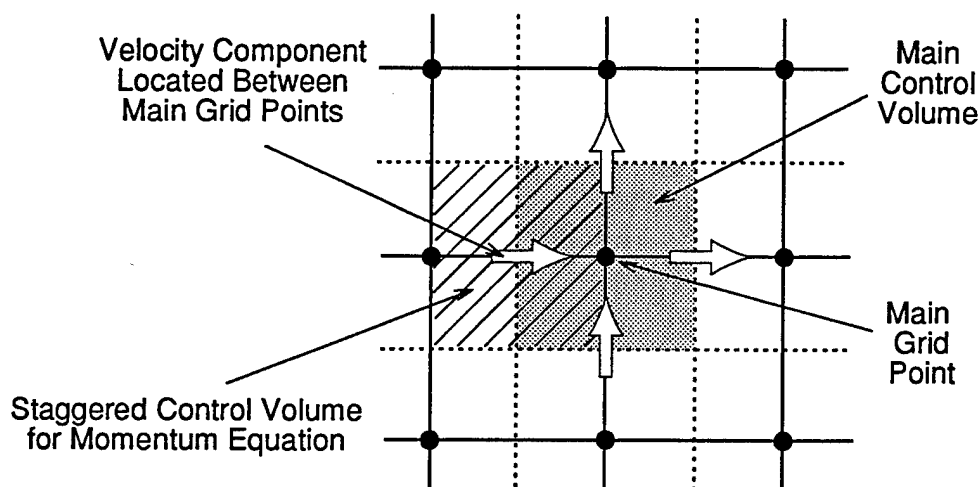


Figure 31. Locations of staggered control volume for velocity components and momentum equations.

The discretization equations are solved within the SIMPLEC main program. Facilities exist within the program to either solve the problem in a steady-state manner, or to incorporate a time step, and solve unsteady problems. Within a time step, or for a steady state problem, nonlinearities are handled by iteration until the solution converges. If after a number of iteration cycles, the values of the calculated variables have not changed by a certain specified small amount, the solution is considered converged, either for the the steady-state problem, or for that particular time step in an time-varying problem. Various schemes such as underrelaxation of specific variables or source terms are available to aid in the convergence of nonlinear problems, and are especially helpful in the case of complex source terms.

Specific problems are set up within a separate user subroutine, several examples of which are given later in Chapters 5, 6, and 7. The user subroutines specify solution sequences and parameter for the problem of interest, and define how source terms and boundary conditions are to be handled. The main program solves the set of equations given by equation 4.1, where the variables, source terms, and diffusion coefficients have all been defined by the user subroutine.

The complete set of convection/diffusion finite-difference equations used by SIMPLE, SIMPLER, and SIMPLEC are widely available [81-83]. The complete set of algebraic equations are not presented here, however, an example of the set of finite-difference equations for the unsteady diffusion equation, which does not include the convection term, is presented in the next section.

4.3 Programming of Exponential Scheme for the SIMPLEC Program

The general control-volume discretization method for the general transport equation contained in the SIMPLEC program is based on the fully implicit finite-difference method. For cases we will be examining later, some situations involve diffusion processes only, and there is no need to use the SIMPLEC numerical code to solve the pressure-linked momentum equations. For these situations, there are some reasons to use a finite-difference scheme which may have some advantages over the fully implicit scheme. For example, an alternate discretization method, called the exponential scheme, was developed by Patankar and Baliga [82]. This method was integrated as an option into the SIMPLEC program, and is described below.

We will illustrate the discretization scheme for the partial differential equation given by:

$$\rho \frac{\partial \phi}{\partial t} = \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \phi}{\partial y} \right) + \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \phi}{\partial y} \right) + \bar{S} \quad (4.2)$$

with the source term linearized so that

$$\bar{S} = S_c + S_p \phi_P \quad (4.3)$$

The grid notation was given previously in Figure 30, and for the x -direction only in Figure 32.

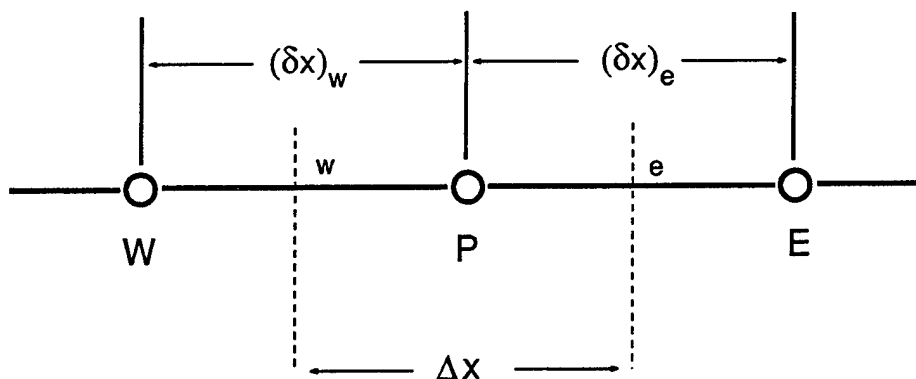


Figure 32. Discretization of grid for x -direction only.

The generalized finite difference equation is given by [78]:

$$\begin{aligned}
 \frac{\rho \Delta x \Delta y}{\Delta t} (\phi_P - \phi_P^0) = & \frac{\Gamma_e \Delta y}{\delta x_e} [f_E \phi_E + (1 - f_E) \phi_E^0] - [f_P \phi_P + (1 - f_P) \phi_P^0] \\
 & - \frac{\Gamma_w \Delta y}{\delta x_w} [f_P \phi_P + (1 - f_P) \phi_P^0] - [f_W \phi_W + (1 - f_W) \phi_W^0] \\
 & + \frac{\Gamma_n \Delta x}{\delta y_n} [f_N \phi_N + (1 - f_N) \phi_N^0] - [f_P \phi_P + (1 - f_P) \phi_P^0] \\
 & - \frac{\Gamma_s \Delta x}{\delta y_s} [f_P \phi_P + (1 - f_P) \phi_P^0] - [f_S \phi_S + (1 - f_S) \phi_S^0] \\
 & + S_c \Delta x \Delta y + S_P \Delta x \Delta y [f_P \phi_P + (1 - f_P) \phi_P^0]
 \end{aligned} \tag{4.4}$$

In these equations the weighting factor f (f_E, f_W, f_N, f_S) refers to the variation of the old (ϕ^0) and new (ϕ) values of the variable ϕ within the current time step.

The numerical values of the weighting factor determine the type of finite-difference scheme. The explicit scheme results if $f = 0.0$, the Crank-Nicolson scheme is the result of using $f = 0.5$, and the fully implicit scheme is obtained by using $f = 1.0$.

We can rearrange the equation by bringing all the terms containing the new value of ϕ to the left-hand-side:

$$\begin{aligned}
 & \left[\frac{\rho \Delta x \Delta y}{\Delta t} + \frac{\Gamma_e \Delta y}{\delta x_e} f_P + \frac{\Gamma_w \Delta y}{\delta x_w} f_P + \frac{\Gamma_n \Delta x}{\delta y_n} f_P + \frac{\Gamma_s \Delta x}{\delta y_s} f_P - S_P \Delta x \Delta y f_P \right] \phi_P \\
 & = \frac{\Gamma_e \Delta y}{\delta x_e} [f_E \phi_E + (1 - f_E) \phi_E^0] + \frac{\Gamma_w \Delta y}{\delta x_w} [f_W \phi_W + (1 - f_W) \phi_W^0] \\
 & + \frac{\Gamma_n \Delta x}{\delta y_n} [f_N \phi_N + (1 - f_N) \phi_N^0] + \frac{\Gamma_s \Delta x}{\delta y_s} [f_S \phi_S + (1 - f_S) \phi_S^0] \\
 & + \left[\frac{\rho \Delta x \Delta y}{\Delta t} + S_P \Delta x \Delta y (1 - f_P) - \frac{\Gamma_e \Delta y}{\delta x_e} (1 - f_P) - \frac{\Gamma_w \Delta y}{\delta x_w} (1 - f_P) - \frac{\Gamma_n \Delta x}{\delta y_n} (1 - f_P) - \frac{\Gamma_s \Delta x}{\delta y_s} (1 - f_P) \right] \phi_P^0 \\
 & + S_c \Delta x \Delta y
 \end{aligned} \tag{4.5}$$

If we define coefficients as:

$$a_E = \frac{\Gamma_e \Delta y}{(\delta x)_e} \quad (4.6)$$

$$a_W = \frac{\Gamma_w \Delta y}{(\delta x)_w} \quad (4.7)$$

$$a_N = \frac{\Gamma_n \Delta x}{(\delta y)_n} \quad (4.8)$$

$$a_S = \frac{\Gamma_s \Delta x}{(\delta y)_s} \quad (4.9)$$

$$a_P^0 = \frac{\rho \Delta x \Delta y}{\Delta t} \quad (4.10)$$

$$a_P = (a_E + a_W + a_N + a_S - S_P \Delta x \Delta y) f_P + a_P^0 \quad (4.11)$$

the discretization equation may be written as:

$$a_P \phi_P = \left\{ \begin{array}{l} a_E f_E \phi_E \\ + a_W f_W \phi_W \\ + a_N f_N \phi_N \\ + a_S f_S \phi_S \end{array} \right\} + \left\{ \begin{array}{l} a_E (1 - f_E) \phi_E^0 \\ + a_W (1 - f_W) \phi_W^0 \\ + a_N (1 - f_N) \phi_N^0 \\ + a_S (1 - f_S) \phi_S^0 \end{array} \right\} \\ + \left[a_P^0 + (1 - f_P) (S_P \Delta x \Delta y - a_E - a_W - a_N - a_S) \right] \phi_P^0 + S_c \Delta x \Delta y \quad (4.12)$$

Another general form of this discretization equation is given by Patankar and Baligar in their paper on the exponential scheme [82] : All the coefficients are given a * symbol to avoid confusion with the discretization equation given previously.

$$\left\{ 1 + f_P a_P^* \frac{\Delta t}{\rho} \right\} \phi_P = \frac{\Delta t}{\rho} \sum a_n^* [f_n \phi_n + (1 - f_n) \phi_n^0] + \left[1 - (1 - f_P) a_P^* \frac{\Delta t}{\rho} \right] \phi_P^0 + \frac{\Delta t}{\rho} S_c \quad (4.13)$$

We can rewrite the coefficient from the exponential scheme as:

$$a_P^* = \frac{1}{\Delta x \Delta y} (a_E + a_W + a_N + a_S - S_P) \quad (4.14)$$

or

$$a_P^* = \frac{a_P}{\Delta x \Delta y} - \frac{\rho}{\Delta t} ; \quad (4.15)$$

this expression uses variables which are all known quantities we already use in the

SIMPLEC program.

The discretization equation may be rewritten as:

$$\left\{ 1 + f_P \left[\left(\frac{a_P}{\Delta x \Delta y} \right) \left(\frac{\Delta t}{\rho} \right) - 1 \right] \right\} \left(\frac{\rho \Delta x \Delta y}{\Delta t} \right) \phi_P = \begin{Bmatrix} a_E f_E \phi_E \\ + a_W f_W \phi_W \\ + a_N f_N \phi_N \\ + a_S f_S \phi_S \end{Bmatrix} + \begin{Bmatrix} a_E (1 - f_E) \phi_E^0 \\ + a_W (1 - f_W) \phi_W^0 \\ + a_N (1 - f_N) \phi_N^0 \\ + a_S (1 - f_S) \phi_S^0 \end{Bmatrix} \quad (4.16)$$

$$+ \left[1 - (1 - f_P) \left(\frac{(a_E + a_W + a_N + a_S)}{\Delta x \Delta y} - S_P \frac{\Delta t}{\rho} \right) \right] \left(\frac{\rho \Delta x \Delta y}{\Delta t} \right) \phi_P^0 + S_c \Delta x \Delta y$$

The weighting function used for the exponential scheme is:

$$f_P = \frac{1}{1 - \exp(-\lambda_P)} - \frac{1}{\lambda_P} \quad (4.16)$$

where the quantity λ_P is defined by:

$$\lambda_P \equiv a_P^* \frac{\Delta t}{\rho} \text{ or } \lambda_P \equiv \left(\frac{a_P}{\Delta x \Delta y} \right) \frac{\Delta t}{\rho} - 1 \quad (4.17)$$

A power law approximation for the weighting function is:

$$f_P = 1 - \frac{1}{\lambda_P} \left[1 - (1 - 0.1 \lambda_P)^5 \right], \quad 0 \leq \lambda_P < 10 \quad (4.18)$$

$$f_P = 1 - \frac{1}{\lambda_P}, \quad \lambda_P \geq 10 \quad (4.19)$$

This new definition for the exponential scheme weighting factor f was incorporated into the SIMPLEC program as an option. The default weighting factor f is set to 1.0 for the fully implicit method. For comparison purposes, the program was also modified to give one the ability to select two other weighting factors, $f = 0.5$ for the Crank-Nicolson scheme, and $f = 0.0$ for the explicit scheme. The ability to select the discretization scheme one desires were only added for equations other than the momentum equation, which has a special solution technique, and which already incorporates a form of the exponential scheme in its solution process.

4.4 Comparison of the Modified SIMPLEC Code with the Exponential and Crank-Nicolson Finite Difference Schemes for the One-Dimensional Diffusion Equation

We may compare the performance of the various finite-difference schemes we have programmed into the SIMPLEC code to each other in terms of sensitivity to time step and grid spacing, to see if the exponential scheme does have any advantages over the implicit and the Crank-Nicolson schemes. The various finite-difference solutions may also be compared to the exact solution for one-dimensional unsteady heat conduction in a slab of thickness L and uniform thermal diffusivity α . This is identical to the problem considered by Patankar and Baliga in their paper [82] on the development of the exponential scheme.

The slab is initially at a uniform temperature T_i , and at time $t=0$ the two faces of the slab at $y=0$ and $y=L$ are brought up to, and held at, a temperature T_b .

We may define dimensionless variables as:

$$\theta = \frac{T - T_i}{T_b - T_i} ; \quad (4.20)$$

$$\tau = \frac{\alpha t}{L^2} ; \quad (4.21)$$

$$X = \frac{x}{L} \quad (4.22)$$

The unsteady heat conduction equation becomes:

$$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial X^2} ; \quad (4.23)$$

and the boundary conditions and geometry are shown in Figure 33:

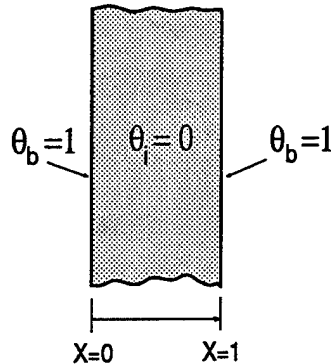


Figure 33. Nondimensional unsteady heat conduction boundary and initial conditions.

We may compare the results from the finite-difference solutions to an analytical solution (for a symmetrical slab subjected to a sudden temperature change on both sides) given by Schneider [83] as:

$$\theta = 1 - \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} e^{-(n^2 \pi^2 \tau)} \sin n\pi X \quad ; \quad n = 1, 3, 5, \dots \quad (4.24)$$

In the paper by Patankar and Baliga [82], results are compared for a fully implicit scheme, a Crank-Nicolson scheme, and the exponential scheme. The basis for comparison is the time step parameter defined as:

$$\lambda = \frac{\alpha \Delta t}{\Delta x^2} \quad (4.25)$$

The number of grid points used was 21, so that there are 20 equally-spaced divisions of the calculation domain. This necessitated some minor changes to the grid numbering scheme of the SIMPLEC program. The SIMPLE, SIMPLER, and SIMPLEC programs are all based on a control volume approach. The calculation domain is divided up by placing control volume faces between main grid points. If a uniform grid is specified, this means that the calculation domain is divided up into control volumes of equal thickness (for one-dimension). This means that the actual grid spacings are not all the same because the two grid points next to the boundaries are displaced to maintain constant control volume spacings. For the purposes of comparison with the results of Patankar and Baliga, the control volume spacings were modified as shown in Figure 34.

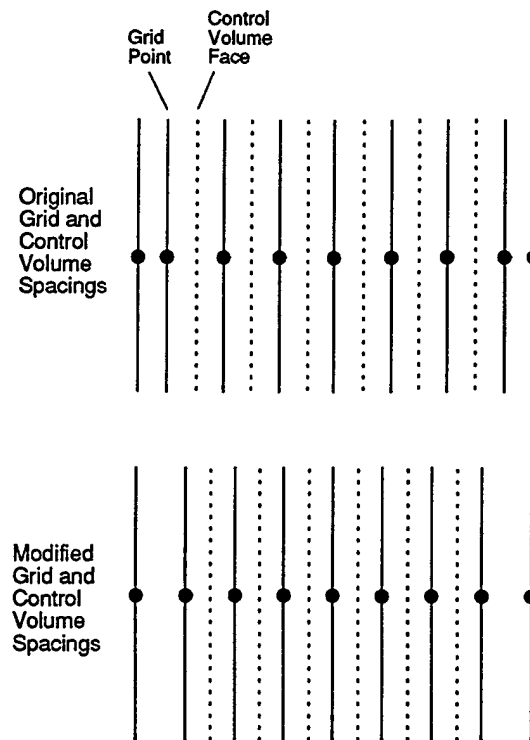


Figure 34. Uniform grid used in SIMPLEC for comparison with results of Patankar and Baliga [82].

We may compare the output from the various finite-difference schemes by using the dimensionless temperature gradient at the surface of the slab after the first time step, which is defined by:

$$\left(\frac{\partial\theta}{\partial X}\right)_0 = \frac{\theta_0 - \theta_1}{\Delta X} \quad (4.26)$$

The results for values of the time step parameter $\lambda = 16$ are shown in Figure 35, along with the results contained in Patankar's and Baliga's paper (results are the same).

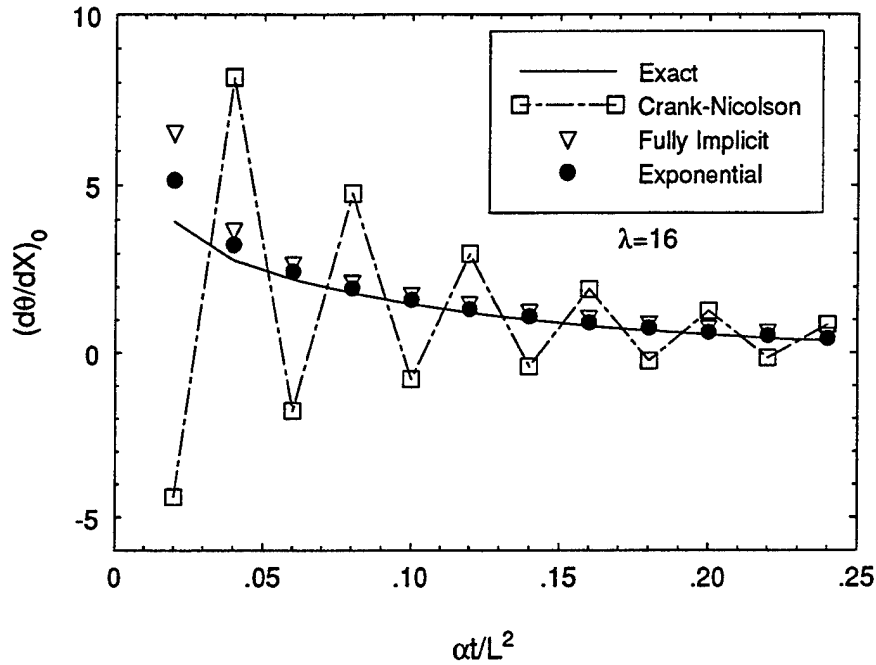


Figure 35. Predicted value of the dimensionless temperature gradient at the surface for time step parameter $\lambda=16$.

4.5 Conclusions

The results obtained with the SIMPLEC program are identical to the results for the fully implicit and exponential finite-difference schemes, obtained by Patankar and Baliga [82]. This gives us some confidence that the mechanics of the code are working correctly, and that there are no gross errors in this modified version of the SIMPLEC program.

The exponential scheme should allow larger time steps, yet retain good stability and accuracy compared to either the implicit or the Crank-Nicolson schemes. It should allow us to investigate diffusion-type problems, and permit us to get faster execution times, and possibly better execution times, by eliminating the need to use very small time steps, or excessively slow relaxation criteria.

CHAPTER 5

ONE-DIMENSIONAL MODELING OF COUPLED DIFFUSION OF ENERGY AND MASS

5.1 Introduction

We shall first examine the one-dimensional case of transient diffusion in hygroscopic porous textiles. Many workers in the textile field have successfully addressed this problem over the years [84-90]. The one-dimensional problem is a convenient starting place to illustrate the numerical solution of partial differential equations describing the coupled diffusion processes for energy and water vapor transport through porous hygroscopic layers.

In Chapter 2, the general governing equations were given (Equations 2.8.1-24). For one-dimensional transient diffusion, the major simplifying assumptions are: 1) there is no liquid or gas phase convection, 2) there is no liquid phase present, 3) the transport is one-dimensional (x -direction).

5.2 One-Dimensional Governing Differential Equations:

Energy equation:

$$\langle \rho \rangle C_p \frac{\partial \langle T \rangle}{\partial t} + (Q_l + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = \frac{\partial}{\partial x} \left(k_{eff} \frac{\partial \langle T \rangle}{\partial x} \right) \quad (5.1)$$

Solid phase continuity equation:

$$\epsilon_\sigma \frac{\partial}{\partial t} (\langle \rho_1 \rangle^\sigma) + \langle \dot{m}_{sv} \rangle = 0 \quad (5.2)$$

Gas phase diffusion equation (component 1--water vapor):

$$\frac{\partial}{\partial t} (\epsilon_\gamma \langle \rho_1 \rangle^\gamma) - \langle \dot{m}_{sv} \rangle = \frac{\partial}{\partial x} \left(\mathcal{D}_{eff} \frac{\partial \langle \rho_1 \rangle^\gamma}{\partial x} \right) \quad (5.3)$$

Volume fraction constraint:

$$\begin{aligned} \epsilon_\gamma + \epsilon_\sigma &= 1 \quad ; \quad \epsilon_\sigma = 1 - \epsilon_\gamma = \epsilon_{bw} + \epsilon_{ds} \\ \epsilon_{bw} &= \text{volume fraction of water dissolved in the solid} \\ \epsilon_{ds} &= \text{volume fraction of the totally dry solid (constant for a particular problem)} \end{aligned} \quad (5.4)$$

Thermodynamic relations

$$\begin{aligned}
 \langle p_2 \rangle^Y &= \langle p_Y \rangle^Y - \langle p_1 \rangle^Y \\
 \langle p_2 \rangle^Y &= \langle \rho_2 \rangle^Y \frac{R}{M_2} \langle T \rangle \\
 \langle p_1 \rangle^Y &= \langle \rho_1 \rangle^Y \frac{R}{M_1} \langle T \rangle \\
 \langle \rho_Y \rangle^Y &= \langle \rho_1 \rangle^Y + \langle \rho_2 \rangle^Y
 \end{aligned} \tag{5.5-5.8}$$

Transport Coefficients and Mixture Properties

$$k_{eff} = \varepsilon_\sigma \left(\frac{k_1 \rho_1 + k_2 \rho_2}{\rho_1 + \rho_2} \right)_\sigma + \varepsilon_Y \left(\frac{k_1 \rho_1 + k_2 \rho_2}{\rho_1 + \rho_2} \right)_Y \tag{5.9}$$

$$\mathcal{D}_{eff} = \frac{\mathcal{D}_a \varepsilon_Y}{\tau} \tag{5.10}$$

$$\langle \rho \rangle = \varepsilon_\sigma (\langle \rho_1 \rangle^\sigma + \langle \rho_2 \rangle^\sigma) + \varepsilon_Y (\langle \rho_1 \rangle^Y + \langle \rho_2 \rangle^Y) \tag{5.11}$$

$$C_p = \frac{\varepsilon_\sigma [\langle \rho_1 \rangle^\sigma (c_p)_{1\sigma} + \langle \rho_2 \rangle^\sigma (c_p)_{2\sigma}] + \varepsilon_Y [\langle \rho_1 \rangle^Y (c_p)_{1Y} + \langle \rho_2 \rangle^Y (c_p)_{2Y}]}{\langle \rho \rangle} \tag{5.12}$$

Sorption Relations

$$Q_l \text{ (J/kg)} = 1.95 \times 10^5 \left(1 - \frac{\langle p_1 \rangle^Y}{p_s} \right) \left(\frac{1}{\left(0.2 + \frac{\langle p_1 \rangle^Y}{p_s} \right)} + \frac{1}{\left(1.05 - \frac{\langle p_1 \rangle^Y}{p_s} \right)} \right) \tag{5.13}$$

$$\frac{\langle \rho_1 \rangle^\sigma}{\langle \rho_2 \rangle^\sigma} = R_f \left(0.55 \frac{\langle p_1 \rangle^Y}{p_s} \right) \left[\frac{1}{\left(0.25 + \frac{\langle p_1 \rangle^Y}{p_s} \right)} + \frac{1}{\left(1.25 - \frac{\langle p_1 \rangle^Y}{p_s} \right)} \right] \tag{5.14}$$

This notation (e.g. $\langle \rho_1 \rangle^Y$, $\langle \rho_2 \rangle^S$) was convenient during the derivation of the equations, but is quite confusing if one is not familiar with the numbering system used to keep track of the different components in each phase. To make the equations easier to follow they are rewritten in terms of a different notation, and the volume-average symbols are dropped. This new notation is also more consistent with a series of papers by Tao, et al. [91-93], which addresses one-dimensional transient diffusion effects in fibrous materials in a manner which also uses volume-averaging concepts. Included below is a new symbol definition list at the beginning of the system of equations.

Symbols and Notation:

Roman Letters

- $(c_p)_{ds}$ constant pressure heat capacity of the dry solid [J/kg-°K]
- $(c_p)_w$ constant pressure heat capacity of liquid water [4182 J/kg-°K (@290 K)]
- $(c_p)_v$ constant pressure heat capacity of water vapor [1862 J/kg-°K (@290 K)]
- $(c_p)_a$ constant pressure heat capacity of dry air [1003 J/kg-°K (@290 K)]
- C_p mass fraction weighted average constant pressure heat capacity [J/kg-°K]
- D_a diffusion coefficient of water vapor in air [m²/sec]
- D_{eff} effective gas phase diffusivity [m²/sec]
- Δh_{vap} enthalpy of vaporization per unit mass [J/kg]
- k_{ds} thermal conductivity of the dry solid [J/sec-m-°K]
- k_w thermal conductivity of liquid water [0.600 J/sec-m-°K (@290 K)]
- k_v thermal conductivity of saturated water vapor [0.0246 J/sec-m-°K (@380K)]
- k_a thermal conductivity of dry air [0.02563 J/sec-m-°K (@290 K)]
- $\dot{m}_{sv} = \langle \dot{m}_{sv} \rangle$ mass rate of desorption from solid phase to vapor phase per unit volume [kg/sec-m³]
- $p_Y = \langle p_Y \rangle^Y$ total gas pressure [N/m²]
- $p_a = \langle p_2 \rangle^Y$ partial pressure of air [N/m²]
- $p_v = \langle p_1 \rangle^Y$ partial pressure of water vapor [N/m²]
- p_s saturation vapor pressure (function of T only) [N/m²]
- $T = \langle T \rangle$ temperature [K]

Greek Letters

ε_{ds}	V_{ds}/V , volume fraction of the dry solid (constant)
$\varepsilon_{bw}(t)$	V_{bw}/V , volume fraction of the water dissolved in the solid phase
$\rho=\langle\rho\rangle$	volume average density [kg/m ³]
ρ_{ds}	density of dry solid [for polymers typically 900 to 1300 kg/m ³]
ρ_w	density of liquid water [approximately 1000 kg/m ³]
ρ_γ	density of gas phase (mixture of air and water vapor) [kg/m ³]
$\rho_v=\langle\rho_1\rangle^\gamma$	density of water vapor in the gas volume (equivalent to mass concentration) [kg/m ³]
$\rho_a=\langle\rho_2\rangle^\gamma$	density of the inert air component in the gas volume (equivalent to mass of air/total gas volume) [kg/m ³]
$\rho_w\left(\frac{\varepsilon_{bw}}{\varepsilon_{bw} + \varepsilon_{ds}}\right) = \langle\rho_1\rangle^\sigma$	species density of water vapor in the solid phase volume (equivalent to mass concentration) [kg/m ³]
$\rho_{ds}\left(\frac{\varepsilon_{ds}}{\varepsilon_{bw} + \varepsilon_{ds}}\right) = \langle\rho_2\rangle^\sigma$	species density of the dry solid in the solid phase volume (equivalent to mass concentration) [kg/m ³]

The equations are rewritten in terms of these new definitions of the variables, which makes them less confusing to follow. The biggest change is that we no longer use the solid phase volume fraction ε_s , but have rewritten everything for the solid phase in terms of the volume fraction of water dissolved in the solid. The water dissolved in the solid phase is sometimes called "bound water", so we call this volume fraction ε_{bw} . This bound water is the only material that crosses the solid phase interface, so it is really the only volume fraction we need to keep track of. This also means we assume that the volume of the solid phase is made up of the volume of the dry solid plus the volume of the "bound water" which is dissolved in the solid phase. An illustration of the bound water volume fraction is shown below in Figure 36.

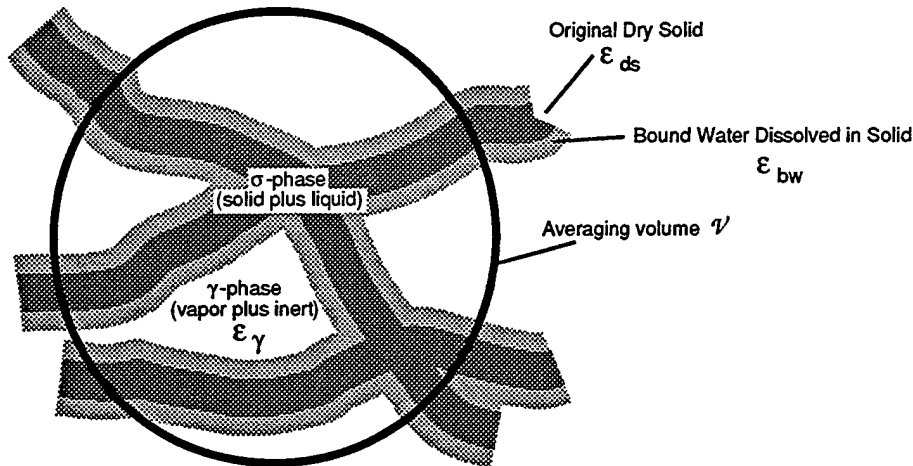


Figure 36. Schematic for definition of "bound water" volume fraction ε_{bw} .

With this new less-confusing nomenclature, the equations can be rewritten.

Governing Differential Equations:

Energy equation:

$$\rho C_p \frac{\partial T}{\partial t} + (Q_l + \Delta h_{vap}) \dot{m}_{sv} = \frac{\partial}{\partial x} \left(k_{eff} \frac{\partial T}{\partial x} \right) \quad (5.15)$$

Solid phase continuity equation:

$$\rho_w \frac{\partial}{\partial t} (\epsilon_{bw}) + \dot{m}_{sv} = 0 \quad \text{or} \quad \frac{\partial}{\partial t} (\epsilon_{bw}) + \frac{\dot{m}_{sv}}{\rho_w} = 0 \quad (5.16)$$

Gas phase diffusion equation:

$$\frac{\partial}{\partial t} (\epsilon_g \rho_v) - \dot{m}_{sv} = \frac{\partial}{\partial x} \left(\mathcal{D}_{eff} \frac{\partial \rho_v}{\partial x} \right) \quad (5.17)$$

Volume fraction constraint:

$$\epsilon_g + \epsilon_{bw} + \epsilon_{ds} = 1 \quad (5.18)$$

Thermodynamic relations

$$p_a = p_g - p_v$$

$$p_a = \rho_a \frac{R}{M_a} T$$

$$p_v = \rho_v \frac{R}{M_w} T$$

(5.19-5.21)

Sorption Relation [94]

$$\epsilon_{bw} = 0.578 R_f \left(\epsilon_{ds} \frac{\rho_{ds}}{\rho_w} \right) (\phi) \left[\frac{1}{(0.321 + \phi)} + \frac{1}{(1.262 - \phi)} \right] \quad (5.22)$$

The 8 unknown variables are: $T, \dot{m}_{sv}, \epsilon_g, \epsilon_{bw}, \rho_v, \rho_a, p_a, p_v$.

The source term due to vapor sorption is modeled by assuming that diffusion into the fiber is quasi-steady state [90]. The polymer at the fiber's surface is assumed to immediately come into equilibrium with the relative humidity of the gas phase within the control volume for that grid point. The mass flux into or out of the fiber is calculated by:

$$\dot{m}_{sv} = \frac{D_{solid} \rho_{ds}}{d_f^2} (R_{total} - R_{skin}) \quad (5.23)$$

R_{total}	instantaneous fiber regain (kg of water / 100 kg of dry polymer) [fraction]
R_{skin}	equilibrium regain at relative humidity within the control volume [fraction]
D_{solid}	apparent solid phase diffusion coefficient [m ²]
d_f	apparent fiber diameter [m]

Transport Coefficients and Mixture Properties

$$\rho = \varepsilon_{bw}\rho_w + \varepsilon_{ds}\rho_{ds} + \varepsilon_\gamma(\rho_v + \rho_a) \quad (5.24)$$

$$C_p = \frac{\varepsilon_{bw}\rho_w(c_p)_w + \varepsilon_{ds}\rho_{ds}(c_p)_{ds} + \varepsilon_\gamma[\rho_v(c_p)_v + \rho_a(c_p)_a]}{\rho} \quad (5.25)$$

For fabrics with significant porosity, an alternate form for the effective thermal conductivity is given by [95,96]:

$$k_{eff} = k_\gamma \left\{ \frac{[1 + (\varepsilon_{bw} + \varepsilon_{ds})]k_\sigma + \varepsilon_\gamma k_\gamma}{\varepsilon_\gamma k_\sigma + [1 + (\varepsilon_{bw} + \varepsilon_{ds})]k_\gamma} \right\} \quad (5.26)$$

$$k_\gamma = \left(\frac{k_v\rho_v + k_a\rho_a}{\rho_v + \rho_a} \right); \quad k_\sigma = \left(\frac{k_w\rho_w\varepsilon_{bw} + k_{ds}\rho_{ds}\varepsilon_{ds}}{\rho_w\varepsilon_{bw} + \rho_{ds}\varepsilon_{ds}} \right) \quad (5.27-5.28)$$

$$\mathcal{D}_{eff} = \frac{\mathcal{D}_a\varepsilon_\gamma}{\tau} \quad (5.29)$$

$$\mathcal{D}_a \text{ (m}^2 \cdot \text{s}^{-1}\text{)} = 2.23 \times 10^{-5} \left(\frac{T}{273.15} \right)^{1.75} \quad (5.30)$$

$$Q_i \text{ (J / kg)} = 1.95 \times 10^5 (1 - \phi) \left(\frac{1}{(0.2 + \phi)} + \frac{1}{(1.05 - \phi)} \right) \quad (5.31)$$

$$\Delta h_{vap} \text{ (J / kg)} = 2.792 \times 10^6 - 160T - 3.43T^2 \quad (5.32)$$

$$p_s \text{ (N} \cdot \text{m}^{-2}\text{)} = 614.3 \exp \left\{ 17.06 \left[\frac{(T - 273.15)}{(T - 40.25)} \right] \right\} \quad (5.33)$$

Initial and Boundary Conditions

Boundary Conditions (L refers to fabric thickness):

$$h_c(T_\infty - T)_{x=0} = -k_{eff} \left. \frac{\partial T}{\partial x} \right|_{x=0} = h_c(T_\infty - T)_{x=L} = -k_{eff} \left. \frac{\partial T}{\partial x} \right|_{x=L} \quad (5.34)$$

$$h_m(\rho_{v\infty} - \rho_v)_{x=0} = -D_{eff} \left. \frac{\partial \rho_v}{\partial x} \right|_{x=0} = h_m(\rho_{v\infty} - \rho_v)_{x=L} = -D_{eff} \left. \frac{\partial \rho_v}{\partial x} \right|_{x=L} \quad (5.35)$$

Initial Conditions:

$$T(x, t = 0) = T_0$$

$$\rho_v(x, t = 0) = \phi_0 p_s(@T_0)$$

$$\varepsilon_{bw}(x, t = 0) = \varepsilon_{bw0}$$

Assumed or Measured Parameters

Fabric Properties (from Appendix A):

Thickness (2 layers)

Apparent Bulk Density (Dry)

Regain at 65% relative humidity R_f

Volume fraction of dry solid ϵ_o

Effective Diffusivity D_{eff}

Fabric Tortuosity ($\tau = \epsilon_o D_a / D_{eff}$)

Dry solid density ρ_{ds}

Constant pressure heat capacity of the dry solid $(c_p)_{ds}$

Thermal conductivity of the dry solid k_{ds} .

The external convective mass transfer coefficient h_m is available from measurements in the DMPC. For the particular flow rates and temperature used in the diffusion experiments contained in Chapter 3, the apparent boundary layer resistance was found in Chapter 3 to be about 95 s/m. Assuming that the boundary layers are equal on the two sides of the sample, each side had a resistance of 47.5 s/m, which results in a mass transfer coefficient h_m of 0.021 m/s.

An estimate of the external convective heat transfer coefficient may be found from boundary layer similarity [97] (assuming laminar flow):

$$h_c = h_m \left[\rho_g (c_p)_g \right] \left(\frac{\alpha}{D_a} \right)^{2/3} \quad (5.36)$$

α = thermal diffusivity of gas [m²/s]

D_a = diffusion coefficient of water vapor in air [m²/s]

ρ_g = gas density

$(c_p)_g$ = gas heat capacity

This resulted in a external convective heat transfer coefficient of $h_c = 21.8 \text{ W}/(\text{m}^2 \cdot \text{s})$.

However, we may also estimate convective heat transfer coefficients using relations developed for flow between parallel plates [98]. Using these relations, we get a heat transfer coefficient of around 40 W/(m²·s). Since the thermal boundary layer over the test sample is not fully developed, the assumptions about boundary layer similarity may not be justified, and the heat transfer coefficient for flow between parallel plates was used. Later in this chapter we compare results obtained using these two different heat transfer coefficients, and we also include thermal radiation effects.

This problem of uncertainty about what boundary heat transfer coefficients to use in our numerical solution will not come up in Chapter 6, since the flow simulation will take care of all the momentum, concentration, and temperature boundary layers in the course of the fluid dynamic and heat/mass transfer calculations for the flowing gas streams.

The assumed properties of effective fiber diameter d_{eff} and effective solid phase diffusion coefficient D_{solid} , which takes care of the actual geometry of the fiber, including fiber/yarn shape and fiber/yarn size distribution, was chosen to fit the experimental data contained in Chapter 3 for the seven hygroscopic porous fabrics listed in Table 3-2 in Chapter 3.

5.3 Comparison of Experimental and Numerical Results

The one-dimensional program written to solve the diffusion equation, mentioned in Chapter 4, was used to solve the system of equations. The main program and the user routine are both contained in Appendix B. The number of grid points used was 11, the time step was 0.01 seconds, and the exponential scheme was used for the finite-difference equations. Later in this chapter, the effects of changing grid size, time step, and finite-difference scheme will be shown for a typical calculation.

The situation modeled was the step change in relative humidity from 0.0 to 1.0 at a temperature of 20°C. The experimental plots were shown as Figure 15 in Chapter 3, and the experimental schematic was shown previously as Figure 13 in Chapter 3.

Figures 37 and 38 show the comparison with the computed temperature history, and the experimental temperature history as a function of time, for the seven hygroscopic fabrics. The experimental data contain more data points in the first minute to resolve the initial fast temperature changes.

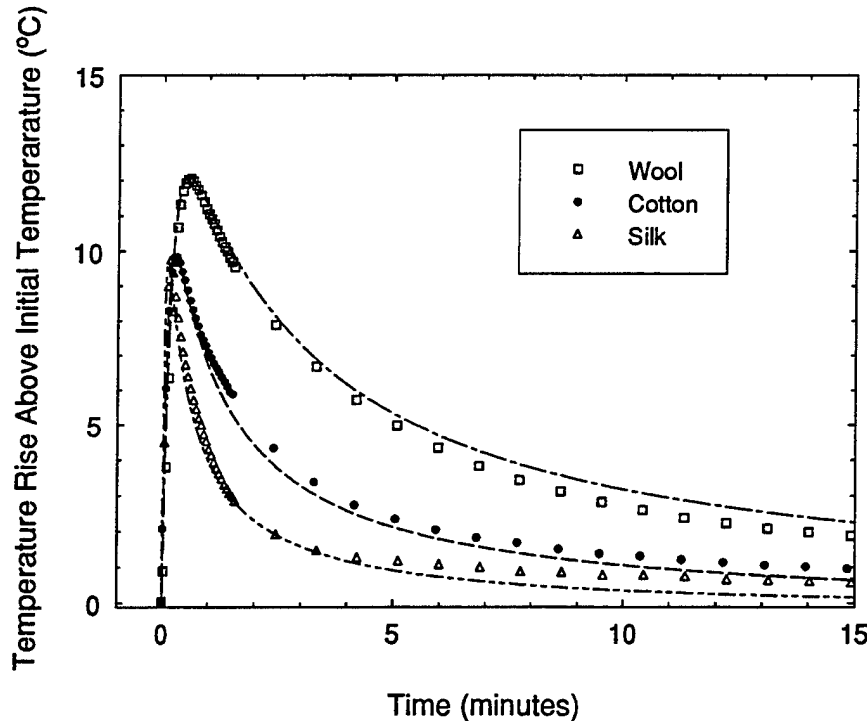


Figure 37. Comparison of numerical predictions to experimental results of centerline temperature of Wool, Cotton, and Silk fabrics subjected to step change in relative humidity.

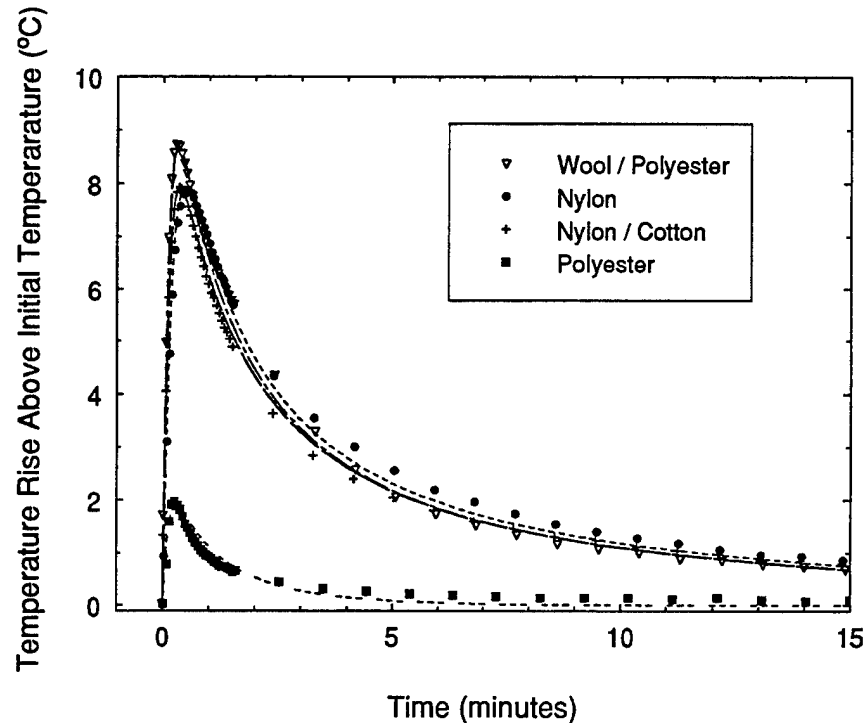


Figure 38. Comparison of numerical predictions to experimental results of centerline temperature of Wool/Polyester, Nylon/Cotton, Nylon and Polyester fabrics subjected to step change in relative humidity.

We see fairly good agreement between measured and predicted temperature transients for all the fabrics. The comparisons shown are for the center thermocouple only, and are thus average values. There is actually a significant temperature and concentration variation down the length of the sample in the DMPC apparatus during the vapor sorption process, which is reflected in the different temperature records from the three thermocouples in the test sample. Later in Chapter 6, we will show how this temperature variation is due to the interaction of the developing concentration and thermal boundary layers as the vapor-laden gas streams flow over the test sample.

The one-dimensional numerical model may be used to examine features of hygroscopic porous textiles such as equilibration time, and weight change due to vapor uptake. This is convenient when data from apparatus such as a sorption balance are available.

Figure 39 shows the performance of the one-dimensional model for the sorption of water vapor, following a step change from 0.0 to 0.65 relative humidity, at 20°C, under the same conditions as given before for the DMPC, for the seven fabrics in Table 3-2. The results plot the fractional approach to equilibrium regain at 65% relative humidity.

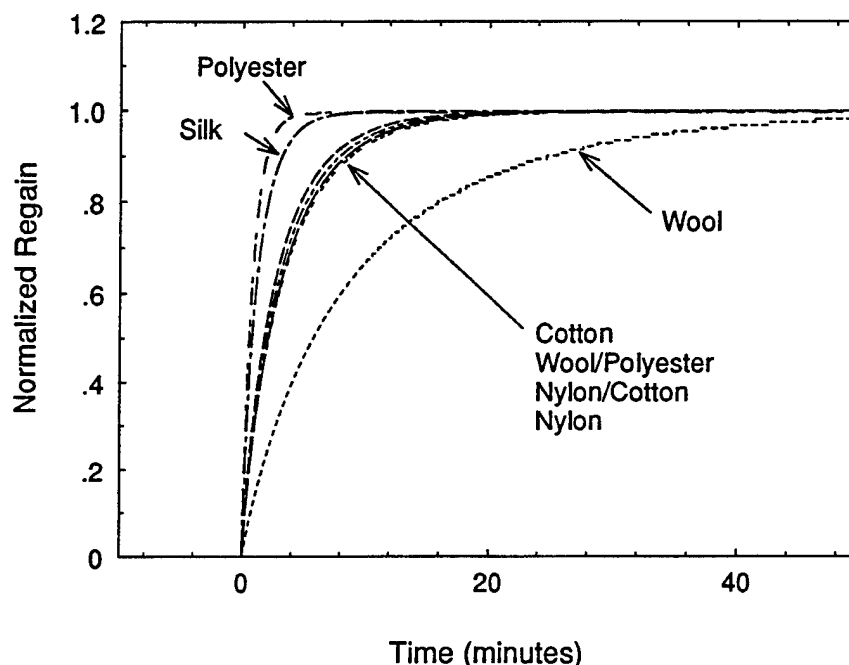


Figure 39. Fractional approach to equilibrium at 65% relative humidity, 20°C, for seven fabrics.

Figure 39 agrees with the trends shown in Figure 18, Chapter 3, which also shows the fractional approach to equilibrium of the seven fabrics, although under conditions of a humidity gradient across the sample. In both cases, the wool fabric is the slowest, the group of four fabrics (cotton, nylon/cotton, nylon, and wool/polyester) are very similar in their time to equilibrate, and the silk fabric is intermediate between this group and the polyester fabric, which equilibrates the fastest. Figure 39 also makes clear that the wool fabric takes over an hour to come to sorptive equilibrium, and that the experimental set points used in the DMPC of about 30 minutes in Chapter 3 were probably too short for the wool fabric.

5.4 Thermal Hysteresis of Hygroscopic Fabrics due to Vapor Sorption

One of the reasons that hygroscopic fibers such as cotton and wool are thought to be more "comfortable" is that they have the ability to buffer concentration or temperature changes occurring between the environment and the human body. This effect will be investigated in more detail in Chapter 7, but we can use the one-dimensional model to observe the effects of a sudden change in temperature on one side of a fabric barrier.

A practical example that is often cited in the literature is that of a material such as a wool fabric which is used for cool weather clothing. If a person wearing a wool coat inside a building, where it is warm and dry, then goes outside to a colder environment, which may have a higher humidity, the wool coat will buffer the temperature change due to the tendency of the wool fibers to absorb water vapor from the air and release the heat of sorption. These types of effects may also be important in applications such as building insulation, where even though fiberglass itself is not terribly hygroscopic, the resinous binders used to stabilize the batts have an affinity for water, and may cause significant temperature transients due to absorption processes [93].

We can use our one-dimensional numerical model to examine this thermal hysteresis effect for a typical hygroscopic textile. The fabric properties are those of two layers of the wool fabric given in Table A-1, Appendix A. The fabric is first equilibrated at 30°C and 40% relative humidity. The ambient temperature and ambient relative humidity on one side of the fabric are then suddenly changed to 5°C and 90%. The external heat and mass transfer coefficients on both sides of the fabric are kept the same as were used in the previous simulations.

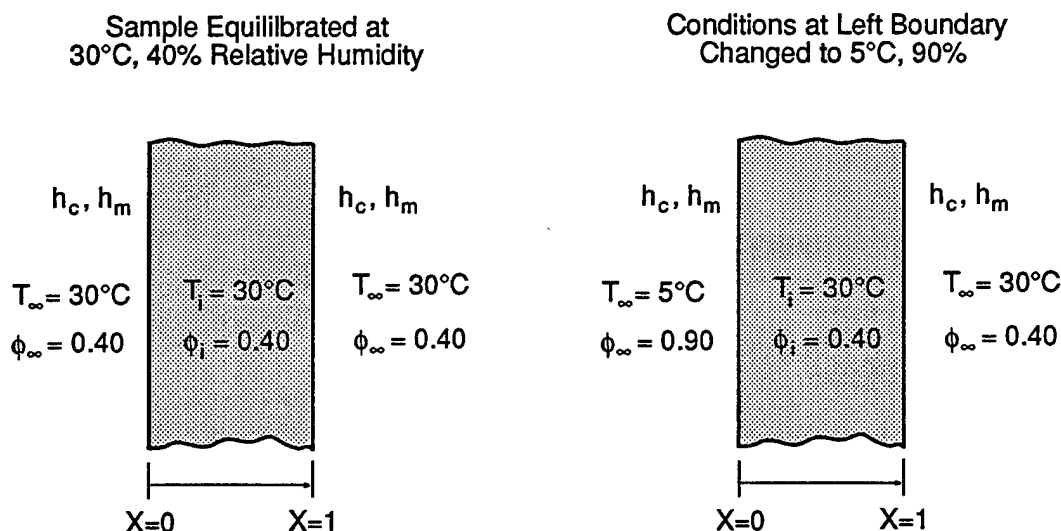


Figure 40. Numerical experiment to simulate sudden temperature and relative humidity change on one side of a fabric.

We examine the performance of the one-dimensional model for two cases. In Case I, we turn off the hygroscopic character of the fibers by setting the fabric regain to zero. In Case II, we turn the normal hygroscopic character of the wool fibers back on by using the proper value for the regain of wool. All the other properties of the fabric remain the same, and we can observe the consequences of vapor sorption as unaffected by any other physical properties (although we should mention that the thermal conductivity and heat capacity of the hygroscopic fabric change slightly over time as the fabric's proportion of bound water increases).

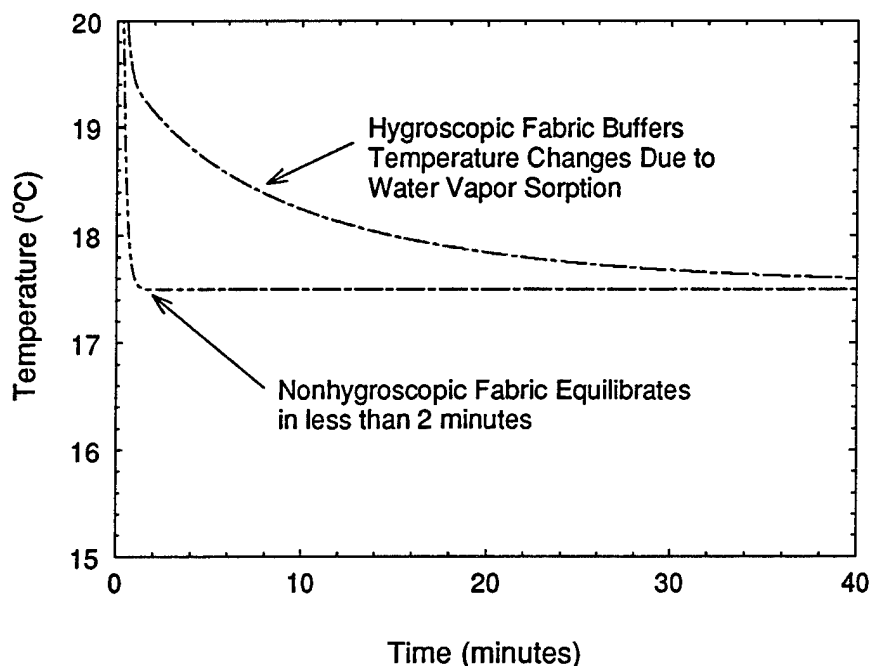


Figure 41. Change in centerline temperature of a hygroscopic and a nonhygroscopic fabric as the ambient conditions are changed on one side.

We see the expected behavior of a hygroscopic fabric as compared to the nonhygroscopic fabric. For Case I, the nonhygroscopic fabric, the material equilibrates very quickly, and the temperature of the center of the fabric quickly assumes the mean temperature of the two sides (17.5°C) valid for steady-state conduction, with constant heat capacity and thermal conductivity. For Case II, the hygroscopic fabric, significant vapor sorption takes place which serves an energy source within the fabric. The equilibration time thus takes much longer: 30 minutes for the case of the hygroscopic fabric versus 2 minutes for the nonhygroscopic fabric.

We may also show the calculated temperature profiles within the fabric at different times after the ambient temperature and humidity are changed on one side of the fabric.

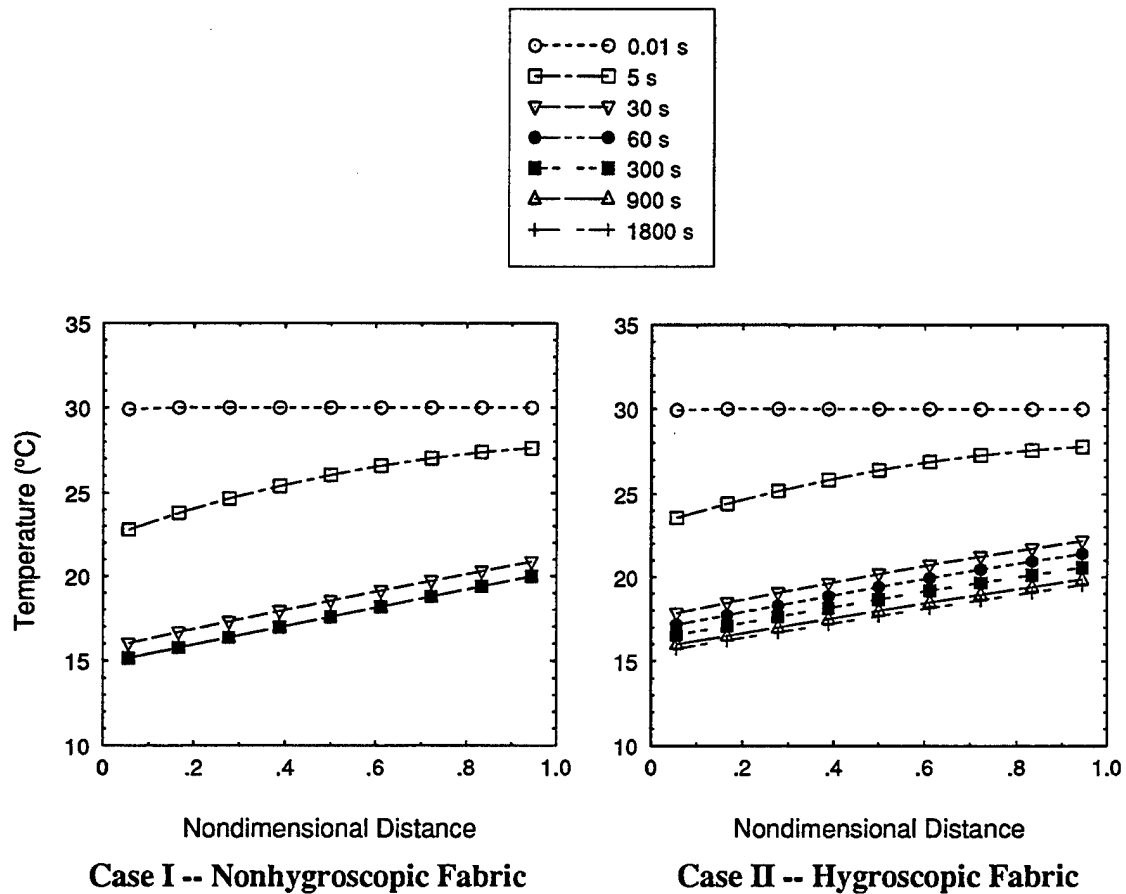


Figure 42. Calculated temperature profiles through the nonhygroscopic and the hygroscopic fabric at various times after ambient conditions on one side are changed.

We may also use the numerical code to examine the gas phase water vapor concentration profile within the fabric. Figure 43 shows the calculated water vapor concentration through the fabric thickness, for the hygroscopic and nonhygroscopic fabrics as a function of time. Vapor concentration is chosen rather than relative humidity, since the saturation vapor pressure is a function of temperature, and relative humidity can be confusing when temperature gradients are involved.

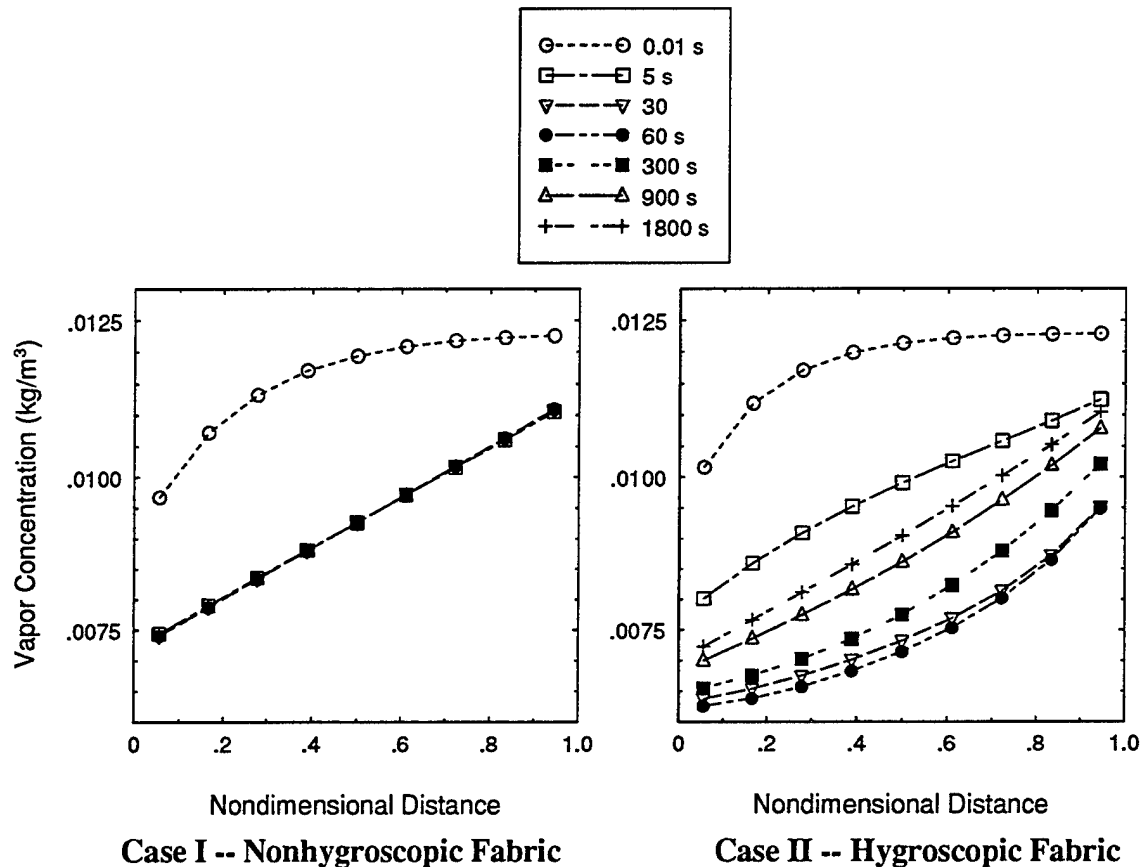


Figure 43. Calculated water vapor concentration profiles through the nonhygroscopic and the hygroscopic fabric at various times after conditions on one side are changed.

Figure 43 shows that the nonhygroscopic fabric equilibrates within a matter of seconds, but that the hygroscopic wool fabric takes much longer due both to the sorption kinetics of the fibers as they come to equilibrium with the local water vapor concentration in the fabric pores, but also because of value of the local temperature, which affects the gas phase partial vapor pressures.

There are a great many more situations which could be examined with this model, e.g. transient behavior under changes in boundary temperatures or changes in external heat and mass transfer coefficients. Most of these situations are deferred to Chapter 6, where a more complete treatment of the problem is presented which includes a two-dimensional geometry, the inclusion of convection through the porous material, and the numerical simulation of the fluid dynamic behavior and heat/mass transfer properties of the gas phase.

5.5 Effect of Grid Size and Time Step on Numerical Results

The grid size and time step were varied systematically to observe the effect on the results obtained with the one-dimensional numerical code. This was done previously in Chapter 4 for the nondimensional diffusion equation, with no source terms. The inclusion of a nonlinear source term, along with the time- and spatially-dependent transport properties, complicate matters significantly. For example, the Crank-Nicolson scheme did not work well when large source terms are included in the finite-difference equations. Although the Crank-Nicolson scheme performs well for the diffusion equation with no source terms, it had a lot of stability and convergence problems when it was used for the hygroscopic sorption case, so it is not mentioned further. The two remaining schemes, the exponential and the fully implicit, both performed identically over the range of grid sizes and time steps explored. Here again, when the source term is the dominant factor, the differences between the various finite-difference schemes disappears, especially when a relaxation method is used within a time step.

The case of the cotton fabric, subjected to a step change of 0.00 to 0.65 relative humidity, at 20°C, was used to examine the effect of both grid size and time step and the numerical results of the one-dimensional code. Since there was little difference between the exponential and the fully implicit schemes, the exponential scheme is used to explore the effects of time step and grid size.

Figure 44 shows that the number of grid points has a small effect on the accuracy of the results.

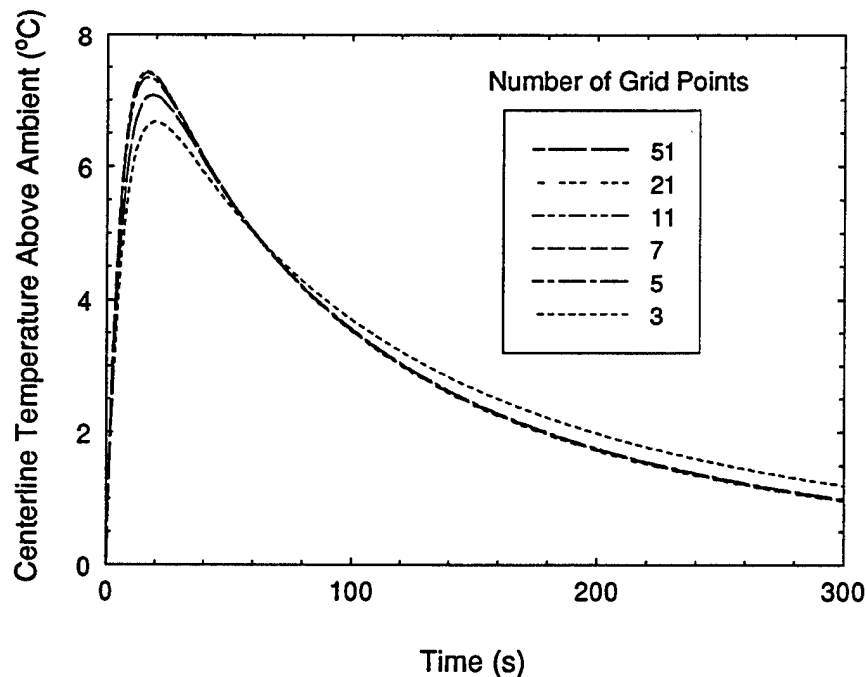


Figure 44. Effect of number of grid points on temperature transients during water vapor sorption of a hygroscopic fabric.

Using the predicted centerline temperature as a guide, Figure 44 shows that once the number of grid points exceeds five, the results are essentially the same. Even for three grid points, the results are about as accurate in terms of the fit to the experimental data, as the larger number of grid points. This is important for the modeling conducted in Chapter 7, where the numerical model for the transient diffusion behavior of hygroscopic clothing is combined with a numerical model of the human thermoregulatory system. The ability to use a small number of grid spacings makes the integrated model run much more quickly.

The effect of the time step used is shown in Figure 45.

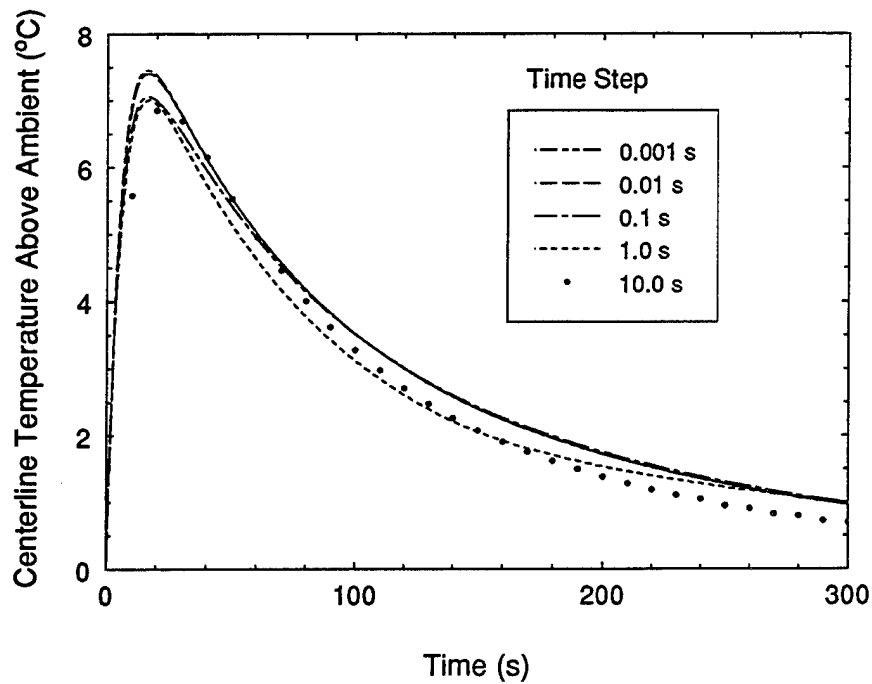


Figure 45. Effect of time step on calculation of temperature transients during water vapor sorption for a hygroscopic fabric.

5.6 Effect of Different Boundary Heat Transfer Coefficients on Numerical Results

In Section 5.2, we noted that we had to assume a heat transfer coefficient, and one could choose values that ranged from about 20 to 40 W/(m²·s). We may use the one-dimensional numerical code to examine the consequences of using various assumptions about the boundary heat transfer coefficients. In addition to varying the convective heat transfer coefficient, we may also include an effective radiative heat transfer coefficient, and examine the importance of thermal radiation heat losses on the calculated temperature transients.

The simplest thermal radiation thermal radiation heat transfer coefficient may be defined as that between two parallel plates at different temperatures [99]. An effective heat transfer coefficient is defined as the sum of the convective heat transfer coefficient (h_c) and a thermal radiative heat transfer coefficient (h_r):

$$h_{eff} = (h_c + h_r) \quad (5.37)$$

We will assume that the emissivity of the fabric and the plastic duct are both equal to 0.9, which allows the thermal radiative heat transfer coefficient to be written as:

$$h_r = \epsilon \sigma (T_f + T_w)(T_f^2 - T_w^2) \quad (5.38)$$

- ϵ emissivity (assumed to be 0.9)
- σ Stefan Boltzmann constant [5.669×10^{-8} W/(m²·K⁴)]
- T_f fabric temperature (K)
- T_w duct wall temperature (K)

Figure 46 shows the effects of three different boundary heat transfer coefficients on the calculated temperature transients for the cotton fabric, with the sorption rate factor adjusted to give equal peak temperatures, as listed in Appendix B. There are some differences between the different assumptions, but referring back to the experimental data scatter contained in Figure 17 in Chapter 3, the differences are fairly minor.

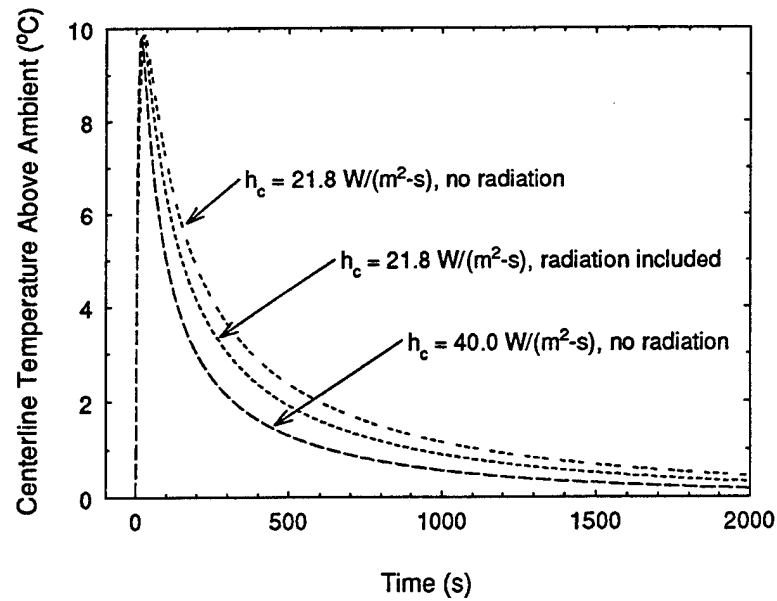


Figure 46. Effect of different boundary heat transfer coefficients on calculation of temperature transients during water vapor sorption for a cotton fabric.

Figure 47 shows how the different heat transfer coefficient assumptions affect the rate at which the cotton fabric approaches sorptive equilibrium. In this case, the differences are due more to the different sorption rate factors, rather than a direct influence of the boundary heat transfer coefficient.

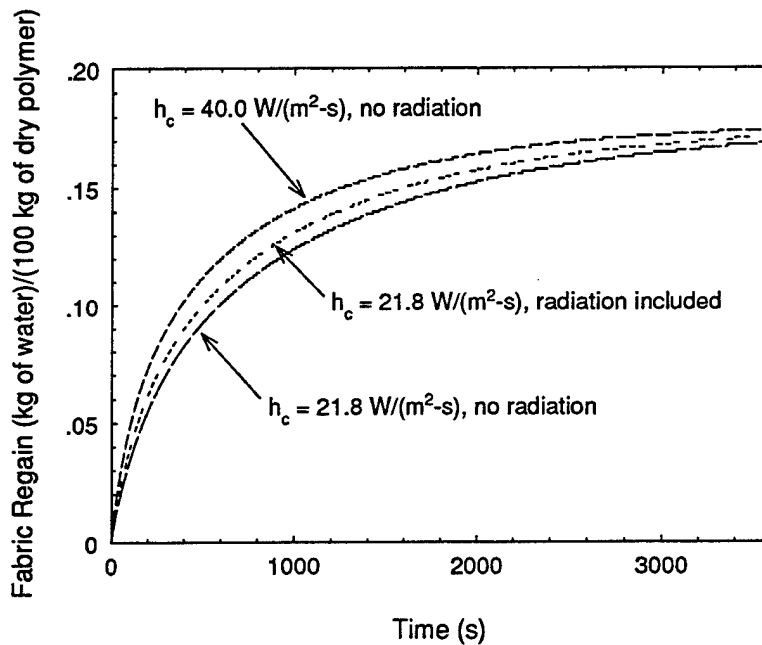


Figure 47. Effect of different boundary heat transfer coefficients on water vapor sorption for a cotton fabric.

5.7 Conclusions

The one-dimensional model of coupled diffusion of mass and energy in hygroscopic porous materials performs well in predicting and simulating temperature changes as a function of time, and weight changes as a function of time -- two quantities which are easily measured experimentally. The one-dimensional model neglects factors such as the convective transfer of energy and mass through the porous structure of textiles, which is often of more practical significance than the diffusion properties of clothing systems. The one-dimensional model also neglects radiative transfer, although this can be incorporated in a fairly simple fashion by an effective thermal conductivity which accounts for radiative transfer. The next chapter will present a more sophisticated model which includes both diffusive and convective transport through porous materials.

CHAPTER 6

TWO-DIMENSIONAL MODELING OF DIFFUSION AND DIFFUSION/CONVECTION PROCESSES IN TEXTILES

6.1 Introduction

The previous chapter addressed one-dimensional transient coupled diffusion. The effects of convective gas flow were not included. Convective heat and mass transfer in porous media is often more important than transport due to diffusion, especially if such materials are used in conditions where a large pressure gradient is present. Laboratory test methods for textiles usually concentrate on one transport mechanism, to the exclusion of the others. Diffusion test methods are particularly easy to perform, and often become the primary ranking and evaluation method for determining the transport properties of textiles. Such test methods can be very misleading for textiles, particularly those which have high air permeability, since a very small pressure gradient can produce large convective flows through the porous structure, which overwhelms any diffusive transport which takes place.

In this chapter, we present the governing equations for the two-dimensional coupled diffusion/convection of energy and mass through hygroscopic porous materials. We solve the system of equations in a computational domain which simulates the geometry of the dynamic moisture permeation cell (DMPC), which was described in Chapter 3. We then compare the numerical results to experimental results for the situations of transient coupled diffusion (similar to Chapter 5), and for the case of convection/diffusion, where both pressure gradients and concentration gradients are present within the DMPC.

6.2 Two-Dimensional Governing Differential Equations

The set of modified equations which describe the coupled convective/diffusive transfer of heat and mass through hygroscopic porous materials are summarized below. The equations derive from the general governing equations given previously (Equations 2.8.1-24 in Chapter 2). In these equations, the major simplifying assumption is that there is no liquid phase present. Since these equations are to be solved using the SIMPLEC code, they are algebraically manipulated until they are each in the general form:

$$\rho \frac{\partial \phi}{\partial t} + \rho u \frac{\partial \phi}{\partial x} + \rho v \frac{\partial \phi}{\partial y} = \frac{\partial}{\partial x} \left(\Gamma_{\phi} \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma_{\phi} \frac{\partial \phi}{\partial y} \right) + S_{\phi} \quad (6.1)$$

Gas phase continuity equation

$$\nabla \cdot \left(\langle \rho_r \rangle^r \langle \bar{v}_r \rangle \right) = \langle \dot{m}_{sv} \rangle - \frac{\partial}{\partial t} \left(\epsilon_r \langle \rho_r \rangle^r \right) \quad (6.2)$$

Gas phase diffusion equations

water vapor:

$$\frac{\partial}{\partial t}(\epsilon_\gamma \langle \rho_v \rangle^\gamma) + \nabla \cdot (\langle \rho_v \rangle^\gamma \langle \vec{v}_\gamma \rangle) - \langle \dot{m}_{sv} \rangle = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \approx \nabla \cdot (\mathcal{D}_{eff} \nabla \langle \rho_v \rangle^\gamma) \quad (6.3)$$

or

$$\frac{\partial}{\partial t}(\langle \rho_v \rangle^\gamma) + \nabla \cdot (\langle \rho_v \rangle^\gamma \langle \vec{v}_\gamma \rangle) \approx \nabla \cdot (\mathcal{D}_{eff} \nabla \langle \rho_v \rangle^\gamma) + \langle \dot{m}_{sv} \rangle \left(1 - \frac{\langle \rho_v \rangle^\gamma}{\rho_w} \right) + (1 - \epsilon_\gamma) \frac{\partial \langle \rho_v \rangle^\gamma}{\partial t} \quad (6.4)$$

It is easier to get the diffusion equation into the general form in terms of mass fraction ρ_v/ρ_γ (which is also more exact):

$$\frac{\partial}{\partial t} \left(\epsilon_\gamma \langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right) + \nabla \cdot \left(\langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \langle \vec{v}_\gamma \rangle \right) = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} + \langle \dot{m}_{sv} \rangle \quad (6.5)$$

or

$$\begin{aligned} \frac{\partial}{\partial t} \left\{ \langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} + \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \langle \vec{v}_\gamma \rangle \right\} &= \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \\ &+ \left[\left[1 - \left(\frac{\langle \rho_\gamma \rangle^\gamma}{\rho_w} \right) \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \langle \dot{m}_{sv} \rangle \right. \\ &\left. + (1 - \epsilon_\gamma) \frac{\partial}{\partial t} \left[\langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \right] \end{aligned} \quad (6.5)$$

or

$$\begin{aligned}
 & \langle \rho_\gamma \rangle^\gamma \frac{\partial}{\partial t} \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) + \langle \rho_\gamma \rangle^\gamma \left\{ \langle u_\gamma \rangle \left[\frac{\partial}{\partial x} \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] + \langle v_\gamma \rangle \left[\frac{\partial}{\partial y} \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \right\} \\
 &= \left\{ \frac{\partial}{\partial x} \left[\langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \frac{\partial}{\partial x} \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] + \frac{\partial}{\partial y} \left[\langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \frac{\partial}{\partial y} \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \right\} + \left\{ \left[1 - \left(\frac{\langle \rho_\gamma \rangle^\gamma}{\rho_w} \right) \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \langle \dot{m}_{sv} \rangle \right. \\
 & \quad \left. + (1 - \epsilon_\gamma) \frac{\partial}{\partial t} \left[\langle \rho_\gamma \rangle^\gamma \left(\frac{\langle \rho_v \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right] \right\} \quad (6.6)
 \end{aligned}$$

air/nitrogen (not required):

$$\frac{\partial}{\partial t} (\epsilon_\gamma \langle \rho_a \rangle^\gamma) + \nabla \cdot (\langle \rho_a \rangle^\gamma \langle \bar{v}_\gamma \rangle) = \nabla \cdot \left\{ \langle \rho_\gamma \rangle^\gamma \mathcal{D}_{eff} \nabla \left(\frac{\langle \rho_a \rangle^\gamma}{\langle \rho_\gamma \rangle^\gamma} \right) \right\} \approx \nabla \cdot (\mathcal{D}_{eff} \nabla \langle \rho_a \rangle^\gamma) \quad (6.7)$$

Total thermal energy equation

$$\langle \rho \rangle \langle C_p \rangle \frac{\partial \langle T \rangle}{\partial t} + \left[(c_p)_\gamma \langle \rho_\gamma \rangle^\gamma \langle \bar{v}_\gamma \rangle \right] \cdot \nabla \langle T \rangle + (Q_i + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle = \nabla \cdot (k_{eff} \nabla \langle T \rangle) \quad (6.8)$$

or

$$\rho C_p \frac{\partial T}{\partial t} + (c_p)_\gamma \rho_\gamma \left(u_\gamma \frac{\partial T}{\partial x} + v_\gamma \frac{\partial T}{\partial y} \right) + (Q_i + \Delta h_{vap}) \dot{m}_{sv} = \frac{\partial}{\partial x} \left(k_{eff} \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k_{eff} \frac{\partial T}{\partial y} \right) \quad (6.9)$$

or

$$\begin{aligned}
 & (c_p)_\gamma \langle \rho_\gamma \rangle^\gamma \frac{\partial \langle T \rangle}{\partial t} + (c_p)_\gamma \langle \rho_\gamma \rangle^\gamma \left(\langle u_\gamma \rangle \frac{\partial \langle T \rangle}{\partial x} + \langle v_\gamma \rangle \frac{\partial \langle T \rangle}{\partial y} \right) = \frac{\partial}{\partial x} \left(k_{eff} \frac{\partial \langle T \rangle}{\partial x} \right) + \frac{\partial}{\partial y} \left(k_{eff} \frac{\partial \langle T \rangle}{\partial y} \right) \\
 & \quad - \left\{ (Q_i + \Delta h_{vap}) \langle \dot{m}_{sv} \rangle + \left[\langle \rho \rangle \langle C_p \rangle - (c_p)_\gamma \langle \rho_\gamma \rangle^\gamma \right] \frac{\partial \langle T \rangle}{\partial t} \right\} \quad (6.10)
 \end{aligned}$$

Gas phase equation of motion for the porous media :

$$\langle \bar{v}_\gamma \rangle = - \left(\frac{K_\gamma}{\mu_\gamma} \right) \varepsilon_\gamma \nabla (\langle p_v \rangle^\gamma + \langle p_d \rangle^\gamma) = - \left(\frac{K_\gamma}{\mu_\gamma} \right) \varepsilon_\gamma \nabla \langle p_\gamma \rangle^\gamma = - \left(\frac{K_\gamma}{\mu_\gamma} \right) \nabla \langle p_\gamma \rangle \quad (6.11)$$

so:

$$\langle u_\gamma \rangle = - \left(\frac{\varepsilon_\gamma K_\gamma}{\mu_\gamma} \right) \left(\frac{\partial \langle p_\gamma \rangle^\gamma}{\partial x} \right) = - \frac{K_\gamma}{\mu_\gamma} \frac{\partial \langle p_\gamma \rangle}{\partial x} \quad (6.12)$$

and

$$\langle v_\gamma \rangle = - \left(\frac{\varepsilon_\gamma K_\gamma}{\mu_\gamma} \right) \left(\frac{\partial \langle p_\gamma \rangle^\gamma}{\partial y} \right) = - \frac{K_\gamma}{\mu_\gamma} \frac{\partial \langle p_\gamma \rangle}{\partial y} \quad (6.13)$$

which results in a general momentum equation :

x-momentum equation:

$$\langle \rho_\gamma \rangle^\gamma \frac{\partial \langle u_\gamma \rangle}{\partial t} + \langle \rho_\gamma \rangle^\gamma \left(\langle u_\gamma \rangle \frac{\partial \langle u_\gamma \rangle}{\partial x} + \langle v_\gamma \rangle \frac{\partial \langle u_\gamma \rangle}{\partial y} \right) = \mu_\gamma \left(\frac{\partial^2 \langle u_\gamma \rangle}{\partial x^2} + \frac{\partial^2 \langle u_\gamma \rangle}{\partial y^2} \right) - \frac{\partial \langle p_\gamma \rangle}{\partial x} - \left(\frac{\mu_\gamma}{K_\gamma} \right) \langle u_\gamma \rangle \quad (6.14)$$

y-momentum equation:

$$\langle \rho_\gamma \rangle^\gamma \frac{\partial \langle v_\gamma \rangle}{\partial t} + \langle \rho_\gamma \rangle^\gamma \left(\langle u_\gamma \rangle \frac{\partial \langle v_\gamma \rangle}{\partial x} + \langle v_\gamma \rangle \frac{\partial \langle v_\gamma \rangle}{\partial y} \right) = \mu_\gamma \left(\frac{\partial^2 \langle v_\gamma \rangle}{\partial x^2} + \frac{\partial^2 \langle v_\gamma \rangle}{\partial y^2} \right) - \frac{\partial \langle p_\gamma \rangle}{\partial y} - \left(\frac{\mu_\gamma}{K_\gamma} \right) \langle v_\gamma \rangle \quad (6.15)$$

Note that the momentum equations given above are really only valid for the bulk fluid. They are not of a proper form to give the time-dependent flow in the porous medium, they will not give the channeling effect at the walls which we know to be present in porous media, and they do not give a pore velocity or pore pressure. But they do assure continuity of flow across the porous media by using only the apparent filter velocity, and they assure continuity of pressure across the porous media. More correct relations for the momentum equation in porous media may be found in References [100-102].

Solid phase continuity equation:

$$\rho_w \frac{\partial}{\partial t} (\varepsilon_{bw}) + \langle \dot{m}_{sv} \rangle = 0 \quad \text{or} \quad \frac{\partial}{\partial t} (\varepsilon_{bw}) + \frac{\langle \dot{m}_{sv} \rangle}{\rho_w} = 0 \quad (6.16)$$

Volume fraction constraint:

$$\varepsilon_\gamma + \varepsilon_{bw} + \varepsilon_{ds} = 1 \quad (6.17)$$

Thermodynamic relations

$$\begin{aligned}
 \langle p_\gamma \rangle^\gamma &= \frac{\langle p_\gamma \rangle}{\epsilon_\gamma} \\
 \langle p_a \rangle^\gamma &= \langle p_\gamma \rangle^\gamma - \langle p_v \rangle^\gamma \\
 \langle p_a \rangle^\gamma &= \langle \rho_a \rangle^\gamma \frac{R}{M_a} \langle T \rangle \\
 \langle p_v \rangle^\gamma &= \langle \rho_v \rangle^\gamma \frac{R}{M_w} \langle T \rangle
 \end{aligned} \tag{6.17-20}$$

Sorption Relation

$$\epsilon_{bw} = 0.578 R_f \left(\epsilon_{ds} \frac{\rho_{ds}}{\rho_w} \right) \left(\frac{\langle p_v \rangle^\gamma}{\langle p_s \rangle^\gamma} \right) \left[\frac{1}{\left(0.321 + \frac{\langle p_v \rangle^\gamma}{\langle p_s \rangle^\gamma} \right)} + \frac{1}{\left(1.262 - \frac{\langle p_v \rangle^\gamma}{\langle p_s \rangle^\gamma} \right)} \right] \tag{6.21}$$

Transport Coefficients and Mixture Properties

The volume-averaged notation is again dropped, with the understanding that the variables have already been expressed as phase averages or intrinsic phase averages (as defined in the Nomenclature section).

As we did in Chapter 5, the source term due to vapor sorption is modeled by assuming that diffusion into the fiber is quasi-steady state. We assume that the polymer at the fiber's surface immediately comes into equilibrium with the relative humidity of the gas phase within the control volume for that grid point. We then calculate the mass flux into or out of the fiber by:

$$\dot{m}_{sv} = \frac{D_{solid} \rho_{solid}}{d_f^2} (R_f - R_{skin}) \tag{6.21}$$

D_{solid} = effective solid phase diffusion coefficient (m^2 / s)

ρ_{solid} = dry solid phase density (kg / m^3)

d_f^2 = effective fiber diameter (m)

R_{skin} = equilibrium regain at fiber surface

R_f = fiber regain from last time step

$$k_{eff} = k_\gamma \left\{ \frac{[1 + (\varepsilon_\beta + \varepsilon_{bw} + \varepsilon_{ds})]k_\sigma + \varepsilon_\gamma k_\gamma}{\varepsilon_\gamma k_\sigma + [1 + (\varepsilon_\beta + \varepsilon_{bw} + \varepsilon_{ds})]k_\gamma} \right\} \quad (6.22)$$

$$k_\sigma = \left(\frac{k_w \rho_\beta \varepsilon_\beta + k_w \rho_\beta \varepsilon_{bw} + k_{ds} \rho_{ds} \varepsilon_{ds}}{\rho_\beta \varepsilon_\beta + \rho_\beta \varepsilon_{bw} + \rho_{ds} \varepsilon_{ds}} \right) \quad (6.23-24)$$

$$k_\gamma = \left(\frac{k_v \rho_v + k_a \rho_a}{\rho_v + \rho_a} \right)$$

$$\mathcal{D}_{eff} = \frac{\mathcal{D}_a \varepsilon_\gamma}{\tau} \quad (6.24)$$

$$\rho_\gamma = \rho_v + \rho_a \quad (6.25)$$

$$(c_p)_\gamma = \frac{[\rho_v (c_p)_v + \rho_a (c_p)_a]}{\rho_\gamma} \quad (6.26)$$

$$\rho = \varepsilon_{bw} \rho_w + \varepsilon_{ds} \rho_{ds} + \varepsilon_\gamma (\rho_v + \rho_a) \quad (6.27)$$

$$C_p = \frac{\varepsilon_{bw} \rho_w (c_p)_w + \varepsilon_{ds} \rho_{ds} (c_p)_{ds} + \varepsilon_\gamma [\rho_v (c_p)_v + \rho_a (c_p)_a]}{\rho} \quad (6.28)$$

$$K_\gamma = \frac{\Delta x}{R(\phi)}$$

$$R(\phi) = R_{dry} + \left(\frac{\varepsilon_{bw}(\phi)}{\varepsilon_{bwsat}} \right) (R_{sat} - R_{dry}) \quad (6.29-30)$$

Initial Constants and Values Required:

$$T_b, \phi_b, T_0, \phi_0, \varepsilon_{bw0}$$

$$\tau, \varepsilon_{ds}, \rho_w, \rho_{ds}, R_f, (c_p)_{ds}, (c_p)_w, (c_p)_v, (c_p)_a, k_{ds}, k_w, k_v, k_a, M_a, M_w, R$$

$$\mathcal{D}_a \text{ (m}^2 \cdot \text{s}^{-1}\text{)} = 2.23 \times 10^{-5} \left(\frac{T}{273.15} \right)^{1.75}$$

$$Q_l \text{ (J/kg)} = 1.95 \times 10^5 \left(1 - \frac{p_v}{p_s} \right) \left(\frac{1}{\left(0.2 + \frac{p_v}{p_s} \right)} + \frac{1}{\left(1.05 - \frac{p_v}{p_s} \right)} \right) \quad (6.31)$$

$$\Delta h_{vap} \text{ (J/kg)} = 2.792 \times 10^6 - 160T - 3.43T^2 \quad (6.32)$$

$$p_s \text{ (N} \cdot \text{m}^{-2}) = 614.3 \exp \left\{ 17.06 \left[\frac{(T - 273.15)}{(T - 40.25)} \right] \right\} \quad (6.33)$$

Each main equation has been expressed in the general form:

$$\rho \frac{\partial \phi}{\partial t} + \rho u \frac{\partial \phi}{\partial x} + \rho v \frac{\partial \phi}{\partial y} = \frac{\partial}{\partial x} \left(\Gamma_\phi \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma_\phi \frac{\partial \phi}{\partial y} \right) + S_\phi \quad (6.34)$$

or:

$$\frac{\partial \rho \phi}{\partial t} + \nabla(\rho \bar{v} \phi) = \nabla \cdot (\Gamma_\phi \nabla \phi) + S_\phi \quad (6.35)$$

In tabular form, the coefficients for the general form of the momentum, mass, and energy tranport equations are:

Table 6-1. Coefficients for General Transport Equations

<i>Equation</i>	ϕ	ρ	Γ_ϕ	S_ϕ
<i>x-momentum</i>	u	ρ_γ	μ_γ	$\left[-\frac{\partial p_\gamma}{\partial x} \right] - u \left(\frac{\mu_\gamma}{K_\gamma} \right)$
<i>y-momentum</i>	v	ρ_γ	μ_γ	$\left[-\frac{\partial p_\gamma}{\partial y} \right] - v \left(\frac{\mu_\gamma}{K_\gamma} \right)$
<i>Energy</i>	T	$\rho_\gamma (c_p)_\gamma$	k_{eff}	$\left[(c_p)_\gamma \rho_\gamma - \rho C_p \right] \frac{\partial T}{\partial t} - (Q_l + \Delta h_{vap}) \dot{m}_{sv}$
<i>Diffusion</i>	$\frac{\rho_v}{\rho_\gamma}$	ρ_γ	$\rho_\gamma D_{eff}$	$\left[1 - \left(\frac{\rho_\gamma}{\rho_w} \right) \left(\frac{\rho_v}{\rho_\gamma} \right) \right] \langle \dot{m}_{sv} \rangle + (1 - \epsilon_\gamma) \frac{\partial}{\partial t} \left[\rho_\gamma \left(\frac{\rho_v}{\rho_\gamma} \right) \right]$

For the special case of the gas phase continuity equation, we have:

$$\nabla \cdot (\rho_y \bar{v}_y) = \dot{m}_{sv} - \frac{\partial}{\partial t} (\epsilon_y \rho_y) \quad ; \quad S_\phi = \dot{m}_{sv} - \frac{\partial}{\partial t} (\epsilon_y \rho_y) \quad (6.36)$$

6.2 Two-Dimensional Modeling of Convection/Diffusion

The SIMPLEC code and the USER subroutine DIFFCON.F (contained in Appendix C) were used to solve the system of equations 6.2-6.33. The geometry is modeled as shown below in Figure 48.

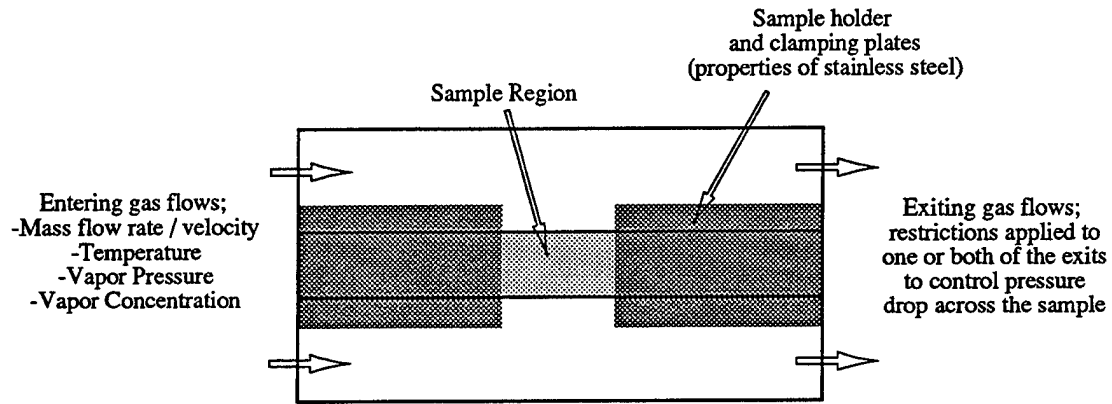


Figure 48. Model geometry to simulate convection/diffusion processes in the DMPC.

The subroutine DIFFCON.F, in Appendix C, has the appropriate grid generation routines, boundary condition specifiers, etc., to define all the properties of each control volume as to the appropriate diffusion coefficients, permeabilities, and boundary conditions (inflow/outflow, adiabatic, impermeable, etc.). For the inlet conditions, the properties of the entering gas flows, such as mass flow rate, temperature, vapor concentration, etc., are specified. The SIMPLEC code then solves for the momentum, concentration, and temperature equations for the entire computational domain. Facilities exist within the SIMPLEC code to solve problems in both a steady-state and a transient mode. Examples of both modes for both transient and steady-state diffusion/convection will be presented in this chapter.

The easiest situation to compare numerical and experimental results with is steady-state diffusion/convection in the DMPC. The experimental situation was shown previously in Figure 19 of Chapter 3. Equal mass flow rates of nitrogen enter two sides of the DMPC, with one flow entering at a relative humidity of 0.0 and the other flow entering at a relative humidity of 1.0. The restrictor valves at the exits of the cell are used to systematically vary the pressure drop across the sample to produce varying amounts of convective flows across the sample. For the fabrics which have a humidity-dependent porosity/Darcy permeability, the relationship between the measured relative humidity and the pressure drop is a good test of the ability of the numerical code to simulate the experimental results.

The seven fabrics listed in Table 3-2, Chapter 3, were tested in the DMPC under convection/diffusion conditions. The SIMPLEC and DIFFCON code given in Appendix C were used to simulate this experiment, using the fabric properties given in Appendix A.

The factor most affecting the results is the humidity-dependent Darcy flow resistance property, defined in equations 6.29-30. Figure 49 shows the comparison of the numerical flow simulation with the experimental results for the cotton fabric. Also included on this plot are numerical results where the Darcy flow resistance was not allowed to vary, but was specified as either R_{dry} (the value obtained using dry nitrogen) or R_{sat} (the value obtained using a relative humidity of 1.0). The use of these constant values for the Darcy flow resistance allow one to bracket the results between the two extremes of measured permeability.

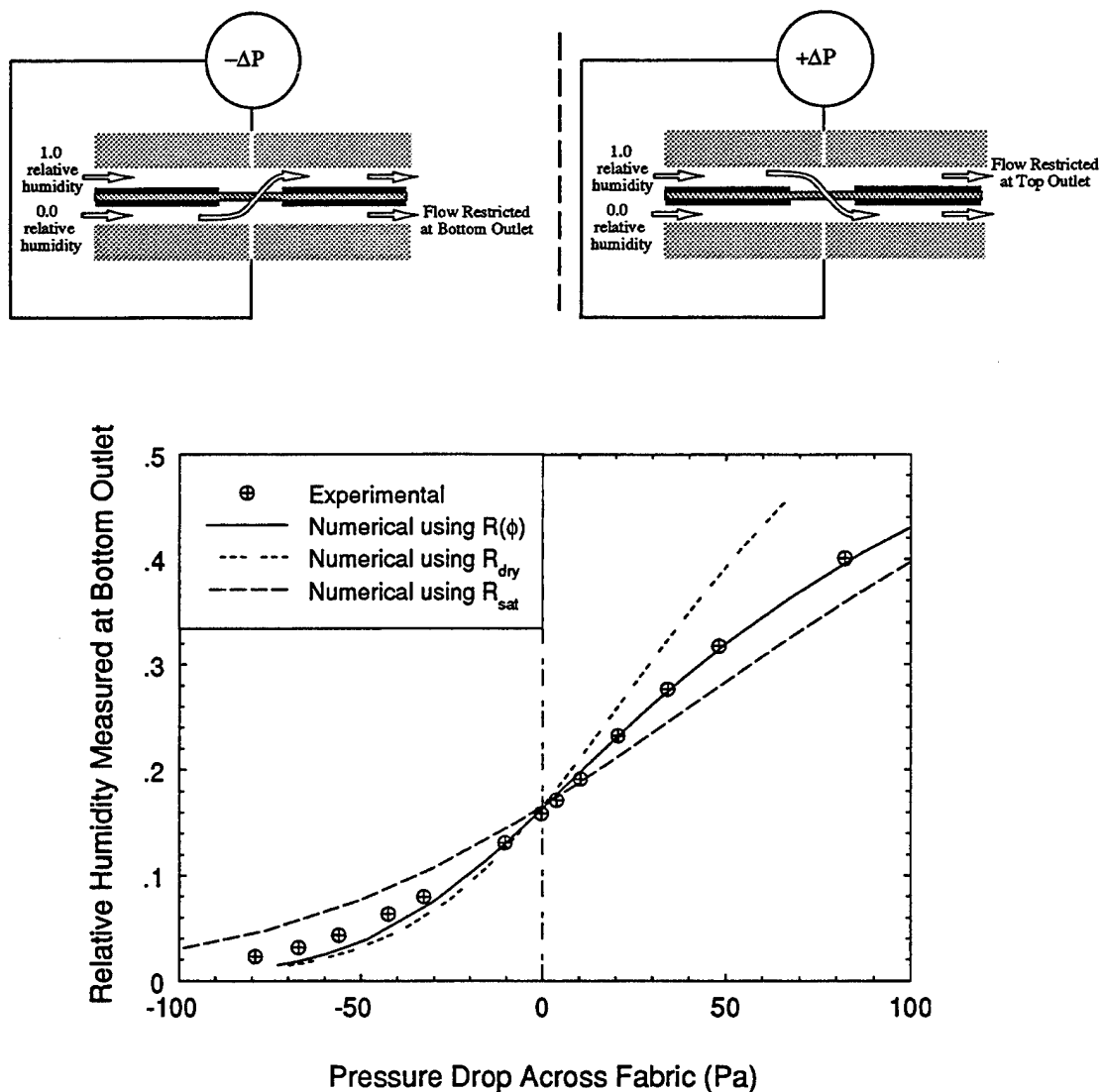


Figure 49. Comparison of experimental versus numerical results for simulation of convection/diffusion in the DMPC for the cotton fabric.

Figure 49 shows that the approximation for the humidity-dependent permeability works well for the situation where the high humidity flow is being forced through the fabric (positive pressure drop), but does not match the experimental data as closely for the situation where the low humidity flow is being forced through the fabric (negative pressure drop). Judging by the fact that the experimental data is just about in the middle of the two bracketing curves which used the value for R_{dry} or R_{sat} , an approximation for the relative humidity-dependent Darcy permeability of just taking the mean or average relative humidity on the two sides of the fabric may work just as well as the more complicated sorption scaling approach given by equations 6.29-30.

The numerical results for the wool fabric, another highly hygroscopic material, are shown in Figure 50. What we notice from this plot is that the data seem to be represented more by the results from using the Darcy flow resistance measured at saturation conditions, than by the sorption scaling approach. Another possible error, due to the experimental method, is the use of equilibration times which were too short for wool (which really should have an hour minimum at each set point).

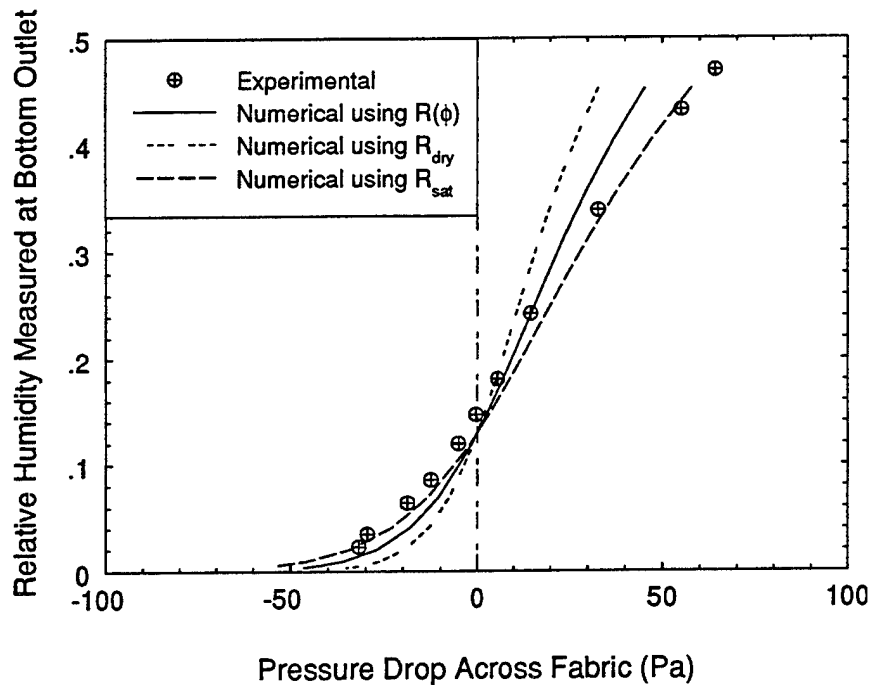


Figure 50. Comparison of experimental versus numerical results for simulation of convection/diffusion in the DMPC for the wool fabric.

We can show comparable results for the other five of our test fabrics, shown in Figures 51 and 52.

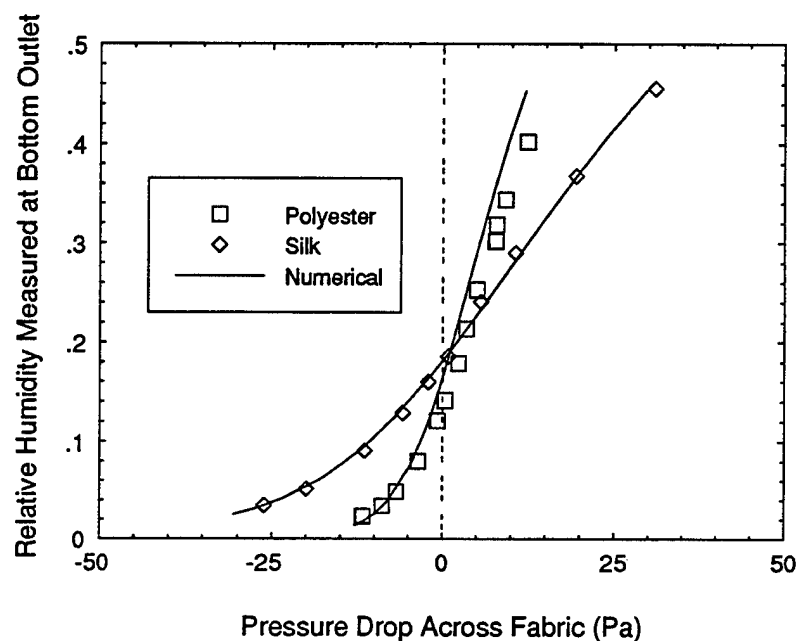


Figure 51. Comparison of experimental versus numerical results for simulation of convection/diffusion in the DMPC for the silk and polyester fabrics.

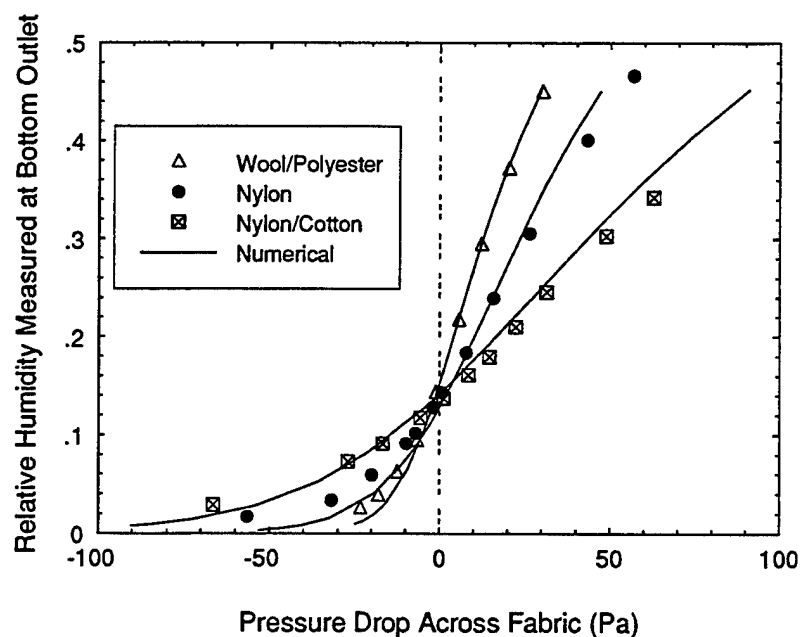


Figure 52. Comparison of experimental versus numerical results for simulation of convection/diffusion in the DMPC for the wool/polyester, nylon/cotton, and nylon fabrics.

We see good agreement of the experimental with the numerical results, especially for the tests which have a positive pressure drop driving convective flow from the high humidity to the low humidity side of the DMPC. The agreement is not as good for the test conditions which have a negative pressure drop, which drives the dry gas against the diffusion gradient present in the DMPC. We can think of several possible explanations for this disagreement. Sorption hysteresis may be responsible for some of the differences. The air permeability tests were conducted starting at the dry condition and working up to the saturated condition. The convection/diffusion experiments in the DMPC were conducted in the opposite order, starting at the high humidity flowing through the sample with a positive pressure gradient, and working down to the negative pressure gradient. Since we did not include a factor to account for sorption hysteresis in our numerical approximation of the sorption isotherm, this could be a cause of the error (in addition to the possible errors due to the sorption scaling approach used for the Darcy resistance as a function of bound water volume fraction). Another possible source of error is the equilibration time allowed at each setpoint. For wool in particular, one should allow equilibration times of at least an hour, but shorter times were used for many of the experimental setpoints for the wool fabric (with the intention of maximizing the number of tests per day)

We have seen generally good agreement between experimental and numerical results which simulate fluid dynamic/structural interactions in hygroscopic porous textiles under steady-state conditions. Since the two dimensional code solves for the variables of temperature, pressure, velocity, concentration, rate of phase change, etc., we have the ability to look at the entire flow field in the DMPC by plotting variables such as velocity and relative humidities.

In Figure 53, we show the entire flow field in the DMPC simulation, for the variables of velocity (indicated by vectors) and relative humidity (indicated by the shaded contours). The two situations shown are for the cotton fabric, under conditions of no pressure drop across the sample (pure diffusion), and under the situation of a positive pressure drop across the sample, which forces the humidified gas to flow across the sample to the low humidity side of the cell.

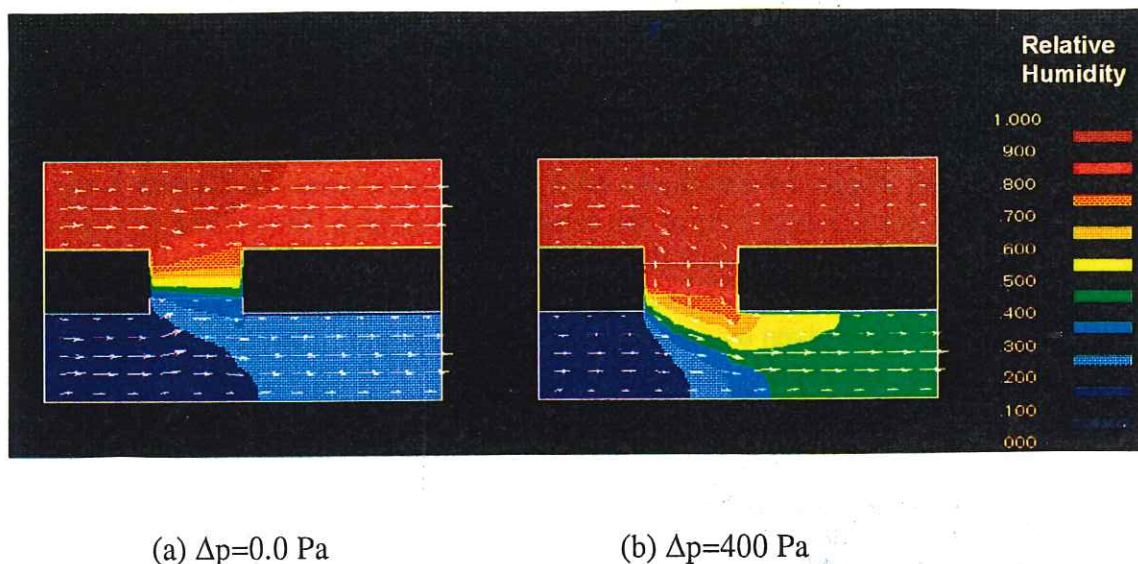


Figure 53. Flow field simulation obtained using numerical solution for cotton fabric in DMPC at two different pressure drop conditions. Gas phase velocity vectors, and contours of relative humidity shown on plot

Figure 53 makes it easier to see some of the phenomena we've been describing; namely, that the flow in a test such as this is two-dimensional in nature, and that the formation of boundary layers results in gradients down the length of the sample. This will be illustrated further in the next section of this chapter which deals with transient diffusion.

6.3 Two-Dimensional Modeling of Transient Diffusion/Sorption

The two-dimensional system of governing equations may also be solved to simulate the case of transient diffusion/sorption, as was done previously for the one-dimensional situation in Chapter 5. We will present only the results for the cotton fabric; qualitatively, there was nothing much different seen in the two-dimensional results than for the one-dimensional results. The experimental results for the transient diffusion/sorption case were obtained using two layers of fabric instrumented with three thermocouples (see Figure 13 in Chapter 3). The sample equilibrated at a relative humidity of 0.0 on both sides under constant nitrogen flow at 20°C (293 K). The relative humidity was then changed to 1.0 on both sides, and the temperature rise and fall due to vapor sorption was recorded as a benchmark for the computer simulations. A modified version of the numerical codes given in Appendix C were used to simulate this situation in the DMPC. Figure 54 below shows the computed and the experimentally-measured temperatures of the fabric as a function of time for the simulation. The numerical results match the trend of the experimental results, particularly the fact that there is a difference between the upstream, center, and downstream thermocouples, due to the formation of concentration and temperature boundary layers down the length of the sample. Figure 54 also shows what we have seen before in Chapter 5; the numerical results do not fall as quickly as the experimental results. We believe this failure to match the rate of temperature drop more precisely is due to 1) the two-dimensional model doesn't include radiation heat transfer, and 2) the very simple model of the sorption kinetics given by equation 6.21 doesn't do a very good job of mimicking the true sorption behavior of cotton or wool fibers.

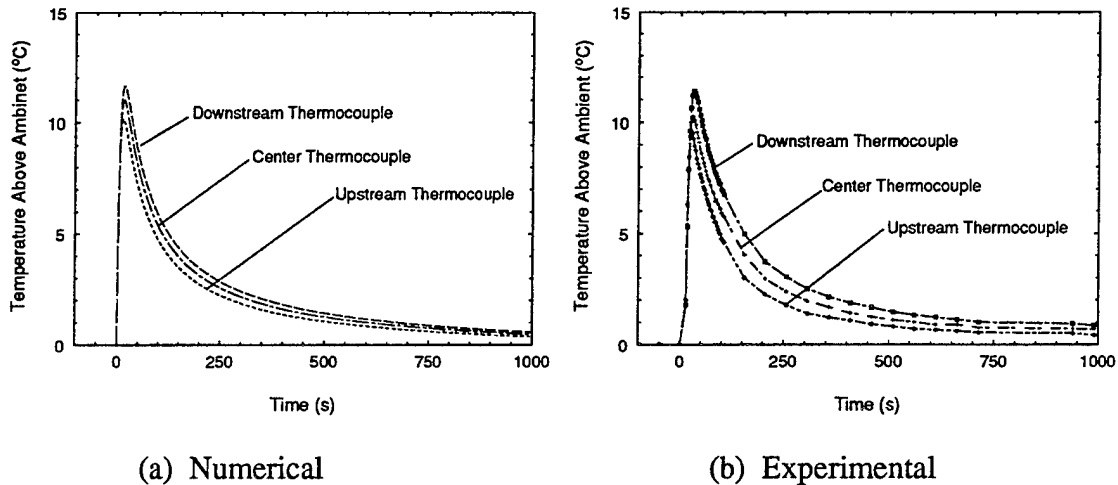


Figure 54. (a) Numerical and (b) experimentally measured temperature transients for a hygroscopic cotton fabric subjected to a sudden change in relative humidity.

Using the two-dimensional information from the flow simulation, we can also plot the entire computational domain as a "snapshot" in time. Figure 55 shows the computed velocity field and the temperatures at different times after the relative humidity of the gas flow is changed to 1.0. We can clearly see the fabric heating up as it absorbs water vapor, and then cooling back down as it comes back to thermal equilibrium. Figure 55 illustrates the interaction between the concentration and thermal boundary layers in the DMPC, and how the flow field of the gas influences the experimental results we obtain in such an apparatus.

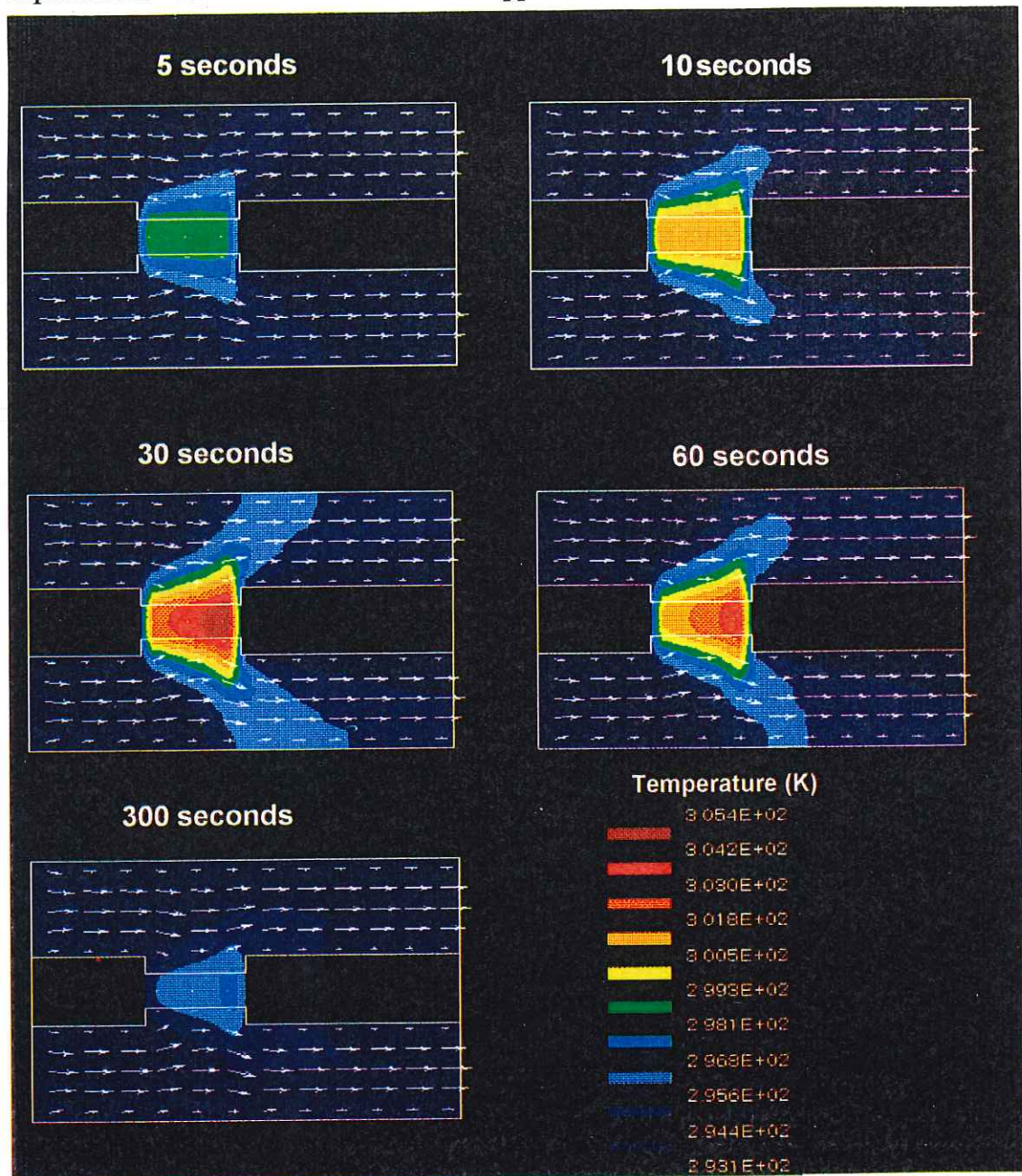


Figure 55. Numerical simulation of flow field and temperature for cotton fabric undergoing water vapor sorption under relative humidity step change from 0.0 to 1.0 at 20°C (293K).

6.4 Conclusions

The ability to conduct the two-dimensional numerical modeling and the experimental testing together is especially valuable in addressing many of the questions we have about the range of applicability of the volume-averaging method for textiles, and the effect of the test equipment itself on the results. For example, we have ignored for the most part the possible existence of microscale pore-level heat and mass transfer coefficients. Our assumption of local thermal equilibrium assumes that the fiber and the gas are at the same temperature, which eliminates the heat transfer coefficient (although such coefficients may exist in reality), but our treatment of mass transfer between fiber and gas implicitly recognizes the existence of a pore-level mass transfer coefficient, which we have not considered in the interest of maintaining simplicity. The effective solid phase diffusivity lumps microscale transfer coefficients, fiber geometry, surface area, and solid phase diffusion coefficients into one parameter. The ability to conduct testing under conditions of different pore flow velocities, and the observation of the equilibration times for a quantity such as pressure drop, is one way of directly observing the relative importance of factors such as a microscale mass transfer coefficient between the gas phase volume and the solid phase volume within the porous material.

The ability to apply the proper form of the two-dimensional transport equations to a particular test device eliminates much of the need to guess or estimate boundary heat and mass transfer coefficients, since they are taken care of automatically in the course of the computational fluid dynamic and heat transfer calculations involved in solving for the gas flow field. This allows one to get a clearer picture of the behavior of the porous material itself, without the complications of the behavior of the particular experimental apparatus intruding themselves into the results.

CHAPTER 7

INTEGRATION OF A HUMAN THERMAL PHYSIOLOGY CONTROL MODEL WITH A NUMERICAL MODEL FOR COUPLED HEAT AND MASS TRANSFER THROUGH HYGROSCOPIC POROUS TEXTILES

7.1 Introduction

The comprehensive numerical model developed for the coupled transport of energy and mass through porous hygroscopic materials was integrated with an existing human thermal physiology model to provide appropriate boundary conditions for the clothing model. The human thermal control model provides skin temperature, core temperature, skin heat flux, and water vapor flux, along with liquid water accumulation at the skin surface. The integrated model couples the dynamic behavior of the clothing system to the human physiology of heat regulation. This model provided the opportunity to systematically examine a number of clothing parameters, which are traditionally not included in steady-state thermal physiology studies, and to determine their potential importance under various conditions of human work rates and environmental conditions.

7.2 Description of Human Thermal Model

We selected a particularly simple human thermal physiology model to integrate with the clothing model. The model includes appropriate control functions for variables such as sweating rate, metabolic activity, blood flow effects, etc. The model is one-dimensional in nature and treats the body as a single thermal zone. In effect, to put it indelicately, the model behaves like a living sweaty slab of tissue that responds to the environment in a manner similar to the way the human body responds. Our reasons for selecting the simplest model we could find was that we wanted to demonstrate the feasibility of applying the comprehensive clothing model to a standard physiological model, and we wanted to do all the programming of the code ourselves, so we did not have to rely on a computer program written by someone else, with all its quirks and special numerical optimization schemes which are often known only to the programmer. We would expect that the next step is to integrate the clothing model with more sophisticated multi-zone human thermal models [103,104], in a manner similar to that done by Jones [105].

The body is modeled as a one-dimensional slab [106,107]. The thermal conductivity of the body is a function of the average body temperature. Blood flow effects are included in the effective thermal conductivity for the tissue. The body is divided into three layers. The core layer generates basal metabolic energy. The muscle layer generates energy due to exercise, or due to shivering. The skin layer produces no heat, but generates sweat at rates specified by the physiological control functions. The body is assumed symmetrical so that the computational domain extends from the body center of symmetry outward to the skin. A schematic of the body model is shown in Figure 56.

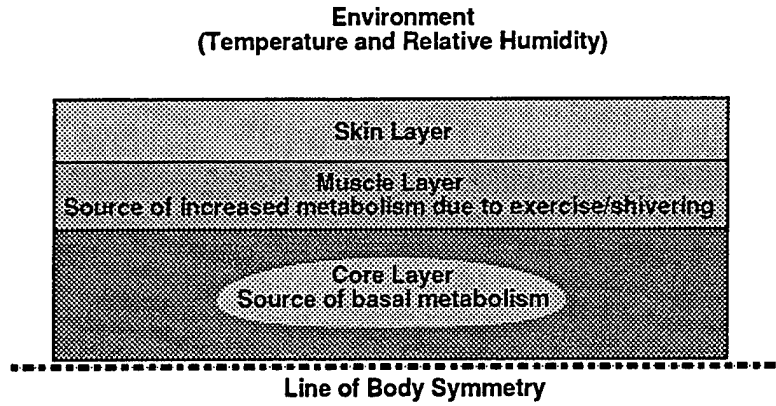


Figure 56. One-dimensional thermoregulatory model of the human body.

The governing equations for the thermal physiology model are given below:

Energy Equation

$$\rho_B c_{pB} \frac{\partial T}{\partial t} = k_B \frac{\partial^2 T}{\partial y^2} + q_m \quad (7.1)$$

ρc_{pB} product of density and heat capacity of body tissues ($4.187 \times 10^6 \text{ J/m}^3\text{-K}$)
 k_B thermal conductivity of body tissue (W/m-K)
 q_m rate of heat generation due to metabolism (W/m³-s)
 t time (s)
 T temperature (K)
 y distance from body center of symmetry (m)

Initial conditions:

$$\text{at } t=0: T=T(y) \quad (7.2)$$

Boundary Conditions:

$$\text{at } y=0: \frac{\partial T}{\partial y} = 0 \quad (7.3)$$

$$\text{at } y=L: k_B \frac{\partial T}{\partial y} = \dot{m} \Delta h_{vap} + h_c (T_{sk} - T_a) \quad (7.4)$$

L	total half-thickness of system (0.056 m)
$\dot{m}$	mass flux of water vapor from skin surface (kg/m ² -s)
Δh_{vap}	enthalpy of vaporization (J/kg)
h_c	convective heat transfer coefficient (W/m ² -s)
T_{sk}	skin temperature (K)
T_a	ambient air temperature (K)

Average body temperature is defined as:

$$T_B = \frac{1}{L} \int_0^L T(\xi, t) d\xi \quad (7.5)$$

T_B average body temperature (K)

The deviation from the body's setpoint is defined as:

$$\Delta T_B = T_B - 35.75 \text{ }^\circ\text{C} = T_B - 308.9 \text{ K} \quad (7.6)$$

ΔT_B deviation from the body's setpoint (K)

The body's thermal conductivity, which varies to simulate vasodilation effects, is defined in terms of a reference thermal conductivity, and proportional and rate control functions which kick in when the average body temperature deviates from the setpoint (defined as 35.75 °C).

For temperatures above the body's setpoint:

$$k_B = k_{B0} \left(1 + \alpha_{k1} \Delta T_B + \gamma \frac{dT_B}{dt} \right) \leq 1.511 k_{B0} \quad \Delta T_B > 0 \quad (7.7)$$

k_{B0} reference body tissue thermal conductivity (0.498 W/m-K)
 α_{k1} thermal proportional control coefficient (1/K)
 γ thermal rate control coefficient (s/K)

For temperatures below the body's setpoint:

$$k_B = k_{B0} \left(1 + \alpha_{k2} \Delta T_B + \gamma \frac{dT_B}{dt} \right) \geq 0.675 k_{B0} \quad \Delta T_B < 0 \quad (7.8)$$

α_{k2} thermal proportional control coefficient (1/K)

The sweating rate from the skin is also defined in terms of deviations from body setpoint.

For temperatures above the setpoint, the sweating rate is given by:

$$s = s_0 + \alpha_{s1}\Delta T_B + \alpha_{s2}(\Delta T_B)^4 \leq 60s \quad \Delta T_B > 0 \quad (7.9)$$

For temperatures below setpoint, the sweating rate is given by the basal rate:

$$s = s_0 \quad \Delta T_B < 0 \quad (7.10)$$

s	sweating rate (kg/m ² -s)
s_0	reference basal sweating rate (2.80 x10 ⁶ kg/m ² -s)
α_{s1}	sweating proportional control coefficient (kg/m ² -s-K)
α_{s2}	sweating proportional control coefficient (kg/m ² -s-K ⁴)

For no clothing layer, with a given mass transfer coefficient, mass flux of water vapor from the skin surface is given by:

$$\dot{m} = \begin{cases} h_m(c_{sk} - c_a) & M_{avl} > 0 \\ \text{minimum of } [s, h_m(c_{sk} - c_a)] & M_{avl} = 0 \end{cases} \quad (7.11)$$

h_m	convective mass transfer coefficient (m/s)
c_{sk}	water vapor concentration at skin surface (kg/m ³)
c_a	water vapor concentration of ambient atmosphere (kg/m ³)
M_{avl}	liquid water available on skin surface for evaporation (kg/m ²)

For a clothing layer, the mass transfer coefficient is replaced by the effective diffusion coefficient of the fabric or air space combined with the thickness of the first control volume above the skin surface.

The mass of liquid water available at the skin surface (M_{avl}) is given by:

$$M_{avl} = M_{avl}|_{t=0} + \int_0^t [s(\xi) - \dot{m}(\xi)] d\xi \quad (7.12)$$

ξ dummy integration variable

Metabolic heat generation is distributed between the various layers:

--No heat generation occurs in the outer skin layer.

--Basal metabolic heat generation takes place in the core layer:

$$(q_m = 1339 \text{ W/m}^3).$$

The muscle layer produces heat due to exercise, or due to shivering:

$$q_m = E \quad \Delta T_B > 0 \quad (7.13)$$

$$q_m = E + \Delta Q_{sh} \quad \Delta T_B < 0 \quad (7.14)$$

$$\Delta Q_{sh} = -\alpha_{sh1} \Delta T_B + \alpha_{sh2} (\Delta T_B)^2 \quad -1 \leq \Delta T_B < 0 \quad (7.15)$$

$$\Delta Q_{sh} = 660.3 (\text{W/m}^3) - \alpha_{sh2} \Delta T_B \quad \Delta T_B < -1 \quad (7.16)$$

E metabolic energy generated in muscle layer due to exercise ($\text{W/m}^3\text{-s}$)

ΔQ_{sh} metabolic energy generated due to shivering ($\text{W/m}^3\text{-s}$)

α_{sh1} shivering proportional control coefficient ($\text{W/m}^3\text{-K}$)

α_{sh2} shivering proportional control coefficient ($\text{W/m}^3\text{-K}^2$)

α_{sh3} shivering proportional control coefficient ($\text{W/m}^3\text{-K}$)

7.3 Integrated Clothing and Human Thermal Model

A schematic of the integrated model is shown below in Figure 57. We use the existing human thermal model, and place two material layers on the skin surface. By specifying the properties of these two layers, we can have the material layers behave like still air layers, or like textile layers. For most of these simulations, we are only looking at the coupled diffusion of heat and water vapor. If there is any extra liquid sweat which builds up on the skin surface, it does not drip off or wick into the fabric. Later in this report we discuss a simulation which uses a very simple way of including liquid sweat wicking. We are also ignoring gas convection due to pressure differences, which can arise either due to body movement, or due to external air movement (wind).

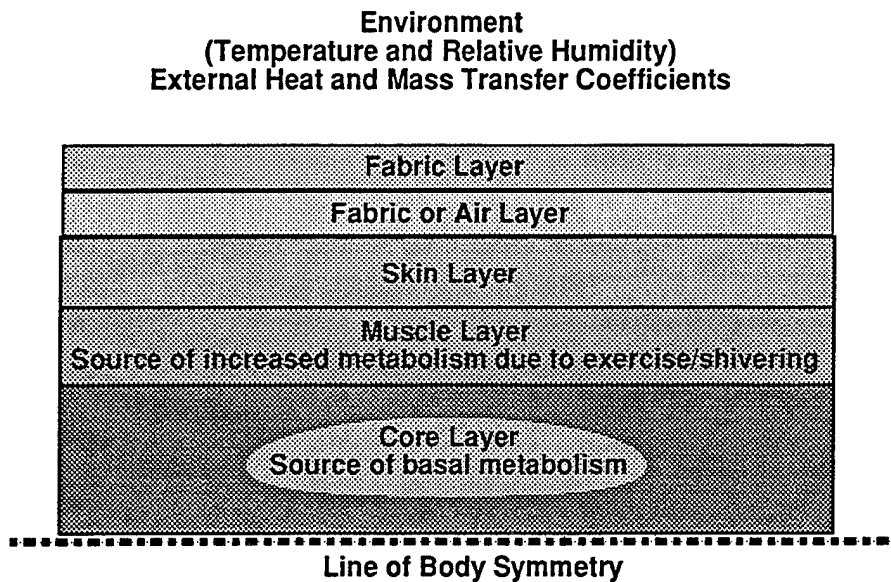


Figure 57. Human thermal control system model combined with clothing material model.

The clothing model uses the one-dimensional model given in Chapter 5, in equations (5.15-33). The original one-dimensional material model used convective heat and mass transfer coefficients at the boundaries. The boundary conditions for the fabric's external surface remain the same in the integrated model, but the boundary conditions for the fabric's inner surface are determined by the vapor flux and heat flux from the human thermal control model. The material properties used for the numerical simulations are given in Table 7-1, and are similar to those for a typical cotton fabric, such as that given in Table 3-2, Chapter 3.

Table 7-1. Assumed Fabric Layer Properties

Thickness:	$9.14 \times 10^{-4} \text{ m}$
Apparent Bulk Density (Dry):	438 kg/m^3
Regain at 65% relative humidity R_f :	0.06
Effective Diffusivity D_{eff} :	$6.09 \times 10^{-6} \text{ m}^2/\text{sec}$
Fabric Tortuosity τ :	2.9
Effective solid phase diffusion coefficient D_{solid} :	$4.0 \times 10^{-14} \text{ m}^2/\text{sec}$
Effective fiber diameter d_f :	$3.6 \times 10^{-6} \text{ m}$
Density of dry polymer ρ_{ds} :	1500 kg/m^3
Heat capacity of dry polymer $(c_p)_{ds}$:	1200 J/kg-K
Thermal conductivity of dry polymer k_{ds} :	0.16 J/sec-m-K

An example of the general model, contained in the USER subroutine ONEREV.F is given in Appendix D. The program is run when compiled with the general one-dimensional equation solver EXP1D.F contained in Appendix B.

7.4 Effect of Changes in Environmental Relative Humidity

We first examine the performance of the integrated model for the situation where the environmental relative humidity is changed, while the environmental temperature remains constant. This situation is chosen to illustrate the difference between a hygroscopic fabric, which actively absorbs water vapor and releases the heat of sorption, and a nonhygroscopic fabric, which remains passive.

One of the ways in which we experimentally verified the material model in the laboratory was the case where we looked at the temperature rise of a hygroscopic fabric due to step changes in relative humidity. It is interesting to conduct the same experiment numerically using the integrated clothing/human thermal physiology model.

We examine two cases for the clothing model. One case is a highly hygroscopic fabric which has a high heat of sorption (Table 7-1). The second case is to take the exact same fabric properties, except to let the regain equal zero at all conditions, which turns the fabric into a nonhygroscopic fabric such as polyester. The nonhygroscopic fabric has all the other properties of the hygroscopic fabric, such as thermal conductivity, heat capacity, thickness, effective diffusivity, etc., but it will not absorb water vapor from the atmosphere.

We start off by letting the human thermal model, with the clothing layer, equilibrate for 3 hours under warm and dry conditions of 35°C and 10% relative humidity. The human work rate is kept constant at 50 Watt/m², which corresponds to a fairly light work rate. To observe the differences between the two fabrics we suddenly change the relative humidity to 90% for 1 hour, beginning at hour 3, and then change the relative humidity back to 10%, at hour 4.

By doing this, we can get a feel for how much skin temperature (which relates to comfort) would be affected by the different regain properties of a hygroscopic versus a nonhygroscopic fabric. In this case we are only looking at environmental changes. We will get some of the same effects if the environmental conditions remain the same, and if we change the metabolic energy generation rate due to exercise, which will change the sweating rate, which will also show up in the heat generated/absorbed in the hygroscopic clothing layer.

Two cases are shown below in Figure 58 for a hygroscopic and a nonhygroscopic fabric: for a human work rate of 50 Watt/m², and an environmental temperature of 35°C, the relative humidity was changed from 10% to 90% for one hour.

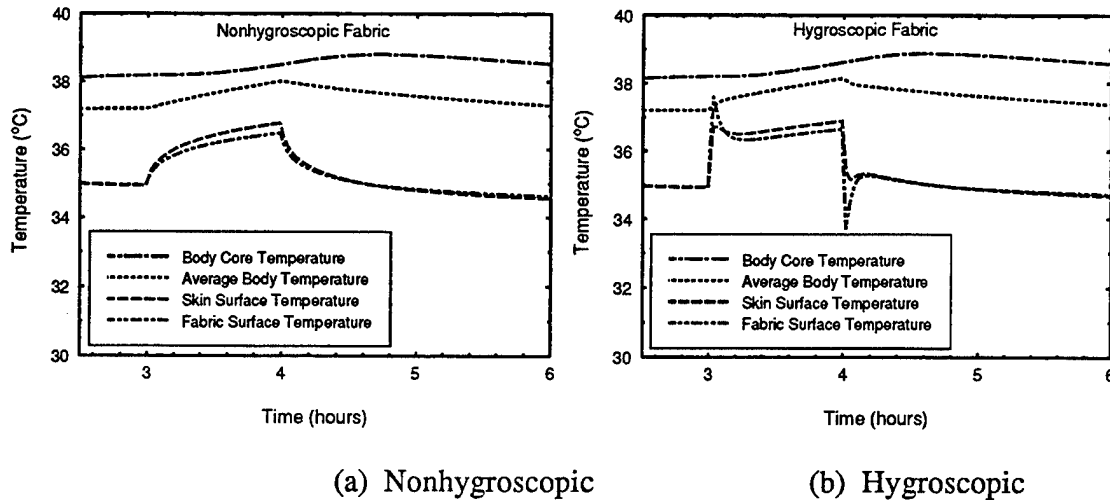


Figure 58. Calculated temperatures for nonhygroscopic and hygroscopic fabrics covering sweating human, when subjected to large changes in environmental relative humidity.

The large differences in skin temperature profiles seen in Figure 58 imply differences in perceived comfort of these two fabrics. For the thin fabrics used in this simulation, we see very little difference between the two fabrics for the overall thermal balance, as reflected by the nearly identical body core temperatures for the two situations. We would expect that some of these short-term transient effects, which affect comfort, surface temperature, etc., would be much less important to the overall steady-state heat transfer.

Figure 58 shows that we can get changes in fabric surface temperature just due to changes in environment. Most standard human thermal models which only use steady-state clothing properties, would not pick up changes such as these. These are surface temperature differences between the two cases of several °C, which can be detected by infrared sensors, so it is possible that the ability to model the surface temperatures of clothing in this way would be useful in the study of the thermal signature of clothed humans.

Besides the temperature variables shown in Figure 58, the integrated model gives explicit values for a great many other variables, such as sweating rate, humidities, vaporization rate, porosity changes, percent water absorbed as a function of location in the clothing system, etc. In Figures 59 and 60, we may observe a comparison of the hygroscopic versus nonhygroscopic case for the three quantities of skin relative humidity, liquid sweat accumulation at the skin surface, and volumetric vaporization rate at the skin surface. The volumetric vaporization rate is equal to the vaporization rate per unit skin area divided by the thickness of the first control volume at the skin surface.

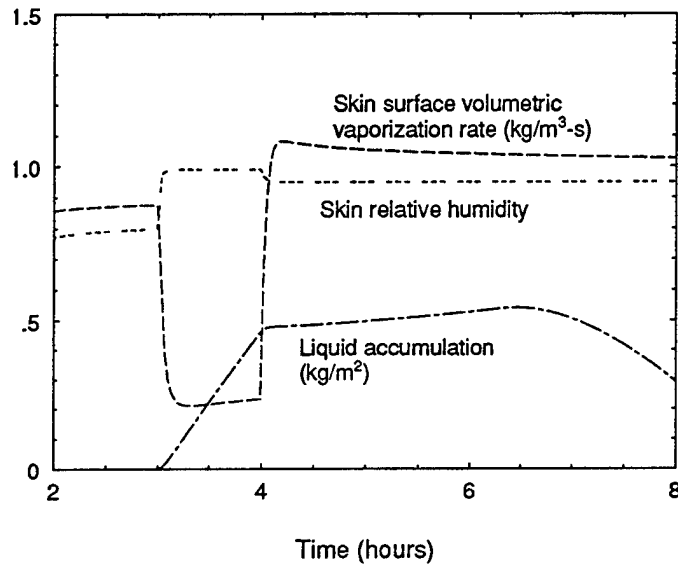


Figure 59. Skin surface relative humidity, liquid sweat accumulation at skin surface, and vaporization rate at skin surface, for hygroscopic fabric case.

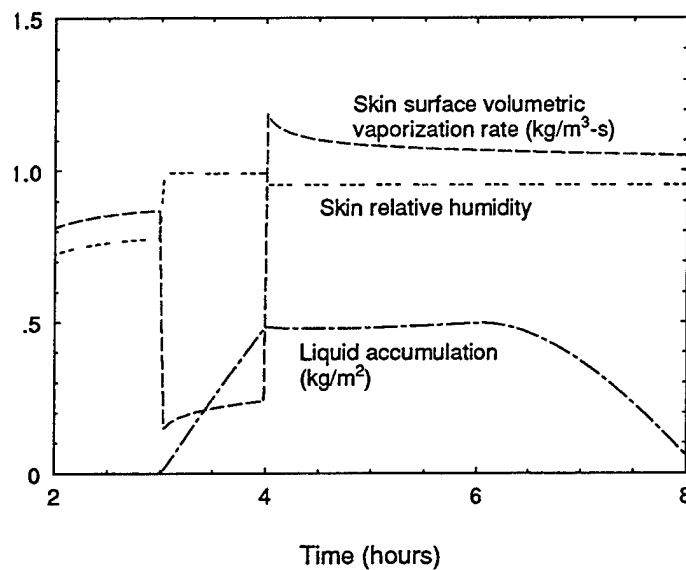


Figure 60. Skin surface relative humidity, liquid sweat accumulation at skin surface, and vaporization rate at skin surface, for nonhygroscopic fabric case.

7.5 Effect of Order of Clothing Layers

Keeping in mind the extreme simplicity of the assumptions incorporated into our single-zone human thermal physiology model, we can immediately start to use the integrated model to look at factors such as the arrangement and properties of fabric layers, and how they might affect how a fabric system performs. One particularly easy thing to try is to vary the arrangement of two fabric layers on the body. Such questions about system design often come up in cold weather clothing and chemical protective clothing systems, where we may know the total thermal or water vapor resistance of the clothing system, but we also have options about the order of arrangement of those layers.

We will place the two fabric layers on the body model, and only vary the hygroscopic nature of each fabric layer. We again examine the case where the relative humidity is changed from 10% to 90% relative humidity for one hour at hour 3, and then back to 10% relative humidity at hour 4. The four cases examined are listed below:

Case I -- both layers nonhygroscopic

Case II -- outer layer hygroscopic, inner layer nonhygroscopic

Case III -- inner layer hygroscopic, outer layer nonhygroscopic

Case IV -- both layers hygroscopic

Cases I and IV were shown previously in Figure 58, if we assume that the total thickness of those fabrics were equal to that of two layers. In Figure 61 we show the calculated fabric surface temperatures, for each of the four cases, and in Figure 62 we show the calculated skin surface temperatures.

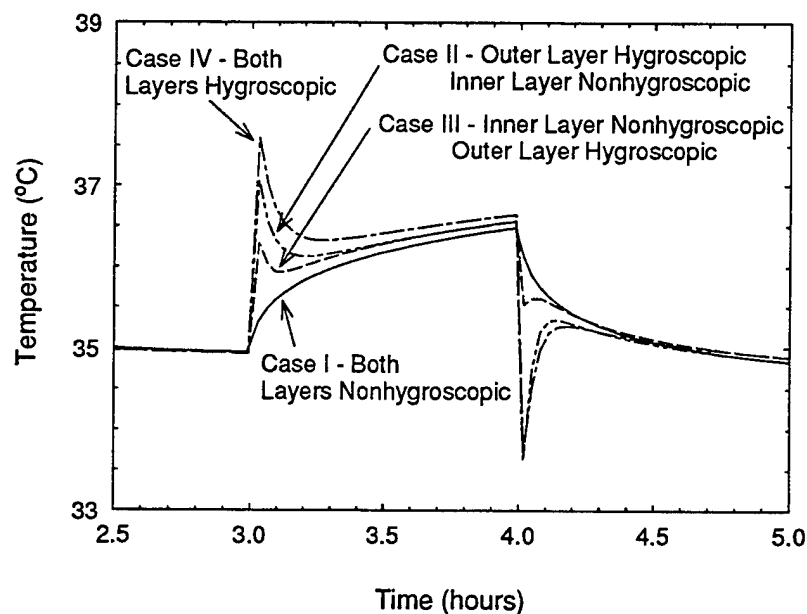


Figure 61. Calculated fabric surface temperatures, for four layering arrangements.

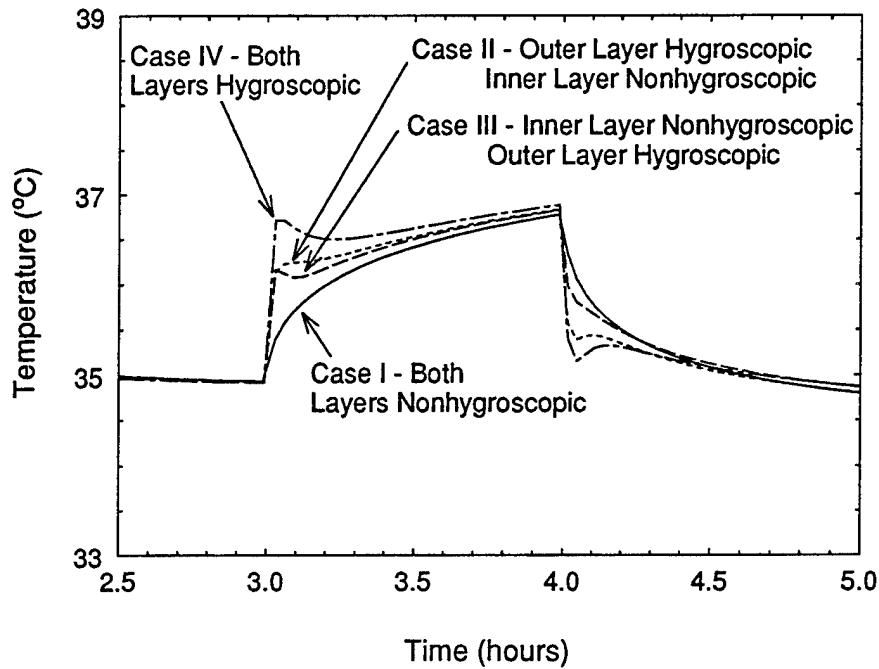


Figure 62. Calculated skin surface temperatures, for four layering arrangements.

We can see a steady progression of relative differences in the fabric surface temperature and skin temperature due to the differences in fabric regain and the arrangement of the two layers, but again we didn't see much difference in core temperature, which means these differences have the most impact on perceived comfort, rather than overall heat stress. We also emphasize again that we are not letting liquid sweat wick into the clothing layers, but only allow it to build up in a layer on the skin surface.

7.6 Effect of Changes in Human Work Rate

We can also use the combined model to look at how hygroscopic clothing properties interact with increases in sweating rate and exercise levels due to changes in human work rate, while the environmental factors remain constant.

The first situation we examine is a change in human work rate from 20 W/m^2 (light work) to 100 W/m^2 (fairly heavy work). The environmental conditions are 30°C , 65% r.h. We again examine the difference between clothing layers which differ only in their hygroscopic nature. No air layers are present, and again the fabric does not wick liquid sweat, so we are only looking at the diffusion of heat and water vapor. Human work rate is changed to the higher level between hours 3 and 4.

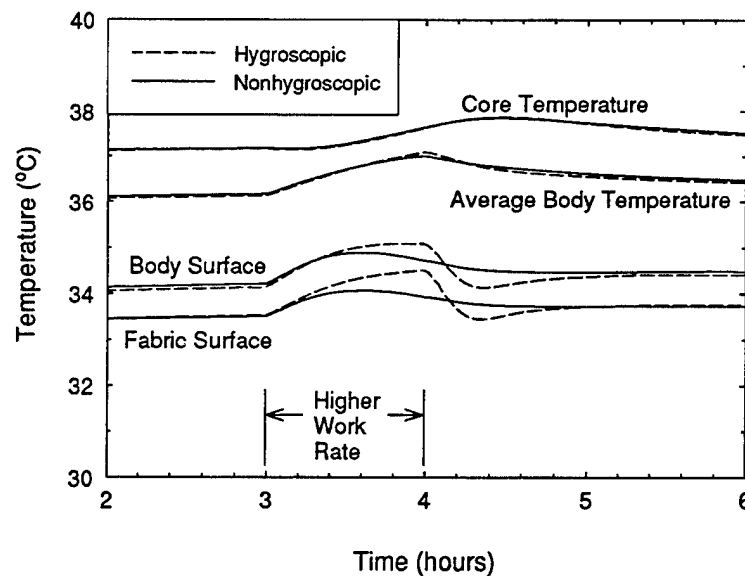


Figure 63. Differences in calculated temperatures between hygroscopic and nonhygroscopic fabrics, for a change in work rate from 20 to 100 W/m^2 .

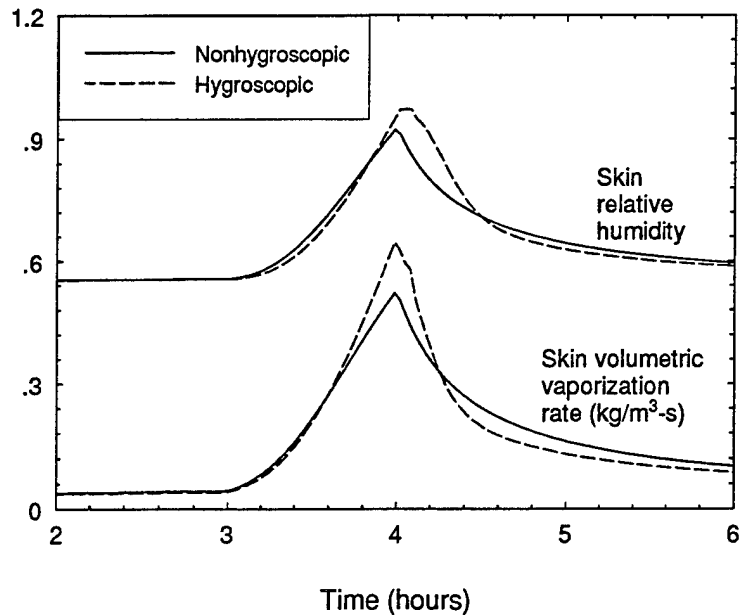


Figure 64. Differences in vaporization rate and skin relative humidity for hygroscopic and nonhygroscopic fabrics, for a change in work rate from 20 to 100 W/m².

Figure 63 and 64 show some of the same effects of fiber hygroscopicity that we saw for the change in environmental conditions, but in this case, they are influenced by the increases in vapor concentration caused by increased sweating and vaporization rates at the skin surface.

Up to this point, we have neglected sweat accumulation within the clothing layers, which is quite ludicrous for most normal fabric materials. The large amount of sweat generated by the body would certainly be absorbed by the clothing, which would produce much higher vaporization rates than those predicted by the pure diffusion model. Fortunately it is quite easy to incorporate simple wicking effects into the clothing model, which is covered in the next section.

7.7 A Simple Model of Wicking Effects in Clothing

To incorporate liquid sweat movement and accumulation in clothing layers, we must modify the one-dimensional governing equations to be in the form given in Chapter 2, Equations (2.8.1-24). The major simplifying assumptions are that we again neglect gas phase convective transport, and only consider one dimension. The major change is that we may now have the liquid phase present within the averaging control volume, and we must keep track of the liquid volume fraction, and the rate of liquid-to-vapor phase change in the control volume. The modified set of one-dimensional equations are:

Energy equation:

$$\rho C_p \frac{\partial T}{\partial t} + \left[\rho_\beta (c_p)_\beta \langle \bar{v}_\beta \rangle \right] \frac{\partial T}{\partial x} + \Delta h_{vap} \dot{m}_{lv} + (Q_l + \Delta h_{vap}) \dot{m}_{sv} = \frac{\partial}{\partial x} \left(k_{eff} \frac{\partial T}{\partial x} \right) \quad (7.17)$$

Solid phase continuity equation:

$$\rho_\beta \frac{\partial}{\partial t} (\epsilon_{bw}) + \dot{m}_{sv} = 0 \quad \text{or} \quad \frac{\partial}{\partial t} (\epsilon_{bw}) + \frac{\dot{m}_{sv}}{\rho_\beta} = 0 \quad (7.18)$$

Gas phase diffusion equation:

$$\frac{\partial}{\partial t} (\epsilon_\gamma \rho_v) - \dot{m}_{lv} - \dot{m}_{sv} = \frac{\partial}{\partial x} \left(\mathcal{D}_{eff} \frac{\partial \rho_v}{\partial x} \right) \quad (7.19)$$

Liquid phase continuity equation:

$$\frac{\partial \epsilon_\beta}{\partial t} + \frac{\partial \langle \bar{v}_\beta \rangle}{\partial x} + \frac{\dot{m}_{lv}}{\rho_\beta} = 0 \quad \text{or} \quad \frac{\partial \epsilon_\beta}{\partial t} = - \frac{\partial \langle \bar{v}_\beta \rangle}{\partial x} - \frac{\dot{m}_{lv}}{\rho_\beta} \quad (7.20)$$

Liquid phase equation of motion:

$$\langle \bar{v}_\beta \rangle = - \left(\frac{K_\beta}{\mu_\beta} \right) \nabla [(p_a + p_v) - P_c] \quad (7.21)$$

Volume fraction constraint:

$$\epsilon_\gamma + \epsilon_\beta + \epsilon_{bw} + \epsilon_{ds} = 1 \quad (7.22)$$

Thermodynamic relations

$$p_a = p_\gamma = p_v$$

$$p_a = \rho_a \frac{R}{M_a} T$$

$$p_v = \rho_v \frac{R}{M_w} T$$

(7.23-25)

Sorption Relation

$$\epsilon_{bw} = 0.578 R_f \left(\epsilon_{ds} \frac{\rho_{ds}}{\rho_w} \right) \left(\frac{p_v}{p_s} \right) \left[\frac{1}{\left(0.321 + \frac{p_v}{p_s} \right)} + \frac{1}{\left(1.262 - \frac{p_v}{p_s} \right)} \right] \quad (7.26)$$

The 11 unknown variables for the clothing layer: $T, \dot{m}_{sv}, \dot{m}_{lv}, \epsilon_\gamma, \epsilon_\beta, \epsilon_{bw}, \rho_v, \rho_a, p_a, p_v, \langle \bar{v}_\beta \rangle$.

As we have done before, the source term due to vapor sorption is modeled by assuming that diffusion into the fiber is quasi-steady state. The polymer at the fiber's surface is assumed to immediately come into equilibrium with the relative humidity of the gas phase within the control volume for that grid point. The mass flux into or out of the fiber is calculated by:

$$\dot{m}_{sv} = \frac{D_{solid} \rho_{ds}}{d_f^2} (R_{total} - R_{skin}) \quad (7.27)$$

R_{total} instantaneous fiber regain (kg of water / 100 kg of dry polymer) [fraction]
 R_{skin} equilibrium regain at relative humidity within the control volume [fraction]
 D_{solid} apparent solid phase diffusion coefficient [m^2]
 d_f apparent fiber diameter [m]

In a similar manner, we determine a source term for evaporation or condensation from the liquid phase to gas phase by assuming that if there is any liquid phase present, the partial vapor pressure is saturated. If the vapor pressure is above saturation, then condensation takes place until the partial vapor pressure is at saturation.

Transport Coefficients and Mixture Properties

The transport coefficients and mixture properties are of the same form as were used in Chapter 5, but must be modified to include the liquid phase volume fraction.

$$k_{eff} = k_\gamma \left\{ \frac{\left[1 + (\epsilon_\beta + \epsilon_{bw} + \epsilon_{ds}) \right] k_\sigma + \epsilon_\gamma k_\gamma}{\epsilon_\gamma k_\sigma + \left[1 + (\epsilon_\beta + \epsilon_{bw} + \epsilon_{ds}) \right] k_\gamma} \right\} \quad (7.28)$$

$$k_\sigma = \left(\frac{k_w \rho_\beta \epsilon_\beta + k_w \rho_\beta \epsilon_{bw} + k_{ds} \rho_{ds} \epsilon_{ds}}{\rho_\beta \epsilon_\beta + \rho_\beta \epsilon_{bw} + \rho_{ds} \epsilon_{ds}} \right) \quad (7.29)$$

$$k_\gamma = \left(\frac{k_v \rho_v + k_a \rho_a}{\rho_v + \rho_a} \right) \quad (7.30)$$

$$\mathcal{D}_{eff} = \frac{\mathcal{D}_a \varepsilon_\gamma}{\tau} \quad (7.31)$$

$$\rho = \varepsilon_\beta \rho_\beta + \varepsilon_{bw} \rho_\beta + \varepsilon_{ds} \rho_{ds} + \varepsilon_\gamma (\rho_v + \rho_a) \quad (7.32)$$

$$C_p = \frac{\varepsilon_\beta \rho_\beta (c_p)_\beta + \varepsilon_{bw} \rho_\beta (c_p)_\beta + \varepsilon_{ds} \rho_{ds} (c_p)_{ds} + \varepsilon_\gamma [\rho_v (c_p)_v + \rho_a (c_p)_a]}{\rho} \quad (7.33)$$

The set of equations (7.17-33) given above are of a form compatible with those often used by people modeling moisture migration or drying processes in porous material. There are several simplifying assumption we can make which will allow us to proceed to a very simple assumption about wicking effects in clothing, but we shall first discuss two very important concepts: capillary pressure, and irreducible saturation.

The capillary pressure P_c is often a function of the fraction of the void space occupied by the liquid. Liquid present in a porous material may be either in a pendular state, or in a continuous state. If the liquid is in a pendular state, it is in discrete drops or regions which are unconnected to other regions of liquid. If liquid is in the pendular state, there is no liquid flow, since the liquid does not form a continuous phase. There may be significant capillary pressure present, but until the volume fraction of liquid rises to a critical level to form a continuous phase, there will be no liquid flow. This implies that there is a critical saturation level, which we can think of as the relative proportion of liquid volume within the gas phase volume, which must be reached before liquid movement may begin.

Experimentally measured liquid capillary curves often show significant hysteresis depending on whether liquid is advancing (imbibition) or receding (drainage) through the porous material. A typical curve [108] is shown below in Figure 65.

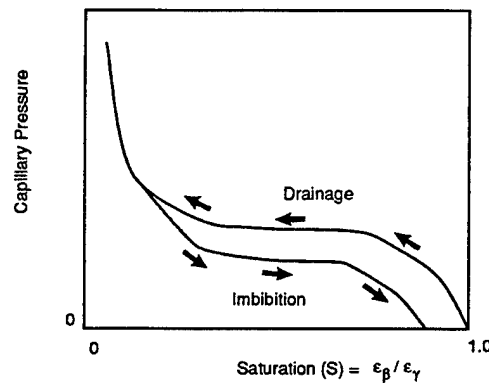


Figure 65. Typical appearance of capillary pressure curves as a function of liquid saturation for porous materials.

We may take a definition for liquid saturation as:

$$S = \frac{V_\beta}{V_\gamma + V_\beta} = \frac{\epsilon_\beta}{\epsilon_\beta + \epsilon_\gamma} \quad (7.44)$$

The point at which the liquid phase becomes discontinuous is often called the irreducible saturation (s_{ir}) [109]. When the irreducible saturation is reached, the flow is discontinuous, which implies that liquid flow ceases when:

$$\epsilon_\beta < s_{ir} [1 - (\epsilon_{ds} + \epsilon_{bw})] \quad (7.45)$$

An empirical equation given by Stanish, et al. [110], suggests a form for the equation for capillary pressure as a function of fraction of void space occupied by liquid:

$$P_c = a \left(\frac{\epsilon_\beta}{\epsilon_\gamma} \right)^{-b}, \text{ where } a \text{ and } b \text{ are empirical constants} \quad (7.46)$$

For liquid permeability as a function of saturation [110]:

$$K_\beta = \begin{cases} 0; & (\epsilon_\beta / \epsilon_\gamma) < s_{ir} \\ K_\beta^s \left\{ 1 - \cos \left[\frac{\pi (\epsilon_\beta / \epsilon_\gamma - s_{ir})}{2 (1 - s_{ir})} \right] \right\}; & (\epsilon_\beta / \epsilon_\gamma) \geq s_{ir} \end{cases} \quad (7.47)$$

where K_β^s is the liquid phase Darcy permeability when fully saturated.

Equations 7.46 and 7.47 can be incorporated in a straightforward manner into the one-dimensional model; however, the technical problem of measuring the empirical constants contained in these equation is considerable.

Another way to look at a moisture diffusivity equation is to consider the moisture distribution throughout the porous material as akin to a diffusion process. By combining the conservation of mass and Darcy's equation, a differential equation for the local saturation S may be written as [111]:

$$\frac{\partial S}{\partial t} = \frac{\partial}{\partial x} \left[F(s) \frac{\partial S}{\partial x} \right] \quad (7.48)$$

where the "moisture diffusivity" is given by:

$$F(s) = \frac{\left(\frac{K_\beta}{\mu_\beta} \right) \left(\frac{dP_c}{dS} \right)}{(\epsilon_\beta + \epsilon_\gamma)} \quad (7.49)$$

If we rewrite the saturation variable S in terms of its original definition:

$$S = \frac{V_\beta}{V_\gamma + V_\beta} = \frac{\epsilon_\beta}{\epsilon_\beta + \epsilon_\gamma} ;$$

the differential equation for liquid migration under the influence of capillary pressure may be written as:

$$\frac{\partial}{\partial t} \left(\frac{\epsilon_\beta}{(\epsilon_\beta + \epsilon_\gamma)} \right) = \frac{\partial}{\partial x} \left[\frac{\left(\frac{K_\beta}{\mu_\beta} \right) \left(\frac{dP_c}{dS} \right)}{(\epsilon_\beta + \epsilon_\gamma)} \frac{\partial}{\partial x} \left(\frac{\epsilon_\beta}{(\epsilon_\beta + \epsilon_\gamma)} \right) \right] \quad (7.50)$$

Although we have available the form of these relations for the capillary pressures, and permeability as a function of saturation and irreducible saturation, we have very little data available on the actual values of permeabilities for representative textile fabrics. Wicking studies on fabrics are usually carried out parallel to the plane of the fabric by cutting a strip, dipping one end in water, and studying liquid motion as it wicks up the strip [112,113]. However, wicking from the body through the fabric layer takes place perpendicular to the plane of the fabric, where the transport properties quite different due to the highly anisotropic properties of woven fabrics.

The usefulness of the relations contained in equations (7.44-50) are that they allow one to model the drying behavior of porous materials by accounting for both a constant drying rate period and a falling rate period. In the constant rate drying period, evaporation takes place at the surface of the porous material, and capillary forces bring the liquid to the surface. When irreducible saturation is reached in regions of the porous solid, drying becomes limited by the necessity for diffusion to take place through the porous structure of the material, which is responsible for the "falling rate" period of drying. These effects are most important for materials which are thick, or of low porosity. For materials of the porosity and thickness typical of woven fabrics, almost all drying processes are in the constant rate regime, which suggests that many of the complicating factors which are important for thicker materials can be safely ignored. Studies on the drying rates of fabrics [114-116] suggest that simply assuming drying times proportional to the original liquid water content are a good predictor of the drying behavior of both hygroscopic and nonhygroscopic fabrics. Wicking processes perpendicular to the plane of the fabric wicking take place very quickly, and the falling rate period is very short once most of the liquid has evaporated from the interior portions of fabrics. The one-dimensional modeling we have done of fabric drying also suggests that this is true: for thin fabrics, the drying rate is constant for most of the drying time, and the falling rate period is very short. We have also found that extremely small time steps must be taken computationally to capture the falling rate period.

For these reasons, we decided that it may not be productive at this point to include numerical values for the empirical factors and constants contained in equations 7.44-50 in the integrated clothing/human thermal physiology model. However, it is a very simple matter to assume a very high liquid permeability and very high capillary pressures, which cause any liquid sweat remaining at the skin surface to instantly be distributed within the free porosity of the fabric. This allows us to look at two different clothing materials which are identical in all their properties except that one material will wick sweat away from the skin surface, while the other does not allow wicking through its structure. When liquid is present, wicking effects quickly overwhelm any of the other diffusive effects, both due to the evaporation of liquid water within the clothing, and the increase in thermal conductivity of the porous matrix due to the liquid water which builds up within the clothing layers. An example is shown in Figure 66 for the case of a wicking versus a nonwicking fabric, when a human goes from a light work rate (20 Watt/m²) to a heavy work rate (200 Watt/m²) for 1 hour, and then back to a light work rate. Environmental conditions in both cases are air temperature of 30°C and relative humidity of 65%.

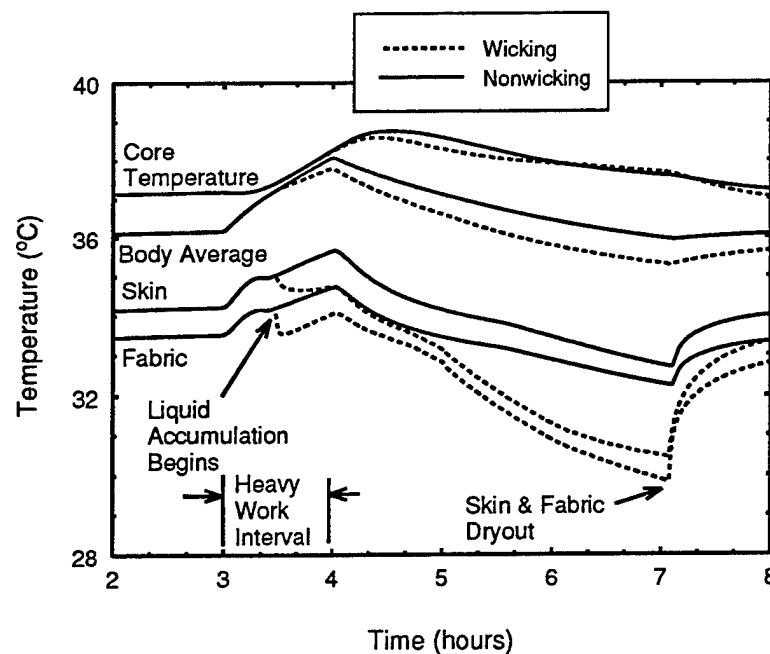


Figure 66. Comparison of a wicking versus a nonwicking fabric (other properties identical) during changes in human work rate.

7.8 Conclusions

The integration of a model of the human thermal control system with a material model for the coupled diffusion of mass and energy through hygroscopic porous materials has made it possible to examine phenomena such as transient changes in skin and clothing surface temperature as influenced by the hygroscopicity of fabric layers. Factors important to the design of clothing systems, such as the placement and order of fabric layers, and the relative importance of wicking versus nonwicking fabrics, may be assessed with this simple model.

The clothing model and the physiological model are relatively independent. This should make it possible to incorporate the clothing model into more realistic multi-zone models of the human thermophysiological control system.

CHAPTER 8

CONCLUSIONS / RECOMMENDATIONS

The combination of test methods and modeling described in this report provides a consistent overall picture of the transport of energy and mass through protective clothing systems which are subjected to a wide range of human sweat generation rates and ambient environmental conditions. The integrated numerical model described here is a useful analysis tool for studies of the various transport phenomena affecting the thermal balance of clothed humans under different environmental conditions. It addresses the current need for more realistic models of coupled heat and mass transfer through hygroscopic porous textile materials, which include phenomena such as liquid water wicking, condensation/evaporation within textile layers, concentration-dependent permeation behavior of semipermeable membrane laminates, and the sorption behavior of hygroscopic textile fibers.

Conclusions

1. Whitaker's theory of coupled heat and mass transfer through porous media, when modified to include hygroscopic porous materials which can absorb liquid into the solid matrix, provides a very useful framework to describe transport phenomena which take place in clothing materials. The system of equations described in this report makes it possible to evaluate the time-dependent transport properties of hygroscopic and non-hygroscopic clothing materials by including many important factors which are usually ignored in the analysis of heat and mass transfer through textile materials. The set of equations allows for the unsteady capillary wicking of sweat through fabric structure, condensation and evaporation of sweat within various layers of the clothing system, forced gas phase convection through the porous structure of a textile layer, and the swelling and shrinkage of fibers and yarns as they absorb/desorb liquid water and water vapor.
2. The dynamic moisture permeation cell (DMPC) permits rapid testing of small quantities of permeable fabrics and semipermeable laminates under a wide variety of test conditions. This allows determination of a material's transport behavior under test conditions which are difficult or impossible to reach using existing standard laboratory test methods.
3. Water vapor permeation results obtained with the DMPC are in excellent agreement with those from the ISO 11092 sweating guarded hot plate test method, and from a modified inverted ASTM E 96 cup test.

4. The DMPC is particularly useful for characterizing transient diffusion of hygroscopic fabrics subjected to step changes in relative humidity. It provides excellent control over temperatures, pressures, and vapor concentrations, which is necessary for accurately determining the time-dependent and nonlinear transport properties of clothing materials. The DMPC also provides a controlled means of testing porous fabrics under conditions of combined diffusion and convective transport, in a way that provides experimental data to verify numerical models of the transport processes occurring in hygroscopic porous textiles.

5. The automated air permeability apparatus provides experimental data on convective gas flow transport properties, e.g. Darcy permeability or Darcy flow resistance. It provides a means of exploring and characterizing the way in which fabric structural changes, caused by fiber swelling at high humidities, can drastically change the air permeability of fabrics incorporated into clothing systems.

6. The one-dimensional model of coupled diffusion of mass and energy in hygroscopic porous materials performs well in predicting and simulating temperature changes as a function of time, and weight changes as a function of time -- two quantities which are easily measured experimentally. However, the one-dimensional model neglects factors such as the convective transfer of energy and mass through the porous structure of textiles, which is often of more practical significance than the diffusion properties of clothing systems.

7. The two-dimensional model of gas phase convection/diffusion through porous hygroscopic materials successfully simulates steady-state and time-dependent phenomena which are seen experimentally. The ability to apply the proper form of the two-dimensional transport equations to a particular test device eliminates much of the need to guess or estimate boundary heat and mass transfer coefficients, since they are taken care of automatically in the course of the computational fluid dynamic and heat transfer calculations involved in solving for the gas flow field. This allows one to get a clearer picture of the behavior of the porous material itself, without the complications of the behavior of the particular experimental apparatus intruding themselves into the results.

8. The integration of a model of the human thermal control system with a material model for the coupled diffusion of mass and energy through hygroscopic porous materials made it possible to examine phenomena such as transient changes in skin and clothing surface temperature as influenced by the hygroscopicity of fabric layers. Factors important to the design of clothing systems, such as the placement and order of fabric layers, and the relative importance of wicking versus nonwicking fabrics, may be assessed with this simple model.

Recommendations

1. The ability of the model to account for the swelling/shrinkage of textile fibers due to changes in relative humidity, with a corresponding change in fabric porosity and air permeability, is being explored further, by generating experimental results on a wide variety of hygroscopic and nonhygroscopic fabrics. Inertial effects at higher flow rates will be incorporated, and the experimental results will be analyzed based on assumptions about structural factors such as porosity, effective pore size, and the relative importance of intrayarn versus interyarn porosity.

2. The wicking and liquid transport portion of the model should be extended and exercised with experimental data in as extensive a manner as was done for diffusion/convection processes in the gas phase. The anisotropic nature of the liquid transport process can be taken into account with the two-dimensional model, but the generation of experimental data for wicking through the plane of the fabric will require significant effort.

3. The two-dimensional material model should be applied directly to chemical agent testing conducted on chemical protective uniform materials, particularly laboratory test methods. The diffusion/convection of water vapor/air/chemical agent/simulant mixtures may be modeled and used to explain/simulate issues which arise in test methods which evaluate samples under conditions of constant pressure drop or constant flow rate conditions.

4. The two-dimensional model could be applied to a cylindrical geometry in crossflow conditions in a wind tunnel. This would allow modeling and simulation of realistic two-dimensional flow fields, and the interaction of variable fabric permeability with the air flow through and over a simulated clothing system covering the human body.

5. The direct incorporation of the enthalpy of phase changes in the basic thermal energy equations of the model should make it possible to analyze the important parameters of new clothing items incorporating phase change materials such as encapsulated paraffins, which is a topic of ongoing research programs at Natick and in industry.

6. The application of the integrated clothing/human thermal physiology model to the area of camouflage/detection, e.g., modeling the varying thermal signature of clothed humans should be explored. The model may be especially applicable when there are rapid changes in either the environment (exiting vehicles/shelters), or in human work/metabolic rate.

7. The integrated clothing/human thermal physiology model should incorporate a more realistic and useful version of the human thermoregulatory system; at minimum one which models the body as a set of different thermal zones.

To summarize, the present design and evaluation process for a new protective clothing system is very expensive and time-consuming. Textile materials, which are the basic building blocks of any garment, are usually selected based on a few steady-state properties determined from laboratory tests. A long process of design iteration using heated manikins, and human wear trials in environmentally-controlled chambers, is then begun in order to evaluate the dynamic interactions between the human, clothing, and the environment. Computational power has progressed to the point where it is now possible to approach the modeling of the interaction of clothing/human performance in a truly integrated fashion. Computational fluid dynamic/heat transfer finite-element techniques, which capture the irregular geometries involved, can model the time-dependent flows through and over the clothing system. Biomechanical models of body motion, integrated with human thermoregulatory models, can now be coupled to sophisticated models of heat and mass transfer through clothing, resulting in a three-dimensional model of the human/clothing system, which incorporates body motion, external environmental conditions, and human thermal physiology.

This document reports research undertaken at the U.S. Army Soldier Systems Command, Natick Research, Development and Engineering Center and has been assigned No. NATICK/TR-97/005 in the series of reports approved for publication.

CHAPTER 9

REFERENCES

1. Henry, P., Diffusion in absorbing media, *Proceedings of the Royal Society of London*, Vol. 171 A, pp. 215-241, 1939.
2. Crank, J., *The Mathematics of Diffusion*, 2nd ed., Oxford University Press, London, 1975.
3. Nordon, P., David, H.G., Coupled diffusion of moisture and heat in hygroscopic textile materials, *International Journal of Heat and Mass Transfer*, Vol. 10, pp. 853-866, 1967.
4. Farnworth, B., A numerical model of the combined diffusion of heat and water vapour through clothing, *Textile Research Journal*, Vol. 56, 653-665, 1986.
5. Wehner, J.A., Miller, B., Rebenfeld, L., Dynamics of water vapor transmission through fabric barriers, *Textile Research Journal*, Vol. 58, pp. 581-592, 1988.
6. Li, Y., Holcombe, B.V., A two-stage sorption model of the coupled diffusion of moisture and heat in wool fabrics, *Textile Research Journal* Vol. 62, pp. 211-217, 1992.
7. Berger, D., Pei, C., Drying of hygroscopic capillary porous solids - A theoretical approach, *International Journal of Heat and Mass Transfer*, Vol. 16, pp. 293-301, 1973.
8. Eckert, E.R.G., Faghri, M., A general analysis of moisture migration caused by temperature differences in an unsaturated porous medium, *International Journal of Heat and Mass Transfer*, Vol. 23, pp. 1613-1623, 1980.
9. Stanish, M., Schajer, G., Kayihan, F., A mathematical model of drying for hygroscopic porous media, *AIChE Journal*, Vol. 32, pp. 1301-1311, 1986.
10. Luikov, A.V., Heat and mass transfer in capillary-porous bodies, *Advances in Heat Transfer*, Vol 1, Academic Press, New York, pp. 123-184, 1964.
11. Whitaker, S., Simultaneous heat, mass, and momentum transfer in porous media: A theory of drying, *Advances in Heat Transfer*, Vol. 13, Academic Press, New York, pp. 119-200, 1977.

12. Simonson, C.J., Tao, Y.X., Besant, R.W., Thermal hysteresis in fibrous insulation, *International Journal of Heat and Mass Transfer*, Vol. 36, pp. 4433-4441, 1993.
13. Ma, Y.H., Rust, L.W., Larson, R.E., Spano, L.A., Mathematical simulation of simultaneous energy and mass transfer process in a clothing-airspace-skin system, *Proceedings, Summer Computer Simulation Conference, N.Y. Association for Computing Machinery*, Vol. 2, pp. 859-867, 1970.
14. Li, Y., Holcombe, B.V., Schneider, A.M., Apcar, F., Mathematical modelling of the coolness to touch of hygroscopic fabrics, *Journal of the Textile Institute*, Vol. 84, pp. 267-274, 1993.
15. Jones, B. W., Transient interaction between the human and the thermal environment, *ASHRAE Transactions*, Vol. 98, pp. 189-195, 1992.
16. Whitaker, S., Quintard, M., One- and two-equation models for transient diffusion processes in two-phase systems, *Advances in Heat Transfer*, Vol. 23, Academic Press, New York, pp. 369-464, 1993.
17. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing, Washington, 1980.
18. Vafai, K., Whitaker, S., Simultaneous heat and mass transfer accompanied by phase change in porous insulation, *Journal of Heat Transfer*, Vol. 108, pp. 667-675, 1986.
19. Tao, Y., Besant, R., Rezkallah, K., The transient thermal response of a glass-fiber insulation slab with hygroscopicity effects, *International Journal of Heat and Mass Transfer*, Vol. 35, pp. 1155-1167, 1992.
20. Patankar, S.V., Baliga, B.R., A new finite-difference scheme for parabolic differential equations, *Numerical Heat Transfer*, Vol. 1, pp. 27-37, 1978.
21. Stolwijk, J., Hardy, J., Control of body temperature, in *Handbook of Physiology, Section 9*, edited by S. Geiger, Waverly Press, Baltimore, pp. 45-68, 1977.
22. Wissler, E., Mathematical simulation of human thermal behavior using whole body models. In *Heat and Mass Transfer in Medicine and Biology* (Edited by Shitzer, A., and Eberhart, R.C.), Plenum Press, New York, pp. 325-373, 1985.

23. Gagge, A., Fobelets, A., Berglund, L., A standard predictive index of human response to the thermal environment, *ASHRAE Transactions*, Vol. 82, No. 2.
24. Gagge, A., A two-node model of human temperature regulation in Fortran. In: *Bioastronautics Data Book* (2nd ed.) edited by J.F. Parker, Jr., and V.R. West, NASA, Washington, D.C., pp. 142-148, 1973.
25. Whitaker, S., A Theory of Drying in Porous Media, in *Advances in Heat Transfer* 13, Academic Press, New York, 1977, pp. 119-203.
26. Jomaa, W., Puiggali, J., Drying of Shrinking Materials: Modellings with Shrinkage Velocity, *Drying Technology* 9, no. 5, 1991, pp. 1271-1293.
27. Crapiste, G., Rostein, E., Whitaker, S., Drying Cellular Material. I: Mass Transfer Theory, *Chem. Eng. Sci.* 43, 2919-2928. II: Experimental and Numerical Results, *Chem. Eng. Sci.* 43, 2929-2936, 1988.
28. Slattery, J., *Momentum, Energy, and Mass Transfer in Continua*, McGraw-Hill, New York, 1972.
29. Ibid., p. 19.
30. Gray, W., A Derivation of the Equations for Multi-Phase Transport, *Chemical Engineering Science* 30, 1975, pp. 229-233.
31. Morton, W., Hearle, J., *Physical Properties of Textile Fibres*, John Wiley & Sons, New York, 1975, pp. 178.
32. Lotens, W., *Heat Transfer From Humans Wearing Clothing*, Doctoral Thesis, published by TNO Institute for Perception, Soesterberg, The Netherlands, 1993, pp. 34-37.
33. Nordon, P., David, H.G., Coupled Diffusion of Moisture and Heat in Hygroscopic Textile Materials, *Int. J. Heat Mass Transfer* 10, pp. 853-866, 1967.
34. Stanish, M., Schajer, G., Kayihan, F., A Mathematical Model of Drying for Hygroscopic Porous Media, *AIChE Journal* 32, no. 8, pp. 1301-1311, 1986.
35. Duillien, F., *Porous Media: Fluid Transport and Pore Structure*, Academic Press, London, 1979, Chapters 4 and 6.

36. Henry, P., Diffusion in Absorbing Media, Proceeding of the Royal Society of London **171A**, pp. 215-241, 1939.
37. Li, Y., Holcombe, B., A Two-Stage Sorption Model of the Coupled Diffusion of Moisture and Heat in Wool Fabrics, *Textile Research Journal* **62**, no. 4, pp. 211-217, 1992.
38. American Society for Testing and Materials (ASTM) E96-80, Standard Test Methods for Water Vapor Transmission of Materials (1984).
39. International Organization for Standardization, International Standard ISO 11092, 1993. "Textiles -- Physiological Effects, Part 1: Measurement of Thermal and Water Vapour Resistance under Steady-State Conditions (Sweating Guarded-Hotplate Test)," prepared by ISO Technical Committee ISO TC 38.
40. Pye, D., Hoehn, H., Panar, M., Measurement of gas permeability of polymers. II. Apparatus for determination of permeabilities of mixed gases and vapors, *Journal of Applied Polymer Science* **20**, pp. 287-301, 1976.
41. Tadlaoui, N., Clement, B., Namiesnik, J., Torres, L., A device for determining the permeability of polymer films used for food products packaging, *Polymer Testing* **12**, pp. 195-206, 1993.
42. O'Brien, K., Koros, W., Barbari, T., A new technique for the measurement of multicomponent gas transport through polymeric films, *Journal of Membrane Science* **29**, pp. 229-238, 1986.
43. Husband, R., Petter, P., An infrared instrument for the rapid measurement of water vapor permeation through barrier webs, *TAPPI Journal* (Technical Association of the Pulp and Paper Industry) **49**, pp. 565-572, 1966.
44. Ahlen, A., Diffusion of sorbed water vapor through paper and cellulose film, *TAPPI Journal* (Technical Association of the Pulp and Paper Industry) **53**, pp. 1320-1326, 1970.
45. Demorest, R., Recent developments in testing the permeability of good barriers, *Journal of Plastic Film and Sheeting* **8**, pp. 109-115, 1992.

46. TAPPI (Technical Association of the Pulp and Paper Industry) Test Method T523 om-93 "Dynamic measurement of water vapor transfer through sheet materials", *TAPPI Test Methods*, TAPPI Press, Atlanta, Georgia, 1994.
47. ASTM (American Society for Testing and Materials) Test Method F 372-73 "Standard test method for water vapor transmission rate of flexible barrier materials using an infrared detection technique", *Annual Book of ASTM Standards*, Vol 04.06.
48. Wehner, J., Miller, B., Rebenfeld, L., Dynamics of water vapor transmission through fabric barriers, *Textile Research Journal* **58**, pp. 581-592, 1988.
49. Chen, R.Y., Flow in the entrance region at low Reynolds number, *Journal of Fluids Engineering*, **95**, March, 1973, pp. 153-158.
50. Geankopolis, G., *Mass Transport Phenomena*, Holt, Rinehart, and Winston, Inc., New York, 1972, pp. 277-278.
51. Military Specification, MIL-C-43468, Cloth, Camouflage Pattern, Wind Resistant Poplin, Cotton.
52. Military Specification, MIL-C-44031D, Cloth, Camouflage pattern: Woodland, Cotton and Nylon, Class 2 - Camouflage Printed and Quarpel Treated.
53. Military Specification, MIL-C-43468, Cloth, Camouflage Pattern, Wind Resistant Poplin, Cotton, Type VI, Quarpel Treated.
54. Military Specification, MIL-C-29462, Cloth, Laminated, Polyester Tricot Knit, Activated Carbon Sphere, Chemical Protective.
56. Military Specification, MIL-S-29461, Suit, Chemical and Biological Protective (Carbon Sphere), Class 2 - Desert Camouflage Printed.
57. Military Specification, MIL-C-43858B(GL), Cloth, Laminated, Nylon Tricot Knit, Polyurethane Foam Laminate, Chemical Protective and Flame Resistant, Type III.
58. Military Specification, MIL-S-43926H, Suit, Chemical Protective, Class 1-Woodland Camouflage.
59. Military Specification, MIL-C-44187B, Cloth, Laminate, Waterproof and Moisture Vapor Permeable.

60. Farnworth, B., Lotens, W., and Wittgen, P., "Variation of Water Vapor Resistance of Microporous and Hydrophilic Films with Relative Humidity," *Textile Research Journal* **60**, pp. 50-53, 1990.
61. Oszcewski, R., "Water Vapor Transfer Through a Hydrophilic Film at Subzero Temperatures," *Textile Research Journal* **66**, pp. 24-29, 1996.
62. Gibson, P., Auerbach, M., Giblo, J., Teal, W., Endrusick, T., Interlaboratory Evaluation of a New Sweating Guarded Hot Plate Test Method (ISO 11092), *Journal of Thermal Insulation and Building Envelopes* **18**, pp. 182-200, October, 1994.
63. Military Specification, Mil-C-823, Cloth, Serge, Wool, Wool and Nylon, Polyester and Wool
64. Military Specification, Mil-C-21115, Cloth, Tropical: Wool, Polyester/Wool
65. Wehner, J. A., Miller, B., Rebenfeld, L., "Moisture induced changes in fabric structure as evidenced by air permeability measurements," *Textile Research Journal*, **57**, pp. 247-256, 1987.
66. Armour, J., Cannon, J., Fluid Flow Through Woven Screens, *AIChE Journal* **14**, no. 3., pp. 415-420, 1968.
67. Kyan, C., Wasan, D., Kintner, R., Flow of Single-Phase Fluids through Fibrous Beds, *Industrial Engineering and Chemistry Fundamentals* **9**, no. 4, pp. 596-603, 1970.
68. Kulichenko, A., Langenhove, L., The Resistance to Flow Transmission of Porous Materials, *Journal of the Textile Institute* **83**, no. 1, pp. 127-132, 1992.
69. Brekel, L., Jong, E., Hydrodynamics in Packed Textile Beds, *Textile Research Journal* **59**, pp. 443-440, 1989.
70. Chen, C., Filtration of Aerosols by Fibrous Media, *Chemical Reviews* **55**, p. 595-623, 1955.
71. Dullien, F., *Porous Media - Fluid Transport and Pore Structure*, Academic Press, New York, p. 157, 1979.

72. Kind, J., Jenkins, J., Seddigh, F., Experimental investigation of heat transfer through wind-permeable clothing, *Cold Regions Science and Technology* **20**, pp. 39-49, 1991.
73. Take-uchi, M., Analysis of wind effect on the thermal resistance of clothing with the aids of Darcy's law and heat transfer equation, *Sen-i Gakkaishi* **39**, no. 3, pp. 39-48, 1989.
74. Stuart, I., Denby, E., Wind induced transfer of water vapor and heat through clothing, *Textile Research Journal* **57**, pp. 247-256, 1987.
75. Fedele, P., Bergman, W., McCallen, R., Sutton, S., Hydrodynamically induced aerosol transport through clothing, *Proceedings of the 1986 Army Science Conference*, Vol. I, pp. 279-293, 1986.
76. Dullien, F., *Porous Media - Fluid Transport and Pore Structure*, Academic Press, New York, p. 81, 1979.
77. Morton, W., Hearle, J., *Physical Properties of Textile Fibers*, John Wiley and Sons, New York, pp. 227, 1975.
78. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere, 1980.
79. *Documentation of COMPACT-3D Version 3.1*, Innovative Research, Inc., Minneapolis, Minnesota, 1993.
80. Charmchi, M., *Analysis of Gas-Cooled Particle-Bed Reactor, The Theory and Numerical Techniques Applied in the SIMBED Codes*, Technical Report prepared for Brookhaven National Laboratory, July, 1993.
81. Robertson, S., An Axisymmetric, Turbulent Flow Analysis of Contaminant Infiltration into a Pressurized Structure with a Fabric Endcap, U.S. Army Natick RDE Center Technical Report *NATICK/TR-93/008*, November, 1992.
82. Patankar, S.V., Baliga, B.R., A new finite-difference scheme for parabolic differential equations, *Numerical Heat Transfer*, vol. 1, pp. 27-37, 1978.
83. Schneider, P.J., *Conduction Heat Transfer*, Addison-Wesley, Cambridge, Massachusetts, 1955, p. 235.

84. Henry, P., Diffusion in absorbing media, *Proceedings of the Royal Society of London*, Vol. 171 A, pp. 215-241, 1939.
85. Nordon, P., David, H.G., Coupled diffusion of moisture and heat in hygroscopic textile materials, *International Journal of Heat and Mass Transfer*, Vol. 10, pp. 853-866, 1967.
86. Farnworth, B., A numerical model of the combined diffusion of heat and water vapour through clothing, *Textile Research Journal*, **56**, pp. 653-665, 1986.
87. Wehner, J., Miller, B., Rebenfeld, L., Dynamics of water vapor transmission through fabric barriers, *Textile Research Journal* **58**, pp. 581-592, 1988.
88. Li, Y., Holcombe, B.V., A two-stage sorption model of the coupled diffusion of moisture and heat in wool fabrics, *Textile Research Journal* **62**, pp. 211-217, 1992.
89. Jones, B., Ito, M., McCullough, E., "Transient thermal response of clothing system," *Proceedings of the International Conference on Environmental Ergonomics*, Austin, Texas, October, pp. 66-67, 1990.
90. Le, C., Ly, N., "Heat and Moisture Transfer in Textile Assemblies, Part I: Steaming of Wool, Cotton, Nylon, and Polyester Fabric Beds," *Textile Research Journal* **65**, No. 4, pp. 203-212, 1995.
91. Tao, Y., Besant, R., Rezkallah, K., Unsteady heat and mass transfer with phase changes in an insulation slab: frosting effects, *International Journal of Heat and Mass Transfer* **34**, 1991, pp. 1593-1603.
92. Tao, Y., Besant, R., Rezkallah, K., The transient thermal response of a glass-fiber insulation slab with hygroscopicity effects, *International Journal of Heat and Mass Transfer* **35**, 1992, pp. 1155-1167.
93. Tao, Y., Besant, R., Rezkallah, K., Thermal hysteresis in porous insulation, *International Journal of Heat and Mass Transfer* **36**, 1993, pp. 4433-4441.
94. Lotens, W., Havenith, G., "Effects of moisture absorption in clothing on the human heat balance," *Ergonomics* **38**, No. 6, pp. 1092-1113, 1994.
95. Kerner, E. The electrical conductivity of composite media, *Proceedings of the Physical Society*, B69, 1956, p. 802.

96. Progelhof, R., Throne, J., Ruetsch, R., Methods for predicting the thermal conductivity of composite systems: a review, *Polymer Engineering and Science* **16**, no. 9, pp. 615-625, 1976.
97. Holman, J., *Heat Transfer*, McGraw-Hill, New York, p. 609, 1990.
98. Kays, W., Crawford, M., *Convective Heat and Mass Transfer*, McGraw-Hill, New York, 1993, pp. 125-140.
99. Kreith, F., Bohn, M., *Principles of Heat Transfer*, Harper & Row Publishers, New York, pp. 496-497, 1986.
100. Vafai, K., Tien, C., Boundary and inertia effects on convective mass transfer in porous media, *International Journal of Heat and Mass Transfer* **25**, pp. 1183-1190, 1980.
101. Vafai, K., Tien, C., Boundary and inertia effects on flow and heat transfer in porous media, *International Journal of Heat and Mass Transfer* **24**, pp. 195-203, 1981.
102. Vafai, K., Convective flow and heat transfer in variable-porosity media, *Journal of Fluid Mechanics* **147**, pp. 233-259, 1984.
103. Stolwijk, J., Hardy, J., Control of body temperature, in *Handbook of Physiology, Section 9*, edited by S. Geiger, Waverly Press, Baltimore, pp. 45-68, 1977.
104. Wissler, E., Mathematical simulation of human thermal behavior using whole body models. In *Heat and Mass Transfer in Medicine and Biology* (Edited by Shitzer, A., and Eberhart, R.C.), Plenum Press, New York, pp. 325-373, 1985.
105. Jones, B. W., Transient interaction between the human and the thermal environment, *ASHRAE Transactions*, Vol. 98, pp. 189-195, 1992.
106. Crosbie, R., Hardy, J., Fessenden, E., Electrical Analog Simulation of Temperature Regulation in Man. In *Temperature: Its Measurement and Control in Science and Industry, Volume 3, Part 3, Biology and Medicine*, Hardy, J. ed., Herzfeld, C., editor-in-chief, Reinhold Publishing Corporation, New York, pp. 627-635, 1963.

107. Shitzer, A., Chato, C., Thermal Interaction with Garments. In *Heat and Mass Transfer in Medicine and Biology : Analysis and Applications, Volume I*, (Edited by Shitzer, A., and Eberhart, R.C.), Plenum Press, New York, pp. 375-381, 1985.
108. Whitaker, S., A Theory of Drying in Porous Media, *Advances in Heat Transfer* **13**, Academic Press, New York, 1977, pp. 185.
109. Kaviani, M., *Principles of Heat Transfer in Porous Media*, Springer-Verlag, New York, 1991, pp. 428-431.
110. Stanish, M., Schajer, G., Kayihan, F., A Mathematical Model of Drying for Hygroscopic Porous Media, *AIChE Journal* **32**, no. 8, August, 1986, pp. 1301-1311.
111. Chatterjee, P., *Absorbency*, Elsevier Science Publishing Co., Inc., New York, pp. 46-47.
112. Ghali, K., Jones, B., Tracy, J., Modeling heat and mass transfer in fabrics, *International Journal of Heat and Mass Transfer* **38**, no. 1, pp. 13-21, 1995.
113. Gahli, K., Jones, B., Tracy, J., Modeling Moisture Transfer in Fabrics, *Experimental Thermal and Fluid Science* **9**, pp. 330-336, 1994.
114. Crow, R., Moisture, Liquid and Textiles -- A Critical Review, Defense Research Establishment Ottawa, DREO Report No. 970, June 1987.
115. Crow, R., Oszcewski, R., The Effect of Fibre and Fabric Properties on Fabric Drying Times, Defense Research Establishment Ottawa, DREO Report No. 1182, August, 1993.
116. Crow, R., Dewar, M., The Vertical and Horizontal Wicking of Water in Fabrics, Defense Research Establishment Ottawa, DREO Report No. 1180, July, 1993.
117. American Association of Textile Technology AATT Monograph No. 109, *Textile Fibers and Their Properties*.
118. Morton, S., Hearle, J., *Physical Properties of Textile Fibres*, John Wiley and Sons, New York, 1975, p. 586.
119. Morton, S., Hearle, J., *Physical Properties of Textile Fibres*, John Wiley and Sons, New York, 1975, p. 591.

APPENDICES

Appendix A

Fabric Physical Properties

For the purposes of this study, it was desirable to have the physical properties of the textile samples determined in the dry state, with no absorbed water present, rather than at the standard textile test conditions of 65% relative humidity. Fabric samples were dried in a forced-convection oven for two days, and then the mass and thickness were measured. Thickness measurements in textiles are often problematic, since we don't want to deform the material we're measuring, since in the test methods the material will not be under any pressure. Thicknesses were therefore measured under the lightest possible pressure to give consistent measurements.

From measurements of mass, thickness, and area, one may calculate a bulk density ρ_b (kg/m^3) and an areal density (kg/m^2). From property tables, one may find the solid density ρ_{ds} of the polymers making up the textile. The material dry solid volume fraction, ϵ_{ds} , is defined as: $\epsilon_{ds} = \rho_b / \rho_{ds}$. The dry gas phase volume fraction is usually defined as $1 - \epsilon_{ds}$, but this can be misleading as shown in Figure A.1 below. The gas phase volume fraction is actually made up of the interyarn porosity, through which most convective and diffusional transport takes place, and the intrayarn porosity, which participates much less in transport across the fabric layer.

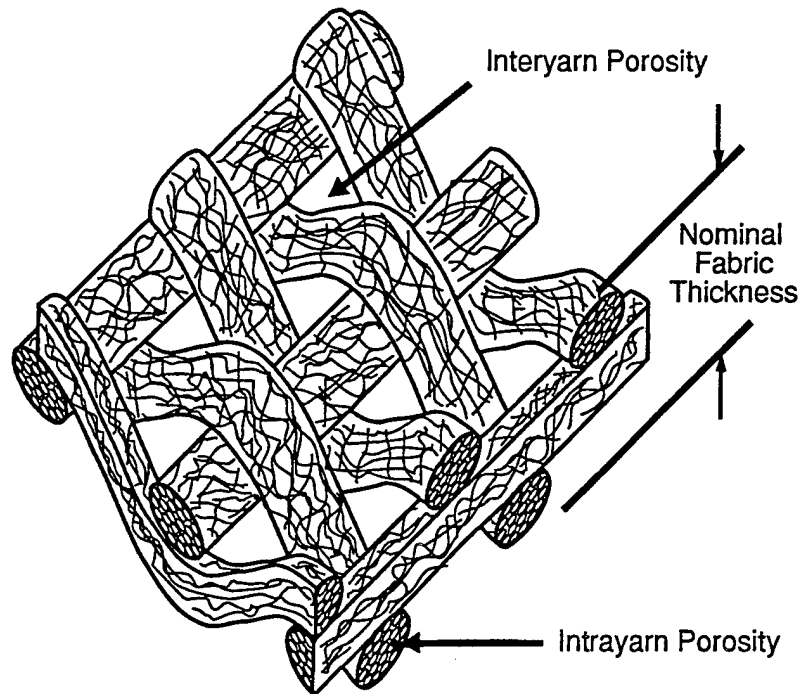


Figure A-1. Schematic of fabric illustrating intrayarn and interyarn volume fraction.

For materials which are a mixture of two or more fibers, an effective solid density is assigned based on the mass fractions of the respective solid materials making up the material.

Fabric regain at 65% relative humidity is also available from property tables [117]. For materials which are a mixture of two or more fibers, an effective regain is calculated based again on mass fraction weighting of the respective solid materials making up the material.

Table A-1. Fabric Physical Properties

	Thickness (m)	Areal Density (kg/m ²)	Bulk Density (kg/m ³)	Dry Solid Density (kg/m ³) (ρ_{ds}) [1]	Dry Solid Volume Fraction (ϵ_{ds})	Regain (@ $\phi=0.65$) (R_f)
Wool	6.43×10^{-4}	0.318	495	1300	0.381	0.150
Silk	1.35×10^{-4}	0.0688	484	1340	0.361	0.100
Cotton	3.84×10^{-4}	0.200	521	1550	0.336	0.070
Wool/Polyester	4.42×10^{-4}	0.230	520	1354	0.384	0.062
Nylon/Cotton	4.63×10^{-4}	0.255	550	1345	0.409	0.056
Nylon	8.61×10^{-4}	0.353	410	1140	0.360	0.041
Polyester	5.89×10^{-4}	0.240	407	1390	0.293	0.004

Table A-2 presents diffusion properties determined in the DMPC at 20°C, with top humidity at 0.60, bottom humidity at 0.0. The apparent boundary layer resistance was taken to be 95 s/m.

Table A-2 Diffusion Properties of Two Fabric Layers

Sample Identification	Total Diffusion Resistance (includes boundary layer resistances) (s/m)	Apparent Fabric Diffusivity (m ² /s)	Apparent Tortuosity Factor
Wool	289	6.63 E-6	2.35
Silk	161	4.09 E-6	3.94
Cotton	196	7.60 E-6	2.12
Wool / Polyester	217	7.24 E-6	2.14
Nylon / Cotton	250	5.97 E-6	2.49
Nylon	289	8.87 E-6	1.82
Polyester	194	1.19 E-5	1.50
PTFE Film	112	--	--

Table A-3. Textile Fiber Thermal Properties

	Heat Capacity [J/(kg-K)] [118]	Thermal Conductivity [J/(s-m-K)] [119]
Wool	1360	0.20**
Silk	1380	0.20**
Cotton	1210	0.16
Wool/Polyester*	1348	0.16
Nylon/Cotton*	1320	0.20
Nylon	1430	0.25
Polyester	1340	0.14

* Mass fraction weight averages

** estimated

Table A-4. Darcy Flow Resistance Properties

	$R_{dry} (\phi=0)$ [m ⁻¹]	$R_{sat} (\phi=1)$ [m ⁻¹]
Wool	0.614 × 10 ⁸	1.08 × 10 ⁸
Silk	0.353 × 10 ⁸	0.760 × 10 ⁸
Cotton	1.23 × 10 ⁸	2.40 × 10 ⁸
Wool/Polyester	0.425 × 10 ⁸	0.595 × 10 ⁸
Nylon/Cotton	1.55 × 10 ⁸	1.85 × 10 ⁸
Nylon	0.930 × 10 ⁸	0.868 × 10 ⁸
Polyester	0.226 × 10 ⁸	0.226 × 10 ⁸

Appendix B

One-Dimensional Numerical Code to Solve Transient Coupled Diffusion Problem

Table B-1. Sorption Rate Factors

Material	Value of Fitting Parameter [$h_c = 40.0 \text{ W}/(\text{m}^2 \cdot \text{s})$]	Value of Fitting Parameter [$h_c = 21.8 \text{ W}/(\text{m}^2 \cdot \text{s})$]
	$\frac{D_{solid}}{d_f^2}$	$\frac{D_{solid}}{d_f^2}$
Wool	0.0231	0.00813
Silk	0.0906	0.0363
Cotton	0.0344	0.0144
Wool/Polyester	0.0319	0.0138
Cotton/Nylon	0.0269	0.0113
Nylon	0.0481	0.0194
Polyester	0.0489	0.0256

One-Dimensional Main Program -- EXP1D.F

```

C_____
C ——— THIS IS THE 1-D VERSION ———
C
C — This version can handle mass source term —
C   and unsteady situation
C
C_____
C—— This Program is made unsteady ——
C—— ON ——
C—— January 25, 1993 ——
C
C—— Revised October, 1994——
C—— to include the exponential scheme——
C ——
C—— Modified March, 1995 to be——
C—— One-Dimensional Only——
C_____

      PROGRAM MAIN
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL LSTOP,LSTEADY
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      CALL DEFALT
      CALL GRID
      CALL SETUP1
      CALL START
      CALL UNSTDY
10  CALL DENSE
      CALL BOUND
      CALL OUTPUT
      IF(LSTOP) STOP
      CALL SETUP2
      GO TO 10
      END

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SOLVE
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1  AJP(JD),AJM(JD),AP(JD),
2  Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3  YCVR(JD),YCVRS(JD)
      COMMON PT(MIJ),QT(MIJ)
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
C*****
C      print *, 'SOLVE'

```

```

c      print *,nf
      JSTF=JST-1
      JT1=M2+JST
      JT2=M3+JST
C-----
      N=NF
C----- TDMA -----
      PT(JSTF)=0.
      QT(JSTF)=F(JSTF,N)
      DO 270 J=JST,M2
      DENOM=AP(J)-PT(J-1)*AJM(J)
      PT(J)=AJP(J)/DENOM
      TEMP=CON(J)
      QT(J)=(TEMP+AJM(J)*QT(J-1))/DENOM
270 CONTINUE
      DO 280 JJ=JST,M2
      J=JT1-JJ
280 F(J,N)=F(J+1,N)*PT(J)+QT(J)
      RETURN
C*****
      ENTRY RESET
      DO 400 J=1,M1
      CON(J)=0.0
      AP(J)=0.0
400 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SETUP
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1 AJP(JD),AJM(JD),AP(JD),
2 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3 YCVR(JD),YCVRS(JD)
      COMMON PT(MIJ),QT(MIJ)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/FLOW,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(JD,NFD),FNEW(JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(JD),FP(JD)
      DIMENSION U(JD),V(JD),PC(JD),P(JD)
      EQUIVALENCE(F(1,1),U(1)),(F(1,2),V(1)),(F(1,3),PC(1))
      EQUIVALENCE(F(1,4),P(1))
C*****
1 FORMAT(//,15X,'COMPUTATION IN CARTESIAN COORDINATES')
2 FORMAT(//,15X,'COMPUTATION FOR AXISYMMETRIC SITUATION')

```



```

3 FORMAT(//,15X,'COMPUTATION   IN   POLAR   COORDINATES')
4 FORMAT(14X,40(1H*),//)
C-----
      ENTRY DEFALT
      NP=4
      NFMIN=5
      NFMAX=20
      NRHO=NFMAX+1
      NGAM=NFMAX+2
      NCON=NFMAX+3
      LSTOP=.FALSE.
      LAST=5
      TIME=0.0
      ITER=0
      JITER=0
      LSCHEME=1
      LSTEADY=.FALSE.
      DT=1.0D+20
      JPREF=1
      RHOCON=1.0
      DO 877 IL=1,NFP3
      LSOLVE(IL)=.FALSE.
      LPRINT(IL)=.FALSE.
      LBLK(IL)=.TRUE.
      RELAX(IL)=1.0
877  CONTINUE
      DO 878 K=1,NFD
      EPS(K)=1.0D+20
      IFLAG(K)=1
878  CONTINUE
      RETURN
C-----
      ENTRY SETUP1
      M2=M1-1
      M3=M2-1
      Y(1)=YV(2)
      DO 10 J=2,M2
10   Y(J)=0.5*(YV(J+1)+YV(J))
      Y(M1)=YV(M1)
      DO 35 J=2,M1
35   YDIF(J)=Y(J)-Y(J-1)
      DO 40 J=2,M2
40   YCV(J)=YV(J+1)-YV(J)
CON,AP,U,V,RHO,PC AND P ARRAYS ARE INITIALIZED HERE
      DO 95 J=1,M1
      PC(J)=0.
      U(J)=0.
      V(J)=0.
      CON(J)=0.
      AP(J)=0.
      RHO(J)=RHOCON
      P(J)=0.
95  CONTINUE
      PRINT 4

```

```

      RETURN
C-----
      ENTRY SETUP2
c      print *, 'SETUP2'
COEFFICIENTS FOR THE U EQUATION
      NF=1
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 100
100 CONTINUE
COEFFICIENTS FOR THE V EQUATION-----
      NF=2
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 200
      IF(LSTEADY) IFLAG(NF)=0
200 CONTINUE
      NF=3
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 300
      IF(LSTEADY) IFLAG(NF)=0
300 CONTINUE
COEFFICIENTS FOR THE PRESSURE EQUATION-----
      NF=NP
      IFLAG(NF)=1
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 500
500 CONTINUE
COEFFICIENTS FOR OTHR EQUATIONS-----
c      print *, 'other equations'
c      print *,nf
      JST=2
      DO 600 NF=NFMIN,NFMAX
      IF(.NOT.LSOLVE(NF)) GO TO 600
      IF(LSTEADY) IFLAG(NF)=0
      CALL RESET
      CALL GAMSOR
c      print *, 'other equations after not lsolve, nf, lsolve'
c      print *, nf,lsolve(nf)
      REL=1.-RELAX(NF)
      DIFF=GAM(1)/YDIF(2)
      AJM(2)=DIFF
      DO 603 J=2,M2
      IF(J.EQ.M2) GO TO 606
      DIFF=2.*GAM(J)*GAM(J+1)/(YCV(J)*GAM(J+1)
1+YCV(J+1)*GAM(J))
      GO TO 607
606 DIFF=GAM(M1)/YDIF(M1)
607 AJM(J+1)=DIFF
      AJP(J)=AJM(J+1)
      VOL=YCV(J)
      APT=VOL*RHO(J)/DT
      CON(J)=CON(J)*VOL/APT
      AP(J)=-AP(J)*VOL+AJP(J)+AJM(J)
      ALAMBDA(J)=AP(J)/APT
      AJP(J)=AJP(J)/APT

```

```

      AJM(J)=AJM(J)/APT
603  CONTINUE
      CALL FACTOR
      DO 608 J=2,M2
      AP(J)=1.0+ALAMBDA(J)*FP(J)
      CON(J)=CON(J)+AJP(J)*(1.-FP(J+1))*FOLD(J+1,NF)
      1+AJM(J)*(1.-FP(J-1))*FOLD(J-1,NF)
      CON(J)=CON(J)+(1.0+FP(J)*ALAMBDA(J)-ALAMBDA(J))*
      1FOLD(J,NF)
      AJP(J)=AJP(J)*FP(J+1)
      AJM(J)=AJM(J)*FP(J-1)
      AP(J)=AP(J)/RELAX(NF)
      REL=1.-RELAX(NF)
      CON(J)=CON(J)+REL*AP(J)*F(J,NF)
608  CONTINUE
C-----
      DO 730 J=JST,M2
      FNEW(J)=F(J,NF)
730  CONTINUE
C-----
      CALL SOLVE
C-----
      DIFMAX=0.0
      DO 731 J=2,M2
      DELTAT=DABS((F(J,NF)-FNEW(J))/(F(J,NF)+1.D-200))
      DIFMAX=DMAX1(DIFMAX,DELTAT)
731  CONTINUE
      IF(DIFMAX.LE.EPS(NF)) IFLAG(NF)=1
      call output
C-----
600  CONTINUE
      JITER=JITER+1
      IF(JITER.LT.2) RETURN
C*****
      JFLAG=0
      DO 732 N=1,NFD
      JFLAG=JFLAG+IFLAG(N)
732  CONTINUE
      IF(JFLAG.EQ.NFD) GO TO 742
      IF(JITER.LT.JLAST) RETURN
      CALL BOUND
      CALL SAVE1
      CALL OUTTOO
      CALL SAVEPC
      print *, 'Number of Iterations Exceeds JLAST'
      LSTOP=.TRUE.
      RETURN
C*****
742  CONTINUE
      JITER=0
      ITER=ITER+1
      IF(LSTEADY) TIME=TIME+DT
      CALL UNSTDY
      CALL BOUND

```



```

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1 AJP(JD),AJM(JD),AP(JD),
2 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3 YCVR(JD),YCVRS(JD)
      COMMON PT(MIJ),QT(MIJ)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/Flow,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(JD,NFD),FNEW(JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(JD),FP(JD)
C*****1-exponential scheme, 2-implicit, 3-Crank-Nicolson
      GO TO (1,2,3,4)LScheme
C*****EXPONENTIAL SCHEME
1 DO 10 J=2,M2
  IF (ALAMBDA(J).GE.0..AND.ALAMBDA(J).LT.10.) THEN
    FP(J)=1.-(1.-(1.-.1*ALAMBDA(J))**5)/ALAMBDA(J)
  END IF
  IF (ALAMBDA(J).GT.10.) FP(J)=1.-1./ALAMBDA(J)
  FP(1)=1.0
  FP(M1)=1.0
10 CONTINUE
  RETURN
C*****IMPLICIT SCHEME
2 DO 20 J=1,M1
  FP(J)=1.0
20 CONTINUE
  RETURN
C*****CRANK-NICOLSON SCHEME
3 DO 30 J=1,M1
  FP(J)=0.5
30 CONTINUE
C*****EXPLICIT SCHEME
4 DO 40 J=1,M1
  FP(J)=0.0
40 CONTINUE
  RETURN
  END

```

User Program -- DIFFUSE.F

```

C*****APRIL 1995
      SUBROUTINE DIFFUSE
C*****This is 1-d test case for vapor diffusion problem
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE*60
      CHARACTER MARKER*4
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY,IBUG
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      PARAMETER (ISLAB=21)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1  AJP(JD),AJM(JD),AP(JD),
2  Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3  YCVR(JD),YCVRS(JD)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/Flow,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(JD,NFD),FNEW(JD),JLAST,
1  IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(JD),FP(JD)
      DIMENSION U(JD),V(JD),PC(JD),P(JD)
      EQUIVALENCE(F(1,1),U(1)),(F(1,2),V(1)),(F(1,3),PC(1))
      EQUIVALENCE(F(1,4),P(1))
C*****
C
C*****UNSTEADY DIFFUSION PROBLEM*****
      COMMON/PROPS/datafile,iwrite,nprint,RF,TAU,EDS,RHODS,
1RHOW,TKDS,TKW,TKV,TKA,CPDS,CPW,CPV,CPA,PATM,RGAS,XMW,TKEFF1,
1XMA,CHTC,CMTC,DSOLID,METHOD,EPSRHOV,aL,aL2,DDRY,DWET,THICK
      COMMON/BNDRY/PSATB,RHBND,TBND,PVBND,RHOVBND,PABND,RHOABND,
1PSATBB,PSATBT,RHBNDB,RHBNDT,TBNDB,TBNDT,PVBNDB,PVBNDT,
1RHOVBNDB,RHOVBNDT,PABNDB,PABNDT,RHOABNDB,RHOABNDT,
1WEIGHTI,WEIGHT,WEIGHTG,RHINT,EBWINT,EBWSAT,TINT
      COMMON/DSORC/cf2(101),cfol2(101),tsoln2(101),cfdim(101),
1ta2(101),tb2(101),tc2(101),trr2(101),alpha(101),
1cfnew(JD,islab),cfold(JD,islab)
      DIMENSION T(JD),RHOV(JD),EBW(JD),XMDOT(JD),
1EGAM(JD),RHOAIR(JD),PV(JD),PA(JD),PSAT(JD)
      EQUIVALENCE(F(1,5),RHOV(1)),(F(1,6),T(1)),
1(F(1,7),EBW(1)),(F(1,8),XMDOT(1)),
1(F(1,9),EGAM(1)),(F(1,10),RHOAIR(1)),
1(F(1,11),PV(1)),(F(1,12),PA(1)),(F(1,13),PSAT(1))
      common /phil/ dxs,dtS
      DATA TITLE(5)/7H      RHOV/
      DATA TITLE(6)/7H      TEMP/

```

```

DATA TITLE(7)/7H    EBW/
DATA TITLE(8)/7H  XMDOT/
DATA TITLE(9)/7H   EGAM/
DATA TITLE(10)/7H RHOAIR/
DATA TITLE(11)/7H  PVAP/
DATA TITLE(12)/7H  PAIR/
DATA TITLE(13)/7H  PSAT/
DATA IBUG/.false./
C*****
ENTRY GRID
IF (IBUG) PRINT *, "ENTRY GRID"
C*****FABRIC THICKNESS*****
CALL PROPERTY
YL=THICK
C*****NUMBER OF GRID POINTS
M1=11
CALL UGRID
RETURN
C*****
C
ENTRY START
IF (IBUG) PRINT *, "ENTRY START"
c*****input the data file name
nprint=100
iwrite=nprint
open(unit=10,file='filename')
read(10,*)datafile
close(10)
marker='a'
C*****LScheme=1-exponential,2-implicit,3-Crank-Nicolson
LSTEADY=.TRUE.
LScheme=1
C*****MASS SOURCE METHOD; 1-EQUILIBRIUM, 2-SOLID DIFFUSION,
C*****SEMI-INFINITE SOLID, 3-SOLID DIFFUSION, FINITE SLAB,
C*****4-SORPTION EQUILIBRIA,5-variable diffusion coefficient,
C*****6-alternate2
METHOD=4
LSOLVE(5)=.TRUE.
LSOLVE(6)=.TRUE.
DT=1.D-2
EPS(5)=1.D-3
EPS(6)=1.D-3
JLAST=10000
LAST=360000
RELAX(5)=.5D0
RELAX(6)=1.D0
c    CALL PROPERTY
C*****INITIAL CONDITIONS
TINT=293.15D0
RHINT=.00D0
EBWINT=(0.578D0*RF*EDS*RHODS/RHOW)
1*(RHINT)*((1.D0/(.321D0+(RHINT))))
1+(1.D0/(1.262D0-(RHINT))))
RH100=1.D0

```

```

      EBWSAT=(0.578D0*RF*EDS*RHODS/RHOW)
      1*(RH100)*((1.D0/(.321D0+(RH100)))
      1+(1.D0/(1.262D0-(RH100))))
      PSATI=614.3D0*DEXP(17.06D0*((TINT-273.15D0)
      1/(TINT-40.25D0)))
      PVINT=RHINT*PSATI
      RHOVINT=PVINT*XMW/(RGAS*TINT)
C*****BOUNDARY CONDITIONS
C*****TOP OF SLAB
      RHBNDT=0.99D0
      TBNDT=293.15D0
      PSATBT=614.3D0*DEXP(17.06D0*((TBNDT-273.15D0)
      1/(TBNDT-40.25D0)))
      PVBNDT=PSATBT*RHBNDT
      PABNDT=PATM-PVBNDT
      RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
      RHOVBNDT=PVBNBT*XMW/(RGAS*TBNDT)
C*****BOTTOM OF SLAB
      RHBNDB=.99D0
      TBNDB=293.15D0
      PSATBB=614.3D0*DEXP(17.06D0*((TBNDB-273.15D0)
      1/(TBNDB-40.25D0)))
      PVBNDB=PSATBB*RHBNDB
      PABNDB=PATM-PVBNDB
      RHOABNDB=PABNDB*XMA/(RGAS*TBNDB)
      RHOVBNDB=PVBNDB*XMW/(RGAS*TBNDB)
      DO 100 J=1,M1
      T(J)=TINT
      XMDOT(J)=0.D0
      EBW(J)=EBWINT
      IF(METHOD.EQ.3) THEN
      DO 102 L1=1,ISLAB
      CFOLD(J,L1)=RHOW*EBW(J)/EDS
      CFNEW(J,L1)=RHOW*EBW(J)/EDS
102 CONTINUE
      END IF
      EGAM(J)=1.D0-EDS-EBW(J)
      PV(J)=PVINT
      RHOV(J)=RHOVINT
      PA(J)=PATM-PV(J)
      RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
      PSAT(J)=614.3D0*DEXP(17.06D0*((T(J)
      1-273.15D0)/(T(J)-40.25D0)))
      IF(J.EQ.1.) THEN
      T(J)=TBNDB
      PV(J)=PVBNDB
      RHOV(J)=RHOVBNDDB
      PA(J)=PABNDB
      RHOAIR(J)=RHOABNDB
      PSAT(J)=PSATBB
      END IF
      IF(J.EQ.M1) THEN
      T(J)=TBNDT
      PV(J)=PVBNDT

```



```

        RHOV(J)=RHOVBNDT
        PA(J)=PABNDT
        RHOAIR(J)=RHOABNDT
        PSAT(J)=PSATBT
        END IF
100 CONTINUE
        LPRINT(5)=.TRUE.
        LPRINT(6)=.TRUE.
        LPRINT(7)=.TRUE.
        LPRINT(8)=.TRUE.
        DO 101 J=1,M1
        DO 101 NF=1,NFD
        FOLD(J,NF)=F(J,NF)
101 CONTINUE
C*****CALCULATE INITIAL WEIGHT (MASS/SQ. METER)
        WEIGHTI=0.0D0
        DO 103 J=2,M2
        RHOAVG=RHOW*EBW(J)+RHODS*EDS+
1 (RHOV(J)+RHOAIR(J))*EGAM(J)
        VOLUME=YCV(J)
        WEIGHTI=WEIGHTI+VOLUME*RHOAVG
103 CONTINUE
C*****record run variables
        open(10,access='append',file=datafile)
        write(10,*)"dt,eps(5),eps(6),rel(5),rel(6)"
        write(10,190)marker,dt,eps(5),eps(6),relax(5),
1relax(6)
        write(10,*)"M1,LScheme,METHOD,JLAST,LAST"
        write(10,191)marker,M1,lscheme,method,jlast,last
        write(10,*)"Thickness,Rf,aL,DSOLID,WEIGHTI,EBWINT"
        write(10,189)marker,YL,RF,aL,dsolid,weighti,ebwint
        write(10,*)"rhint,rhbndt,rhbndb,tint,tbndb"
        write(10,190)marker,rhint,rhbndt,rhbndb,tint,tbndb
        write(10,*)"time,t,rhov,ebw,xmdot,weight,% change"
        write(10,*)"//nc"
        close(10)
189 format(a4,6e12.4)
190 format(a4,5e12.4)
191 format(a4,5i10)
        RETURN

C*****
C
        ENTRY UNSTDY
C*****FOLD=FNEW
        IF (IBUG) PRINT *, "ENTRY UNSTEADY"
        DO 200 J=1,M1
        DO 200 NF=1,NFD
        FOLD(J,NF)=F(J,NF)
200 CONTINUE
        IF (METHOD.EQ.3) THEN
        DO 201 J=1,M1
        DO 201 J1=1,ISLAB
        CFNEW(J,J1)=CFOLD(J,J1)

```

```

201 CONTINUE
    END IF
    RETURN

C*****
C
    ENTRY DENSE
    IF (IBUG) WRITE (*,*) "ENTRY DENSE"
    if(ibug)print *,nf
    CALL THERMO
    CALL SOURCE
    RETURN

C*****
C
    ENTRY BOUND
    IF(IBUG)print *, 'entry bound'
    RETURN

C*****
C
    ENTRY OUTPUT
    if(ibug)print *, 'ENTRY OUTPUT'
    if(ibug)print *,nf
    ncent=(M1/2)+1
    RETURN

    ENTRY OUTPUT2
    RETURN

    ENTRY OUTPUT3
    RETURN

C*****
C
    ENTRY GAMSOR
    IF (IBUG) print *, 'entry gamsor'
    if(ibug)print *,nf
    GO TO (699,699,699,699,610,600) NF
    GO TO 699
c*****energy equation, nf=6
    600 CONTINUE
c*****call ERHO to get GAM and RHO
    CALL ERHO
    DO 601 J=2,M2
    DHVAP=(2.792D+6)-160.D0*t(j)-3.43D0
    1*t(j)*t(j)
    rhumid=pv(j)/psat(j)
    QL=(1.95D+5)*(1.D0-rhumid)
    1*((1.D0/(0.2D0+(rhumid)))+
    1(1.D0/(1.05D0-(rhumid))))
c*****ALL SOURCE TERMS LUMPED INTO CON(J) (LAZY WAY)
    CON(J)=- (DHVAP+QL)*XMDOT(J)
    AP(J)=0.D0
    601 CONTINUE

```

```

c*****CREATE CONVECTIVE BOUNDARY CONDITIONS
  GDYM=GAM(1)/YDIF(2)
  GDYP=GAM(M1)/YDIF(M1)
  AREAM=1.0D0/((1.0D0/CHTC+1.0D0/GDYM)*YCV(2))
  AREAP=1.0D0/((1.0D0/CHTC+1.0D0/GDYP)*YCV(M2))
  GAM(1)=0.0D0
  GAM(M1)=0.0D0
  CON(2)=CON(2)+AREAM*TBNDB
  AP(2)=AP(2)-AREAM
  CON(M2)=CON(M2)+AREAP*TBNDT
  AP(M2)=AP(M2)-AREAP
  GO TO 699

C*****CALCULATE THERMODYNAMIC PROPERTIES AND XMDOT

  610 CONTINUE

      DO 611 J=1,M1
C*****DENSITY TERM FOR MASS DIFFUSION EQUATION
      RHO(J)=fold(j,9)
c      rho(j)=ebw(j)
C*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
      DAIR=(2.20D-5)*((t(j)/273.15D0)**1.75D0)
      DEFF=DAIR*fold(j,9)/TAU
      DEFF=DAIR*egam(j)/TAU
      GAM(J)=DEFF
      xsource=xmdot(j)-rhov(j)*(egam(j)-fold(j,9))/dt
c      xsource=xmdot(j)-fold(j,5)*(egam(j)-fold(j,9))/dt
      if(xsource.lt.0.)then
        con(j)=0.D0
        ap(j)=xsource/rhov(j)
      end if
      if(xsource.ge.0.)then
        con(j)=xsource
        ap(j)=0.D0
      end if
      611 CONTINUE
C*****CREATE CONVECTIVE BOUNDARY CONDITIONS
      GDYM=GAM(1)/YDIF(2)
      GDYP=GAM(M1)/YDIF(M1)
      AREAM=1.0D0/((1.0D0/CMTC+1.0D0/GDYM)*YCV(2))
      AREAP=1.0D0/((1.0D0/CMTC+1.0D0/GDYP)*YCV(M2))
      GAM(1)=0.0D0
      GAM(M1)=0.0D0
      CON(2)=CON(2)+AREAM*RHOVBNDDB
      AP(2)=AP(2)-AREAM
      CON(M2)=CON(M2)+AREAP*RHOVBNDT
      AP(M2)=AP(M2)-AREAP
      699 RETURN

C*****
C
      ENTRY SAVE1

```

RETURN

C*****

C

ENTRY OUTTOO

LPRINT(5)=.TRUE.

LPRINT(6)=.TRUE.

if(time.gt.30.)nprint=200

if (iwrite.eq.nprint)then

ncent=(M1/2)+1

if(iter.eq.1)weightg=0.

pctchnng=weightg/weighti

open(10,access='append',file=datafile)

t1=t(2)-tint

t2=t(ncent)-tint

reffk=y1/tkeff1

write(*,813)time,t1,t2,rhov(ncent),

1ebw(ncent),xmdot(ncent),weight

write(10,811)time,t1,t2,rhov(ncent),

1ebw(ncent),xmdot(ncent),weight,pctchnng

close(10)

c do 810 j=1,M1

c rhumid=pv(j)/psat(j)

c write(*,812)t(j),ebw(j),xmdot(j),rhov(j),rhumid

c 810 continue

iwrite=0

endif

iwrite=iwrite+1

811 format(1e10.4,1x,f6.2,1x,f6.2,5e10.3)

813 format(1e10.4,1x,f6.2,1x,f6.2,4e11.4)

812 format (5e11.4)

RETURN

C*****

C

ENTRY SAVEPC

RETURN

C*****

C

ENTRY SAVE2

RETURN

ENTRY PROPERTY

if(ibus)print *,'entry property'

open(unit=10,file='fabric.prp')

C*****FABRIC PROPERTIES

C*****THICKNESS OF ONE FABRIC LAYER

THICK1=6.43D-4

read(10,531)thick1

```

c*****THESIS EXPERIMENTS USED TWO FABRIC LAYERS
      THICK=2.*THICK1
c*****VOLUME FRACTION OF DRY SOLID
      EDS=0.381D0
      read(10,531)eds
c*****SOLID PHASE DIFFUSION COEFFICIENT OF WATER
      DSOLID=4.D-13
      read(10,531)dsolid
      DDRY=4.D-14
      DWET=4.D-10
c*****EFFECTIVE FIBER DIAMETER (m)
      aL=4.D-6
      read(10,531)aL
      aL2=1.5D-5
c*****MASS FRACTION OF WATER TO DRY SOLID AT 65% r.h.
      RF=0.15D0
      read(10,531)rf
c*****TORTUOSITY
      TAU=2.35D0
      read(10,531)tau
c*****SOLID POLYMER DRY DENSITY (KG/M^3)
      RHODS=1300.
      read(10,531)rhods
c*****THERMAL CONDUCTIVITIES OF DRY POLYMER (J/s-m-K)
      TKDS=0.20D0
      read(10,531)tkds
c*****HEAT CAPACITY OF DRY POLYMER (J/kg-K)
      CPDS=1360.D0
      read(10,531)cpds
      close(10)
      531 format(e15.5)
c*****PROBLEM CONSTANTS
c*****CONVECTIVE HEAT TRANSFER COEFFICIENTS AT BOUNDARIES
c***** (usually at 21.8 for DMPC simulations)
      CHTC=21.8D0
c*****CONVECTIVE MASS TRANSFER COEFFICIENTS AT BOUNDARIES
c***** (measured from apparent boundary layer resistance in DMPC)
c***** usually at 0.021 m/s
      CMTC=.021D0
c*****DENSITY OF LIQUID WATER
      RHOW=1000.D0
c*****THERMAL CONDUCTIVITIES OF LIQUID WATER, WATER VAPOR, AND AIR
      TKW=0.6D0
      TKV=0.0246D0
      TKA=0.02563D0
c*****HEAT CAPACITIES OF LIQUID WATER, WATER VAPOR, AND AIR
      CPW=4182.D0
      CPV=1862.D0
      CPA=1003.D0
c*****TOTAL GAS PRESSURE (ATMOSPHERIC)
      PATM=101325.D0
c*****UNIVERSAL GAS CONSTANT
      RGAS=8314.5D0
c*****MOLECULAR WEIGHTS

```

```

XMW=18.015D0
XMA=28.97D0

RETURN

ENTRY THERMO
if(ibug)print *,'entry thermo'
DO 1021 J=1,M1
  PSAT(J)=614.3D0*DEXP(17.06D0*((T(J)-273.15D0)/
1(T(J)-40.25D0)))
  PV(J)=RHOV(J)*RGAS*T(J)/XMW
  PA(J)=PATM-PV(J)
  RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
1021 CONTINUE
RETURN

ENTRY SOURCE
if(ibug)print *,'entry source'
DO 1029 J=2,M2
  RHUMID=PV(J)/PSAT(J)
  EBW(J)=(.578D0*RF*EDS*RHODS/RHOW)*RHUMID*
1(1.D0/(.321D0+RHUMID)+1.D0/(1.262D0-RHUMID))
  XMDOT(J)=(FOLD(J,7)-EBW(J))*RHOW/DT
1029 CONTINUE

C*****GO TO PROPER MASS SOURCE CALCULATION METHOD
      GOTO(1025,1030,1035,1030,1030,1030)METHOD

C*****EQUILIBRIUM METHOD-SOURCE TERM DOMINANT
1025 DO 1026 J=2,M2
  XFRAC=0.6
  IF(XMDOT(J).LT.0.)THEN
C    PHILIM=1.D-150
C    PHILIM=(1.D-200)+(RHOV(J))*(1.-XFRAC)
C    CON(J)=(XMDOT(J)*PHILIM)/(PHILIM-RHOV(J))
C    AP(J)=-XMDOT(J)/(PHILIM-RHOV(J))
C    CON(J)=0.
C    AP(J)=XMDOT(J)/RHOV(J)
  END IF
  IF(XMDOT(J).GE.0.)THEN
C    RHOVSAT=(PSAT(J)*XMW)/(RGAS*T(J))
C    PHILIM=RHOVSAT
C    PHILIM=RHOV(J)+(RHOVSAT-RHOV(J))*XFRAC
C    CON(J)=(XMDOT(J)*PHILIM)/(PHILIM-RHOV(J))
C    AP(J)=-XMDOT(J)/(PHILIM-RHOV(J))
C    CON(J)=XMDOT(J)
C    AP(J)=0.
  END IF
C    EBW(J)=FOLD(J,7)-XMDOT(J)*DT/RHOW
  EGAM(J)=1.-EDS-EBW(J)
1026 CONTINUE
      GOTO 1050

C*****SOLID DIFFUSION METHOD, SEMI-INFINITE SOLID

```

```

1030 DO 1031 J=2,M2
    CSKIN=RHOW*EBW(J)/(EBW(J)+EDS)
    CINSIDE=RHOW*FOLD(J,7)/(FOLD(J,7)+EDS)
    PI=3.141592654D0
    XMDOT(J)=(CINSIDE-CSKIN)*DSQRT(DSOLID/(DT*PI))
    IF (METHOD.EQ.4) THEN
        REQ=EBW(J)*EDS*RHODS/RHOW
        RINST=FOLD(J,7)*EDS*RHODS/RHOW
        XMDOT(J)=DSOLID*RHODS*(RINST-REQ)/(aL*aL)
    END IF
    IF (METHOD.EQ.5) THEN
        REQ=EBW(J)*EDS*RHODS/RHOW
        RINST=FOLD(J,7)*EDS*RHODS/RHOW
        DSEFF=DDRY+(DWET-DDRY)*EBW(J)/EBWSAT
        XMDOT(J)=DSEFF*RHODS*(RINST-REQ)/(aL2*aL2)
    END IF
    IF (METHOD.EQ.6) THEN
        REQ=EBW(J)*EDS*RHODS/RHOW
        RINST=FOLD(J,7)*EDS*RHODS/RHOW
        XMDOT(J)=DSOLID*RHODS*(RINST-REQ)
    END IF
C    XMDOT(J)=(1.-RHOV(J)/RHOW)*XMDOT(J)
C    IF (XMDOT(J).LT.0.) THEN
C        CON(J)=0.D0
C        AP(J)=XMDOT(J)/RHOV(J)
C    END IF
C    IF (XMDOT(J).GE.0.) THEN
C        CON(J)=XMDOT(J)
C        AP(J)=0.0D0
C    END IF
    EBW(J)=FOLD(J,7)-XMDOT(J)*DT/RHOW
    EGAM(J)=1.D0-EDS-EBW(J)
1031 CONTINUE
    GOTO 1050

C*****SOLID DIFFUSION METHOD, FINITE SLAB
1035 DO 1036 J=2,M2
    CSKIN=RHOW*EBW(J)/EDS
C*****define the problem constants (thickness)
    sl=1.
C*****parameters from MAIN
    aL=eds*ycv(j)
C*****input time and space step intervals
    tfd=dt
    tf=tfd*dsolid/(aL*aL)
    il=islabs
    nslab=il-1
    kcenter=((il-1)/2)+1
    xlambda=1.
C*****set the initial conditions
C*****cf is nondimensional fiber concentration
    do 1037 i=1,il
        alpha(i)=1.
        cf2(i)=(cfold(j,i)-cfold(j,kcenter))/

```

```

        1(cskin-cfold(j,kcenter)+1.D-100)
        cfol2(i)=(cfold(j,i)-cfold(j,kcenter))/
        1(cskin-cfold(j,kcenter)+1.D-100)
1037 continue
        cfbnd=1.
c*****now compute the important parameters
        nm1=nslab-1
        nm2=nslab-2
        times=0.
        dxs=s1/nslab
        dts=xlambda*dxs*dxs/2.
        tf10=tf/10.
        if(dts.ge.tf10)dts=tf/10.
c*****implement boundary conditions
        call tbcond(cf2,il,cfbnd)
c*****increment the iteration counters and check for the maximum limit
1044 times=times+dts
        tfend=tf
        if(times.gt.tfend)goto 1045
        do 1038 i=1,il
c*****parameters in the finite difference equations
        greekx=dts/(2.*dxs*dxs)
1038 continue
c*****implement boundary conditions
        call tbcond(cf2,il,cfbnd)
c*****PATANKAR'S EXPONENTIAL SOLUTION
c*****form the tridiagonal system of equations
        call fntpat(alpha,il,cfol2,ta2,tb2,tc2,trr2)
        nt=il-2
c*****invert the tridiagonal system of equations
        call tdig2(ta2,tb2,tc2,trr2,tsoln2,nt)
c*****save the old solution
        do 1039 i=2,il-1
        cf2(i)=tsoln2(i)
1039 continue
        do 1040 i=1,il
        cfol2(i)=cf2(i)
1040 continue
c*****impose the boundary condition
        call tbcond(cf2,il,cfbnd)
        xdim=0.
        do 1041 i=1,il
        dxdim=dxs*aL
        tdim=aL*aL*times/dsolid
        cfdim(i)=cf2(i)*(cskin-cfold(j,kcenter))
        1+cfold(j,kcenter)
        cfnew(j,i)=cfdim(i)
        xdim=xdim+dxdim
1041 continue
        go to 1044
c*****integrate concentration to get total mass
c*****use Trap. Rule
1045 totnew=0.
        totold=0.

```



```

      nslab1=nslab-1
      do 1046 i=2,nslab1
        totold=totold+((cfold(j,i)+cfold(j,i+1))/2.)*dxdim
        totnew=totnew+((cfnew(j,i)+cfnew(j,i+1))/2.)*dxdim
1046    continue
      totnew=totnew+dxdim*(cfnew(j,2)+cfnew(j,nslab1))
      totold=totold+dxdim*(cfold(j,2)+cfold(j,nslab1))
      totold2=rhow*fold(j,7)*ycv(j)
      ebw(j)=totnew/(rhow*ycv(j))
      ebdiff1=(fold(j,7)-ebw(j))/(fold(j,7)+1.D-100)
      ebdiff=dabs(fold(j,7)-ebw(j))/(fold(j,7)+1.D-100)
      eblim=1.
      if(ebdiff.gt.eblim)then
        ebw(j)=fold(j,7)-(ebdiff1/ebdiff)*eblim*fold(j,7)
      end if
      xmdot(j)=(totold-totnew)/(ycv(j)*dt)
      XMDOT(J)=(FOLD(J,7)-EBW(J))*RHOW/DT
      XMDOT(J)=(1.-RHOV(J)/RHOW)*XMDOT(J)
      IF(XMDOT(J).LT.0.) THEN
        CON(J)=0.
        AP(J)=XMDOT(J)/RHOV(J)
      END IF
      IF(XMDOT(J).GE.0.)THEN
        CON(J)=XMDOT(J)
        AP(J)=0.0
      END IF
      EBW(J)=FOLD(J,7)-XMDOT(J)*DT/RHOW
      EGAM(J)=1.-EDS-EBW(J)
1036 CONTINUE

1050 RETURN

      ENTRY ERHO
      WEIGHT=0.0
      DO 1231 J=1,M1
C*****DENSITY AND HEAT CAPACITY FOR ENERGY EQUATION
      RHOAVG=RHOW*fold(j,7)+RHODS*EDS+
      1(fold(j,5)+fold(j,10))*fold(j,9)
      CPAVG=(RHOW*fold(j,7)*CPW+RHODS*EDS*CPDS+
      1(fold(j,5)*CPV+fold(j,10)*CPA)*fold(j,9))/RHOAVG
      RHO(J)=RHOAVG*CPAVG
C*****CALCULATE CURRENT WEIGHT (MASS/SQ. METER)
C*****AND WEIGHT GAIN OR LOSS FROM INITIAL CONDITION
      VOLUME=YCV(J)
      WEIGHT=WEIGHT+VOLUME*RHOAVG
      WEIGHTG=WEIGHT-WEIGHTI
C*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
      TKF=((TKW*RHOW*ebw(j)+TKDS*RHODS*EDS)
      1/(RHOW*ebw(j)+RHODS*EDS))
      TKG=((TKV*rhov(j)+TKA*rhoair(j))/
      1(rhov(j)+rhoair(j)))
      VF=EDS+ebw(j)
      TKPERP=TKF*TKG/(VF*TKG+(1.D0-VF)*TKF)
      TKPRLL=(1.D0-VF)*TKG+VF*TKF

```

```

      WOMEQA=1.D0
      TKEFF=WOMEQA*TKPRL+(1.0D0-WOMEQA)*TKPERP
c*****Cylinder Model
c      TKEFF=TKG*((TKF*(1.D0+VF)+TKG*(1.D0-VF))/
c      1*(TKF*(1.D0-VF)+TKG*(1.+VFD0)))
      GAM(J)=TKEFF
      ncent=(M1/2)+1
      if(j.eq.ncent)tkeff1=tkeff
1231 CONTINUE

      RETURN
C*****

      END

      subroutine tbcond(tc,il,cfbnd)
c*****this subroutine implements the conc. boundary conditions
      implicit double precision (a-h,o-z)
      dimension tc(101)
c*****CONSTANT conc. BOUNDARY CONDITIONS
c*****left boundary
      tc(1)=cfbnd
c*****right boundary
      tc(il)=cfbnd
      return
      end

      subroutine tdig2(a2,b2,c2,r2,soln2,n)
c*****this subroutine inverts a tridiagonal matrix by the thomas algorithm
      implicit double precision (a-h,o-z)
      dimension a2(101),b2(101),c2(101),r2(101),soln2(101),bn2(101)
      do 10 i=1,n
      bn2(i)=b2(i)
10 continue
      do 21 i=2,n
      d=a2(i)/bn2(i-1)
      bn2(i)=bn2(i)-c2(i-1)*d
      r2(i)=r2(i)-r2(i-1)*d
21 continue
      soln2(n+1)=r2(n)/bn2(n)
      do 30 i=1,n-1
      j=n-i
      soln2(j+1)=(r2(j)-c2(j)*soln2(j+2))/bn2(j)
30 continue
      return
      end

      subroutine fmpat(alphaa2,il,fold2,a2,b2,c2,r2)
      implicit double precision (a-h,o-z)
      common /phil/ dxs,dt
      dimension fold2(101),r2(101),a2(101),b2(101),c2(101)
      dimension ae2(101),aw2(101),ap2(101),fp2(101),alphaa2(101)
      call coeff2(alphaa2,il,aw2,ae2,ap2)
      call fpat(il,ap2,fp2)

```

```

do 10 i=2,il-1
a2(i-1)=-aw2(i)*fp2(i-1)*dts
c2(i-1)=-ae2(i)*fp2(i+1)*dts
10 continue
do 22 i=2,il-1
b2(i-1)=1.+fp2(i)*ap2(i)*dts
z12=(aw2(i)*((1.-fp2(i-1))*dts))*fold2(i-1)
z22=(1.-((1.-fp2(i))*ap2(i)*dts))*fold2(i)
z32=(ae2(i)*((1.-fp2(i+1))*dts))*fold2(i+1)
r2(i-1)=z12+z22+z32
c*****incorporate the appropriate boundary conditions
if(i.eq.2)r2(i-1)=r2(i-1)-a2(i-1)*fold2(i-1)
if(i.eq.il-1)r2(i-1)=r2(i-1)-c2(i-1)*fold2(i+1)
22 continue
return
end

subroutine coeff2(alpha2,il,aw2a,ae2a,ap2a)
implicit double precision (a-h,o-z)
common /phil/ dxs,dts
dimension alpha2(101),aw2a(101),ae2a(101),ap2a(101)
zeta=dxs*dxs
do 22 i=2,il-1
aw2a(i)=(2.*alpha2(i)*alpha2(i-1))/(alpha2(i)+alpha2(i-1))/zeta
ae2a(i)=(2.*alpha2(i)*alpha2(i+1))/(alpha2(i)+alpha2(i+1))/zeta
ap2a(i)=aw2a(i)+ae2a(i)
22 continue
return
end

subroutine fpat(il,ap2aa,fp2a)
implicit double precision (a-h,o-z)
common /phil/ dxs,dts
dimension fp2a(101),ap2aa(101)
do 22 i=2,il-1
xlmbdap=ap2aa(i)*dts
if(xlmbdap.ge.0..and.xlmbdap.lt.10.)then
fp2a(i)=1.-(1./xlmbdap)*(1.-((1.-(0.1*xlmbdap))**5.))
end if
if(xlmbdap.ge.10.)fp2a(i)=(1.-(1./xlmbdap))
22 continue
return
end

```


Appendix C

Two-Dimensional Numerical Code to Solve Transient and Steady-State Coupled Diffusion/Convection Problem

User Program -- DIFFCON.F

```

c*****DECEMBER 1995
      SUBROUTINE diffcon
c*****Diffusion/Convection in the DMPC
c*****THIS IS USER ROUTINE FOR SIMPLEC
c*****this routine includes the sample holder step
c*****This is flow simulation for DMPC cell
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
c*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE1*60
      CHARACTER DATAFILE2*60
      character datafile3*60
      character datafile4*60
      character datafile5*60
      character header*64
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY,IBUG
      PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
      COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1 AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2 X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4 YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5 R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
      COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1 FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/FLOW,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(ID,JD,NFD),FNEW(ID,JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(ID,JD),FP(ID,JD)
      COMMON/DEBUG/IBUG
      COMMON/OTHER/GAM5(ID,JD),RHO5(ID,JD),GAM6(ID,JD),RHO6(ID,JD)
      DIMENSION U(ID,JD),V(ID,JD),PC(ID,JD),P(ID,JD)
      EQUIVALENCE(F(1,1,1),U(1,1)),(F(1,1,2),V(1,1)),(F(1,1,3),PC(1,1))
      EQUIVALENCE(F(1,1,4),P(1,1))
c*****
c
c*****UNSTEADY DIFFUSION PROBLEM*****
      COMMON/PROPS/RF,TAU,RHODS,RHOW,TKDS,TKW,TKV,TKA,
1CPDS,CPW,CPV,CPA,PATM,RGAS,XMW,XMA,CHTC,CMTC,dsolid,
1method,epsrhov,iwrite,nprint,aL,aL2,ddry,dwet,dhvap,
1q1,header,tkss,cpss,rhoss,edss,edsa,edsh,amu,amus,xmn2,
1ykdar,c,kdar,c,deltap,pinch,facttop,factbot,rddry,rdwet,
1thick,rfl

```

```

common/datafile/datafile1,datafile2,datafile3,datafile4,
1datafile5
COMMON/BNDRY/PSATB,RHBND,TBND,PVBND,RHOVBND,PABND,RHOABND
1,psatbb,psatbt,rhbndb,rhbndt,tbndb,tint,ebwint,ebwsat,
1tbndt,pvbndb,pvbndt,rhovbndb,rhovbndt,pabndb,pabndt,
1rhoabndb,rhoabndt,weighti,weight,rhint,tintt,tintb
COMMON/DUCT/RHOA,DUCTH,DUCTA,DUCTW,RHON2,VNOM,
1dn2,deff,rhovtop,rhovbot,rhoat,rhoab,isbeg,isend,
1jsbeg,jsend,jhlbeg,jhlend,jhubeg,jhuend,vnomt,vnomb,
1xmassti,xmassbi,xmassto,xmassbo,xtotin,xtotout,xmasssi,
1xmassso,xsample,ysample,xqn2mks,dcssi,dctopi,dcboti,
1dcboto,dctopo,xqst,xqsb,icent,jcent,convcte,ihend
1,ltype,ivalvbeg,ivalvend,jvalvbeg,jvalvend
common/vapor/xmvt,xmvtb,xmn2t,xmn2b,xmn2out,xmvout,rhbo
common/solid/kplot(nfp3),isolid(id,jd),tplot
DIMENSION T(ID,JD),RHOV(ID,JD),EBW(ID,JD),XMDOT(ID,JD),
1EGAM(ID,JD),RHOAIR(ID,JD),PV(ID,JD),PA(ID,JD),PSAT(ID,JD),
1EDS(ID,JD),TKEFF(ID,JD),RHUMIDA(ID,JD),vapmf(id,jd)
1,xsolid(id,jd),diffg(id,jd),rkdarcy(id,jd)
EQUIVALENCE(F(1,1,5),vapmf(1,1)),(F(1,1,6),T(1,1)),
1(F(1,1,7),EBW(1,1)),(F(1,1,8),XMDOT(1,1)),
1(F(1,1,9),EGAM(1,1)),(F(1,1,10),RHOAIR(1,1)),
1(F(1,1,11),PV(1,1)),(F(1,1,12),PA(1,1)),
1(F(1,1,13),PSAT(1,1)),(F(1,1,14),EDS(1,1)),(F(1,1,15),
1TKEFF(1,1)),(F(1,1,16),RHUMIDA(1,1)),(f(1,1,17),rhov(1,1))
1,(F(1,1,18),xsolid(1,1)),(f(1,1,19),diffg(1,1)),(f(1,1,20),
1rkdarcy(1,1))
DATA TITLE(1)/7H VEL U/
DATA TITLE(2)/7H VEL V/
data title(3)/7H Str.F./
data title(4)/7H Press./
DATA TITLE(5)/7H vapmf/
DATA TITLE(6)/7H TEMP/
DATA TITLE(7)/7H EBW/
DATA TITLE(8)/7H XMDOT/
DATA TITLE(9)/7H EGAM/
DATA TITLE(10)/7H RHOAIR/
DATA TITLE(11)/7H PVAP/
DATA TITLE(12)/7H PAIR/
DATA TITLE(13)/7H PSAT/
DATA TITLE(14)/7H EDS/
DATA TITLE(15)/7H TKEFF/
DATA TITLE(16)/7H R.H./
DATA TITLE(17)/7H rhov/
DATA TITLE(18)/7H xsolid/
DATA TITLE(19)/7H diffg/
DATA TITLE(20)/7H kdarcy/
DATA TITLE(21)/7H none/
DATA TITLE(22)/7H none/
DATA TITLE(23)/7H none/
DATA IBUG/.false./
c DATA IBUG/.true./
C*****
ENTRY GRID

```

```

        IF (IBUG) PRINT *, "ENTRY GRID"
c*****DEFINE MATERIAL PROPERTIES
        CALL PROPERTY
c*****define grid and various regions of geometry
        call gridgen
        RETURN
c*****
C
        ENTRY START
        IF (IBUG) PRINT *, "ENTRY START"
c*****input the data file names
        nprint=1
        iwrite=nprint
c*****LScheme=1-exponential,2-implicit,3-Crank-Nicolson
        LSTEADY=.false.
        LSCHEME=2
        LSOLVE(1)=.TRUE.
        LSOLVE(2)=.TRUE.
        LSOLVE(3)=.TRUE.
        LSOLVE(4)=.TRUE.
        LSOLVE(5)=.TRUE.
        LSOLVE(6)=.true.
c        print *, 'input relaxation factor for velocities'
c        read(*,*)relxx
        relxx=.9
        relax(1)=relxx
        relax(2)=relxx
        relax(6)=.1
        LAST=60
c*****place reference pressure at outlet of cell
        ipref=11
        jpref=1
c*****INITIAL CONDITIONS FOR GAS FLOWS
c*****initial conditions for flow on top and bottom of cell
c*****TEMPERATURES (K)
        TINT=293.15D0
        TINTT=TINT
        TINTB=TINT
c*****RELATIVE HUMIDITIES
        RHTOPIN=0.99D0
        RHBOTIN=0.00D0
        rhint=0.
c*****SATURATION VAPOR PRESSURES
        PSATI=614.3D0*DEXP(17.06D0*((TINT-273.15D0)
1/(TINT-40.25D0)))
        PSATIT=614.3D0*DEXP(17.06D0*((TINTt-273.15D0)
1/(TINTt-40.25D0)))
        PSATIB=614.3D0*DEXP(17.06D0*((TINTb-273.15D0)
1/(TINTb-40.25D0)))
c*****VAPOR PRESSURES/PARTIAL VAPOR PRESSURE (PA)
        PVTOPIN=RHTOPIN*PSATIT
        PVBOTIN=RHBOTIN*PSATIB
c*****VAPOR CONCENTRATIONS (KG/M^3)
        RHOVTOP=PVTOPIN*XMW/(RGAS*TINTt)

```



```

      RHOVBOT=PVBOTIN*XMW/(RGAS*TINTb)
C*****PARTIAL PRESSURE OF INERT GAS (AIR OR NITROGEN)  (PA)
      PATOP=PATM-PVTOPIN
      PABOT=PATM-PVBOTIN
C*****INERT GAS CONCENTRATION (KG/M^3)
      RHOAT=PATOP*XMA/(RGAS*TINTt)
      RHOAB=PABOT*XMA/(RGAS*TINTb)
C*****TOTAL MIXTURE DENSITY
      RHOTOP=RHOAT+RHOVTOP
      RHOBOT=RHOAB+RHOVBOT
c*****specific humidity ratio
      spgamt=(pvtopin/patop)*(xmww/xma)
      spgamb=(pvbotin/pabot)*(xmww/xma)
c*****initial conditions for the sample
      rh100=1.D0
c*****Nominal Entrance Velocity (m/s)
c*****duct width (m)
      DUCTW=.025
c*****duct area (m^2)
      DUCTA=DUCTH*DUCTW
c*****incoming dry nitrogen flow (cm^3/min) at 273 K
      CMFLOWt=2000.
      cmflowb=2000.
c      print *, 'input cmflowt'
c      read (*,*)cmflowt
c      print *, 'input cmflowb'
c      read (*,*)cmflowb
c*****density of nitrogen at 0 deg C
      RHON20=patm*xma/(rgas*273.15)
      xqst=cmflowt*((1./100.)**3)*(1./60.)
      xqsb=cmflowb*((1./100.)**3)*(1./60.)
c*****mass flow rates of nitrogen (kg/s)
      xmn2t=xqst*RHON20
      xmn2b=xqsb*RHON20
c      print *,xmn2t,xmn2b
c      print *,xqst,xqsb

c*****entering mass flow of water vapor  (kg/s)
      xmnt=(rhovtop/rhoat)*xmn2t
      xmvb=(rhovbot/rhoab)*xmn2b
c      print *, 'rhovtop,rhovbot'
c      print *, 'rhoat,rhoab'
c      print *, 'xmnt,xmvb'
c      print *,rhovtop,rhovbot
c      print *,rhoat,rhoab
c      print *,xmnt,xmvb
c*****total mass flow rates (kg/s) of vapor + air entering
      xmtott=xmn2t+xmnt
      xmtotb=xmn2b+xmvb
c      print *, 'xmtott,xmtotb'
c      print *,xmtott,xmtotb

c*****ENTERING VELOCITIES
c*****mdot=rho*V*A... V=mdot/(rho*A)

```

```

        vnomt=xmtott/(rhotop*ducta)
        vnomb=xmtotb/(rhobot*ducta)
c      print *,rhon20
c      print *,'vnom bottom, vnom top'
c      print *,vnomb,vnomt
c*****Normalize all mass flows by width of duct
        xmtott=xmtott/ductw
        xmtotb=xmtotb/ductw
        xmn2t=xmn2t/ductw
        xmn2b=xmn2b/ductw
        xmvb=xmvb/ductw
        xmvb=xmvb/ductw
        xgst=xgst/ductw
        xgsb=xgsb/ductw
c*****option to start off at atmospheric pressure
        do 1236 i=1,11
        do 1236 j=1,m1
c      p(i,j)=patm
1236 continue
c      print *,p(ipref,jpref)

c*****flow in bottom portion of cell
        DO 100 I=1,L1
        DO 100 J=1,jhlbeg-1
        t(i,j)=tintb
        xmdot(i,j)=0.D0
        EBW(i,j)=0.D0
        EGAM(i,j)=1.D0
        EDS(i,j)=edsa
        pv(i,j)=pvbotin
        rhov(i,j)=rhovbot
        pa(i,j)=pabot
        rhoair(i,j)=rhoab
        psat(i,j)=psatib
        rho(i,j)=rhov(i,j)+rhoair(i,j)
        vapmf(i,j)=rhov(i,j)/rho(i,j)
c*****velocities
        if(i.gt.1.and.j.gt.1)then
            u(i,j)=vnomb
            u(2,j)=vnomb
c*****option to use parabolic entrance profile
c      u(i,j)=6.*VNOMB*(-((y(j)**2)/DUCTH**2))+y(j)/DUCTH)
        v(i,j)=0.
        u(i,1)=0.
        u(2,1)=0.
        end if
100 CONTINUE
c*****FLOW IN LOWER CUT-OUT
        DO 101 I=isbeg,isend
        DO 101 J=jhlbeg,jhlend
        t(i,j)=tintb
        xmdot(i,j)=0.D0
        EBW(i,j)=0.D0
        EGAM(i,j)=1.D0

```

```

      EDS(i,j)=edsa
      pv(i,j)=pvbotin
      rhov(i,j)=rhovbot
      pa(i,j)=pabot
      rhoair(i,j)=rhoab
      psat(i,j)=psatib
      rho(i,j)=rhov(i,j)+rhoair(i,j)
      vapmf(i,j)=rhov(i,j)/rho(i,j)
c*****VELOCITIES
      if(i.gt.1.and.j.gt.1)then
        u(i,j)=0.D0
        u(i+1,j)=0.d0
c*****option to use parabolic entrance profile
c      u(i,j)=6.*VNOMB*(-((y(j)**2)/DUCTH**2))+y(j)/DUCTH)
        v(i,j)=0.
      end if
101 CONTINUE

c*****flow in top portion of cell
      DO 105 I=1,L1
      DO 105 J=JHUEND+1,M1
        t(i,j)=tintt
        xmdot(i,j)=0.D0
        eds(i,j)=edsa
        EBW(i,j)=0.D0
        EGAM(i,j)=1.D0
        pv(i,j)=pvtopin
        rhov(i,j)=rhovtop
        pa(i,j)=patop
        rhoair(i,j)=rhoat
        psat(i,j)=psatit
        rho(i,j)=rhov(i,j)+rhoair(i,j)
        vapmf(i,j)=rhov(i,j)/rho(i,j)
c*****VELOCITIES
        if(i.gt.1.and.j.gt.1)then
          u(i,j)=vnomt
          u(2,j)=vnomt
          u(i,m1)=0.d0
          u(2,m1)=0.d0
c*****option to use parabolic entrance profile
c      u(i,j)=6.*VNOMT*(-((y(j)**2)/DUCTH**2))+y(j)/DUCTH)
          v(i,j)=0.
        end if
105 CONTINUE
c*****FLOW IN UPPER CUT-OUT
      DO 106 I=isbeg,isend
      DO 106 J=jhubeg,jhuend
        t(i,j)=tintt
        xmdot(i,j)=0.D0
        EBW(i,j)=0.D0
        EGAM(i,j)=1.D0
        EDS(i,j)=edsa
        pv(i,j)=pvtopin
        rhov(i,j)=rhovtop

```

```

        pa(i,j)=patop
        rhoair(i,j)=rhoat
        psat(i,j)=psatit
        rho(i,j)=rhov(i,j)+rhoair(i,j)
        vapmf(i,j)=rhov(i,j)/rho(i,j)
c*****VELOCITIES
        if(i.gt.1.and.j.gt.1)then
            u(i,j)=0.d0
            u(i+1,j)=0.d0
c*****option to use parabolic entrance profile
c        u(i,j)=6.*VNOMT*(-((y(j)**2)/DUCTH**2))+y(j)/DUCTH)
            v(i,j)=0.
            end if
106 CONTINUE

c*****flow in sample section of cell
        DO 103 I=isbeg,isend
        DO 103 J=JSBEG,JSEND
            t(i,j)=tint
            xmdot(i,j)=0.D0
            EDS(i,j)=edss
c*****INITIAL SAMPLE WATER CONTENT
            ebwint=(0.578D0*rf*eds(i,j)*rhods/rhow)
            1*(rhint)*((1.d0/(.321D0+(rhint))))
            1+(1.D0/(1.262D0-(rhint))))
            ebwsat=(0.578D0*rf1*eds(i,j)*rhods/rhow)
            1*(rh100)*((1.d0/(.321D0+(rh100))))
            1+(1.D0/(1.262D0-(rh100))))
            EBW(i,j)=ebwint
            EGAM(i,j)=1.D0-eds(i,j)-ebw(i,j)
            pv(i,j)=pvbotin
            rhov(i,j)=rhovbot
            pa(i,j)=pabot
            rhoair(i,j)=rhoab
            psat(i,j)=psati
            rho(i,j)=rhov(i,j)+rhoair(i,j)
            vapmf(i,j)=rhov(i,j)/rho(i,j)
c*****velocities
            if(i.gt.1.and.j.gt.1)then
                u(i,j)=0.d0
                u(i+1,j)=0.0
c            v(i,j)=-1.d-10
                end if
103 CONTINUE

c*****flow in solid sections
        DO 102 I=1,11
        DO 102 J=1,m1
            if(isolid(i,j).eq.1)then
                t(i,j)=tint
                xmdot(i,j)=0.D0
                EDS(i,j)=edsh
                EBW(i,j)=0.D0

```

```

        EGAM(i,j)=0.D0
        pv(i,j)=0.
        rhov(i,j)=0.
        pa(i,j)=0.
        rhoair(i,j)=rhon20
        psat(i,j)=psati
        rho(i,j)=rhov(i,j)+rhoair(i,j)
        vapmf(i,j)=0.d0
c*****VELOCITIES
        if(i.gt.1.and.j.gt.1)then
            u(i,j)=0.0
            v(i,j)=0.0
        end if
        end if
102 CONTINUE

        RETURN

C*****
C
        ENTRY UNSTDY
C*****FOLD=FNEW
        IF (IBUG) PRINT *, "ENTRY UNSTEADY"
        DO 200 I=1,L1
        DO 200 J=1,M1
        DO 200 NF=1,NFD
        FOLD(I,J,NF)=F(I,J,NF)
200 CONTINUE
        RETURN

C*****
C
        ENTRY DENSE
        IF (IBUG) WRITE (*,*) "ENTRY DENSE"
        CALL THERMO
        CALL SOURCE
        RETURN

C*****
C
        ENTRY BOUND
        IF(IBUG)print *, 'entry bound'

c*****option to use more precise pressures
c*****fix up pressures
        pave=0.0
        do 974 j=2,m2
            pave=pave+p(12,j)*ycv(j)
974 continue
        pave=pave/y1
        do 975 i=1,l1
        do 975 j=1,m1
c            p(i,j)=p(i,j)-pave+patm
975 continue

```

```

c*****concentration equation
c*****top and bottom boundaries of cell
c*****bottom inflow of cell (not really necessary after 1st time)
      do 320 j=2,JHLBEG-1
        rhov(1,j)=rhovbot
        rhoair(1,j)=rhoab
        vapmf(1,j)=rhov(1,j)/(rhov(1,j)+rhoair(1,j))
      320 continue
c*****top inflow of cell (not necessary either)
      do 321 j=JHUEND+1,m2
        rhov(1,j)=rhovtop
        rhoair(1,j)=rhoat
        vapmf(1,j)=rhov(1,j)/(rhov(1,j)+rhoair(1,j))
      321 continue
c*****outflow of cell
      do 322 j=2,m2
        vapmf(11,j)=vapmf(12,j)
        xmfr=vapmf(11,j)/(1.d0-vapmf(11,j))
        rhov(11,j)=(xmfr*patm*xma/(rgas*t(11,j)))
        1/(1.d0+xmfr*(xma/xmw))
        PSAT(11,j)=614.3D0*DEXP(17.06D0*((T(11,j)-273.15D0)
        1/(T(11,j)-40.25D0)))
        pv(11,j)=rhov(11,j)*rgas*t(11,j)/xmw
        pa(11,j)=patm-pv(11,j)
        rhoair(11,j)=pa(11,j)*xma/(rgas*t(11,j))
      322 continue
c*****get impermeable walls at joint between sample and holder
c      do 323 j=JHLBEG,JHUEND
c      do 323 i=1,ISBEG-1
c        rhov(i,j)=rhov(ISBEG,j)
c        rhov(i,j)=1.D-50
c        rhov(i,j)=0.D0
c        vapmf(i,j)=rhov(i,j)
c        pv(i,j)=0.D0
c        rhumida(i,j)=pv(i,j)/psat(i,j)
c 323 continue
c      do 324 j=JHLBEG,JHUEND
c      do 324 i=ISEND+1,ihend
c        rhov(i,j)=rhov(ISEND,j)
c        rhov(i,j)=1.D-50
c        rhov(i,j)=0.D0
c        vapmf(i,j)=rhov(i,j)
c        pv(i,j)=0.D0
c        rhumida(i,j)=pv(i,j)/psat(i,j)
c 324 continue
c*****top and bottom of cell
      do 325 i=1,11
        vapmf(i,1)=vapmf(i,2)
        vapmf(i,m1)=vapmf(i,m2)
        rhov(i,m1)=rhov(i,m2)
        rhoair(i,m1)=rhoair(i,m2)
        rhov(i,1)=rhov(i,2)
        rhoair(i,1)=rhoair(i,2)
      325 continue

```

```

c*****momentum equations
c*****check incoming flows of nitrogen and water vapor
c*****Incoming flows on bottom
c*****flinb is inflow of nitrogen only
c*****flinbe is inflow of water vapor + nitrogen
      flinb=0.0
      flinbe=0.0
      do 300 j=2,jhlbeg-1
        rhot=rhoair(1,j)
        rhote=rhov(1,j)+rhoair(1,j)
        flinbe=flinbe+rhote*u(2,j)*ycv(j)
        flinb=flinb+rhot*u(2,j)*ycv(j)
      300 continue

c*****Incoming flows on top
      flint=0.0
      flinte=0.0
      do 301 j=jhuend+1,m2
        rhot=rhoair(1,j)
        rhote=rhov(1,j)+rhoair(1,j)
        flinte=flinte+rhote*u(2,j)*ycv(j)
        flint=flint+rhot*u(2,j)*ycv(j)
      301 continue

c*****floutt is outflow of nitrogen on top
c*****floutte is outflow of water vapor + nitrogen on top
      floutt=0.0
      floutte=0.0
      do 303 j=jhuend+1,m2
        rhot=rhoair(11,j)
        floutt=floutt+rhot*u(12,j)*ycv(j)
        rhote=rhoair(11,j)+rhov(11,j)
        floutte=floutte+rhote*u(12,j)*ycv(j)
      303 continue

c*****floutb is outflow of nitrogen on bottom
c*****floutbe is outflow of water vapor + nitrogen on bottom
      floutb=0.0
      floutbe=0.0
      do 304 j=2,jhlbeg-1
        rhot=rhoair(11,j)
        floutb=floutb+rhot*u(12,j)*ycv(j)
        rhote=rhoair(11,j)+rhov(11,j)
        floutbe=floutbe+rhote*u(12,j)*ycv(j)
      304 continue

c*****total inflow of nitrogen
      totin=flint+flinb
c*****total inflow of nitrogen and water vapor
      totine=flinte+flinbe
c*****total outflow of nitrogen
      flout=floutt+floutb
c*****total outflow of nitrogen and water vapor
      floute=floutte+floutbe

```

```

c*****flow factors based either on nitrogen only (totin)
c*****or nitrogen + water vapor (totine)
      factor=totin/flout
      factor=totine/floute

c*****desired ratio of upper to lower flow is RATIO
      ratio=pinch
      targtop=(ratio/(1.d0+ratio))*(flinbe+flinte)
      targbot=(flinbe+flinte-targtop)
      facttop=targtop/floutte
      factbot=targbot/floutbe
      do 307 j=jhuend+1,m2
307  u(11,j)=u(12,j)*facttop
      do 308 j=2,jhlbeg-1
308  u(11,j)=u(12,j)*factbot

c*****Option to set velocities in sample holder and/or sample = 0
c      do 310 i=2,12
c      do 310 j=2,m2
c          if(isolid(i,j).eq.1)then
c              u(i,j)=0.d0
c              u(i+1,j)=0.d0
c              v(i,j)=0.d0
c              v(i,j+1)=0.d0
c          end if
c 310 continue

c*****ROUTINE TO PRINT OUT FLOWS AT VARIOUS LOCATIONS TO
c*****CHECK CONTINUITY
      call flwchek1

c*****temperature equation
c*****top and bottom boundaries of cell
c*****bottom inflow of cell
      do 330 j=2,JHLBEG-1
          t(1,j)=tintb
330 continue
c*****top inflow of cell
      do 331 j=JHUEND+1,m2
          t(1,j)=tintt
331 continue
c*****outflow of cell
      do 332 j=2,m2
          t(11,j)=t(12,j)
c*****TEMPERATURE ADDED FOR PLOTTING CONTRAST for mcrogrfx
c*****temperature is usually the peak temperature for sorption
          t(11,jhuend-3)=tplot
332 continue
c*****get adiabatic walls at joint between sample and holder
c      do 334 j=JHLBEG,JHUEND
c          t(11,j)=tint
c 334 continue

```



```

c      do 376 j=JSBEG,JSEND
c      do 376 i=ISEND+1,ihend
c      t(i,j)=t(ISEND,j)
c 376 continue
c*****top and bottom of cell
      do 335 i=1,l1
      t(i,m1)=t(i,m2)
      t(i,1)=t(i,2)
335 continue

      RETURN

C*****
C
      ENTRY OUTPUT
      if(ibug)print *, 'entry output'
      if(ibug)print *,nf
      if(ibug)call print
      RETURN

      ENTRY OUTPUT2
      RETURN

      ENTRY OUTPUT3
      RETURN
C*****
C
      ENTRY GAMSOR
      IF (IBUG) print *, 'entry gamsor'
      if (ibug) print *,nf
      goto(600,610,618,618,620,640)nf
      goto 699

c*****momentum equation nf=1
      600 DO 601 J=1,m1
      DO 601 I=1,L1

c*****CLEAR FLUID REGION
      GAM(I,J)=AMU

c*****IMPERMEABLE REGIONS (very high viscosity)
      if(isolid(i,j).eq.1)then
      gam(i,j)=amus
c      ipin=isend+3
c      jpin=jcent
c      ap(ipin,jpin)=-1.d-40
      end if

c*****SAMPLE REGION (POROUS MATERIAL)
c*****option for very low permeability in plane of fabric
      if(isolid(i,j).eq.2)then
      xkdarc=(1.D-20)*ykdarc
      xkdarc=ykdarc
      if(ykdarc.eq.0.)then

```

```

        xkdarc=1.
        end if

c*****use staggered velocity control volume
        ap(i,j)=ap(i,j)-gam(i,j)/xkdarc
        if(i.eq.isbeg)then
            ap(i,j)=-((x(i)-xu(i))/xcvs(i))*gam(i,j)/xkdarc
        end if
        if(i.eq.isend)then
            ap(i+1,j)=ap(i+1,j)-((xu(i+1)-x(i))/xcvs(i+1))*gam(i,j)/xkdarc
        end if

        if(ykdarc.eq.0.)then
            ap(i,j)=0.
            ap(i,jsend+1)=0.
        end if

        end if

        GAM(L1,J)=0.d0
        rho(i,j)=rhoair(i,j)+rhov(i,j)

601 CONTINUE

c*****momentum equation nf=2
c*****lower half of cell
        610 DO 611 J=1,m1
            DO 611 I=1,L1

c*****CLEAR FLUID REGIONS
            GAM(I,J)=AMU

c*****IMPERMEABLE REGIONS (very high viscosity)
            if(isolid(i,j).eq.1)then
                gam(i,j)=amus
c            ipin=isend+3
c            jpin=jcent
c            ap(ipin,jpin)=-1.d-40
            end if
            rkdarcy(i,j)=0.

c*****SAMPLE REGION (POROUS MATERIAL)
c*****permeability perpendicular to plane of fabric
            if(isolid(i,j).eq.2)then
c*****permeability across fabric
c*****rkdarcy is darcy resistance normalized w.r.t. rddry
                rkdarcy(i,j)=(rddry+(ebw(i,j)/ebwsat)*(rdwet-rddry))/rddry
                rkdarcl=rddry*rkdarcy(i,j)
                ykdarcl=thick/rkdarcl

                if(ykdarc.eq.0.)then
                    ykdarcl=1.
                end if

```

```

c*****use staggered control volumes for velocity source terms
  ap(i,j)=ap(i,j)-gam(i,j)/ykdarcl
  if(j.eq.jsbeg)then
    ap(i,j)=-((y(j)-yv(j))/ycvs(j))*gam(i,j)/ykdarcl
  end if
  if(j.eq.jsend)then
    ap(i,j+1)=ap(i,j+1)-((yv(j+1)-y(j))/ycvs(j+1))*gam(i,j)/ykdarcl
  end if

  if(ykdarc.eq.0.)then
    ap(i,j)=0.
  end if

  end if

  GAM(L1,J)=0.
  rho(i,j)=rhoair(i,j)+rhov(i,j)

611 CONTINUE

  goto 699

c*****source terms for pressure correction and
c*****pressure equations (mass source),nf=3,4
618 DO 619 J=JSBEG,JSEND
  DO 619 I=ISBEG,ISEND
    rho(i,j)=(rhov(i,j)+rhoair(i,j))
    rhoegn=(rhov(i,j)+rhoair(i,j))*egam(i,j)
    rhoego=(fold(i,j,17)+fold(i,j,10))*fold(i,j,9)
    dgascon=(rhoegn-rhoego)/dt
    xsource=xmdot(i,j)-dgascon
    con(i,j)=xsource
    ap(i,j)=0.d0
619 CONTINUE
  goto 699

c*****concentration equation, nf=5
620 CONTINUE
c*****rho and gam for gas flow in cell
  do 623 j=1,m1
    do 623 i=1,l1
      dair=(2.20D-5)*((T(i,j)/273.15D0)**1.75D0)
      rho(i,j)=rhov(i,j)+rhoair(i,j)
      rho1(i,j)=rho(i,j)

c*****CLEAR FLUID REGION
      gam(i,j)=dair
      gam(i,j)=gam(i,j)*rho(i,j)

c*****IMPERMEABLE REGIONS
      if(isolid(i,j).eq.1)then
        gam(i,j)=0.d0
        gam(i,j)=gam(i,j)*rho(i,j)

```

```

        end if

c*****SAMPLE REGIONS (POROUS MATERIAL)
        if(isolid(i,j).eq.2)then
            deff=dair*egam(i,j)/tau
            gam(i,j)=deff
            gam(i,j)=gam(i,j)*rho(i,j)
c*****source terms for sample regions
            rhogam=rho(i,j)
            dvapmfdt=(rhogam*vapmf(i,j)-
1(fold(i,j,17)+fold(i,j,10))*fold(i,j,5))/dt
            xsource=xmdot(i,j)*(1.d0-(rhogam/rhow)*vapmf(i,j))+
1(1.d0-egam(i,j))*dvapmfdt
            if(xsource.lt.0.)then
                con(i,j)=0.D0
                ap(i,j)=xsource/(vapmf(i,j))
            end if
            if(xsource.ge.0.)then
                con(i,j)=xsource
                ap(i,j)=0.D0
            end if
            end if

c*****entrances, exits, and boundaries
            gam(i,1)=0.D0
            gam(i,m1)=0.d0
c            gam(1,j)=0.d0
            gam(11,j)=0.d0

        623 continue

        do 6223 i=1,11
            do 6223 j=1,m1
                dair=(2.20D-5)*((T(i,j)/273.15D0)**1.75D0)
c*****keep track of effective normalized diffusivity
                diffg(i,j)=gam(i,j)/(dair*rho(i,j))
        6223 continue
            go to 699

c*****ENERGY EQUATION, NF=6
        640 CONTINUE
            CALL ERHO
            go to 699

        699 RETURN

C*****
C
        ENTRY SAVE1

        RETURN

C*****
C

```

```

ENTRY OUTTOO

lprint(1)=.TRUE.
lprint(2)=.true.
lprint(3)=.true.
lprint(4)=.true.
LPRINT(5)=.TRUE.
c    LPRINT(6)=.TRUE.
c    LPRINT(7)=.TRUE.
c    LPRINT(8)=.TRUE.
c    LPRINT(9)=.TRUE.
c    LPRINT(10)=.TRUE.
c    LPRINT(11)=.TRUE.
c    LPRINT(12)=.TRUE.
c    LPRINT(13)=.TRUE.
c    LPRINT(14)=.TRUE.
c    LPRINT(15)=.TRUE.
c    LPRINT(16)=.TRUE.
LPRINT(17)=.TRUE.
lprint(18)=.true.
lprint(19)=.true.
lprint(20)=.true.
c    call print

if(iter.eq.50)then
nprint=10
iwrite=nprint
end if

c*****options to go to unsteady calculations
if(iter.ge.50)then
call timestep
end if

if(iwrite.eq.nprint)then

call vaporflx
call flwc hek1
call flwc hek2

if(lsteady)then
else
print *,' '
print *,'iter,smax,ssum'
print *,iter,smax,ssum
end if

iwrite=0
end if

iwrite=iwrite+1

it=int(time)

```

```

        if(it.eq.0)then
        datafile3='0.dat'
c      call mcrogrfx
c      call print
        end if
        if(it.eq.5)then
        datafile3='5.dat'
c      call mcrogrfx
        end if
        if(it.eq.10)then
        datafile3='10.dat'
c      call mcrogrfx
        end if
        if(it.eq.30)then
        datafile3='30.dat'
c      call mcrogrfx
        end if
        if(it.eq.60)then
        datafile3='60.dat'
c      call mcrogrfx
        end if
        if(it.eq.300)then
        datafile3='300.dat'
c      call mcrogrfx
        end if
        if(iter.eq.last)then
        datafile3='last.dat'
c      call mcrogrfx
        end if
        RETURN

C*****
C
        ENTRY SAVEPC
        RETURN

C*****
C
        ENTRY SAVE2

        RETURN
        END

        SUBROUTINE USER2
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
        CHARACTER TITLE*18
        CHARACTER DATAFILE1*60
        CHARACTER DATAFILE2*60
        character datafile3*60
        character datafile4*60
        character datafile5*60
        character header*64

```

```

LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY,IBUG
PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1 AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2 X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4 YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5 R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1 FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
COMMON/CNTL/LSTOP,LSTEADY,LScheme
COMMON/SORC/SMAX,SSUM
COMMON/COEF/FLOW,DIFF,ACOF
COMMON/UNSTEDY/FOLD(ID,JD,NFD),FNEW(ID,JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
COMMON/EXPNT/ALAMBDA(ID,JD),FP(ID,JD)
COMMON/DEBUG/IBUG
COMMON/OTHER/GAM5(ID,JD),RHO5(ID,JD),GAM6(ID,JD),RHO6(ID,JD)
DIMENSION U(ID,JD),V(ID,JD),PC(ID,JD),P(ID,JD)
EQUIVALENCE(F(1,1,1),U(1,1)),(F(1,1,2),V(1,1)),(F(1,1,3),PC(1,1))
EQUIVALENCE(F(1,1,4),P(1,1))
C*****
C
C*****UNSTEADY DIFFUSION PROBLEM*****
COMMON/PROPS/RF,TAU,RHODS,RHOW,TKDS,TKW,TKV,TKA,
1CPDS,CPW,CPV,CPA,PATM,RGAS,XMW,XMA,CHTC,CMTC,dsolid,
1method,epsrhov,iwrite,nprint,aL,aL2,ddry,dwet,dhvap,
1q1,header,tkss,cpss,rhoss,edss,edsa,edsh,amu,amus,xmn2,
1ykdar,ckdar,deltap,pinch,facttop,factbot,rddry,rdwet,
1thick,rfl
common/datafile/datafile1,datafile2,datafile3,datafile4,
1datafile5
COMMON/BNDRY/PSATB,RHBND,TBND,PVBND,RHOVBND,PABND,RHOABND
1,psatbb,psatbt,rhbndb,rhbndt,tbndb,tint,ebwint,ebwsat,
1tbndt,pvbndb,pvbndt,rhovbndb,rhovbndt,pabndb,pabndt,
1rhoabndb,rhoabndt,weighti,weight,rhint,tintt,tintb
COMMON/DUCT/RHOA,DUCTH,DUCTA,DUCTW,RHON2,VNOM,
1dn2,deff,rhovtop,rhovbot,rhoat,rhoab,isbeg,isend,
1jsbeg,jsend,jhlbeg,jhlend,jhubeg,jhuend,vnomt,vnomb,
1xmassti,xmassbi,xmassto,xmassbo,xtotin,xtotout,xmasssi,
1xmassso,xsample,ysample,xqn2mks,dcssi,dctopi,dcboti,
1dcboto,dctopo,xqst,xqsb,icent,jcent,convcte,ihend
1,ltype,ivalvbeg,ivalvend,jvalvbeg,jvalvend
common/vapor/xmvt,xmvtb,xmn2t,xmn2b,xmn2out,xmvout,rhbo
common/solid/kplot(nfp3),isolid(id,jd),tplot
DIMENSION T(ID,JD),RHOV(ID,JD),EBW(ID,JD),XMDOT(ID,JD),
1EGAM(ID,JD),RHOAIR(ID,JD),PV(ID,JD),PA(ID,JD),PSAT(ID,JD),
1EDS(ID,JD),TKEFF(ID,JD),RHUMIDA(ID,JD),vapmf(id,jd)
1,xsolid(id,jd),diffg(id,jd),rkdarcy(id,jd)
EQUIVALENCE(F(1,1,5),vapmf(1,1)),(F(1,1,6),T(1,1)),

```

```

1(F(1,1,7),EBW(1,1)),(F(1,1,8),XMDOT(1,1)),
1(F(1,1,9),EGAM(1,1)),(F(1,1,10),RHOAIR(1,1)),
1(F(1,1,11),PV(1,1)),(F(1,1,12),PA(1,1)),
1(F(1,1,13),PSAT(1,1)),(F(1,1,14),EDS(1,1)),(F(1,1,15),
1TKEFF(1,1)),(F(1,1,16),RHUMIDA(1,1)),(f(1,1,17),rhov(1,1))
1,(F(1,1,18),xsolid(1,1)),(f(1,1,19),diffg(1,1)),(f(1,1,20),
1rkdarcy(1,1))

entry property
if(ibus)print *,'property'
c*****PROBLEM CONSTANTS AND MATERIAL PROPERTIES
open(unit=10,file='fabric.dat')
c*****thickness of one layer (m)
read(10,532)thick1
c*****thickness of two layers (m)
thick=thick1
c*****Darcy flow resistances for one layer (m^-1)
read(10,532)rddry
read(10,532)rdwet
c*****Darcy flow resistances for two layers (m^-1)
rddry=rddry
rdwet=rdwet
ykdarc=thick/rddry
c*****volume fraction of dry solid
read(10,532)edss
c*****solid phase diffusion coefficient of water (m^2/s)
read(10,532)dsolid
c*****effective fiber diameter (m)
read(10,532)aL
c*****mass fraction of water to dry solid at 65% r.h.
read(10,532)rf1
c*****tortuosity
read(10,532)tau
c*****dry polymer solid density (kg/m^3)
read(10,532)rhods
c*****thermal conductivity (J/(s-m-K))
read(10,532)tkds
c*****heat capacity of solid polymer (J/(kg-K))
read(10,532)cpds
c*****temperature added for contrast on Micrografix plots
read(10,532)tplot
close(10)
531 format(i10)
532 format(e15.5)
rf=0.0
c*****solid volume fraction of clear gas
edsa=0.D0
c*****solid volume fraction of the sample holder
edsh=1.0D0
c*****density of liquid water (kg/m^3)
rhov=1000.D0
c*****thermal conductivities of water, vapor, air (J/(s-m-K))
tkw=0.6D0
tkv=0.0246D0

```



```

      tka=0.02563D0
c*****thermal properties of sample holder (stainless steel)
      tkss=14.4
      cpss=461.
      rhoss=7817.
c*****heat capacities of liquid water, vapor, air (J/(kg-K))
      cpw=4182.D0
      cpv=1862.D0
      cpa=1003.D0
c*****nitrogen viscosity (N-s/m^2)
      AMU=1.784D-5
      AMUS=1.D20
c*****TOTAL GAS PRESSURE (ATMOSPHERIC) (Pa)
      PATM=101325.D0
c*****UNIVERSAL GAS CONSTANT (N-m)/(kg-K)
      RGAS=8314.5
c*****MOLECULAR WEIGHTS (kg/kgmole)
      XMW=18.015
      XMA=28.97
      XMN2=28.013
      RETURN

      entry THERMO
      if(ibug)print *, 'entry thermo'
      do 1021 j=1,m1
      do 1021 i=1,l1
      if(isolid(i,j).ne.1)then
c*****new method (assume atmospheric pressure)
      PSAT(i,j)=614.3D0*DEXP(17.06D0*((T(i,j)-273.15D0)
1/(T(i,j)-40.25D0)))
      sphumid=vapmf(i,j)/(1.d0-vapmf(i,j))
      rhov(i,j)=xma*patm*sphumid/
1(rgas*t(i,j)*(1.+(xma/xmw)*sphumid))
      pv(i,j)=rhov(i,j)*rgas*t(i,j)/xmw
      pa(i,j)=patm-pv(i,j)
      rhoair(i,j)=pa(i,j)*xma/(rgas*t(i,j))
      rho(i,j)=rhov(i,j)+rhoair(i,j)
      rhumid=pv(i,j)/psat(i,j)
      rhumida(i,j)=rhumid
      vapcalc=rhov(i,j)/(rhov(i,j)+rhoair(i,j))
      errx=dabs(vapcalc-vapmf(i,j))
c*****alternate method
c      wbar=1./(vapmf(i,j)/xmw+(1.-vapmf(i,j))/xma)
c      rho(i,j)=patm*wbar/(rgas*t(i,j))
c      rhov(i,j)=rho(i,j)*vapmf(i,j)
c      rhoair(i,j)=rho(i,j)*(1.-vapmf(i,j))
c      pv(i,j)=rhov(i,j)*rgas*t(i,j)/xmw
c      pa(i,j)=patm-pv(i,j)
c      rhumid=pv(i,j)/psat(i,j)
c      rhumida(i,j)=rhumid
      end if

1021 continue
      RETURN

```

```

      entry SOURCE
      if(ibug)print *, 'entry source'
      do 1120 j=1,m1
      do 1120 i=1,l1
      xmdot(i,j)=0.d0
1120 continue
      do 1122 j=JSBEG,JSEND
      do 1122 i=ISBEG,ISEND
      rhumid=pv(i,j)/(psat(i,j)+(1.D-10)*psat(i,j))
      ebw(i,j)=(0.578D0*rf*eds(i,j)*rhods/rhow)
      1*(rhumid)*((1.d0/(.321D0+(rhumid)))
      1+(1.D0/(1.262D0-(rhumid))))
      xmdot(i,j)=(fold(i,j,7)-ebw(i,j))*rhow/dt
      req=ebw(i,j)*eds(i,j)*rhods/rhow
      rinst=fold(i,j,7)*eds(i,j)*rhods/rhow
      xmdot(i,j)=dsolid*rhods*(rinst-req)/(aL*aL)
      ebw(i,j)=fold(i,j,7)-xmdot(i,j)*dt/rhow
      egam(i,j)=1.D0-eds(i,j)-ebw(i,j)
1122 continue
      RETURN

      entry ERHO
      if(ibug)print *, 'entry erho'
      weight=0.0
      do 631 j=1,m1
      do 631 i=1,l1
c*****density and heat capacity for energy equation
      rhoavg=rhow*ebw(i,j)+rhods*eds(i,j)+
      1(rhov(i,j)+rhoair(i,j))*egam(i,j)
      cpavg=(rhow*fold(i,j,7)*cpw+rhods*fold(i,j,14)*cpds+
      1(fold(i,j,17)*cpv+fold(i,j,10)*cpa)*fold(i,j,9))/
      1(rhoavg+1.D-20)
      rhogam=rhov(i,j)+rhoair(i,j)
      cpgam=(rhov(i,j)*cpv+rhoair(i,j)*cpa)/rhogam
      rho(i,j)=rhogam*cpgam
      rho1(i,j)=rhoavg*cpavg
      if(isolid(i,j).eq.1)then
      rho(i,j)=cpss*rhoss
      rho1(i,j)=cpss*rhoss
      end if
c*****diffusion coefficients
      tkf=((tkw*rhow*ebw(i,j)+tkds*rhods*eds(i,j))
      1/(rhow*ebw(i,j)+rhods*eds(i,j)+1.D-50))
      tkg=((tkv*rhov(i,j)+tkair*rhoair(i,j)))/
      1(rhov(i,j)+rhoair(i,j)+1.D-20))
      vf=eds(i,j)+ebw(i,j)
      tkperp=tkf*tkg/(vf*tkg+(1.D0-vf)*tkf+1.D-50)
      tkprll=(1.d0-VF)*tkg+vf*tkf
      womega=1.d0
      gam(i,j)=womega*tkprll+(1.d0-womega)*tkperp
      if(isolid(i,j).eq.1)then
      gam(i,j)=tkss
      end if

```

```

c*****entrances, exits, and boundaries
      gam(i,1)=0.D0
      gam(i,m1)=0.d0
c      gam(1,j)=0.d0
      gam(l1,j)=0.d0
      tkeff(i,j)=gam(i,j)
c*****source terms for energy equation
      dhvap=(2.792d6)-160.D0*t(i,j)-3.43d0*t(i,j)*t(i,j)
      ql=(1.95D5)*(1.d0-pv(i,j)/(psat(i,j)+1.D-100))
      1*((1.d0/(0.2d0+(pv(i,j)/(psat(i,j)+1.D-100))))+
      1(1.d0/(1.05d0-(pv(i,j)/(psat(i,j)+1.D-100))))))
      xsource=-(dhvap+ql)*xmdot(i,j)
      if(isolid(i,j).eq.1)then
      xsource=0.d0
      end if
      xsourcep=(rhoavg*cpavg-cpgam*rhogam)*fold(i,j,6)/dt
      xsourcem=(rhoavg*cpavg-cpgam*rhogam)*t(i,j)/dt
      if(xsource.lt.0.)then
      con(i,j)=0.
      ap(i,j)=xsource/t(i,j)
      end if
      if(xsource.ge.0.)then
      con(i,j)=xsource
      ap(i,j)=0.
      end if
631 continue
      RETURN

      entry GRIDGEN
      if(ibus)print *, 'entry gridgen'
c      print *, 'input ratio of exiting upper flow to entering upper flow'
c      read (*,*)pinch
c      pinch=1.0147
      pinch=0.1
      MODE=1
C*****Number of control volumes in each x region
c*****entrance region
      NXVOLS1=2
c*****approach region
      NXVOLS2=4
c*****edge before sample
      NXVOLS3=4
c*****edge right after sample edge (1st quarter of sample)
      NXVOLS4=2
c*****middle region of sample
      NXVOLS5=5
c*****back 1/4 of sample
      NXVOLS6=2
c*****first edge of back holder
      NXVOLS7=4
c*****middle region of holder
      NXVOLS8=4
c*****valve region
      NXVOLS9=2

```

```

c*****mixing region
      NXVOLS10=4
c*****Number of x regions
      NXREG=10
c*****Number of control volumes in each y region
      NYVOLS1=4
c*****center of duct region
      NYVOLS2=4
      NYVOLS3=4
      NYVOLS4=4
c*****sample y region
      NYVOLS5=7
      NYVOLS6=4
      NYVOLS7=4
      NYVOLS8=4
      NYVOLS9=4
c*****Number of y regions
      NYREG=9
c*****SAMPLE DIMENSIONS
      XSAMPLE=4.D-2
      YSAMPLE=thick
c*****DISTANCE ALONG DUCT FOR FIRST GRID (m)
      XDUCT1=1.D-2
c*****DISTANCE FROM DUCT ENTRANCE TO SAMPLE EDGE (m)
      XDUCT2=4.5D-2
c*****DISTANCE FROM SAMPLE BACK EDGE TO HOLDER EDGE
      XDUCT3=4.5D-2
c*****DISTANCE FROM HOLDER BACK EDGE TO DUCT EXIT
      XDUCT4=4.0D-2
c*****TOTAL DUCT LENGTH (m)
      XL=XDUCT2+XDUCT3+XDUCT4+XSAMPLE
c*****DUCT HEIGHT (m)
      DUCTH=2.5D-3
c*****valve region
      xvalve=ducth
      xvalve=.1*xduct3
c*****SAMPLE HOLDER THICKNESS (m)
      YHOLDER=4.572D-4
c*****TOTAL THICKNESS in Y DIRECTION (m)
      YL=2.*DUCTH+YSAMPLE+2.*YHOLDER
      NXVOLS=NXVOLS1+NXVOLS2+NXVOLS3+NXVOLS4+NXVOLS5+NXVOLS6
      1+NXVOLS7+NXVOLS8+NXVOLS9+NXVOLS10
      NYVOLS=NYVOLS1+NYVOLS2+NYVOLS3+NYVOLS4+NYVOLS5+NYVOLS6
      1+NYVOLS7+NYVOLS8+NYVOLS9
      L1=NXVOLS+2
      M1=NYVOLS+2
c*****use uniform grids for each section
      XU(2)=0.D0
      NXBEG=3
      NXEND=NXVOLS1+2
      XL1=XDUCT1
      DX=XL1/DFLOAT(NXVOLS1)
      DO 1 I=NXBEG,NXEND
      XU(I)=XU(I-1)+DX

```

```

1 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS2
  XL2=XDUCT2-(XSAMPLE/4.D0)
  DX=(XL2-XL1)/DFLOAT(NXVOLS2)
  DO 2 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
2 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS3
  XL3=XDUCT2
  DX=(XL3-XL2)/DFLOAT(NXVOLS3)
  DO 3 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
3 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS4
C*****SAMPLE BEGINS AT NODE ISBEG
  ISBEG=NXBEG-1
  XL4=XDUCT2+(XSAMPLE/4.D0)
  DX=(XL4-XL3)/DFLOAT(NXVOLS4)
  DO 4 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
4 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS5
  XL5=(XDUCT2+XSAMPLE)-(XSAMPLE/4.D0)
  DX=(XL5-XL4)/DFLOAT(NXVOLS5)
  DO 5 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
5 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS6
C*****SAMPLE ENDS AT NODE ISEND
  ISEND=NXEND-1
  XL6=(XDUCT2)+XSAMPLE
  DX=(XL6-XL5)/DFLOAT(NXVOLS6)
  DO 6 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
6 CONTINUE
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS7
  XL7=(XDUCT2)+XSAMPLE+(XSAMPLE/4.D0)
  DX=(XL7-XL6)/DFLOAT(NXVOLS7)
  DO 7 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX
7 CONTINUE
C*****MIDDLE OF SAMPLE HOLDER
  NXBEG=NXEND+1
  NXEND=NXEND+NXVOLS8
  XL8=XDUCT2+XSAMPLE+XDUCT3-xvalve
  DX=(XL8-XL7)/DFLOAT(NXVOLS8)
  DO 8 I=NXBEG,NXEND
    XU(I)=XU(I-1)+DX

```

```

      8 CONTINUE
C*****VALVE REGION
c*****the valve is not used any more, but the grids
c*****are retained
      NXBEG=NXEND+1
      NXEND=NXEND+NXVOLS9
C*****valve begins at node IVALVBEG
C*****valve ends at node IVALVEND
      ivalvbeg=nxbeg-1
      ivalvend=nxend-1
C*****SAMPLE HOLDER ENDS AT IHEND
      ihend=ivalvend
      XL9=XDUCT2+XSAMPLE+XDUCT3
      DX=(XL9-XL8)/DFLOAT(NXVOLS9)
      DO 88 I=NXBEG,NXEND
        XU(I)=XU(I-1)+DX
88 CONTINUE
      NXBEG=NXEND+1
      NXEND=NXEND+NXVOLS10
      XL10=XL
      DX=(XL10-XL9)/DFLOAT(NXVOLS10)
      DO 89 I=NXBEG,NXEND
        XU(I)=XU(I-1)+DX
89 CONTINUE

      ductc=.5*ducth
      YV(2)=0.D0
      NYBEG=3
      NYEND=NYVOLS1+2
      YL1=(ducth-ductc)/2.d0
      dy=YL1/DFLOAT(NYVOLS1)
      DO 9 J=NYBEG,NYEND
        yv(j)=yv(j-1)+dy
9 CONTINUE
      NYBEG=NYEND+1
      NYEND=NYEND+NYVOLS2
      YL2=ductc+(ducth-ductc)/2.d0
      dy=(YL2-YL1)/DFLOAT(NYVOLS2)
      DO 10 J=NYBEG,NYEND
        yv(j)=yv(j-1)+dy
10 CONTINUE
      NYBEG=NYEND+1
      NYEND=NYEND+NYVOLS3
      YL3=ducth
      dy=(YL3-YL2)/DFLOAT(NYVOLS3)
      DO 11 J=NYBEG,NYEND
        yv(j)=yv(j-1)+dy
11 CONTINUE
      NYBEG=NYEND+1
      NYEND=NYEND+NYVOLS4
C*****LOWER SAMPLE HOLDER BEGINS AT NODE JHLBEG
      JHLBEG=NYBEG-1
C*****LOWER SAMPLE HOLDER ENDS AT NODE JHLEND
      JHLEND=NYEND-1

```

```

        YL4=ducth+YHOLDER
        dy=(YL4-YL3)/DFLOAT(NYVOLS4)
        DO 12 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
12     CONTINUE
        NYBEG=NYEND+1
        NYEND=NYEND+NYVOLS5
C*****SAMPLE BEGINS AT NODE JSBEG
        JSBEG=NYBEG-1
C*****SAMPLE ENDS AT NODE JSEND
        JSEND=NYEND-1
        YL5=ducth+YHOLDER+ysample
        dy=(YL5-YL4)/DFLOAT(NYVOLS5)
        DO 13 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
13     CONTINUE
        NYBEG=NYEND+1
        NYEND=NYEND+NYVOLS6
C*****UPPER SAMPLE HOLDER BEGINS AT NODE JHUBEG
        JHUBEG=NYBEG-1
C*****UPPER SAMPLE HOLDER ENDS AT NODE JHUEND
        JHUEND=NYEND-1
        YL6=ducth+ysample+2.*YHOLDER
        dy=(YL6-YL5)/DFLOAT(NYVOLS6)
        DO 14 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
14     CONTINUE
        NYBEG=NYEND+1
        NYEND=NYEND+NYVOLS7
c*****lower valve body begins at jhuend+1
c*****lower valve body ends at jvalvbeg
        jvalvbeg=nyend-1
        YL7=ducth+ysample+2.*YHOLDER+(ducth-ductc)/2.d0
        dy=(YL7-YL6)/DFLOAT(NYVOLS7)
        DO 15 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
15     CONTINUE
        NYBEG=NYEND+1
        NYEND=NYEND+NYVOLS8
        YL8=2.*ducth+ysample+2.*YHOLDER-(ducth-ductc)/2.d0
        dy=(YL8-YL7)/DFLOAT(NYVOLS8)
        DO 16 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
16     CONTINUE
        NYBEG=NYEND+1
        NYEND=NYEND+NYVOLS9
c*****upper valve body begins at jvalvend
        jvalvend=nybeg-1
        YL9=YL
        dy=(YL9-YL8)/DFLOAT(NYVOLS9)
        DO 17 J=NYBEG,NYEND
            yv(j)=yv(j-1)+dy
17     CONTINUE

```

```

c      print *, 'l1,m1'
c      print *, l1,m1
c      print *, 'isbeg,isend,jsbeg,jsend,ihend'
c      print *, isbeg,isend,jsbeg,jsend,ihend
c      print *, 'jhlbeg,jhlend,jhubeg,jhuend'
c      print *, jhlbeg,jhlend,jhubeg,jhuend
c      print *, 'ivalvbeg,ivalvend,jvalvbeg,jvalvend'
c      print *, ivalvbeg,ivalvend,jvalvbeg,jvalvend
c      icent=isbeg+(isend-isbeg+1)/2
c      jcent=jsbeg+(jsend-jsbeg+1)/2
c      print *, 'icent,jcent'
c      print *, icent,jcent

c*****define the various regions of the domain in terms of isolid
c*****1=solid, 0=clear fluid, 2=sample , 3=valve region
      do 40 i=1,l1
      do 40 j=1,m1
      isolid(i,j)=0
      i1=isbeg-1
      if(i.le.i1.and.j.ge.jhlbeg.and.j.le.jhuend)then
      isolid(i,j)=1
      end if
      i2=isend+1
      if(i.ge.i2.and.j.ge.jhlbeg.and.j.le.jhuend)then
      isolid(i,j)=1
      end if

      if(i.ge.isbeg.and.i.le.isend.and.j.ge.jsbeg.and.j.le.
1jsend)then
      isolid(i,j)=2
      end if

      xsolid(i,j)=1.*isolid(i,j)

40 continue

      RETURN

C*****

30 return

      entry flwchek1
c*****check mass flows coming in on top and bottom
c*****bottom coming in
      xmassbi=0.0
      dcboti=0.0
      do 326 j=2,jhlbeg-1
      rhot=rhov(1,j)+rhoair(1,j)
      xmassbi=xmassbi+rhot*u(2,j)*ycv(j)
      dcboti=dcboti+rhov(1,j)*ycv(j)
326 continue
      dcboti=dcboti/ducth
c*****top coming in

```



```

      xmassti=0.0
      dctopi=0.0
      do 327 j=jhuend+1,m2
        rhot=rhov(1,j)+rhoair(1,j)
        xmassti=xmassti+rhot*u(2,j)*ycv(j)
        dctopi=dctopi+rhov(1,j)*ycv(j)
327    continue
      dctopi=dctopi/ducth
c*****incoming in sample holder region (should be 0)
      xmasssi=0.0
      dcssi=0.0
      do 341 j=jhlbeg,jhuend
        rhot=rhov(1,j)+rhoair(1,j)
        xmasssi=xmasssi+rhot*u(2,j)*ycv(j)
        dcssi=dcssi+rhov(1,j)*ycv(j)
341    continue
c*****total mass entering at left hand side entrance
      xtotin=xmassbi+xmassti+xmasssi

c*****check mass flow crossing the sample due to convection
c*****convcte is air plus vapor
c*****convct is air (nitrogen) only
      convct=0.d0
      convcte=0.d0
      jcent=jsbeg+(jsend-jsbeg+1)/2
      do 351 i=isbeg,isend
        jcent2=jcent+1
        rhot=rhoair(i,jcent)
        convct=convct+rhot*v(i,jcent2)*xcv(i)
        rhote=rhov(i,jcent)+rhoair(i,jcent)
        convcte=convcte+rhote*v(i,jcent2)*xcv(i)
351    continue

c*****flow in top exit portion
      floutt=0.d0
      do 3001 j=jhuend+1,m2
        lmid=ihend-3
        rhot=rhoair(lmid,j)
        floutt=floutt+rhot*u(lmid+1,j)*ycv(j)
3001    continue
c*****flow in middle of bottom exit portion
      floutb=0.d0
      do 3002 j=2,jhlbeg-1
        rhot=rhoair(lmid,j)
        floutb=floutb+rhot*u(lmid+1,j)*ycv(j)
3002    continue

c*****check mass flows going out on top and bottom
      lmid=ihend-3
      lm3=lmid+1
      xmassso=0.0
      dcso=0.0
      do 340 j=jhlbeg,jhuend
        rhot=rhov(lm3,j)+rhoair(lm3,j)

```

```

        xmassso=xmassso+rhot*u(lmid,j)*ycv(j)
        dcsso=dcsso+rhov(lm3,j)
340 continue
        xmassbo=0.0
        dcboto=0.0
        vapmidt=0.0
        do 328 j=2,jhlbeg-1
            rhot=rhov(lm3,j)+rhoair(lm3,j)
            xmassbo=xmassbo+rhot*u(lmid,j)*ycv(j)
            vapmidt=vapmidt+rhov(lm3,j)*u(lmid,j)*ycv(j)
            dcboto=dcboto+rhov(lm3,j)*ycv(j)
328 continue
        dcboto=dcboto/ducth
        xmassto=0.0
        dctopo=0.0
        vapmidb=0.
        do 329 j=jhuend+1,m2
            rhot=rhov(lm3,j)+rhoair(lm3,j)
            xmassto=xmassto+rhot*u(lmid,j)*ycv(j)
            vapmidb=vapmidb+rhov(lm3,j)*u(lmid,j)*ycv(j)
            dctopo=dctopo+rhov(lm3,j)*ycv(j)
329 continue
        dctopo=dctopo/ducth

        xtout=xmassbo+xmassto+xmassso

        return

    entry vaporflx
c*****calculations after flow has passed over sample
c*****keep track of vapor and nitrogen concentrations in flow
        rhbi=0.
        do 3451 j=2,jhlbeg-1
            rhbi=rhbi+rhumida(1,j)*ycv(j)
3451 continue
        rhbi=rhbi/ducth
        rhti=0.
        do 3452 j=jhuend+1,m2
            rhti=rhti+rhumida(1,j)*ycv(j)
3452 continue
        rhti=rhti/ducth
        lmid=ihend-3
        lm3=lmid+1
        dcboto=0.0
        dcn2bo=0.0
        rhbo=0.
        vapmidt=0.0
        do 3281 j=2,jhlbeg-1
            vapmidt=vapmidt+rhov(lm3,j)*u(lmid,j)*ycv(j)
            dcboto=dcboto+rhov(lm3,j)*ycv(j)
            rhbo=rhbo+rhumida(lm3,j)*ycv(j)
            dcn2bo=dcn2bo+rhoair(lm3,j)*ycv(j)
3281 continue
        dcboto=dcboto/ducth

```

```

        dcn2bo=dcn2bo/ducth
        rhbo=rhbo/ducth
        dctopo=0.0
        dcn2to=0.
        rhto=0.
        vapmidb=0.
        do 3291 j=jhuend+1,m2
        vapmidb=vapmidb+rhov(lm3,j)*u(lmid,j)*ycv(j)
        dctopo=dctopo+rhov(lm3,j)*ycv(j)
        dcn2to=dcn2to+rhoair(lm3,j)*ycv(j)
        rhto=rhto+rhumida(lm3,j)*ycv(j)
3291 continue
        dctopo=dctopo/ducth
        dcn2to=dcn2to/ducth
        rhto=rhto/ducth

c*****Flux and Resistance Calculations
c*****concentration differences (fluxbot)
        fluxtop=dabs(dctopi-dctopo)*xqst
        1*tint/(273.15*xsample)
        fluxbot=dabs(dcboti-dcboto)*xqsb
        1*tint/(273.15*xsample)
        PSATN=614.3D0*DEXP(17.06D0*((tint-273.15D0)
        1/(tint-40.25D0)))
c*****relative humidity differences (fluxbot1a)
        fluxtop1a=(rhti-rhto)*xqst*psatn*xmw/(rgas*273.15*xsample)
        fluxbot1a=(rhbi-rhbo)*xqsb*psatn*xmw/(rgas*273.15*xsample)
c*****alternate concentration differences (fluxbot2)
        RHON20=patm*xma/(rgas*273.15)
        fluxtop1=rhon20*xqst*(dctopi/rhoat-dctopo/dcn2to)/xsample
        fluxbot1=rhon20*xqsb*(dcboti/rhoab-dcboto/dcn2bo)/xsample
c*****mass balance (fluxvbt)
c*****Computer mass balance (vapor in minus vapor out on each side)
        fluxvtp=xmvt-vapmidt
        fluxvbt=xmvt-vapmidb
        fluxtop2=fluxvtp/xsample
        fluxbot2=fluxvbt/xsample
c*****concentration differences
        xmdca=dabs(dctopi-dcboti)
        xmdcb=dabs(dctopo-dcboto)
        xlgdct=dabs(xmdca-xmdcb)/(dlog(xmdca/xmdcb))
c*****original DMPC flux
        rtop=xlgdct/(fluxtop+1.d-20)
        rbot=xlgdct/(fluxbot+1.d-20)
        rtopr=xmdca/(fluxtop+1.d-20)
        rbotr=xmdca/(fluxbot+1.D-20)
c*****original DMPC flux using delta r.h.
        rtop1a=xlgdct/(fluxtop1a+1.d-20)
        rbot1a=xlgdct/(fluxbot1a+1.d-20)
        rtopr1a=xmdca/(fluxtop1a+1.d-20)
        rbotr1a=xmdca/(fluxbot1a+1.D-20)
c*****improved DMPC flux
        rtop1=xlgdct/(fluxtop1+1.d-20)
        rbot1=xlgdct/(fluxbot1+1.d-20)

```

```

      rtopr1=xmdca/(fluxtop1+1.d-20)
      rbotr1=xmdca/(fluxbot1+1.D-20)
c*****computer calculated flux based on mass balance for water vapor
      rtop2=xlgdct/(fluxtop2+1.d-20)
      rbot2=xlgdct/(fluxbot2+1.d-20)
      rtopr2=xmdca/(fluxtop2+1.d-20)
      rbotr2=xmdca/(fluxbot2+1.D-20)

c*****pressure drop across sample
      dpsamp=p(icent,m2)-p(icent,2)

c*****vapor fluxes another way
      xmn2out=0.
      xmvout=0.
      do 8110 j=2,m2
        xmn2out=xmn2out+rhoair(11,j)*u(12,j)*ycv(j)
        xmvout=xmvout+rhov(11,j)*u(12,j)*ycv(j)
8110 continue

      vaporin=xmvt+xmvtb
      avapout=vapmidt+vapmidb
      vapdiff=dabs(vaporin-xmvout)/vaporin
      xmn2in=xmn2t+xmn2b
      errn2=dabs(xmn2in-xmn2out)/xmn2in

c*****Print out values every nprint iteration for steady-state
      if(lsteady)then
        open(unit=10,access='append',file=datafile2)
        write(10,8197)time,rbot1,dpsamp,fluxbot,fluxbot1
        close(10)
8197 format(5e13.4)
      else
        print *, 'mass fluxes, top and bottom, original DPMC'
        write(*,3334)fluxtop,fluxbot
        print *, 'fluxes, top and bottom, original based on r.h.'
        write(*,3334)fluxtop1a,fluxbot1a
        print *, 'Improved DMPC calculation for fluxes'
        write(*,3334)fluxtop2,fluxbot2
        print *, 'fluxtop,fluxbot computer mass flow balance'
        write(*,3334)fluxvtp,fluxvbt
        print *, 'xmvt,xmvtb,vapmidt,vapmidb'
        write(*,3336)xmvt,xmvtb,vapmidt,vapmidb
3334 format(2e15.4)
3335 format(3e15.4)
3336 format(4e15.4)
3337 format(5e15.4)
        print *, ' '
        print *, 'Original DMPC, Improved DMPC, computer mass balance'
        print *, 'top lgm resistance, bottom lgm resistance'
        write(*,3334)rtop,rbot
        write(*,3334)rtop1,rbot1
        write(*,3334)rtop2,rbot2
        print *, 'top resistance, bottom resistance'
        write(*,3334)rtopr,rbotr

```

```

write(*,3334)rtopr1,rbotr1
write(*,3334)rtopr2,rbotr2
print *, ' '
print *,'total water vapor in xmv+xmvt,total out,'
print *,'total out after sample total,and error (percent out-in)'
write (*,3336)vaporin,xmvout,avapout,vapdiff
print *,'total nitrogen in xmn2t+xmn2b,total out,error n2'
write(*,3335)xmn2in,xmn2out,errn2
print *,'humidity,vapor density, temperature in sample'
write(*,3335)rhumida(icent,jcent),rhov(icent,jcent),t(icent,jcent)
print *,'humidity outlets, top and bottom'
write(*,3334)rhto,rhbo
end if

return

entry flwchek2
c*****check mass flows coming in on top and bottom
xm1t=0.0
lst1=1
lst2=lst1+1
do 3126 j=2,jhlbeg-1
xm1t=xm1t+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3126 continue
xm2t=0.0
lst1=isbeg-3
lst2=lst1+1
do 3127 j=2,jhlbeg-1
xm2t=xm2t+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3127 continue
xm3t=0.0
lst1=isend+3
lst2=lst1+1
do 3128 j=2,jhlbeg-1
xm3t=xm3t+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3128 continue
xm4t=0.0
lst1=ihend-3
lst2=lst1+1
do 3129 j=2,jhlbeg-1
xm4t=xm4t+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3129 continue
xm5t=0.0
do 3130 j=2,jhlbeg-1
xm5t=xm5t+(rhoair(l1,j)+rhov(l1,j))*u(l2,j)*ycv(j)
3130 continue
xm1tt=0.0
lst1=1
lst2=lst1+1
do 3131 j=jhuend+1,m2
xm1tt=xm1tt+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3131 continue
xm2tt=0.0
lst1=isbeg-3

```

```

        lst2=lst1+1
        do 3132 j=jhuend+1,m2
            xm2tt=xm2tt+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3132 continue
            xm3tt=0.0
            lst1=isend+3
            lst2=lst1+1
            do 3133 j=jhuend+1,m2
                xm3tt=xm3tt+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3133 continue
                xm4tt=0.0
                lst1=ihend-3
                lst2=lst1+1
                do 3134 j=jhuend+1,m2
                    xm4tt=xm4tt+(rhoair(lst1,j)+rhov(lst1,j))*u(lst2,j)*ycv(j)
3134 continue
                    xm5tt=0.0
                    do 3135 j=jhuend+1,m2
                        xm5tt=xm5tt+(rhoair(l1,j)+rhov(l1,j))*u(l2,j)*ycv(j)
3135 continue
3333 format(5e15.6)
            xm1ttt=xm1t+xm1tt
            xm2ttt=xm2t+xm2tt
            xm3ttt=xm3t+xm3tt
            xm4ttt=xm4t+xm4tt
            xm5ttt=(xm5t+xm5tt)

            convct1=xmassto-xmassti
            convct2=xmassbo-xmassbi
            flowerr=dabs((convcte-convct1)/convct1)

c*****calculate relative humidity at bottom exiting stream
            rhbo=0.
            lm31=l2-1
            do 3812 j=2,jhlbeg-1
                rhbo=rhbo+rhumida(lm31,j)*ycv(j)
3812 continue
            rhbo=rhbo/ducth

c*****calculate pressure difference across sample
            dpsamp=p(icent,m2)-p(icent,2)
            dpsamp2=dpsamp*1.45d-4
            dpsamp3=dpsamp*13.6*2.953d-4
            dpsamp=dabs(dpsamp)

c*****calculate apparent convective flow properties
            dpsample=p(icent,jsend+3)-p(icent,jsbeg-3)
            ykcalc=v(icent,jcent)*amu*ysample/dpsample
c*****nondimensional flow resistances
c*****rcalc is from equation, rcalc2 is from actual flows
            rcalc2=rddry*thick/ykcalc
            rcalc=rddry*rkdarcy(icent,jcent)
            dpsamp1=p(icent,jcent)-p(icent,jcent-1)

```

```

      ykcalcl=v(icent,jcent)*amu*ydif(jcent)/dpsamp1

c*****record temperatures at different locations in sample
      jsl=jsbeg-1
      jsen=jsend+1
      t1cnt=t(isbeg+2,jcent)-tint
      t2cnt=t(icent,jcent)-tint
      t3cnt=t(isend-2,jcent)-tint

      if(lsteady)then
        open(unit=10,access='append',file=datafile1)
        write(10,898)time,dpsamp,rhbo,ykcalc,rcalc2,pinch
        write(*,898)time,dpsamp,rhbo,ykcalc,rcalc2,pinch
        close(10)
        open(unit=10,access='append',file=datafile4)
        write(10,898)time,t1cnt,t2cnt,t3cnt,xmdot(icent,jcent),
1rhumida(icent,jcent)
c      write(*,898)time,dpsamp,rhbo,ykcalc,rcalc2,pinch
      close(10)
898 format(6e12.5)
      else
        print *, 'total mass flows at diff. stations (top)'
        write(*,3333)xm1tt,xm2tt,xm3tt,xm4tt,xm5tt
        print *, 'total mass flows at diff. stations (bot)'
        write(*,3333)xm1t,xm2t,xm3t,xm4t,xm5t
        print *, 'total mass flows at diff. stations'
        write(*,3333)xm1ttt,xm2ttt,xm3ttt,xm4ttt,xm5ttt
        print *, ' '
        print *, 'mass flow across sample(convct),calc.(top,bot),error'
        write(*,3336)convcte,convct1,convct2,flowerr
        print *, 'delta p at sample,Pa,psi,inches of H2O'
        write(*,3335)dpsamp,dpsamp2,dpsamp3
        print *, 'calculated permeability,input permeability'
        print *,ykcalcl,ykdarc
        print *, 'calculated resistance, rdry and rwet'
        print *,rcalc2,rcalc,rddry,rdwet
        end if

c      if(flowerr.lt.1.d-5)then
c      last=iter
c      end if

897 format(i10,3e12.4)
      return

      entry mcrogrfx
c*****THIS IS ROUTINE FOR MICROGRAPHICS PLOTS
c      do 1183 j5=1,m1
c      do 1183 i5=1,l1
c      isolid(i5,j5)=0
c      if(j5.le.jhuend.and.j5.ge.jhlbeg.and.
c      i15.lt.isbeg)then
c      isolid(i5,j5)=1
c      end if

```

```

c      if(j5.le.jhuend.and.j5.ge.jhlbeg.and.
c      1i5.gt.isend.and.i5.le.ihend)then
c      isolid(i5,j5)=1
c      end if
c 1183 continue
      do 1192 j5=1,nfp3
        kplot(j5)=0
1192 continue
        open(unit=8,file=datafile3)
        kplot(1)=1
        kplot(2)=1
        kplot(3)=1
        kplot(4)=1
        kplot(5)=1
        kplot(6)=1
        kplot(7)=1
        kplot(8)=1
        kplot(9)=1
        kplot(10)=1
        kplot(11)=1
        kplot(12)=1
        kplot(13)=1
        kplot(14)=1
        kplot(15)=1
        kplot(16)=1
        kplot(17)=1
        kplot(18)=1
        kplot(19)=1
        kplot(20)=1

        ione=0
        itwo=0
        ithree=1
        write(8,116)ione,itwo
        write(8,114)ithree
116 format(2i5)
114 format(i5)
        kgraf=2
        kflow=2*kgraf-1
        write(8,110)header
110 format(a64)
        write(8,111)kflow,11,m1,nfmax,mode,(kplot(i),i=1,nfmax)
111 format(16i5)
        iblok=0
        if(kgraf.eq.2)then
          do 150 j=2,m2
            do 150 i=2,12
              if(isolid(i,j).eq.1)then
                iblok=1
                goto 151
              endif
            150 continue
          151 continue
          write(8,111)iblok

```



```

        endif
        write(8,112) (title(n),n=1,nfmax)
112 format(4A18)
113 format(6e12.5)
        do 156 i1=1,11
c      print *,i1,x(i1),xu(i1)
156 continue
        do 157 j1=1,m1
c      print *,j1,y(j1),yv(j1),r(j1)
157 continue
        write(8,113) (x(i),i=1,11), (y(j),j=1,m1), (xu(i),i=2,11)
1, (yv(j),j=2,m1), (r(j),j=1,m1)
        do 169 n=1,nfmax
            if(kplot(n).ne.0) then
                write(8,113) ((f(i,j,n),i=1,11),j=1,m1)
            endif
169 continue
            if(iblok.eq.1) then
                write(8,111) ((isolid(i,j),i=1,11),j=1,m1)
            endif
            close(8)
            return

ENTRY TIMESTEP
DT=1.D0
c      nprint=10
eps(1)=1.d-2
eps(2)=1.d-2
eps(3)=1.d-2
eps(4)=1.d-2
EPS(5)=1.D-2
EPS(6)=1.D-2
JLAST=10000
LAST=700000

lsteady=.true.
rf=rfl
relax(5)=.9
relax(6)=.9

c*****calculate relative humidity at bottom exiting stream
rhbo=0.
lm31=l2-1
do 3811 j=2,jhlbeg-1
    rhbo=rhbo+rhumida(lm31,j)*ycv(j)
3811 continue
rhbo=rhbo/ducth

c*****calculate pressure difference across sample
dpsamp=p(icent,m2)-p(icent,2)
dpsamp2=dpsamp*1.45d-4
dpsamp3=dpsamp*13.6*2.953d-4
dpsamp=dabs(dpsamp)
rcalc=rddry*rkdarcy(icent,jcent)

```

```

c*****calculate apparent convective flow properties
      dpsample=p(icent,jsend+3)-p(icent,jsbeg-3)
      ykcalc=v(icent,jcent)*amu*ysample/dpsample
c*****nondimensional flow resistances
c*****rcalc is from equation, rcalc2 is from actual flows
      rcalc2=rddry*thick/ykcalc
      rcalc=rddry*rkdarcy(icent,jcent)
      dpsamp1=p(icent,jcent)-p(icent,jcent-1)
      ykcalc1=v(icent,jcent)*amu*ydif(jcent)/dpsamp1

      if(time.eq.2000)then
        pinch=.10
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if
      if(time.eq.4000)then
        pinch=.20
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if
      if(time.eq.6000)then
        pinch=.30
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if
      if(time.eq.8000)then
        pinch=.40
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if
      if(time.eq.10000)then
        pinch=.50
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if
      if(time.eq.12000)then
        pinch=.60
        open(unit=10,access='append',file=datafile5)
        write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
        close(10)
        iter=1000
      end if

```

```

    if(time.eq.14000)then
    pinch=.80
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.16000)then
    pinch=.90
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.18000)then
    pinch=1.014
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.20000.)then
    pinch=1.1
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
8978 format(4e12.5)
    close(10)
    iter=1000
    end if
    if(time.eq.22000.)then
    pinch=1.2
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.24000.)then
    pinch=1.3
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.26000.)then
    pinch=1.4
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
    close(10)
    iter=1000
    end if
    if(time.eq.28000.)then
    pinch=1.5
    open(unit=10,access='append',file=datafile5)
    write(10,8978)dpsamp,rhbo,rcalc2,ykcalc

```

```

close(10)
iter=1000
end if
if(time.eq.30000.)then
pinch=1.6
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.32000.)then
pinch=1.7
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.34000.)then
pinch=1.8
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.36000.)then
pinch=1.9
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.38000.)then
pinch=1.95
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.40000)then
pinch=2.0
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.42000)then
pinch=2.05
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if
if(time.eq.44000)then
pinch=2.10

```

```
open(unit=10,access='append',file=datafile5)
write(10,8978)dpsamp,rhbo,rcalc2,ykcalc
close(10)
iter=1000
end if

if(time.gt.46000.)then
last=iter
iter=last
end if

RETURN

end
```

Main Program -- EXPONENT.F

```

C-----
C  ----- THIS IS THE SIMPLEC PROGRAM -----
C
C - This version can handle mass source term -
C   and unsteady situation
C
C-----
C----- This Program is made unsteady -----
C----- ON -----
C----- January 25, 1993 -----
C
C----- Revised October, 1994 -----
C----- to include the exponential scheme -----
C-----
      PROGRAM MAIN
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL LSTOP,LSTEADY
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      CALL DEFALT
      CALL GRID
      CALL SETUP1
      CALL START
      CALL UNSTDY
10  CALL DENSE
      CALL BOUND
      CALL OUTPUT
      IF(LSTOP) STOP
      CALL SETUP2
      GO TO 10
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE DIFLOW
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/COEF/Flow,DIFF,ACOF
C*****
      ACOF=DIFF
      IF(Flow.EQ.0.) RETURN
      TEMP=DIFF-DABS(Flow)*0.1
      ACOF=0.
      IF(TEMP.LE.0.) RETURN
      TEMP=TEMP/DIFF
      ACOF=DIFF*TEMP**5
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SOLVE
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE*60
      LOGICAL  LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY

```

```

    PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
    COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1  AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2  X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3  Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4  YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5  R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
    COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1  FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
    COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
    DIMENSION D(MIJ),VAR(MIJ),VARM(MIJ),VARP(MIJ),PHIBAR(MIJ)
C*****
C      print *,nf
      ISTF=IST-1
      JSTF=JST-1
      IT1=L2+IST
      IT2=L3+IST
      JT1=M2+JST
      JT2=M3+JST
C-----
      NTIMER=NTIMES(NF)
      DO 999 NT=1,NTIMER
      DO 391 N=NF,NF
      IF(.NOT.LBLK(NF)) GO TO 60
C-----
C-  COME HERE TO DO BLOCK CORRECTION
C-----
C-  SUMMING IN I DIRECTION
C-----
      DO 22 J=JST,M2
      VAR(J)=0.
      VARP(J)=0.
      VARM(J)=0.
      D(J)=0.
      DO 33 I=IST,L2
      VAR(J)=VAR(J)+AP(I,J)
      IF(I.NE.IST) VAR(J)=VAR(J)-AIM(I,J)
      IF(I.NE.L2) VAR(J)=VAR(J)-AIP(I,J)
      VARM(J)=VARM(J)+AJM(I,J)
      VARP(J)=VARP(J)+AJP(I,J)
      D(J)=D(J)+CON(I,J)+AIP(I,J)*F(I+1,J,N)+AIM(I,J)*
1  F(I-1,J,N)+AJP(I,J)*F(I,J+1,N)+AJM(I,J)*F(I,J-1,N)-
2  AP(I,J)*F(I,J,N)
33 CONTINUE
22 CONTINUE
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VAR(4)=1.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VARP(4)=0.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VARM(4)=0.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) D(4)=0.
      PHIBAR(M1)=0.
      PHIBAR(JSTF)=0.

```

```

      PT(JSTF)=0.
      QT(JSTF)=PHIBAR(JSTF)
      DO 44 J=JST,M2
      DENOM=VAR(J)-PT(J-1)*VARM(J)
      PT(J)=VARP(J)/(DENOM)
      TEMP=D(J)
      QT(J)=(TEMP+VARM(J)*QT(J-1))/(DENOM)
44  CONTINUE
      DO 45 JJ=JST,M2
      J=JT1-JJ
45  PHIBAR(J)=PHIBAR(J+1)*PT(J)+QT(J)
      DO 47 I=IST,L2
      DO 47 J=JST,M2
47  F(I,J,N)=F(I,J,N)+PHIBAR(J)
C-----
C SUMMING IN J DIRECTION
C-----
C
      DO 51 I=IST,L2
      VAR(I)=0.
      VARP(I)=0.
      VARM(I)=0.
      D(I)=0.
      DO 53 J=JST,M2
      VAR(I)=VAR(I)+AP(I,J)
      IF(J.NE.JST) VAR(I)=VAR(I)-AJM(I,J)
      IF(J.NE.M2) VAR(I)=VAR(I)-AJP(I,J)
      VARP(I)=VARP(I)+AIP(I,J)
      VARM(I)=VARM(I)+AIM(I,J)
      D(I)=D(I)+CON(I,J)+AIP(I,J)*F(I+1,J,N)+
1    AIM(I,J)*F(I-1,J,N)+AJP(I,J)*F(I,J+1,N)+AJM(I,J)*
2    F(I,J-1,N)-AP(I,J)*F(I,J,N)
53  CONTINUE
51  CONTINUE
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VAR(4)=1.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VARP(4)=0.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) VARM(4)=0.
      IF((NF.EQ.3).OR.(NF.EQ.NP)) D(4)=0.
      PHIBAR(L1)=0.
      PHIBAR(ISTF)=0.
      PT(ISTF)=0.
      QT(ISTF)=PHIBAR(ISTF)
      DO 57 I=IST,L2
      DENOM=VAR(I)-PT(I-1)*VARM(I)
      PT(I)=VARP(I)/(DENOM)
      TEMP=D(I)
      QT(I)=(TEMP+QT(I-1)*VARM(I))/(DENOM)
57  CONTINUE
      DO 58 II=IST,L2
      I=IT1-II
58  PHIBAR(I)=PHIBAR(I+1)*PT(I)+QT(I)
      DO 59 I=IST,L2
      DO 59 J=JST,M2
59  F(I,J,N)=F(I,J,N)+PHIBAR(I)

```



```

60 CONTINUE
C -----
C----- TDMA -----
DO 90 J=JST,M2
PT(ISTF)=0.
QT(ISTF)=F(ISTF,J,N)
DO 70 I=IST,L2
DENOM=AP(I,J)-PT(I-1)*AIM(I,J)
PT(I)=AIP(I,J)/(DENOM)
TEMP=CON(I,J)+AJP(I,J)*F(I,J+1,N)+AJM(I,J)*F(I,J-1,N)
QT(I)=(TEMP+AIM(I,J)*QT(I-1))/(DENOM)
70 CONTINUE
DO 80 II=IST,L2
I=IT1-II
80 F(I,J,N)=F(I+1,J,N)*PT(I)+QT(I)
90 CONTINUE
C -----
DO 190 JJ=JST,M3
J=JT2-JJ
PT(ISTF)=0.
QT(ISTF)=F(ISTF,J,N)
DO 170 I=IST,L2
DENOM=AP(I,J)-PT(I-1)*AIM(I,J)
PT(I)=AIP(I,J)/(DENOM)
TEMP=CON(I,J)+AJP(I,J)*F(I,J+1,N)+AJM(I,J)*F(I,J-1,N)
QT(I)=(TEMP+AIM(I,J)*QT(I-1))/(DENOM)
170 CONTINUE
DO 180 II=IST,L2
I=IT1-II
180 F(I,J,N)=F(I+1,J,N)*PT(I)+QT(I)
190 CONTINUE
C -----
DO 290 I=IST,L2
PT(JSTF)=0.
QT(JSTF)=F(I,JSTF,N)
DO 270 J=JST,M2
DENOM=AP(I,J)-PT(J-1)*AJM(I,J)
PT(J)=AJP(I,J)/(DENOM)
TEMP=CON(I,J)+AIP(I,J)*F(I+1,J,N)+AIM(I,J)*F(I-1,J,N)
QT(J)=(TEMP+AJM(I,J)*QT(J-1))/(DENOM)
270 CONTINUE
DO 280 JJ=JST,M2
J=JT1-JJ
280 F(I,J,N)=F(I,J+1,N)*PT(J)+QT(J)
290 CONTINUE
C -----
DO 390 II=IST,L3
I=IT2-II
PT(JSTF)=0.
QT(JSTF)=F(I,JSTF,N)
DO 370 J=JST,M2
DENOM=AP(I,J)-PT(J-1)*AJM(I,J)
PT(J)=AJP(I,J)/(DENOM)
TEMP=CON(I,J)+AIP(I,J)*F(I+1,J,N)+AIM(I,J)*F(I-1,J,N)

```

```

      QT(J)=(TEMP+AJM(I,J)*QT(J-1))/(DENOM)
370 CONTINUE
      DO 380 JJ=JST,M2
        J=JT1-JJ
380 F(I,J,N)=F(I,J+1,N)*PT(J)+QT(J)
390 CONTINUE
391 CONTINUE
C*****
999 CONTINUE
      RETURN
C*****
      ENTRY RESET
      DO 400 J=1,M1
      DO 400 I=1,L1
      CON(I,J)=0.0
      AP(I,J)=0.0
400 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SETUP
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE*60
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
      PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
      COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1 AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2 X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4 YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5 R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
      COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1 FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/FLOW,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(ID,JD,NFD),FNEW(ID,JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(ID,JD),FP(ID,JD)
      DIMENSION U(ID,JD),V(ID,JD),PC(ID,JD),P(ID,JD)
      EQUIVALENCE(F(1,1,1),U(1,1)),(F(1,1,2),V(1,1)),(F(1,1,3),PC(1,1))
      EQUIVALENCE(F(1,1,4),P(1,1))
      DIMENSION AIPNF(ID,JD,3),AIMNF(ID,JD,3),AJPNF(ID,JD,3),
1AJMNF(ID,JD,3),APNF(ID,JD,2),CONNF(ID,JD,2)
C*****
1 FORMAT(/,15X,'COMPUTATION IN CARTESIAN COORDINATES')
2 FORMAT(/,15X,'COMPUTATION FOR AXISYMMETRIC SITUATION')
3 FORMAT(/,15X,'COMPUTATION IN POLAR COORDINATES')

```

4 FORMAT(14X,40(1H*),//)

C

```
ENTRY DEFALT
NP=4
NFMIN=5
NFMAX=20
NRHO=NFMAX+1
NGAM=NFMAX+2
NCON=NFMAX+3
LSTOP=.FALSE.
MODE=1
LAST=5
TIME=0.0
ITER=0
JITER=0
LScheme=1
LSTEADY=.FALSE.
DT=1.0D+20
IPREF=1
JPREF=1
RHOCON=1.0
DO 877 IL=1,NFP3
LSOLVE(IL)=.FALSE.
LPRINT(IL)=.FALSE.
LBLK(IL)=.TRUE.
RELAX(IL)=1.0
NTIMES(IL)=1
877 CONTINUE
DO 878 K=1,NFD
EPS(K)=1.0D+20
IFLAG(K)=1
878 CONTINUE
RETURN
```

C

```
ENTRY SETUP1
L2=L1-1
L3=L2-1
M2=M1-1
M3=M2-1
X(1)=XU(2)
DO 5 I=2,L2
5 X(I)=0.5*(XU(I+1)+XU(I))
X(L1)=XU(L1)
Y(1)=YV(2)
DO 10 J=2,M2
10 Y(J)=0.5*(YV(J+1)+YV(J))
Y(M1)=YV(M1)
DO 15 I=2,L1
15 XDIF(I)=X(I)-X(I-1)
DO 18 I=2,L2
18 XCV(I)=XU(I+1)-XU(I)
DO 20 I=3,L2
20 XCVS(I)=XDIF(I)
XCVS(3)=XCVS(3)+XDIF(2)
```

```

      XCVS(L2)=XCVS(L2)+XDIF(L1)
      DO 22 I=3,L3
        XCVI(I)=0.5*XCV(I)
22    XCVIP(I)=XCVI(I)
        XCVIP(2)=XCV(2)
        XCVI(L2)=XCV(L2)
        DO 35 J=2,M1
35    YDIF(J)=Y(J)-Y(J-1)
        DO 40 J=2,M2
40    YCV(J)=YV(J+1)-YV(J)
        DO 45 J=3,M2
45    YCVS(J)=YDIF(J)
        YCVS(3)=YCVS(3)+YDIF(2)
        YCVS(M2)=YCVS(M2)+YDIF(M1)
        IF(MODE.NE.1) GO TO 55
        DO 52 J=1,M1
        RMN(J)=1.0
52    R(J)=1.0
        GO TO 56
55    DO 50 J=2,M1
50    R(J)=R(J-1)+YDIF(J)
        RMN(2)=R(1)
        DO 60 J=3,M2
60    RMN(J)=RMN(J-1)+YCV(J-1)
        RMN(M1)=R(M1)
56    CONTINUE
        DO 57 J=1,M1
        SX(J)=1.
        SXMN(J)=1.
        IF(MODE.NE.3) GO TO 57
        SX(J)=R(J)
        IF(J.NE.1) SXMN(J)=RMN(J)
57    CONTINUE
        DO 62 J=2,M2
        YCVR(J)=R(J)*YCV(J)
        ARX(J)=YCVR(J)
        IF(MODE.NE.3) GO TO 62
        ARX(J)=YCV(J)
62    CONTINUE
        DO 64 J=4,M3
64    YCVRS(J)=0.5*(R(J)+R(J-1))*YDIF(J)
        YCVRS(3)=0.5*(R(3)+R(1))*YCVS(3)
        YCVRS(M2)=0.5*(R(M1)+R(M3))*YCVS(M2)
        IF(MODE.NE.2) GO TO 67
        DO 65 J=3,M3
        ARXJ(J)=0.25*(1.+RMN(J)/R(J))*ARX(J)
65    ARXJP(J)=ARX(J)-ARXJ(J)
        GO TO 68
67    DO 66 J=3,M3
        ARXJ(J)=0.5*ARX(J)
66    ARXJP(J)=ARXJ(J)
68    ARXJP(2)=ARX(2)
        ARXJ(M2)=ARX(M2)
        DO 70 J=3,M3

```

```

      FV(J)=ARXJP(J)/ARX(J)
70  FVP(J)=1.-FV(J)
      DO 85 I=3,L2
      FX(I)=0.5*XCV(I-1)/XDIF(I)
85  FXM(I)=1.-FX(I)
      FX(2)=0.
      FXM(2)=1.
      FX(L1)=1.
      FXM(L1)=0.
      DO 90 J=3,M2
      FY(J)=0.5*YCV(J-1)/YDIF(J)
90  FYM(J)=1.-FY(J)
      FY(2)=0.
      FYM(2)=1.
      FY(M1)=1.
      FYM(M1)=0.
CON,AP,U,V,RHO,PC AND P ARRAYS ARE INITIALIZED HERE
      DO 95 J=1,M1
      DO 95 I=1,L1
      PC(I,J)=0.
      U(I,J)=0.
      V(I,J)=0.
      CON(I,J)=0.
      AP(I,J)=0.
      RHO(I,J)=RHOCON
      P(I,J)=0.
95  CONTINUE
      IF(MODE.EQ.1) PRINT 1
      IF(MODE.EQ.2) PRINT 2
      IF(MODE.EQ.3) PRINT 3
      PRINT 4
      RETURN
C-----
      ENTRY SETUP2
COEFFICIENTS FOR THE U EQUATION
      NF=1
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 100
      IF(LSTEADY) IFLAG(NF)=0
      IST=3
      JST=2
      CALL GAMSOR
      IF(RELAX(NF).EQ.1.0) RELAX(NF)=0.9
      REL=1.-RELAX(NF)
      DO 102 I=3,L2
      FL=XCVI(I)*V(I,2)*RHO(I,1)
      FLM=XCVIP(I-1)*V(I-1,2)*RHO(I-1,1)
      FLOW=R(1)*(FL+FLM)
      DIFF=R(1)*(XCVI(I)*GAM(I,1)+XCVIP(I-1)*GAM(I-1,1))/YDIF(2)
      CALL DIFLOW
102  AJM(I,2)=ACOF+DMAX1(0.0D0,FLOW)
      DO 103 J=2,M2
      FLOW=ARX(J)*U(2,J)*RHO(1,J)
      DIFF=ARX(J)*GAM(1,J)/(XCV(2)*SX(J))

```

```

      CALL DIFLOW
      AIM(3,J)=ACOF+DMAX1(0.0D0, FLOW)
      DO 103 I=3,L2
      IF(I.EQ.L2) GO TO 104
      FL=U(I,J)*(FX(I)*RHO(I,J)+FXM(I)*RHO(I-1,J))
      FLP=U(I+1,J)*(FX(I+1)*RHO(I+1,J)+FXM(I+1)*RHO(I,J))
      FLOW=ARX(J)*0.5*(FL+FLP)
      DIFF=ARX(J)*GAM(I,J)/(XCV(I)*SX(J))
      GO TO 105
104  FLOW=ARX(J)*U(L1,J)*RHO(L1,J)
      DIFF=ARX(J)*GAM(L1,J)/(XCV(L2)*SX(J))
105  CALL DIFLOW
      AIM(I+1,J)=ACOF+DMAX1(0.0D0, FLOW)
      AIP(I,J)=AIM(I+1,J)-FLOW
      IF(J.EQ.M2) GO TO 106
      FL=XCVI(I)*V(I,J+1)*(FY(J+1)*RHO(I,J+1)+FYM(J+1)*RHO(I,J))
      FLM=XCVIP(I-1)*V(I-1,J+1)*(FY(J+1)*RHO(I-1,J+1)+FYM(J+1)*
1  RHO(I-1,J))
      GM=GAM(I,J)*GAM(I,J+1)/(YCV(J)*GAM(I,J+1)+YCV(J+1)*GAM(I,J)+
1  1.0D-40)*XCVI(I)
      GMM=GAM(I-1,J)*GAM(I-1,J+1)/(YCV(J)*GAM(I-1,J+1)+YCV(J+1)*
1  GAM(I-1,J)+1.0D-40)*XCVIP(I-1)
      DIFF=RMN(J+1)*2.*(GM+GMM)
      GO TO 107
106  FL=XCVI(I)*V(I,M1)*RHO(I,M1)
      FLM=XCVIP(I-1)*V(I-1,M1)*RHO(I-1,M1)
      DIFF=R(M1)*(XCVI(I)*GAM(I,M1)+XCVIP(I-1)*GAM(I-1,M1))/YDIF(M1)
107  FLOW=RMN(J+1)*(FL+FLM)
      CALL DIFLOW
      AJM(I,J+1)=ACOF+DMAX1(0.0D0, FLOW)
      AJP(I,J)=AJM(I,J+1)-FLOW
      VOL=YCVR(J)*XCVS(I)
      APT=(RHO(I,J)*XCVI(I)+RHO(I-1,J)*XCVIP(I-1))
1  /(XCVS(I)*DT)
      AP(I,J)=AP(I,J)-APT
      CON(I,J)=CON(I,J)+APT*FOLD(I,J,NF)
      ANBR=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
      AP(I,J)=((-AP(I,J)*VOL)+ANBR)/RELAX(NF)
      CON(I,J)=CON(I,J)*VOL+REL*AP(I,J)*U(I,J)
      DU(I,J)=VOL/(XDIF(I)*SX(J))
      DU(I,J)=DU(I,J)/(AP(I,J)-ANBR)
103  CONTINUE
      DO 110 I=1,L1
      DO 110 J=1,M1
      AIPNF(I,J,1)=AIP(I,J)
      AIMNF(I,J,1)=AIM(I,J)
      AJPNF(I,J,1)=AJP(I,J)
      AJMNF(I,J,1)=AJM(I,J)
      APNF(I,J,1)=AP(I,J)
      CONNF(I,J,1)=CON(I,J)
110  CONTINUE
C_____ TEMPORARY USE OF PC(I,J) TO STORE UHAT _____
      DO 151 J=2,M2
      DO 151 I=3,L2

```

```

      ANBR=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
      BCON=CON(I,J)-(ANBR*U(I,J))
      PC(I,J)=(AIP(I,J)*U(I+1,J)+AIM(I,J)*U(I-1,J)+AJP(I,J)*U(I,J+1)+
1AJM(I,J)*U(I,J-1)+BCON)/(AP(I,J)-ANBR)
151  CONTINUE
100  CONTINUE
COEFFICIENTS FOR THE V EQUATION-----
      NF=2
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 200
      IF(LSTEADY) IFLAG(NF)=0
      IST=2
      JST=3
      CALL GAMSOR
      IF(RELAX(NF).EQ.1.0) RELAX(NF)=0.9
      REL=1.-RELAX(NF)
      DO 202 I=2,L2
      AREA=R(1)*XCV(I)
      FLOW=AREA*V(I,2)*RHO(I,1)
      DIFF=AREA*GAM(I,1)/YCV(2)
      CALL DIFLOW
202  AJM(I,3)=ACOF+DMAX1(0.0D0, FLOW)
      DO 203 J=3,M2
      FL=ARXJ(J)*U(2,J)*RHO(1,J)
      FLM=ARXJP(J-1)*U(2,J-1)*RHO(1,J-1)
      FLOW=FL+FLM
      DIFF=(ARXJ(J)*GAM(1,J)+ARXJP(J-1)*GAM(1,J-1))/(XDIF(2)*SXMN(J))
      CALL DIFLOW
      AIM(2,J)=ACOF+DMAX1(0.0D0, FLOW)
      DO 203 I=2,L2
      IF(I.EQ.L2) GO TO 204
      FL=ARXJ(J)*U(I+1,J)*(FX(I+1)*RHO(I+1,J)+FXM(I+1)*RHO(I,J))
      FLM=ARXJP(J-1)*U(I+1,J-1)*(FX(I+1)*RHO(I+1,J-1)+FXM(I+1)*
1 RHO(I,J-1))
      GM=GAM(I,J)*GAM(I+1,J)/(XCV(I)*GAM(I+1,J)+XCV(I+1)*GAM(I,J)+
1 1.D-40)*ARXJ(J)
      GMM=GAM(I,J-1)*GAM(I+1,J-1)/(XCV(I)*GAM(I+1,J-1)+XCV(I+1)*
1 GAM(I,J-1)+1.0D-40)*ARXJP(J-1)
      DIFF=2.*(GM+GMM)/SXMN(J)
      GO TO 205
204  FL=ARXJ(J)*U(L1,J)*RHO(L1,J)
      FLM=ARXJP(J-1)*U(L1,J-1)*RHO(L1,J-1)
      DIFF=(ARXJ(J)*GAM(L1,J)+ARXJP(J-1)*GAM(L1,J-1))/(XDIF(L1)*SXMN(J))
205  FLOW=FL+FLM
      CALL DIFLOW
      AIM(I+1,J)=ACOF+DMAX1(0.0D0, FLOW)
      AIP(I,J)=AIM(I+1,J)-FLOW
      IF(J.EQ.M2) GO TO 206
      AREA=R(J)*XCV(I)
      FL=V(I,J)*(FY(J)*RHO(I,J)+FYM(J)*RHO(I,J-1))*RMN(J)
      FLP=V(I,J+1)*(FY(J+1)*RHO(I,J+1)+FYM(J+1)*RHO(I,J))*RMN(J+1)
      FLOW=(FV(J)*FL+FVP(J)*FLP)*XCV(I)
      DIFF=AREA*GAM(I,J)/YCV(J)
      GO TO 207

```

```

206 AREA=R(M1)*XCV(I)
    FLOW=AREA*V(I,M1)*RHO(I,M1)
    DIFF=AREA*GAM(I,M1)/YCV(M2)
207 CALL DIFLOW
    AJM(I,J+1)=ACOF+DMAX1(0.0D0, FLOW)
    AJP(I,J)=AJM(I,J+1)-FLOW
    VOL=YCVRS(J)*XCV(I)
    APT=(ARXJ(J)*RHO(I,J)*0.5*(SX(J)+SXMN(J))+ARXJP(J-1)*RHO(I,J-1)*
1 0.5*(SX(J-1)+SXMN(J)))/(YCVRS(J)*DT)
    AP(I,J)=AP(I,J)-APT
    CON(I,J)=CON(I,J)+APT*FOLD(I,J,NF)
    ANBR=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
    AP(I,J)=((-AP(I,J)*VOL)+ANBR)/RELAX(NF)
    CON(I,J)=CON(I,J)*VOL+REL*AP(I,J)*V(I,J)
    DV(I,J)=VOL/YDIF(J)
    DV(I,J)=DV(I,J)/(AP(I,J)-ANBR)
203 CONTINUE
    DO 210 I=1,L1
    DO 210 J=1,M1
        AIPNF(I,J,2)=AIP(I,J)
        AIMNF(I,J,2)=AIM(I,J)
        AJPNF(I,J,2)=AJP(I,J)
        AJMNF(I,J,2)=AJM(I,J)
        APNF(I,J,2)=AP(I,J)
        CONNF(I,J,2)=CON(I,J)
210 CONTINUE
200 CONTINUE
COEFFICIENTS FOR THE PRESSURE EQUATION_____
    NF=NP
    IFLAG(NF)=1
    CALL RESET
    IF(.NOT.LSOLVE(NF)) GO TO 500
    IST=2
    JST=2
    CALL GAMSOR
    DO 401 J=2,M2
    DO 401 I=2,L2
        VOL=YCVR(J)*XCV(I)
        CON(I,J)=CON(I,J)*VOL
        AP(I,J)=-AP(I,J)*VOL
401 CONTINUE
    DO 402 I=2,L2
        ARHO=R(1)*XCV(I)*RHO(I,1)
        CON(I,2)=CON(I,2)+ARHO*V(I,2)
402 AJM(I,2)=0.
    DO 403 J=2,M2
        ARHO=ARX(J)*RHO(1,J)
        CON(2,J)=CON(2,J)+ARHO*U(2,J)
        AIM(2,J)=0.
    DO 403 I=2,L2
        IF(I.EQ.L2) GO TO 404
        ARHO=ARX(J)*(FX(I+1)*RHO(I+1,J)+FXM(I+1)*RHO(I,J))
        FLOW=ARHO*PC(I+1,J)
        CON(I,J)=CON(I,J)-FLOW

```



```

      CON(I+1,J)=CON(I+1,J)+FLOW
      AIP(I,J)=ARHO*DU(I+1,J)
      AIM(I+1,J)=AIP(I,J)
      GO TO 405
404  ARHO=ARX(J)*RHO(L1,J)
      CON(I,J)=CON(I,J)-ARHO*U(L1,J)
      AIP(I,J)=0.
405  IF(J.EQ.M2) GO TO 406
      ARHO=RMN(J+1)*XCV(I)*(FY(J+1)*RHO(I,J+1)+FYM(J+1)*RHO(I,J))
      ANBR=AIPNF(I,J+1,2)+AIMNF(I,J+1,2)+AJPNF(I,J+1,2)+
1    AJMNF(I,J+1,2)
      BCON=CONNF(I,J+1,2)-(ANBR*V(I,J+1))
      VHAT=(AIPNF(I,J+1,2)*V(I+1,J+1)+AIMNF(I,J+1,2)*V(I-1,J+1)+
1    AJPNF(I,J+1,2)*V(I,J+2)+AJMNF(I,J+1,2)*V(I,J)+
2    BCON)/(APNF(I,J+1,2)-ANBR)
      FLOW=ARHO*VHAT
      CON(I,J)=CON(I,J)-FLOW
      CON(I,J+1)=CON(I,J+1)+FLOW
      AJP(I,J)=ARHO*DV(I,J+1)
      AJM(I,J+1)=AJP(I,J)
      GO TO 407
406  ARHO=RMN(M1)*XCV(I)*RHO(I,M1)
      CON(I,J)=CON(I,J)-ARHO*V(I,M1)
      AJP(I,J)=0.
407  AP(I,J)=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
403  CONTINUE
      DO 411 I=1,L1
      DO 411 J=1,M1
      AIPNF(I,J,3)=AIP(I,J)
      AIMNF(I,J,3)=AIM(I,J)
      AJPNF(I,J,3)=AJP(I,J)
      AJMNF(I,J,3)=AJM(I,J)
411  CONTINUE
      IF(ITER.LE.1) GO TO 409
      DO 408 J=2,M2
      DO 408 I=2,L2
      AP(I,J)=AP(I,J)/RELAX(NP)
      CON(I,J)=CON(I,J)+(1.-RELAX(NP))*AP(I,J)*P(I,J)
408  CONTINUE
409  CONTINUE
      CALL SOLVE
C#####
      NF=1
      IST=3
      JST=2
      DO 415 I=1,L1
      DO 415 J=1,M1
      AIP(I,J)=AIPNF(I,J,1)
      AIM(I,J)=AIMNF(I,J,1)
      AJP(I,J)=AJPNF(I,J,1)
      AJM(I,J)=AJMNF(I,J,1)
      AP(I,J)=APNF(I,J,1)
      CON(I,J)=CONNF(I,J,1)
415  CONTINUE

```

```

      DO 413 J=2,M2
      DO 413 I=3,L2
      ANBR=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
      DUIJ=DU(I,J)*(AP(I,J)-ANBR)
      CON(I,J)=CON(I,J)+DUIJ*(P(I-1,J)-P(I,J))
413  CONTINUE
      CALL SOLVE
C#####
      NF=2
      IST=2
      JST=3
      DO 416 I=1,L1
      DO 416 J=1,M1
      AIP(I,J)=AIPNF(I,J,2)
      AIM(I,J)=AIMNF(I,J,2)
      AJP(I,J)=AJPNF(I,J,2)
      AJM(I,J)=AJMNF(I,J,2)
      AP(I,J)=APNF(I,J,2)
      CON(I,J)=CONNF(I,J,2)
416  CONTINUE
      DO 414 J=3,M2
      DO 414 I=2,L2
      ANBR=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
      DVIJ=DV(I,J)*(AP(I,J)-ANBR)
      CON(I,J)=CON(I,J)+DVIJ*(P(I,J-1)-P(I,J))
414  CONTINUE
      CALL SOLVE
C#####
COEFFICIENTS FOR THE PRESSURE CORRECTION EQUATION-----
      NF=3
      CALL RESET
      IF(.NOT.LSOLVE(NF)) GO TO 500
      IF(LSTEADY) IFLAG(NF)=0
      IST=2
      JST=2
      CALL GAMSOR
      DO 417 J=2,M2
      DO 417 I=2,L2
      VOL=YCVR(J)*XCV(I)
      CON(I,J)=CON(I,J)*VOL
417  CONTINUE
      DO 418 I=1,L1
      DO 418 J=1,M1
      AIP(I,J)=AIPNF(I,J,3)
      AIM(I,J)=AIMNF(I,J,3)
      AJP(I,J)=AJPNF(I,J,3)
      AJM(I,J)=AJMNF(I,J,3)
      AP(I,J)=AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
418  CONTINUE
      SMAX=0.
      SSUM=0.
      DO 474 I=2,L2
      ARHO=R(1)*XCV(I)*RHO(I,1)
474  CON(I,2)=CON(I,2)+ARHO*V(I,2)

```

```

DO 475 J=2,M2
ARHO=ARX(J)*RHO(1,J)
CON(2,J)=CON(2,J)+ARHO*U(2,J)
DO 475 I=2,L2
IF(I.EQ.L2) GO TO 476
ARHO=ARX(J)*(FX(I+1)*RHO(I+1,J)+FXM(I+1)*RHO(I,J))
FLOW=ARHO*U(I+1,J)
CON(I,J)=CON(I,J)-FLOW
CON(I+1,J)=CON(I+1,J)+FLOW
GO TO 477
476 ARHO=ARX(J)*RHO(L1,J)
CON(I,J)=CON(I,J)-ARHO*U(L1,J)
477 IF(J.EQ.M2) GO TO 478
ARHO=RMN(J+1)*XCV(I)*(FY(J+1)*RHO(I,J+1)+FYM(J+1)*RHO(I,J))
FLOW=ARHO*V(I,J+1)
CON(I,J)=CON(I,J)-FLOW
CON(I,J+1)=CON(I,J+1)+FLOW
GO TO 479
478 ARHO=RMN(M1)*XCV(I)*RHO(I,M1)
CON(I,J)=CON(I,J)-ARHO*V(I,M1)
479 PC(I,J)=0.0
SMAX=DMAX1(SMAX,DABS(CON(I,J)))
SSUM=SSUM+CON(I,J)
475 CONTINUE
CALL SOLVE
IF(SMAX.LE.EPS(NF)) IFLAG(NF)=1
IF(SMAX.LE.EPS(NF)) IFLAG(1)=1
IF(SMAX.LE.EPS(NF)) IFLAG(2)=1
COME HERE TO CORRECT THE VELOCITIES-----
DO 501 J=2,M2
DO 501 I=2,L2
IF(I.NE.2) U(I,J)=U(I,J)+DU(I,J)*(PC(I-1,J)-PC(I,J))
IF(J.NE.2) V(I,J)=V(I,J)+DV(I,J)*(PC(I,J-1)-PC(I,J))
501 CONTINUE
500 CONTINUE
COEFFICIENTS FOR OTHER EQUATIONS-----
IST=2
JST=2
DO 600 NF=NFMIN,NFMAX
IF(.NOT.LSOLVE(NF)) GO TO 600
IF(LSTEADY) IFLAG(NF)=0
CALL RESET
CALL GAMSOR
REL=1.-RELAX(NF)
DO 602 I=2,L2
AREA=R(1)*XCV(I)
FLOW=AREA*V(I,2)*RHO(I,1)
DIFF=AREA*GAM(I,1)/YDIF(2)
CALL DIFLOW
602 AJM(I,2)=ACOF+DMAX1(0.0D0, FLOW)
DO 603 J=2,M2
FLOW=ARX(J)*U(2,J)*RHO(1,J)
DIFF=ARX(J)*GAM(1,J)/(XDIF(2)*SX(J))
CALL DIFLOW

```

```

        AIM(2,J)=ACOF+DMAX1(0.0D0, FLOW)
        DO 603 I=2,L2
        IF(I.EQ.L2) GO TO 604
        FLOW=ARX(J)*U(I+1,J)*(FX(I+1)*RHO(I+1,J)+FXM(I+1)*RHO(I,J))
        DIFF=ARX(J)*2.*GAM(I,J)*GAM(I+1,J)/((XCV(I)*GAM(I+1,J)+
1 XCV(I+1)*GAM(I,J)+1.0D-40)*SX(J))
        GO TO 605
604 FLOW=ARX(J)*U(L1,J)*RHO(L1,J)
        DIFF=ARX(J)*GAM(L1,J)/(XDIF(L1)*SX(J))
605 CALL DIFLOW
        AIM(I+1,J)=ACOF+DMAX1(0.0D0, FLOW)
        AIP(I,J)=AIM(I+1,J)-FLOW
        AREA=RMN(J+1)*XCV(I)
        IF(J.EQ.M2) GO TO 606
        FLOW=AREA*V(I,J+1)*(FY(J+1)*RHO(I,J+1)+FYM(J+1)*RHO(I,J))
        DIFF=AREA*2.*GAM(I,J)*GAM(I,J+1)/(YCV(J)*GAM(I,J+1)+
1 YCV(J+1)*GAM(I,J)+1.0D-40)
        GO TO 607
606 FLOW=AREA*V(I,M1)*RHO(I,M1)
        DIFF=AREA*GAM(I,M1)/YDIF(M1)
607 CALL DIFLOW
        AJM(I,J+1)=ACOF+DMAX1(0.0D0, FLOW)
        AJP(I,J)=AJM(I,J+1)-FLOW
        VOL=YCVR(J)*XCV(I)
        APT=VOL*RHO1(I,J)/DT
        CON(I,J)=CON(I,J)*VOL/APT
        AP(I,J)=-AP(I,J)*VOL+AIP(I,J)+AIM(I,J)+AJP(I,J)+AJM(I,J)
        ALAMBDA(I,J)=AP(I,J)/APT
        AIP(I,J)=AIP(I,J)/APT
        AIM(I,J)=AIM(I,J)/APT
        AJP(I,J)=AJP(I,J)/APT
        AJM(I,J)=AJM(I,J)/APT
603 CONTINUE
c   print *, 'call factor'
        CALL FACTOR
        DO 608 I=2,L2
        DO 608 J=2,M2
        AP(I,J)=1.0+ALAMBDA(I,J)*FP(I,J)
        CON(I,J)=CON(I,J)+AIP(I,J)*(1.-FP(I+1,J))*FOLD(I+1,J,NF)
1+AIM(I,J)*(1.-FP(I-1,J))*FOLD(I-1,J,NF)
1+AJP(I,J)*(1.-FP(I,J+1))*FOLD(I,J+1,NF)
1+AJM(I,J)*(1.-FP(I,J-1))*FOLD(I,J-1,NF)
        CON(I,J)=CON(I,J)+(1.0+FP(I,J)*ALAMBDA(I,J)-ALAMBDA(I,J))*
1FOLD(I,J,NF)
        AIP(I,J)=AIP(I,J)*FP(I+1,J)
        AIM(I,J)=AIM(I,J)*FP(I-1,J)
        AJP(I,J)=AJP(I,J)*FP(I,J+1)
        AJM(I,J)=AJM(I,J)*FP(I,J-1)
        AP(I,J)=AP(I,J)/RELAX(NF)
        REL=1.-RELAX(NF)
        CON(I,J)=CON(I,J)+REL*AP(I,J)*F(I,J,NF)
608 CONTINUE
c
        DO 730 I=IST,L2

```

```

        DO 730 J=JST,M2
        FNEW(I,J)=F(I,J,NF)
730  CONTINUE
C-----
c      print *, 'call solve'
      CALL SOLVE
c      print *, 'exit solve'
C-----
      if(nf.eq.5)call output3
      DIFMAX=0.0
      DO 731 J=2,M2
      DO 731 I=2,L2
      DELTAT=DABS((F(I,J,NF)-FNEW(I,J))/(F(I,J,NF)+1.D-40))
      if(nf.eq.5)call output2
      DIFMAX=DMAX1(DIFMAX,DELTAT)
731  CONTINUE
c      print *, 'after difmax'
      IF(DIFMAX.LE.EPS(NF)) IFLAG(NF)=1
      call output
C-----
600  CONTINUE
      JITER=JITER+1
      IF(JITER.LT.2) RETURN
C*****
      JFLAG=0
      DO 732 N=1,NFD
      JFLAG=JFLAG+IFLAG(N)
732  CONTINUE
      IF(JFLAG.EQ.NFD) GO TO 742
      IF(JITER.LT.JLAST) RETURN
      CALL BOUND
      CALL SAVE1
      CALL OUTTOO
      print *, 'Number of iterations exceeds JLAST'
      LSTOP=.TRUE.
      call savepc
      RETURN
C*****
742  CONTINUE
      JITER=0
      ITER=ITER+1
      IF(LSTEADY) TIME=TIME+DT
      CALL UNSTDY
      CALL BOUND
      CALL OUTTOO
      CALL SAVE2
      CALL SAVEPC
      IF(ITER.GE.LAST) LSTOP=.TRUE.
      IF(ITER.GE.LAST)THEN
      print *, 'Number of iterations exceeds LAST'
      ENDIF
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

```

SUBROUTINE SUPPLY
  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
  CHARACTER TITLE*18
  CHARACTER DATAFILE*60
  LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
  PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
  COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1 AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2 X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4 YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5 R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
  COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1 FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
  COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
  DIMENSION U(ID,JD),V(ID,JD),PC(ID,JD),P(ID,JD)
  EQUIVALENCE(F(1,1,1),U(1,1)),(F(1,1,2),V(1,1)),(F(1,1,3),PC(1,1))
  EQUIVALENCE(F(1,1,4),P(1,1))
C*****
10 FORMAT(14(1X,1H*),3X,A10,1X,14(1X,1H*))
20 FORMAT(1X,4H I =,I6,6I9)
30 FORMAT(1X,1HJ)
40 FORMAT(1X,I2,3X,1P7E9.2)
50 FORMAT(1X,1H )
51 FORMAT(1X,'I =',2X,6(I4,9X))
52 FORMAT(1X,'X =',1P6E12.4)
53 FORMAT('TH =',1P6E12.4)
54 FORMAT(1X,'J =',2X,6(I4,9X))
55 FORMAT(1X,'Y =',1P6E12.4)
C*****
C-----
  ENTRY UGRID
  XU(2)=0.
  DX=XL/DFLOAT(L1-2)
  DO 1 I=3,L1
1 XU(I)=XU(I-1)+DX
  YV(2)=0.
  DY=YL/DFLOAT(M1-2)
  DO 2 J=3,M1
2 YV(J)=YV(J-1)+DY
  RETURN
C*****
  ENTRY PRINT
  IF(.NOT.LPRINT(3)) GO TO 80
CALCULATE THE STREAM FUNCTION-----
  F(2,2,3)=0.
  DO 82 I=2,L1
  IF(I.NE.2) F(I,2,3)=F(I-1,2,3)-RHO(I-1,1)*V(I-1,2)
1*R(1)*XCV(I-1)
  DO 82 J=3,M1

```

```

      RHOM=FX(I)*RHO(I,J-1)+FXM(I)*RHO(I-1,J-1)
      82 F(I,J,3)=F(I,J-1,3)+RHOM*U(I,J-1)*ARX(J-1)
      80 CONTINUE
C-----
      IF(.NOT.LPRINT(NP)) GO TO 90
C-----
CONSTRUCT BOUNDARY PRESSURES BY EXTRAPOLATION
      DO 91 J=2,M2
        P(1,J)=(P(2,J)*XCVS(3)-P(3,J)*XDIF(2))/XDIF(3)
      91 P(L1,J)=(P(L2,J)*XCVS(L2)-P(L3,J)*XDIF(L1))/XDIF(L2)
      DO 92 I=2,L2
        P(I,1)=(P(I,2)*YCVS(3)-P(I,3)*YDIF(2))/YDIF(3)
      92 P(I,M1)=(P(I,M2)*YCVS(M2)-P(I,M3)*YDIF(M1))/YDIF(M2)
        P(1,1)=P(2,1)+P(1,2)-P(2,2)
        P(L1,1)=P(L2,1)+P(L1,2)-P(L2,2)
        P(1,M1)=P(2,M1)+P(1,M2)-P(2,M2)
        P(L1,M1)=P(L2,M1)+P(L1,M2)-P(L2,M2)
        PREF=P(IPREF,JPREF)
      DO 93 J=1,M1
      DO 93 I=1,L1
      93 P(I,J)=P(I,J)-PREF
      90 CONTINUE
C-----
C...START PRINTOUT
      PRINT 50
      IEND=0
      301 IF(IEND.EQ.L1) GO TO 310
        IBEG=IEND+1
        IEND=IEND+6
        IEND=MIN0(IEND,L1)
        PRINT 50
        PRINT 51,(I,I=IBEG,IEND)
        IF(MODE.EQ.3) GO TO 302
        PRINT 52,(X(I),I=IBEG,IEND)
        GO TO 303
      302 PRINT 53,(X(I),I=IBEG,IEND)
      303 GO TO 301
      310 JEND=0
        PRINT 50
      311 IF(JEND.EQ.M1) GO TO 320
        JBEG=JEND+1
        JEND=JEND+6
        JEND=MIN0(JEND,M1)
        PRINT 50
        PRINT 54,(J,J=JBEG,JEND)
        PRINT 55,(Y(J),J=JBEG,JEND)
        GO TO 311
      320 CONTINUE
C-----
      DO 999 NF=1,NGAM
        IF(.NOT.LPRINT(NF)) GO TO 999
        PRINT 50
        PRINT 10,TITLE(NF)
        IFST=1

```

```

JFST=1
IF(NF.EQ.1.OR.NF.EQ.3) IFST=2
IF(NF.EQ.2.OR.NF.EQ.3) JFST=2
IBEG=IFST-7
110 CONTINUE
IBEG=IBEG+7
IEND=IBEG+6
IEND=MIN0(IEND,L1)
PRINT 50
PRINT 20,(I,I=IBEG,IEND)
PRINT 30
JFL=JFST+M1
DO 115 JJ=JFST,M1
J=JFL-JJ
PRINT 40,J,(F(I,J,NF),I=IBEG,IEND)
115 CONTINUE
IF(IEND.LT.L1) GO TO 110
999 CONTINUE
RETURN
END

SUBROUTINE FACTOR
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
CHARACTER TITLE*18
CHARACTER DATAFILE*60
LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY
PARAMETER (ID=300,JD=300,NFD=20,NFP3=23,MIJ=300)
COMMON F(ID,JD,NFD),RHO(ID,JD),GAM(ID,JD),CON(ID,JD),
1 AIP(ID,JD),AIM(ID,JD),AJP(ID,JD),AJM(ID,JD),AP(ID,JD),
2 X(ID),XU(ID),XDIF(ID),XCV(ID),XCVS(ID),
3 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),rho1(id,jd),
4 YCVR(JD),YCVRS(JD),ARX(JD),ARXJ(JD),ARXJP(JD),
5 R(JD),RMN(JD),SX(JD),SXMN(JD),XCVI(ID),XCVIP(ID)
COMMON DU(ID,JD),DV(ID,JD),FV(JD),FVP(JD),
1 FX(ID),FXM(ID),FY(JD),FYM(JD),PT(MIJ),QT(MIJ)
COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,L1,L2,L3,M1,M2,M3,
1IST,JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,XL,YL,
2IPREF,JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),MODE,
3NTIMES(NFP3),RHOCON
COMMON/CNTL/LSTOP,LSTEADY,LScheme
COMMON/SORC/SMAX,SSUM
COMMON/COEF/FLOW,DIFF,ACOF
COMMON/UNSTEDY/FOLD(ID,JD,NFD),FNEW(ID,JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
COMMON/EXPNT/ALAMBDA(ID,JD),FP(ID,JD)
C*****1-exponential scheme, 2-implicit, 3-Crank-Nicolson
GO TO (1,2,3)LScheme
C*****EXPONENTIAL SCHEME
1 DO 10 I=2,L2
DO 10 J=2,M2
IF(ALAMBDA(I,J).GE.0..AND.ALAMBDA(I,J).LT.10.)THEN
FP(I,J)=1.-(1.-(1.-.1*ALAMBDA(I,J))**5)/ALAMBDA(I,J)
END IF

```



```

        IF (ALAMBDA(I,J) .GT. 10.) FP(I,J)=1.-1./ALAMBDA(I,J)
        FP(1,J)=1.0
        FP(L1,J)=1.0
        FP(I,1)=1.0
        FP(I,M1)=1.0
10    CONTINUE
        RETURN
C*****IMPLICIT SCHEME
      2 DO 20 I=1,L1
        DO 20 J=1,M1
          FP(I,J)=1.0
20    CONTINUE
        RETURN
C*****CRANK-NICOLSON SCHEME
      3 DO 30 I=1,L1
        DO 30 J=1,M1
          FP(I,J)=0.5
30    CONTINUE
        RETURN
      END

```


Appendix D

Numerical Code for Coupled Heat and Mass Transfer Through Hygroscopic Porous Textiles Integrated with a Human Thermal Physiology Control Model

```

C*****october 1995
      SUBROUTINE onerev
C*****This is 1-d skin model for human thermal control system
C*****which includes a double clothing layer
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE1*60
      CHARACTER DATAFILE2*60
      character datafile3*60
      CHARACTER MARKER*20
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY,IBUG
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1 AJP(JD),AJM(JD),AP(JD),
2 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3 YCVR(JD),YCVRS(JD)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/Flow,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(JD,NFD),FNEW(JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(JD),FP(JD)
      DIMENSION U(JD),V(JD),PC(JD),P(JD)
      EQUIVALENCE(F(1,1),U(1)),(F(1,2),V(1)),(F(1,3),PC(1))
      EQUIVALENCE(F(1,4),P(1))
C*****
C
C*****UNSTEADY DIFFUSION PROBLEM*****
      COMMON/DIFFLI/datafile,RHODS,RHOW,TKDS,TKW,TKV,
1TKA,CPDS,CPW,CPV,CPA,PATM,RGAS,XMW,XMA,CHTC,CMTC,
1PSATB,RHBND,TBND,PVBND,RHOVBND,PABND,RHOABND,iwrite,nprint,
1METHOD,EPSRHov,PSATBB,PSATBT,RHBNDB,RHBNDT,TBNDB,
1TBNDT,PVBNDB,PVBNDT,RHOVBNDB,RHOVBNDT,PABNDB,PABNDT,
1RHOABNDB,RHOABNDT,WEIGHTI,WEIGHT,RHINT,EBWSAT,
1datafile1,datafile2,datafile3,TINT,swtold,swteb,skebw,
1rf1,rf2,tau1,tau2,al1,al2,eds1,eds2,ebwint1,ebwint2,
1dsolid1,dsolid2
      COMMON/SKINMOD/TAIRAMB,ky1,ky2,ky3,basemet,xercise,xkso,
1rhocp,alphakp,alphakm,xgamk,alpham,sweat0,alphswt,xlmbwt,
1ybod,yskin,ymusc,ycore,yair,ycloth,mybod,mycore,myskin,
1mymusc,myair,mycloth,kycore,kymusc,kyskin,kyair,kycloth,
1skwork,dtb,tbody0,rhprev,tau,rf,al,dsolid,skmass,truesrc
      DIMENSION T(JD),RHOV(JD),EBW(JD),XMDOT(JD),
1EGAM(JD),RHOAIR(JD),PV(JD),PA(JD),PSAT(JD),TSKIN(JD),
1TBAVG(1),SWEAT(1),XMAVAI(1),XMFLUX(1),ELW(JD),
1XMDOTLW(JD),EDS(JD),tau(jd),rf(jd),al(jd),dsolid(jd),
1deffg(jd),tkeffg(jd),pgam(jd),rhov2(jd)
      EQUIVALENCE(F(1,5),rhov(1)),(F(1,6),T(1)),
1(F(1,7),EBW(1)),(F(1,8),XMDOT(1)),
1(F(1,9),EGAM(1)),(F(1,10),RHOAIR(1)),

```

```

1(F(1,11),PV(1)),(F(1,12),PA(1)),(F(1,13),PSAT(1)),
1(F(1,14),TSKIN(1)),(F(1,15),TBAVG(1)),(F(2,15),SWEAT(1)),
1(F(3,15),XMAVAI(1)),(F(4,15),XMFLUX(1)),(F(1,16),ELW(1)),
1(F(1,17),XMDOTLW(1)),(F(1,18),EDS(1)),(f(1,19),pgam(1))
DATA TITLE(5)/7H    vap/
DATA TITLE(6)/7H    TEMP/
DATA TITLE(7)/7H    EBW/
DATA TITLE(8)/7H    XMDOT/
DATA TITLE(9)/7H    EGAM/
DATA TITLE(10)/7H   RHOAIR/
DATA TITLE(11)/7H   PVAP/
DATA TITLE(12)/7H   PAIR/
DATA TITLE(13)/7H   PSAT/
DATA TITLE(14)/7H   TSKIN/
DATA TITLE(15)/7H   SKIN/
DATA TITLE(16)/7H   ELW/
DATA TITLE(17)/7H   XMDOTL/
DATA TITLE(18)/7H   EDS/
DATA TITLE(19)/7H   pgam/
DATA IBUG/.false./
C*****
ENTRY GRID
IF (IBUG) PRINT *, "ENTRY GRID"
C*****THICKNESS*****
C*****Body Thickness (m)
YCORE=.032d0
YMUSC=.016d0
YSKIN=.008d0
YBOD=YCORE+YMUSC+YSKIN
C*****Number of Nodes in Body
MYCORE=11
MYMUSC=10
MYSKIN=10
MYBOD=MYCORE+MYMUSC+MYSKIN
C*****Air Layer Thickness (m)
YAIR=1.d-4
c    yair=0.1*yair

C*****Number of Nodes in Air Layer
MYAIR=3
C*****Clothing Thickness (m)
YCLOTH=9.14D-4
c    ycloth=.25d0*ycloth
YCLOTH=.5*YCLOTH
yair=ycloth
C*****Number of Nodes in Clothing Layer
MYCLOTH=4
C*****Total thickness and number of nodes
YL=YBOD+YAIR+YCLOTH
M1=MYBOD+MYAIR+MYCLOTH
yv(2)=0.0d0
c*****CORE LAYER (nodes (1)/2 to 11)
dycore=ycore/DFLOAT(MYCORE-1)
KYcore=MYCORE

```

```

        do 10 j=3,kycore+1
        yv(j)=yv(j-1)+dycore
10 continue
c*****MUSCLE LAYER (nodes 12 to 21)
        dymusc=ymusc/DFLOAT(MYMUSC)
        KYmusc=KYCORE+MYMUSC
        do 11 j=kycore+2,kymusc+1
        yv(j)=yv(j-1)+dymusc
11 continue
c*****SKIN LAYER (nodes 22 to 31)
        dyskin=yskin/DFLOAT(MYSKIN-1)
        KYskin=KYMUSC+MYSKIN
        do 12 j=kymusc+2,kyskin-1
        yv(j)=yv(j-1)+dyskin
12 continue
        yv(kyskin)=yv(kyskin-1)+dyskin-.05*dyskin
        yv(kyskin+1)=yv(kyskin)+.05*dyskin
c*****AIR LAYER (nodes 32 to 35)
        dyair=yair/DFLOAT(MYAIR)
        KYair=KYSKIN+MYAIR
        do 13 j=kyskin+2,kyair+1
        yv(j)=yv(j-1)+dyair
13 continue
c*****CLOTHING LAYER (nodes 36 to 55/(56))
        dycloth=ycloth/DFLOAT(MYCLOTH-1)
        KYcloth=m1
        do 14 j=kyair+2,kycloth
        yv(j)=yv(j-1)+dycloth
14 continue

        RETURN
C*****
C
        ENTRY START
        IF (IBUG) PRINT *, "ENTRY START"
c*****input the data file name
        dt=1.D0
        nprint=int(60./dt)
        nprint=1000
        iwrite=nprint
        datafile1='5.sst'
        datafile2='noair.dyn'
        datafile3='5.ebw'
        marker='a'
c*****LScheme=1-exponential,2-implicit,3-Crank-Nicolson
        LSTEADY=.TRUE.
        lscheme=2
        LSOLVE(5)=.TRUE.
        LSOLVE(6)=.TRUE.
        lsolve(20)=.true.
        EPS(5)=1.D-3
        EPS(6)=1.D-3
        eps(20)=1.d-3
        JLAST=1000

```

```

        LAST=1440000
        relax(5)=.1d0
        relax(6)=.1d0
        relax(5)=1.
        relax(6)=1.
        relax(17)=.1d0
        relax(17)=1.d0
        RELAX(8)=1.d0
        relax(5)=.5
        relax(6)=1.
c   relax (9) is for fiber source, relax(8)is skin source
c*****PROBLEM CONSTANTS
c*****INITIAL EXERCISE LEVEL (W/m^2)
c*****Typical values are:**
c   sleeping = 0.
c   walking
c   .89 m/s = 75.
c   1.34 m/s= 110.
c   1.79 m/s=180.
c   Office Work = 16. to 41.
c   Light Work = 50. to 98.
c   Heavy Work = 98. to 235.
        skwork=50.D0
c*****SKIN CONSTANTS
c*****INITIAL CONDITIONS FOR BODY
c*****read in equilibrium values for 28 degrees C
        open(13,file='prof28.dat')
        do 113 j=1,kyskin
            read(13,*)t(j)
        113 continue
        close(13)
c*****These three quantities not used in this version
        sweat(1)=0.
        xmavai(1)=0.
        xmflux(1)=0.
c*****CONVECTIVE HEAT TRANSFER COEFFICIENTS AT BOUNDARY
        CHTC=16.d0
c*****CONVECTIVE MASS TRANSFER COEFFICIENTS AT BOUNDARY
c*****CHTC=(RHO*CP)*CMTC
        CMTC=.013d0
c*****SOLID PHASE DIFFUSION COEFFICIENT OF WATER
        DSOLID1=4.D-13
        DSOLID2=4.D-13
c*****EFFECTIVE FIBER DIAMETER
        aL1=3.6D-6
        aL2=3.6D-6
c*****MASS FRACTION OF WATER TO DRY SOLID AT 65% r.h.
        RF1=0.06d0
        RF2=0.00d0
c*****TORTUOSITY
        TAU1=1.d0
        TAU2=2.9d0
        tau1=2.9d0
c*****VOLUME FRACTION OF DRY SOLID

```

```

eds1=0.001D0
eds1=0.3D0
eds2=0.3D0
c*****DENSITIES
RHODS=1500.d0
RHOW=1000.d0
c*****THERMAL CONDUCTIVITIES
TKDS=0.16d0
TKW=0.6d0
TKV=0.0246d0
TKA=0.02563d0
c*****HEAT CAPACITIES
CPDS=1200.d0
CPW=4182.d0
CPV=1862.d0
CPA=1003.d0
c*****TOTAL GAS PRESSURE (ATMOSPHERIC)
PATM=101325.d0
c*****UNIVERSAL GAS CONSTANT
RGAS=8314.5d0
c*****MOLECULAR WEIGHTS
XMW=18.015d0
XMA=28.97d0
c*****INITIAL CONDITIONS
TINTC=35.0d0
TINT=TINTC+273.15d0
RHINT=.1d0
rhprev=rhint
EBWINT1=(0.578d0*RF1*EDS1*RHODS/RHOW)
1*(RHINT)*((1.d0/(.321d0+(RHINT))))
1+(1.d0/(1.262d0-(RHINT))))
EBWINT2=(0.578d0*RF2*EDS2*RHODS/RHOW)
1*(RHINT)*((1.d0/(.321d0+(RHINT))))
1+(1.d0/(1.262d0-(RHINT))))
PSATI=614.3d0*DEXP(17.06d0*((TINT-273.15d0)
1/(TINT-40.25d0)))
PVINT=RHINT*PSATI
RHOVINT=PVINT*XMW/(RGAS*TINT)
c*****BOUNDARY CONDITIONS
c*****TOP OF SLAB
RHBNDT=RHINT
TBNDT=TINT
PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
1/(TBNDT-40.25d0)))
PVBNDT=PSATBT*RHBNDT
PABNDT=PATM-PVBNDT
RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
RHOVBNDT=PVBNDT*XMW/(RGAS*TBNDT)
c initial conditions for the body
rhbod=1.d0
do 114 j=1,mybod
PSAT(J)=614.3d0*DEXP(17.06d0*((T(j)-273.15d0)
1/(T(j)-40.25d0)))
XMDOT(J)=0.d0

```



```

EBW(J)=1.D0
EGAM(J)=0.D0
ELW(J)=0.D0
EDS(J)=0.D0
XMDOTLW(J)=0.D0
PV(J)=PSAT(j)*rhbod
RHOV(J)=pv(j)*xmw/(rgas*t(j))
PA(J)=patm-pv(j)
RHOAIR(J)=pa(j)*xma/(rgas*t(j))
pgam(j)=pa(j)+pv(j)
114 continue

```

c*****INITIAL CONDITIONS FOR AIR LAYERS

```

DO 101 J=kyskin+1,kyair
T(J)=TINT
tau(j)=tau1
dsolid(j)=dsolid1
al(j)=al1
rf(j)=rf1
XMDOT(J)=0.d0
xmdotlw(j)=0.d0
EBW(J)=EBWINT1
EDS(J)=EDS1
ELW(J)=0.D0
EGAM(J)=1.d0-EDS(J)-EBW(J)-ELW(J)
XMDOTLW(J)=0.D0
PV(J)=PVINT
RHOV(J)=RHOVINT
PA(J)=PATM-PV(J)
RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
pgam(j)= pa(j)+pv(j)
PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)
1-273.15d0)/(T(J)-40.25d0)))

```

101 CONTINUE

c*****INITIAL CONDITIONS FOR CLOTHING LAYERS

```

DO 102 J=kyair+1,M1
T(J)=TINT
tau(j)=tau2
dsolid(j)=dsolid2
al(j)=al2
rf(j)=rf2
XMDOT(J)=0.d0
xmdotlw(j)=0.d0
EBW(J)=EBWINT2
EDS(J)=EDS2
ELW(J)=0.D0
EGAM(J)=1.d0-EDS(J)-EBW(J)-ELW(J)
XMDOTLW(J)=0.D0
PV(J)=PVINT
RHOV(J)=RHOVINT
PA(J)=PATM-PV(J)
RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)

```

```

1-273.15d0)/(T(J)-40.25d0)))
pgam(j)=pa(j)+pv(j)
IF(J.EQ.M1) THEN
T(J)=TBNDT
PV(J)=PVBNDT
RHOV(J)=RHOVBNDT
PA(J)=PABNDT
RHOAIR(J)=RHOABNDT
PSAT(J)=PSATBT
pgam(j)=pa(j)+pv(j)
END IF
102 CONTINUE
LPRINT(5)=.TRUE.
LPRINT(6)=.TRUE.
LPRINT(7)=.TRUE.
LPRINT(8)=.TRUE.
DO 103 J=1,M1
DO 103 NF=1,NFD
FOLD(J,NF)=F(J,NF)
103 CONTINUE
C*****CALCULATE INITIAL WEIGHT (MASS/SQ. METER)
WEIGHTI=0.0
DO 104 J=kyskin+1,M2
RHOAVG=RHOW*EBW(J)+RHODS*EDS(J)+
1(RHOV(J)+RHOAIR(J))*EGAM(J)
VOLUME=YCV(J)
WEIGHTI=WEIGHTI+VOLUME*RHOAVG
104 CONTINUE
c*****record run variablesrfrn/RF
open(10,access='append',file=datfile1)
write(10,*)"dt,eps(5),eps(6)"
write(10,*)"relax(5),relax(6),scheme"
write(10,*)marker,dt,eps(5),eps(6)
write(10,*)marker,relax(5),relax(6),lscheme
write(10,*)"grid,M1",M1
write(10,*)marker,M1
write(10,*)"method,Thickness,Rf"
write(10,*)"Dsolid,Initial Weight"
write(10,*)marker,METHOD,YL,RF
write(10,*)marker,DSOLID,WEIGHTI
write(10,*)"rhint,rhbndt,rhbndb"
write(10,*)marker,rhint,rhbndt,rhbndb
write(10,*)"time,t,rhov,ebw,xmdot,weight,% change"
write(10,*)"//nc"
write(10,*)"time,tavg,trect,tskin,tfabric,dtb"
close(10)
open(11,access='append',file=datfile2)
write(11,*)"temp,r.h.,xmdot"
write(11,*)"//nc"
close(11)
open(25,access='append',file=datfile3)
write(25,*)"time,sweat,xmavai(1),xmdot(kyskin),ebw,rhsk"
close(25)
RETURN

```

```

C*****
C
    ENTRY UNSTDY
C*****FOLD=FNEW
    IF (IBUG) PRINT *, "ENTRY UNSTEADY"
    DO 200 J=1,M1
    DO 200 NF=1,NFD
    FOLD(J,NF)=F(J,NF)
    200 CONTINUE

    RETURN

C*****
C
    ENTRY DENSE
c    call thermo
c    call pressure
c    call skin
c    call source

    IF (IBUG) WRITE (*,*) "ENTRY DENSE"
    RETURN

C*****
C
    ENTRY BOUND
    IF(IBUG)print *, 'entry bound'
    t(1)=t(2)
c    call pressure
    RETURN

C*****
C
    entry output1
c    print *, 'f(j,15),fold(j,15),f(j,16),fold(j,16)'
    do 1884 j=1,m1
c    print *,f(j,15),fold(j,15),f(j,16),fold(j,16)
    1884 continue
    return

    ENTRY OUTPUT
    if(ibug)print *, 'ENTRY OUTPUT'
    call outtoo
c    difmax=0.0
c    if(nf.eq.5)then
c    jstrt=kyskin+1
c    jends=m2
c    end if
c    if(nf.eq.6)then
c    jstrt=2
c    jends=m2
c    end if
c    if(nf.eq.5.or.nf.eq.6)then

```

```

cd      do 731 j=jstrt,jends
c      deltat=dabs((f(j,nf)-fnew(j))/(f(j,nf)+1.d-50))
c      difmax=dmax1(difmax,deltat)
c 731 continue
c      if(difmax.le.eps(nf)) iflag(nf)=1
c      if(jiter.gt.200)then
c      print *,jiter,nf
c      iflag(nf)=1
c      print *,'did it'
c      end if
c      if(ibug)print *,nf
c      RETURN

      ENTRY OUTPUT2
      RETURN

      ENTRY OUTPUT3
      RETURN
C*****
C
      ENTRY GAMSOR
      call thermo
c      call skin
c      call source

      IF (IBUG) print *, 'entry gamsor'
      GO TO (699,699,699,699,610,600) NF
c      if(nf.eq.20)goto 610
      go to 699

600 CONTINUE

C*****SKIN CALCULATIONS
C*****CALCULATE AVERAGE BODY TEMPERATURE AND DTB

C*****use rectangles to integrate temperature
      tbody0=308.9d0
      ttskin=0.d0
      DO 670 J=2,kyskin
      ttskin=ttskin+t(j)*ycv(j)
670 CONTINUE
      TBAVG(1)=(1.d0/YBOD)*ttskin
      DTB=TBAVG(1)-tbody0
      DIFFTB=(TBAVG(1)-FOLD(1,15))/DT

C*****CONSTANTS
C*****basal metabolic rate (W/m^3)
      BASEMET=1339.d0
C*****exercise muscular heat generation (W/m^2)
      XERCISE=skwork/YMUSC
C*****reference skin conductivity (W/m-K)
      XKSO=.498d0
C*****density*heat capacity (J/m^3-K)

```

```

        RHOCp=4.1868D6
c*****thermal proportional control coefficients (1/K)
        ALPHAKP= .729d0
        ALPHAKM= .326d0
c*****thermal rate control coefficients (sec/K)
        XGAMK=1106.d0
c*****shivering energy generation constant (1/K)
        alpham=0.d0
c*****reference sweat generation (kg/m^2-sec)
        sweat0=2.80D-6
c*****sweating proportional control constant (kg/m^2-sec-K)
        alphswt=6.98D-6
c*****sweating rate control coefficient (kg/m^2-sec-K^4)
        xlmbswt=2.69D-5

        do 664 j=1,kyskin
        rho(j)=rhocp
        if(dtb.gt.0.d0)then
        gam(j)= xkso*(1.d0+alphakp*dtb+xgamk*difftb)
        xkplim=1.511d0*xkso
        if(gam(j).ge.xkplim)then
        gam(j)=xkplim
        end if
        end if
        if(dtb.le.0.d0)then
        gam(j)= xkso*(1.d0+alphakm*dtb+xgamk*difftb)
        xkmlim=.675d0*xkso
        if(gam(j).le.xkmlim)then
        gam(j)=xkmlim
        end if
        end if
        664 continue
c*****insulated condition at body center for symmetry
        gam(1)=0.0d0
        t(1)=t(2)

c*****body core
        do 661 j1=1,kycore
        con(j1)=BASEMET
        ap(j1)=0.0d0
        661 continue

c*****muscle layer
        do 662 j2=kycore+1,kymusc
c*****shivering control coefficients (W/m^3-K) or (W/m^3-K^2)
c*****shivering heat generation is (W/M^3)
        ashiv1=324.1d0
        ashiv2=383.8d0
        ashiv3=966.9d0
        if(dtb.gt.0.d0)deltmet=XERCISE
        if(dtb.le.0.d0)then
        if(dtb.le.0.d0.and.dtb.ge.-1.0d0)then
        SHIVER=-ashiv1*dtb+ashiv2*dtb*dtb
        end if

```

```

        if(dtb.lt.-1.0d0)then
          SHIVER=660.256d0-ashiv3*(dtb+1.0d0)
        end if
        deltmel=SHIVER+XERCISE
      end if
      con(j2)=deltmet
      ap(j2)=0.0d0
662 continue

c*****skin layer
      do 663 j=kymusc+1,kyskin
        con(j)=0.0d0
        ap(j)=0.0d0
        if(j.eq.kyskin)then
          sknevap=xmflux(1)
          jsk=kyskin+1
          dhvap=(2.792D6)-160.d0*fold(jsk,6)-3.43d0*fold(jsk,6)
          1*fold(jsk,6)
          xsource=-dhvap*sknevap
c          if(rhumid.ge.1.0d0)then
c            print *,rhumid
c            xsource=0.d0
c          end if
          if(xsource.ge.0.d0)then
            con(j)=xsource
            ap(j)=0.d0
          end if
          if(xsource.lt.0.d0)then
            con(j)=0.d0
            ap(j)=xsource/t(j)
          end if
        end if
c        print *,xsource
c        write(*,815)j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
663 continue

c*****Sweating Production and Rate
      if(dtb.ge.0.d0)then
        Sweat(1)=sweat0+alphswt*dtb+xlmbswt*((dtb)**4)
        swtlim=60.d0*sweat0
        if(Sweat(1).gt.swtlim)then
          Sweat(1)=swtlim
        end if
      end if
      if(dtb.lt.0.d0)then
        Sweat(1)=sweat0
      end if

      WEIGHT=0.0d0
      DO 602 J=kyskin+1,m1

```

```

C*****DENSITY AND HEAT CAPACITY FOR ENERGY EQUATION
  RHOAVG=RHOW*(fold(j,7)+fold(j,16))+RHODS*EDS(J)+
  1(fold(j,5)+fold(j,10))*fold(j,9)
  CPAVG=(RHOW*(fold(j,7)+fold(j,16))*CPW+RHODS*EDS(J)*CPDS+
  1(fold(j,5)*CPV+fold(j,10)*CPA)*fold(j,9))/RHOAVG
c    RHOAVG=RHOW*(ebw(j)+elw(j))+RHODS*EDS(J)+
c    1(rhov(j)+rhoair(j))*egam(j)
c    CPAVG=(RHOW*(ebw(j)+elw(j))*CPW+RHODS*EDS(J)*CPDS+
c    1(rhov(j)*CPV+rhoair(j)*CPA)*egam(j))/RHOAVG
  RHO(J)=RHOAVG*CPAVG
C*****CALCULATE CURRENT WEIGHT (MASS/SQ. METER)
C*****AND WEIGHT GAIN OR LOSS FROM INITIAL CONDITION
  VOLUME=YCV(J)
  WEIGHT=WEIGHT+VOLUME*RHOAVG
  WEIGHTG=WEIGHT-WEIGHTI
C*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
  TKF=((TKW*RHOW*(fold(j,7)+fold(j,16))+TKDS*RHODS*EDS(J))
  1/(RHOW*(fold(j,7)+fold(j,16))+RHODS*EDS(J)))
  TKG=((TKV*fold(j,5)+TKA*fold(j,10))/
  1(fold(j,5)+fold(j,10)))
c    TKF=((TKW*RHOW*(ebw(j)+elw(j))+TKDS*RHODS*EDS(J))
c    1/(RHOW*(ebw(j)+elw(j))+RHODS*EDS(J)))
c    TKG=((TKV*rhov(j)+TKA*rhoair(j))/
c    1(rhov(j)+rhoair(j)))
  VF=EDS(J)+ebw(j)+elw(j)
  vf=eds(j)+fold(j,7)+fold(j,16)
  TKPERP=TKF*TKG/(VF*TKG+(1.-VF)*TKF)
  TKPRLL=(1.d0-VF)*TKG+VF*TKF
  WOMEA=1.d0
  TKEFF=WOMEA*TKPRLL+(1.0d0-WOMEA)*TKPERP
c*****Cylinder Model
c    TKEFF=TKG*((TKF*(1.+VF)+TKG*(1.-VF)))/
c    1(TKF*(1.-VF)+TKG*(1.+VF)))
  GAM(J)=TKEFF
  DHVAP=(2.792D+6)-160.d0*fold(J,6)-3.43d0*fold(J,6)*fold(J,6)
  rhlim=0.95d0
  rhumid=fold(j,11)/fold(j,13)
  if(rhumid.gt.rhlim)then
    rhumid=rhlim
  end if
  QL=(1.95D+5)*(1.d0-rhumid)
  1*((1.d0/(0.2d0+(rhumid)))+
  1(1.d0/(1.05d0-(rhumid))))
c*****ALL SOURCE TERMS LUMPED INTO CON(J) (LAZY WAY)
  xhyg=-(dhvap+ql)*xmdot(j)
  xliq=-dhvap*xmdotlw(j)
  xsource=xhyg+xliq
  if(xsource.ge.0.d0)then
    con(j)=xsource
    ap(j)=0.d0
  end if
  if(xsource.lt.0.d0)then
    con(j)=0.d0
    ap(j)=xsource/t(j)

```

```

        end if
602 CONTINUE
c*****CREATE CONVECTIVE BOUNDARY CONDITIONS
c      GDYM=GAM(1)/YDIF(2)
      GDYP=GAM(M1)/YDIF(M1)
c      AREAM=1.0d0/((1.0d0/CHTC+1.0/GDYM)*YCV(2))
      AREAP=1.0d0/((1.0d0/CHTC+1.0d0/GDYP)*YCV(M2))
c      GAM(1)=0.0
      GAM(M1)=0.0d0
c      CON(2)=CON(2)+AREAM*TBNDB
c      AP(2)=AP(2)-AREAM
      CON(M2)=CON(M2)+AREAP*TBNDT
      AP(M2)=AP(M2)-AREAP
c      call output1
c      print *,nf
c      print *, 'j,gam,xmdot,xmdotlw,elw,ebw,egam'
c      do 1239 j=1,m1
c      print *,j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
c 1239 continue
      GO TO 699

610 CONTINUE
c      call thermo
c      call skin
      call source
c*****DO NOT INCLUDE BODY LAYER IN VAPOR SOLUTION
      do 609 j=1,kyskin
      gam(j)=0.0d0
      con(j)=0.0d0
      ap(j)=0.0d0
609 continue

      jstart=kyskin+1
      DO 611 J=jstart,M1
c*****DENSITY TERM FOR MASS DIFFUSION EQUATION
      RHO(J)=EGAM(J)
      rho(j)=fold(j,9)
c*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
      DAIR=(2.20D-5)*((fold(J,6)/273.15d0)**1.75)
      DEFF=DAIR*EGAM(J)/TAU(J)
      deff=dair*fold(j,9)/tau(j)
      GAM(J)=DEFF

      xsource=(xmdot(j)+xmdotlw(j))
c      xsource=(1.d0-f(j,5))*xsource+
c      1(1.d0-egam(j))*(rhov(j)+rhoair(j))*(f(j,5)-fold(j,5))/dt
      xsource=xsource-rhov(j)*(egam(j)-fold(j,9))
      if(j.eq.jstart)then
      xsource=xsource+xmflux(1)
      end if

c*****source-term dominated calculations
      pvvapor=psat(j)

```



```

rhoosat=pvapor*xmw/(rgas*t(j))
paair=pgam(j)-pvapor
rhoasat=paair*xma/(rgas*t(j))
xmsat=rhoosat/(rhoosat+rhoasat)
rhumid=pv(j)/psat(j)

IF(xsource.LT.0.d0) THEN
CON(J)=0.d0
AP(J)=xsource/(rhov(J))
if(elw(j).gt.0.d0)then
xmflim=0.99D0*xmsat
c   con(j)=xsource*xmflim/(xmflim-xmf(j))
c   ap(j)=-xsource/(xmflim-xmf(j))
c   con(j)=(1.D50)*xmflim
c   ap(j)=-1.D50
end if
jj1=kyskin+1
if(xnavai(1).gt.0.d0.and.j.eq.jj1)then
con(j)=xsource*rhoosat/(rhoosat-rhov(j))
ap(j)=-xsource/(rhoosat-rhov(j))
truesrc=con(j)+ap(j)*rhov(j)
end if
END IF
IF(xsource.GE.0.d0) THEN
CON(J)=xsource
AP(J)=0.0d0
c   if(elw(j).gt.0.d0)then
c   if(j.eq.kyskin+1.and.xnavai(1).gt.0.d0)then
xmflim=.99D0*xmsat
c   con(j)=xsource*xmflim/(xmflim-xmf(j))
c   ap(j)=-xsource/(xmflim-xmf(j))
c   con(j)=(1.D50)*xmflim
c   ap(j)=-1.D50
end if
jj1=kyskin+1
if(xnavai(1).gt.0.d0.and.j.eq.jj1)then
con(j)=xsource*rhoosat/(rhoosat-rhov(j))
ap(j)=-xsource/(rhoosat-rhov(j))
truesrc=con(j)+ap(j)*rhov(j)
end if

END IF

611 CONTINUE
c*****CREATE CONVECTIVE BOUNDARY CONDITIONS
c   GDYM=GAM(1)/YDIF(2)
GDYP=GAM(M1)/YDIF(M1)
c   AREAM=1.0/((1.0/CMTC+1.0/GDYM)*YCV(2))
AREAP=1.0d0/((1.0d0/CMTC+1.0d0/GDYP)*YCV(M2))
c   GAM(1)=0.0
GAM(M1)=0.0d0
c   CON(2)=CON(2)+AREAM*RHOVBND
c   AP(2)=AP(2)-AREAM
CON(M2)=CON(M2)+AREAP*RHOVBNDT

```

```

      AP(M2)=AP(M2)-AREAP
c      call output1
c      print *,nf
c      print *,'j,gam,xmdot,xmdotlw,elw,ebw,egam'
c      do 1279 j=1,m1
c      print *,j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
c 1279 continue
      GOTO 699

```

```

699 RETURN

```

```

C*****

```

```

C

```

```

      ENTRY SAVE1

```

```

      RETURN

```

```

C*****

```

```

C

```

```

      ENTRY OUTTOO

```

```

      LPRINT(5)=.TRUE.

```

```

      LPRINT(6)=.TRUE.

```

```

      time1=time/3600.

```

```

c      if(time1.lt.2.)goto 899

```

```

      if(time1.gt.8.)goto 9999

```

```

      if(time1.gt.3.)then

```

```

      dt=1.d-1

```

```

      relax(6)=.1d0

```

```

      relax(5)=.1d0

```

```

      relax(17)=.5d0

```

```

      relax(5)=.5

```

```

      relax(6)=1.

```

```

      relax(17)=1.

```

```

      nprint=1

```

```

      nprint=int(60./dt)

```

```

      nprint=10000

```

```

c      iwrite=nprint

```

```

      if(time1.gt.3.25)then

```

```

      dt=1.d-1

```

```

      nprint=10000

```

```

      end if

```

```

      skwork=50.D0

```

```

      rhlim=0.9d0

```

```

      tchange=600.d0

```

```

      rhchange=0.6d0

```

```

      delrhdt=rhchange/tchange

```

```

      rhprev=rhprev+delrhdt*dt

```

```

      if(rhprev.gt.rhlim)then

```

```

      rhprev=rhlim

```

```

      end if

```

```

      rhprev=0.9d0

```

```

      relax(17)=1.d0

```

```

      RHBNDT=rhprev

```

```

TINTC=35.0d0
TINT=TINTC+273.15d0
TBNDT=TINT
PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
1/(TBNDT-40.25d0)))
PVBNDT=PSATBT*RHBNDT
PABNDT=PATM-PVBNDT
RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
RHOVBNDT=PVBNDT*XMW/(RGAS*TBNDT)
rhov(m1)=rhovbndt
pv(m1)=pvbndt
psat(m1)=psatbt
pa(m1)=pabndt
rhoair(m1)=rhoabndt
t(m1)=tbndt
end if
if(time1.gt.4.d0)then
dt=1.d-1
nprint=10000
c    relax(5)=.5
c    relax(6)=.5
c    relax(8)=.1
c    relax(8)=1.
if(time1.gt.4.25d0)then
dt=1.d-1
c    nprint=int(dt)*60
c    relax(8)=1.
end if
skwork=50.D0
RHBNDT=0.10d0
TINTC=35.d0
TINT=TINTC+273.15d0
TBNDT=TINT
PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
1/(TBNDT-40.25d0)))
PVBNDT=PSATBT*RHBNDT
PABNDT=PATM-PVBNDT
RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
RHOVBNDT=PVBNDT*XMW/(RGAS*TBNDT)
rhov(m1)=rhovbndt
pv(m1)=pvbndt
psat(m1)=psatbt
pa(m1)=pabndt
rhoair(m1)=rhoabndt
pgam(m1)=pv(m1)+pa(m1)
t(m1)=tbndt
end if
if (iwrite.eq.nprint)then
open(10,access='append',file=datafile1)
temp1=tbavg(1)-273.15d0
temp2=t(1)-273.15d0
temp3=t(kyskin)-273.15d0
temp4=t(m2)-273.15d0
write(*,*)"time tbody tcore tskin tc1surf dtb ebwskin"

```

```

        write(*,817)time1,temp1,temp2,temp3,temp4,
        1dtb,ebw(kyskin+1)
        write(10,816)time1,temp1,temp2,temp3,temp4,
        1dtb
816 format(5e11.5,1x,e11.5)
817 format(7f11.5)
818 format(i10,3e11.5)
        close(10)
        open(25,access='append',file=datafile3)
        rhumid=pv(kyskin+1)/psat(kyskin+1)
        write(25,816)time1,sweat(1),xmavai(1),xmflux(1),
        1elw(kyskin+1),rhumid
        close(25)
c      write(*,*)"iter,temp(K),temp(C),dens,humid,xmdot"
        open(11,access='append',file=datafile2)
        write(*,*)"iter,sweat,mass left on skin"
        write(*,*)iter,sweat(1),xmavai(1)
        write(*,*)"original sweat,sweat evaporated,sweat left"
        xsleft=xmavai(1)
        swteb=xmflux(1)*dt*ycv(kyskin+1)
        swtbeg=fold(3,15)+sweat(1)*dt
        write(*,822)skmass,swteb,xsleft
        print *,'source used (total), truesrc'
        print *,xmflux(1),xmdot(kyskin),xmdotlw(kyskin),truesrc
c      write(*,*)"j,temp,elw,rhumid,xmflux(1),xmdotlw,egam"
822 format(3e15.4)

c      print *,'j,temp,rhov(j),rhumid,xmdot(j),ebw(j),xmlw(j),elw(j)'
        print *,'j,temp,rhov(j),rhumid,xmdot(j),ebw(j),pv(j),pgam(j)'
c      print *,'j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)'
        jend=m1
        jstrt=kyskin-1
        do 897 j=jstrt,jend
            temp=t(j)-273.15
            rhumid=pv(j)/(psat(j)+1.D-50)
            flux=xmdot(j)
            if(j.eq.kyskin+1)then
                flux=xmflux(1)
            end if
c          write(*,815)j,temp,rhov(j),rhumid,flux,ebw(j)
c          1,xmdotlw(j),elw(j)
            write(*,815)j,temp,rhov(j),rhumid,flux,ebw(j)
            1,pv(j),pgam(j)
c          write(11,818)j,temp,rhumid,xmdot(j)
897 continue
c          print *,'j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)'
            do 898 j=kyskin-1,m1
                temp=t(j)-273.15
                rhumid=pv(j)/(psat(j)+1.D-50)
                if(j.eq.kyskin+1)then
c                  write(*,815)j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)
                    end if
c                  write(*,815)j,egam(j),ebw(j),elw(j),xmdot(j),xmdotlw(j),xmf(j)
c                  write(11,818)j,temp,rhumid,xmdot(j)

```

```

898 continue
      close(11)
815 format(1x,i7,1f9.3,e9.3,5e9.3)
      iwrite=0
      endif
      iwrite=iwrite+1
811 format(1e10.3,1x,e10.5,1x,e10.5,4e10.3)
812 format (5e11.4)
899 RETURN

C*****
C
      ENTRY SAVEPC

      RETURN

C*****
C
      ENTRY SAVE2

      RETURN

      entry THERMO
c      if(ibug)print *, 'entry thermo'
      do 1021 j=kyskin,m1
c      xmfr=xmf(j)/(1.d0-xmf(j))
c      rhov(j)=(xmfr*(patm*xma/(rgas*t(j))))
c      1/(1.d0+xmfr*(xma/xmw))

      PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)-273.15d0)/
1(T(J)-40.25d0)))
      PV(J)=RHOV(J)*RGAS*T(J)/XMW
      PA(J)=PATM-PV(J)
      pgam(j)=pa(j)+pv(j)
      RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
1021 continue
      return

      entry pressure
      do 1026 j=kyskin,m1
      Pgam(J)=PATM
      PV(J)=RHOV(J)*RGAS*T(J)/XMW
      pa(j)=pgam(j)-pv(j)
      RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
1026 continue
      return

      entry source
      if(ibug)print *, 'entry source'

c*****properties for the two layers
      do 1932 j=kyskin+1,kyair

```

```

        tau(j)=tau1
        eds(j)=eds1
        rf(j)=rf1
        al(j)=al1
        dsolid(j)=dsolid1
1932 continue
        do 1933 j=kyair+1,m2
            tau(j)=tau2
            eds(j)=eds2
            rf(j)=rf2
            al(j)=al2
            dsolid(j)=dsolid2
1933 continue

```

C*****FIBER SOLID-STATE SOURCE TERM

```

        DO 1031 J=kyskin+1,M2
        RHUMID=PV(J)/PSAT(J)
        rhlim=0.95D0
        if(rhumid.gt.rhlim)then
            rhumid=rhlim
        end if
        EBW(J)=(.578d0*RF(J)*EDS(J)*RHODS/RHOW)*RHUMID*
1(1.d0/(.321d0+RHUMID)+1.d0/(1.262d0-RHUMID))
        XMDOT(J)=(FOLD(J,7)-EBW(J))*RHOW/DT
1031 continue

        do 1032 j=kyskin+1,m2
        CSKIN=RHOW*EBW(J)/(EBW(J)+EDS(J))
        CINSIDE=RHOW*FOLD(J,7)/(FOLD(J,7)+EDS(J))
        REQ=EBW(J)*EDS(J)*RHODS/RHOW
        RINST=FOLD(J,7)*EDS(J)*RHODS/RHOW
        XMDOT(J)=DSOLID(J)*RHODS*(RINST-REQ)/(aL(j)*aL(j))
        xmdot(j)=relax(8)*xmdot(j)+(1.d0-relax(8))*fold(j,8)

        EBW(J)=FOLD(J,7)-XMDOT(J)*DT/RHOW

1032 CONTINUE

        DO 1042 J=kyskin+1,m2
c        goto 1742
c        xmdot(j)=0.d0
c        xmdotlw(j)=0.d0
c        if(j.eq.kyskin)goto 1742
        pvvapor=psat(j)
        rhovsat=ppvapor*xmw/(rgas*t(j))
        if(rhov(j).gt.rhovsat)THEN
            fog=elw(j)
            xmdotlw(j)=fold(j,9)*(rhovsat-rhov(j))/dt
            xmdotlw(j)=egam(j)*(rhovsat-rhov(j))/dt
            xmdotlw(j)=relax(17)*xmdotlw(j)+(1.d0-relax(17))*fold(j,17)
            elw(j)=fog-(xmdotlw(j)*dt/rhow)
        end if

```

```

        if(rhov(j).lt.rhovsat.and.ELW(J).GT.0.D0) THEN
            fog=elw(j)
            xmdotlw(j)=egam(j)*(rhovsat-rhov(j))/dt
            xmdotlw(j)=relax(17)*xmdotlw(j)+(1.d0-relax(17))*fold(j,17)
            elw(j)=fog-(xmdotlw(j)*dt/rhow)
            if(elw(j).lt.0.d0) then
                elw(j)=0.d0
            xmdotlw(j)=fog*rhow/dt
            end if
        end if

1742  egam(j)=1.d0-eds(j)-ebw(j)-elw(j)

        jswitch=kyskin+1
        if(j.eq.jswitch) then
c*****use saturation properties at skin surface temperature
            pvvapor=psat(j-1)
            rhovsat=pvvapor*xmw/(rgas*t(j-1))
c*****use saturation properties at 1st interface temperature
            pvvapor=psat(j)
            rhovsat=pvvapor*xmw/(rgas*t(j))
            xmflux(1)=0.d0
            xmold=fold(3,15)
            xmswt=sweat(1)*dt
            skmass=xmold+xmswt
c*****use vapor properties at skin temperature
            DAIR=(2.20d-5)*((fold(j,6)/273.16d0)**1.75d0)
            deff=dair*fold(j,9)/tau(j)
            dforce=rhovsat-rhov(j)
            if(dforce.gt.0.d0.and.elw(j).eq.0.d0) then
                xmflux(1)=(1.d0/ycv(j))*deff*(rhovsat-rhov(j))/(ycv(j)/2.d0)
c            xmflux(1)=egam(j)*dforce/dt
                xmaxflx=(skmass/dt)*(1.d0/ycv(j))
                xmflux(1)=dmin1(xmflux(1),xmaxflx)
                xmleft=skmass-(xmflux(1)*dt*ycv(j))
                xmavai(1)=xmleft
            end if
c        end if
            if(dforce.le.0.d0) then
                xmflux(1)=0.d0
                xmavai(1)=skmass
            end if
        end if
1042  continue
        return

9999  stop
        end

```

```

C*****USER routine work.f
C***** (changes in work rate at constant environmental conditions)
C*****in HP-700 directory ../skin
C*****october 1995
      SUBROUTINE onerev
C*****This is 1-d skin model for human thermal control system
C*****which includes a double clothing layer
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C*****
      CHARACTER TITLE*18
      CHARACTER DATAFILE1*60
      CHARACTER DATAFILE2*60
      character datafile3*60
      CHARACTER MARKER*20
      LOGICAL LSOLVE,LPRINT,LBLK,LSTOP,LSTEADY,IBUG
      PARAMETER (JD=100,NFD=20,NFP3=23,MIJ=100)
      COMMON F(JD,NFD),RHO(JD),GAM(JD),CON(JD),
1 AJP(JD),AJM(JD),AP(JD),
2 Y(JD),YV(JD),YDIF(JD),YCV(JD),YCVS(JD),
3 YCVR(JD),YCVRS(JD)
      COMMON/INDX/NF,NFMIN,NFMAX,NP,NRHO,NGAM,M1,M2,M3,
1JST,ITER,JITER,LAST,TITLE(NFP3),RELAX(NFP3),TIME,DT,YL,
2JPREF,LSOLVE(NFP3),LPRINT(NFP3),LBLK(NFP3),RHOCON
      COMMON/CNTL/LSTOP,LSTEADY,LScheme
      COMMON/SORC/SMAX,SSUM
      COMMON/COEF/FLOW,DIFF,ACOF
      COMMON/UNSTEDY/FOLD(JD,NFD),FNEW(JD),JLAST,
1 IFLAG(NFD),EPS(NFD),DIFMAX,JFLAG,deltat
      COMMON/EXPNT/ALAMBDA(JD),FP(JD)
      DIMENSION U(JD),V(JD),PC(JD),P(JD)
      EQUIVALENCE(F(1,1),U(1)),(F(1,2),V(1)),(F(1,3),PC(1))
      EQUIVALENCE(F(1,4),P(1))
C*****
C
C*****UNSTEADY DIFFUSION PROBLEM*****
      COMMON/DIFFLI/datafile,RHODS,RHOW,TKDS,TKW,TKV,
1TKA,CPDS,CPW,CPV,CPA,PATM,RGAS,XMW,XMA,CHTC,CMTC,
1PSATB,RHBND,TBND,PVBND,RHOVBND,PABND,RHOABND,iwrite,nprint,
1METHOD,EPSRHOF,PSATBB,PSATBT,RHBNDB,RHBNDT,TBNDB,
1TBNDT,PVBNDB,PVBNDT,RHOVBNDT,RHOVBNDT,PABNDB,PABNDT,
1RHOABNDB,RHOABNDT,WEIGHTI,WEIGHT,RHINT,EBWSAT,
1datafile1,datafile2,datafile3,TINT,swtold,swteb,skebw,
1rf1,rf2,tau1,tau2,al1,al2,eds1,eds2,ebwint1,ebwint2,
1dsolid1,dsolid2
      COMMON/SKINMOD/TAIRAMB,ky1,ky2,ky3,basemet,xercise,xkso,
1rhocp,alphakp,alphakm,xgamk,alpham,sweat0,alphswt,xlmbwt,
1lybod,yskin,ymusc,ycore,yair,ycloth,mybod,mycore,myskin,
1mymusc,myair,mycloth,kycore,kymusc,kyskin,kyair,kycloth,
1skwork,dtb,tbody0,rhprev,tau,rf,al,dsolid,skmass,truerc
      DIMENSION T(JD),RHOV(JD),EBW(JD),XMDOT(JD),
1EGAM(JD),RHOAIR(JD),PV(JD),PA(JD),PSAT(JD),TSKIN(JD),
1TBVG(1),SWEAT(1),XMAVAI(1),XMFLUX(1),ELW(JD),
1XMDOTLW(JD),EDS(JD),tau(jd),rf(jd),al(jd),dsolid(jd),
1deffg(jd),tkeffg(jd),pgam(jd),rhov2(jd)

```



```

EQUIVALENCE(F(1,5),rhov(1)),(F(1,6),T(1)),
1(F(1,7),EBW(1)),(F(1,8),XMDOT(1)),
1(F(1,9),EGAM(1)),(F(1,10),RHOAIR(1)),
1(F(1,11),PV(1)),(F(1,12),PA(1)),(F(1,13),PSAT(1)),
1(F(1,14),TSKIN(1)),(F(1,15),TBAVG(1)),(F(2,15),SWEAT(1)),
1(F(3,15),XMAVAI(1)),(F(4,15),XMFLUX(1)),(F(1,16),ELW(1)),
1(F(1,17),XMDOTLW(1)),(F(1,18),EDS(1)),(f(1,19),pgam(1))
DATA TITLE(5)/7H    vap/
DATA TITLE(6)/7H    TEMP/
DATA TITLE(7)/7H    EBW/
DATA TITLE(8)/7H    XMDOT/
DATA TITLE(9)/7H    EGAM/
DATA TITLE(10)/7H  RHOAIR/
DATA TITLE(11)/7H   PVAP/
DATA TITLE(12)/7H   PAIR/
DATA TITLE(13)/7H   PSAT/
DATA TITLE(14)/7H   TSKIN/
DATA TITLE(15)/7H   SKIN/
DATA TITLE(16)/7H   ELW/
DATA TITLE(17)/7H  XMDOTL/
DATA TITLE(18)/7H   EDS/
DATA TITLE(19)/7H   pgam/
DATA IBUG/.false./
C*****
ENTRY GRID
IF (IBUG) PRINT *, "ENTRY GRID"
C*****THICKNESS*****
C*****Body Thickness (m)
YCORE=.032d0
YMUSC=.016d0
YSKIN=.008d0
YBOD=YCORE+YMUSC+YSKIN
C*****Number of Nodes in Body
MYCORE=11
MYMUSC=10
MYSKIN=10
MYBOD=MYCORE+MYMUSC+MYSKIN
C*****Air Layer Thickness (m)
YAIR=1.d-4
c    yair=0.1*yair

C*****Number of Nodes in Air Layer
MYAIR=3
C*****Clothing Thickness (m)
YCLOTH=9.14D-4
c    ycloth=.25d0*ycloth
YCLOTH=.5*YCLOTH
yair=ycloth
C*****Number of Nodes in Clothing Layer
MYCLOTH=4
C*****Total thickness and number of nodes
YL=YBOD+YAIR+YCLOTH
M1=MYBOD+MYAIR+MYCLOTH
yv(2)=0.0d0

```

```

c*****CORE LAYER (nodes (1)/2 to 11)
    dycore=ycore/DFLOAT(MYCORE-1)
    KYcore=MYCORE
    do 10 j=3,kycore+1
        yv(j)=yv(j-1)+dycore
    10 continue
c*****MUSCLE LAYER (nodes 12 to 21)
    dymusc=ymusc/DFLOAT(MYMUSC)
    KYmusc=KYCORE+MYMUSC
    do 11 j=kycore+2,kymusc+1
        yv(j)=yv(j-1)+dymusc
    11 continue
c*****SKIN LAYER (nodes 22 to 31)
    dyskin=yskin/DFLOAT(MYSKIN-1)
    KYskin=KYMUSC+MYSKIN
    do 12 j=kymusc+2,kyskin-1
        yv(j)=yv(j-1)+dyskin
    12 continue
    yv(kyskin)=yv(kyskin-1)+dyskin-.05*dyskin
    yv(kyskin+1)=yv(kyskin)+.05*dyskin
c*****AIR LAYER (nodes 32 to 35)
    dyair=yair/DFLOAT(MYAIR)
    KYair=KYSKIN+MYAIR
    do 13 j=kyskin+2,kyair+1
        yv(j)=yv(j-1)+dyair
    13 continue
c*****CLOTHING LAYER (nodes 36 to 55/(56))
    dycloth=ycloth/DFLOAT(MYCLOTH-1)
    KYcloth=m1
    do 14 j=kyair+2,kycloth
        yv(j)=yv(j-1)+dycloth
    14 continue

    RETURN
C*****
C
    ENTRY START
    IF (IBUG) PRINT *, "ENTRY START"
c*****input the data file name
    dt=1.D0
    nprint=int(60./dt)
    nprint=1000
    iwrite=nprint
    datafile1='worknh2.sst'
    datafile2='noair.dyn'
    datafile3='worknh2.ebw'
    marker='a'
c*****LScheme=1-exponential,2-implicit,3-Crank-Nicolson
    LSTEADY=.TRUE.
    lscheme=2
    LSOLVE(5)=.TRUE.
    LSOLVE(6)=.TRUE.
    lsolve(20)=.true.
    EPS(5)=1.D-3

```

```

      EPS(6)=1.D-3
      eps(20)=1.d-3
      JLAST=1000
      LAST=1440000
      relax(5)=.1d0
      relax(6)=.1d0
      relax(5)=1.
      relax(6)=1.
      relax(17)=.1d0
      relax(17)=1.d0
      RELAX(8)=1.d0
      relax(5)=.5
      relax(6)=1.
c   relax (9) is for fiber source, relax(8)is skin source
c*****PROBLEM CONSTANTS
c*****INITIAL EXERCISE LEVEL (W/m^2)
c*****Typical values are:**
c   sleeping = 0.
c   walking
c   .89 m/s = 75.
c   1.34 m/s= 110.
c   1.79 m/s=180.
c   Office Work = 16. to 41.
c   Light Work = 50. to 98.
c   Heavy Work = 98. to 235.
      skwork=20.D0
c*****SKIN CONSTANTS
c*****INITIAL CONDITIONS FOR BODY
c*****read in equilibrium values for 28 degrees C
      open(13,file='prof28.dat')
      do 113 j=1,kyskin
        read(13,*)t(j)
      113 continue
      close(13)
c*****These three quantities not used in this version
      sweat(1)=0.
      xmavai(1)=0.
      xmflux(1)=0.
c*****CONVECTIVE HEAT TRANSFER COEFFICIENTS AT BOUNDARY
      CHTC=16.d0
c*****CONVECTIVE MASS TRANSFER COEFFICIENTS AT BOUNDARY
c*****CHTC=(RHO*CP)*CMTC
      CMTC=.013d0
c*****SOLID PHASE DIFFUSION COEFFICIENT OF WATER
      DSOLID1=4.D-13
      DSOLID2=4.D-13
c*****EFFECTIVE FIBER DIAMETER
      aL1=3.6D-6
      aL2=3.6D-6
c*****MASS FRACTION OF WATER TO DRY SOLID AT 65% r.h.
      RF1=0.00d0
      RF2=0.00d0
c*****TORTUOSITY
      TAU1=1.d0

```

```

        TAU2=2.9d0
        tau1=2.9d0
c*****VOLUME FRACTION OF DRY SOLID
        eds1=0.001D0
        eds1=0.3D0
        eds2=0.3D0
c*****DENSITIES
        RHODS=1500.d0
        RHOW=1000.d0
c*****THERMAL CONDUCTIVITIES
        TKDS=0.16d0
        TKW=0.6d0
        TKV=0.0246d0
        TKA=0.02563d0
c*****HEAT CAPACITIES
        CPDS=1200.d0
        CPW=4182.d0
        CPV=1862.d0
        CPA=1003.d0
c*****TOTAL GAS PRESSURE (ATMOSPHERIC)
        PATM=101325.d0
c*****UNIVERSAL GAS CONSTANT
        RGAS=8314.5d0
c*****MOLECULAR WEIGHTS
        XMW=18.015d0
        XMA=28.97d0
c*****INITIAL CONDITIONS
        TINTC=30.0d0
        TINT=TINTC+273.15d0
        RHINT=.650d0
        rhprev=rhint
        EBWINT1=(0.578d0*RF1*EDS1*RHODS/RHOW)
        1*(RHINT)*((1.d0/ (.321d0+(RHINT))))
        1+(1.d0/(1.262d0-(RHINT))))
        EBWINT2=(0.578d0*RF2*EDS2*RHODS/RHOW)
        1*(RHINT)*((1.d0/ (.321d0+(RHINT))))
        1+(1.d0/(1.262d0-(RHINT))))
        PSATI=614.3d0*DEXP(17.06d0*((TINT-273.15d0)
        1/(TINT-40.25d0)))
        PVINT=RHINT*PSATI
        RHOVINT=PVINT*XMW/(RGAS*TINT)
c*****BOUNDARY CONDITIONS
c*****TOP OF SLAB
        RHBNDT=RHINT
        TBNDT=TINT
        PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
        1/(TBNDT-40.25d0)))
        PVBNDT=PSATBT*RHBNDT
        PABNDT=PATM-PVBNDT
        RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
        RHOVBNDT=PVBNBT*XMW/(RGAS*TBNDT)
c        initial conditions for the body
        rhbod=1.d0
        do 114 j=1,mybod

```

```

    PSAT(J)=614.3d0*DEXP(17.06d0*((T(j)-273.15d0)
1/(T(j)-40.25d0)))
    XMDOT(J)=0.d0
    EBW(J)=1.D0
    EGAM(J)=0.D0
    ELW(J)=0.D0
    EDS(J)=0.D0
    XMDOTLW(J)=0.D0
    PV(J)=PSAT(j)*rhubd
    RHOV(J)=pv(j)*xmw/(rgas*t(j))
    PA(J)=patm-pv(j)
    RHOAIR(J)=pa(j)*xma/(rgas*t(j))
    pgam(j)=pa(j)+pv(j)
114 continue

```

```

c*****INITIAL CONDITIONS FOR AIR LAYERS
DO 101 J=kyskin+1,kyair
    T(J)=TINT
    tau(j)=tau1
    dsolid(j)=dsolid1
    al(j)=al1
    rf(j)=rf1
    XMDOT(J)=0.d0
    xmdotlw(j)=0.d0
    EBW(J)=EBWINT1
    EDS(J)=EDS1
    ELW(J)=0.D0
    EGAM(J)=1.d0-EDS(J)-EBW(J)-ELW(J)
    XMDOTLW(J)=0.D0
    PV(J)=PVINT
    RHOV(J)=RHOVINT
    PA(J)=PATM-PV(J)
    RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
    pgam(j)= pa(j)+pv(j)
    PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)
1-273.15d0)/(T(J)-40.25d0)))
101 CONTINUE
c*****INITIAL CONDITIONS FOR CLOTHING LAYERS
DO 102 J=kyair+1,M1
    T(J)=TINT
    tau(j)=tau2
    dsolid(j)=dsolid2
    al(j)=al2
    rf(j)=rf2
    XMDOT(J)=0.d0
    xmdotlw(j)=0.d0
    EBW(J)=EBWINT2
    EDS(J)=EDS2
    ELW(J)=0.D0
    EGAM(J)=1.d0-EDS(J)-EBW(J)-ELW(J)
    XMDOTLW(J)=0.D0
    PV(J)=PVINT
    RHOV(J)=RHOVINT

```

```

    PA(J)=PATM-PV(J)
    RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
    PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)
1-273.15d0)/(T(J)-40.25d0)))
    pgam(j)=pa(j)+pv(j)
    IF(J.EQ.M1) THEN
    T(J)=TBNDT
    PV(J)=PVBNDT
    RHOV(J)=RHOVBNDT
    PA(J)=PABNDT
    RHOAIR(J)=RHOABNDT
    PSAT(J)=PSATBT
    pgam(j)=pa(j)+pv(j)
    END IF
102 CONTINUE
    LPRINT(5)=.TRUE.
    LPRINT(6)=.TRUE.
    LPRINT(7)=.TRUE.
    LPRINT(8)=.TRUE.
    DO 103 J=1,M1
    DO 103 NF=1,NFD
    FOLD(J,NF)=F(J,NF)
103 CONTINUE
C*****CALCULATE INITIAL WEIGHT (MASS/SQ. METER)
    WEIGHTI=0.0
    DO 104 J=kyskin+1,M2
    RHOAVG=RHOW*EBW(J)+RHODS*EDS(J)+
1(RHOV(J)+RHOAIR(J))*EGAM(J)
    VOLUME=YCV(J)
    WEIGHTI=WEIGHTI+VOLUME*RHOAVG
104 CONTINUE
C*****record run variablesrfrn/RF
    open(10,access='append',file=datfile1)
    write(10,*)"dt,eps(5),eps(6)"
    write(10,*)"relax(5),relax(6),scheme"
    write(10,*)marker,dt,eps(5),eps(6)
    write(10,*)marker,relax(5),relax(6),lscheme
    write(10,*)"grid,M1",M1
    write(10,*)marker,M1
    write(10,*)"method,Thickness,Rf"
    write(10,*)"Dsolid,Initial Weight"
    write(10,*)marker,METHOD,YL,RF
    write(10,*)marker,DSOLID,WEIGHTI
    write(10,*)"rhint,rhbndt,rhbndb"
    write(10,*)marker,rhint,rhbndt,rhbndb
    write(10,*)"time,t,rhov,ebw,xmdot,weight,% change"
    write(10,*)"//nc"
    write(10,*)"time,tavg,trect,tskin,tfabric,dtb"
    close(10)
    open(11,access='append',file=datfile2)
    write(11,*)"temp,r.h.,xmdot"
    write(11,*)"//nc"
    close(11)
    open(25,access='append',file=datfile3)

```

```

        write(25,*)"time,sweat,xmavai(1),xmdot(kyskin),ebw,rhsk"
        close(25)
        RETURN

C*****
C
        ENTRY UNSTDY
C*****FOLD=FNEW
        IF (IBUG) PRINT *, "ENTRY UNSTEADY"
        DO 200 J=1,M1
        DO 200 NF=1,NFD
        FOLD(J,NF)=F(J,NF)
        200 CONTINUE

        RETURN

C*****
C
        ENTRY DENSE
c        call thermo
c        call pressure
c        call skin
c        call source

        IF (IBUG) WRITE (*,*) "ENTRY DENSE"
        RETURN

C*****
C
        ENTRY BOUND
        IF(IBUG)print *, 'entry bound'
        t(1)=t(2)
c        call pressure
        RETURN

C*****
C
        entry output1
c        print *, 'f(j,15),fold(j,15),f(j,16),fold(j,16)'
        do 1884 j=1,m1
c        print *,f(j,15),fold(j,15),f(j,16),fold(j,16)
        1884 continue
        return

        ENTRY OUTPUT
        if(ibug)print *, 'ENTRY OUTPUT'
        call outtoo
c        difmax=0.0
c        if(nf.eq.5)then
c        jstrt=kyskin+1
c        jends=m2
c        end if
c        if(nf.eq.6)then
c        jstrt=2

```

```

c      jends=m2
c      end if
c      if(nf.eq.5.or.nf.eq.6)then
cd      do 731 j=jstrt,jends
c      deltat=dabs((f(j,nf)-fnew(j))/(f(j,nf)+1.d-50))
c      difmax=dmax1(difmax,deltat)
c 731 continue
c      if(difmax.le.eps(nf)) iflag(nf)=1
c      if(jiter.gt.200)then
c      print *,jiter,nf
c      iflag(nf)=1
c      print *,'did it'
c      end if
c      if(ibug)print *,nf
c      RETURN

      ENTRY OUTPUT2
      RETURN

      ENTRY OUTPUT3
      RETURN
C*****
C
      ENTRY GAMSOR
      call thermo
c      call skin
c      call source

      IF (IBUG) print *, 'entry gamsor'
      GO TO (699,699,699,699,610,600) NF
c      if(nf.eq.20)goto 610
c      go to 699

      600 CONTINUE

C*****SKIN CALCULATIONS
C*****CALCULATE AVERAGE BODY TEMPERATURE AND DTB

c*****use rectangles to integrate temperature
      tbody0=308.9d0
      ttskin=0.d0
      DO 670 J=2,kyskin
      ttskin=ttskin+t(j)*ycv(j)
      670 CONTINUE
      TBAVG(1)=(1.d0/YBOD)*ttskin
      DTB=TBAVG(1)-tbody0
      DIFFTB=(TBAVG(1)-FOLD(1,15))/DT

C*****CONSTANTS
c*****basal metabolic rate (W/m^3)
      BASEMET=1339.d0
c*****exercise muscular heat generation (W/m^2)
      XERCISE=skwork/YMUSC

```



```

c*****reference skin conductivity (W/m-K)
      XKSO=.498d0
c*****density*heat capacity (J/m^3-K)
      RHOCp=4.1868D6
c*****thermal proportional control coefficients (1/K)
      ALPHAKP= .729d0
      ALPHAKM= .326d0
c*****thermal rate control coefficients (sec/K)
      XGAMK=1106.d0
c*****shivering energy generation constant (1/K)
      alpham=0.d0
c*****reference sweat generation (kg/m^2-sec)
      sweat0=2.80D-6
c*****sweating proportional control constant (kg/m^2-sec-K)
      alphswt=6.98D-6
c*****sweating rate control coefficient (kg/m^2-sec-K^4)
      xlmbwt=2.69D-5

      do 664 j=1,kyskin
      rho(j)=rhocp
      if(dtb.gt.0.d0)then
      gam(j)= xkso*(1.d0+alphakp*dtb+xgamk*difftb)
      xkplim=1.511d0*xkso
      if(gam(j).ge.xkplim)then
      gam(j)=xkplim
      end if
      end if
      if(dtb.le.0.d0)then
      gam(j)= xkso*(1.d0+alphakm*dtb+xgamk*difftb)
      xkmlim=.675d0*xkso
      if(gam(j).le.xkmlim)then
      gam(j)=xkmlim
      end if
      end if
      664 continue
c*****insulated condition at body center for symmetry
      gam(1)=0.0d0
      t(1)=t(2)

c*****body core
      do 661 j1=1,kycore
      con(j1)=BASEMET
      ap(j1)=0.0d0
      661 continue

c*****muscle layer
      do 662 j2=kycore+1,kymusc
c*****shivering control coefficients (W/m^3-K)or(W/m^3-K^2)
c*****shivering heat generation is (W/M^3)
      ashiv1=324.1d0
      ashiv2=383.8d0
      ashiv3=966.9d0
      if(dtb.gt.0.d0)deltmet=XERCISE
      if(dtb.le.0.d0)then

```

```

        if(dtb.le.0.d0.and.dtb.ge.-1.0d0)then
            SHIVER=-ashiv1*dtb+ashiv2*dtb*dtb
        end if
        if(dtb.lt.-1.0d0)then
            SHIVER=660.256d0-ashiv3*(dtb+1.0d0)
        end if
        deltmel=SHIVER+XERCISE
        end if
        con(j2)=deltmet
        ap(j2)=0.0d0
662 continue

c*****skin layer
do 663 j=kymusc+1,kyskin
    con(j)=0.0d0
    ap(j)=0.0d0
    if(j.eq.kyskin)then
        sknevap=xmflux(1)
        jsk=kyskin+1
        dhvap=(2.792D6)-160.d0*fold(jsk,6)-3.43d0*fold(jsk,6)
        1*fold(jsk,6)
        xsource=-dhvap*sknevap
c        if(rhumid.ge.1.0d0)then
c            print *,rhumid
c            xsource=0.d0
c        end if
        if(xsource.ge.0.d0)then
            con(j)=xsource
            ap(j)=0.d0
        end if
        if(xsource.lt.0.d0)then
            con(j)=0.d0
            ap(j)=xsource/t(j)
        end if
        end if
c        print *,xsource
c        write(*,815)j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
663 continue

c*****Sweating Production and Rate
        if(dtb.ge.0.d0)then
            Sweat(1)=sweat0+alphswt*dtb+xlmbwt*((dtb)**4)
            swtlim=60.d0*sweat0
            if(Sweat(1).gt.swtlim)then
                Sweat(1)=swtlim
            end if
        end if
        if(dtb.lt.0.d0)then
            Sweat(1)=sweat0
        end if

```

```

WEIGHT=0.0d0
DO 602 J=kyskin+1,m1
C*****DENSITY AND HEAT CAPACITY FOR ENERGY EQUATION
  RHOAVG=RHOW*(fold(j,7)+fold(j,16))+RHODS*EDS(J)+
  1(fold(j,5)+fold(j,10))*fold(j,9)
  CPAVG=(RHOW*(fold(j,7)+fold(j,16))*CPW+RHODS*EDS(J)*CPDS+
  1(fold(j,5)*CPV+fold(j,10)*CPA)*fold(j,9))/RHOAVG
c    RHOAVG=RHOW*(ebw(j)+elw(j))+RHODS*EDS(J)+
c    1(rhov(j)+rhoair(j))*egam(j)
c    CPAVG=(RHOW*(ebw(j)+elw(j))*CPW+RHODS*EDS(J)*CPDS+
c    1(rhov(j)*CPV+rhoair(j)*CPA)*egam(j))/RHOAVG
  RHO(J)=RHOAVG*CPAVG
C*****CALCULATE CURRENT WEIGHT (MASS/SQ. METER)
C*****AND WEIGHT GAIN OR LOSS FROM INITIAL CONDITION
  VOLUME=YCV(J)
  WEIGHT=WEIGHT+VOLUME*RHOAVG
  WEIGHTG=WEIGHT-WEIGHTI
C*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
  TKF=((TKW*RHOW*(fold(j,7)+fold(j,16))+TKDS*RHODS*EDS(J))
  1/(RHOW*(fold(j,7)+fold(j,16))+RHODS*EDS(J)))
  TKG=((TKV*fold(j,5)+TKA*fold(j,10))/
  1(fold(j,5)+fold(j,10)))
c    TKF=((TKW*RHOW*(ebw(j)+elw(j))+TKDS*RHODS*EDS(J))
c    1/(RHOW*(ebw(j)+elw(j))+RHODS*EDS(J)))
c    TKG=((TKV*rhov(j)+TKA*rhoair(j))/
c    1(rhov(j)+rhoair(j)))
  VF=EDS(J)+ebw(j)+elw(j)
  vf=eds(j)+fold(j,7)+fold(j,16)
  TKPERP=TKF*TKG/(VF*TKG+(1.-VF)*TKF)
  TKPRLL=(1.d0-VF)*TKG+VF*TKF
  WOMEGA=1.d0
  TKEFF=WOMEGA*TKPRLL+(1.0d0-WOMEGA)*TKPERP
C*****Cylinder Model
c    TKEFF=TKG*((TKF*(1.+VF)+TKG*(1.-VF)))/
c    1(TKF*(1.-VF)+TKG*(1.+VF)))
  GAM(J)=TKEFF
  DHVAP=(2.792D+6)-160.d0*fold(J,6)-3.43d0*fold(J,6)*fold(J,6)
  rhlim=0.95d0
  rhumid=fold(j,11)/fold(j,13)
  if(rhumid.gt.rhlim)then
    rhumid=rhlim
  end if
  QL=(1.95D+5)*(1.d0-rhumid)
  1*((1.d0/(0.2d0+(rhumid)))+
  1(1.d0/(1.05d0-(rhumid))))
C*****ALL SOURCE TERMS LUMPED INTO CON(J) (LAZY WAY)
  xhyg=-(dhvap+ql)*xmdot(j)
  xliq=-dhvap*xmdotlw(j)
  xsorce=xhyg+xliq
  if(xsorce.ge.0.d0)then
    con(j)=xsorce
    ap(j)=0.d0
  end if

```

```

        if(xsource.lt.0.d0)then
            con(j)=0.d0
            ap(j)=xsource/t(j)
        end if
602 CONTINUE
c*****CREATE CONVECTIVE BOUNDARY CONDITIONS
c      GDYM=GAM(1)/YDIF(2)
      GDYP=GAM(M1)/YDIF(M1)
c      AREAM=1.0d0/((1.0d0/CHTC+1.0/GDYM)*YCV(2))
      AREAP=1.0d0/((1.0d0/CHTC+1.0d0/GDYP)*YCV(M2))
c      GAM(1)=0.0
      GAM(M1)=0.0d0
c      CON(2)=CON(2)+AREAM*TBNDDB
c      AP(2)=AP(2)-AREAM
      CON(M2)=CON(M2)+AREAP*TBNDT
      AP(M2)=AP(M2)-AREAP
c      call output1
c      print *,nf
c      print *, 'j,gam,xmdot,xmdotlw,elw,ebw,egam'
c      do 1239 j=1,m1
c      print *,j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
c 1239 continue
      GO TO 699

610 CONTINUE
c      call thermo
c      call skin
      call source
c*****DO NOT INCLUDE BODY LAYER IN VAPOR SOLUTION
      do 609 j=1,kyskin
          gam(j)=0.0d0
          con(j)=0.0d0
          ap(j)=0.0d0
609 continue

      jstart=kyskin+1
      DO 611 J=jstart,M1
c*****DENSITY TERM FOR MASS DIFFUSION EQUATION
      RHO(J)=EGAM(J)
      rho(j)=fold(j,9)
c*****DIFFUSION COEFFICIENTS AND SOURCE TERMS
      DAIR=(2.20D-5)*((fold(J,6)/273.15d0)**1.75)
      DEFF=DAIR*EGAM(J)/TAU(J)
      deff=dair*fold(j,9)/tau(j)
      GAM(J)=DEFF

      xsource=(xmdot(j)+xmdotlw(j))
c      xsource=(1.d0-f(j,5))*xsource+
c      1(1.d0-egam(j))*(rhov(j)+rhoair(j))*(f(j,5)-fold(j,5))/dt
      xsource=xsource-rhov(j)*(egam(j)-fold(j,9))
      if(j.eq.jstart)then
          xsource=xsource+xmflux(1)
      end if

```

```

c*****source-term dominated calculations
    pvvapor=psat(j)
    rhovsat=pvvapor*xmw/(rgas*t(j))
    paair=pgam(j)-pvvapor
    rhoasat=paair*xma/(rgas*t(j))
    xmsat=rhovsat/(rhovsat+rhoasat)
    rhumid=pv(j)/psat(j)

    IF(xsource.LT.0.d0) THEN
    CON(J)=0.d0
    AP(J)=xsource/(rhov(J))
    if(elw(j).gt.0.d0) then
    xmflim=0.99D0*xmsat
c    con(j)=xsource*xmflim/(xmflim-xmf(j))
c    ap(j)=-xsource/(xmflim-xmf(j))
c    con(j)=(1.D50)*xmflim
c    ap(j)=-1.D50
    end if
    jj1=kyskin+1
    if(xmavai(1).gt.0.d0.and.j.eq.jj1) then
    con(j)=xsource*rhovsat/(rhovsat-rhov(j))
    ap(j)=-xsource/(rhovsat-rhov(j))
    truesrc=con(j)+ap(j)*rhov(j)
    end if
    END IF
    IF(xsource.GE.0.d0) THEN
    CON(J)=xsource
    AP(J)=0.0d0
c    if(elw(j).gt.0.d0) then
    if(j.eq.kyskin+1.and.xmavai(1).gt.0.d0) then
    xmflim=.99D0*xmsat
c    con(j)=xsource*xmflim/(xmflim-xmf(j))
c    ap(j)=-xsource/(xmflim-xmf(j))
c    con(j)=(1.D50)*xmflim
c    ap(j)=-1.D50
    end if
    jj1=kyskin+1
    if(xmavai(1).gt.0.d0.and.j.eq.jj1) then
    con(j)=xsource*rhovsat/(rhovsat-rhov(j))
    ap(j)=-xsource/(rhovsat-rhov(j))
    truesrc=con(j)+ap(j)*rhov(j)
    end if

    END IF

611 CONTINUE
c*****CREATE CONVECTIVE BOUNDARY CONDITIONS
c    GDYM=GAM(1)/YDIF(2)
    GDYP=GAM(M1)/YDIF(M1)
c    AREAM=1.0/((1.0/CMTC+1.0/GDYM)*YCV(2))
    AREAP=1.0d0/((1.0d0/CMTC+1.0d0/GDYP)*YCV(M2))
c    GAM(1)=0.0
    GAM(M1)=0.0d0

```

```

c      CON(2)=CON(2)+AREAM*RHOVBNDDB
c      AP(2)=AP(2)-AREAM
      CON(M2)=CON(M2)+AREAP*RHOVBNDT
      AP(M2)=AP(M2)-AREAP
c      call output1
c      print *,nf
c      print *,'j,gam,xmdot,xmdotlw,elw,ebw,egam'
c      do 1279 j=1,m1
c      print *,j,gam(j),xmdot(j),xmdotlw(j),elw(j),ebw(j),egam(j)
c 1279 continue
      GOTO 699

```

```

699 RETURN

```

```

C*****
C
      ENTRY SAVE1

      RETURN

```

```

C*****
C
      ENTRY OUTTOO
      LPRINT(5)=.TRUE.
      LPRINT(6)=.TRUE.
      time1=time/3600.
c      if(time1.lt.2.)goto 899
      if(time1.gt.8.)goto 9999
      if(time1.gt.3.)then
        dt=1.d-1
        relax(6)=.1d0
        relax(5)=.1d0
        relax(17)=.5d0
        relax(5)=.5
        relax(6)=1.
        relax(17)=1.
        nprint=1
        nprint=int(60./dt)
        nprint=10000
c      iwrite=nprint
      if(time1.gt.3.25)then
        dt=1.d-1
        nprint=10000
      end if
      rhlim=0.9d0
      tchange=600.d0
      rhchange=0.6d0
      delrhdt=rhchange/tchange
      rhprev=rhprev+delrhdt*dt
      if(rhprev.gt.rhlim)then
        rhprev=rhlim
      end if
      rhprev=0.65d0
      relax(17)=1.d0

```

```

skwork=200.D0
RHBNDT=rhprev
TINTC=30.0d0
TINT=TINTC+273.15d0
TBNDT=TINT
PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
1/(TBNDT-40.25d0)))
PVBNDT=PSATBT*RHBNDT
PABNDT=PATM-PVBNDT
RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
RHOVBNDT=PVBNDT*XMW/(RGAS*TBNDT)
rhov(m1)=rhovbndt
pv(m1)=pvbndt
psat(m1)=psatbt
pa(m1)=pabndt
rhoair(m1)=rhoabndt
t(m1)=tbndt
end if
if(time1.gt.4.d0)then
dt=1.d-1
nprint=10000
c   relax(5)=.5
c   relax(6)=.5
c   relax(8)=.1
c   relax(8)=1.
if(time1.gt.4.25d0)then
dt=1.d-1
c   nprint=int(dt)*60
c   relax(8)=1.
end if
skwork=20.D0
RHBNDT=0.650d0
TINTC=30.d0
TINT=TINTC+273.15d0
TBNDT=TINT
PSATBT=614.3d0*DEXP(17.06d0*((TBNDT-273.15d0)
1/(TBNDT-40.25d0)))
PVBNDT=PSATBT*RHBNDT
PABNDT=PATM-PVBNDT
RHOABNDT=PABNDT*XMA/(RGAS*TBNDT)
RHOVBNDT=PVBNDT*XMW/(RGAS*TBNDT)
rhov(m1)=rhovbndt
pv(m1)=pvbndt
psat(m1)=psatbt
pa(m1)=pabndt
rhoair(m1)=rhoabndt
pgam(m1)=pv(m1)+pa(m1)
t(m1)=tbndt
end if
if(iwrite.eq.nprint)then
open(10,access='append',file=datfile1)
temp1=tbavg(1)-273.15d0
temp2=t(1)-273.15d0

```

```

temp3=t(kyskin)-273.15d0
temp4=t(m2)-273.15d0
write(*,*)"time tbody tcore tskin tcslsurf dtb ebwskin"
write(*,817)time1,temp1,temp2,temp3,temp4,
ldtb,ebw(kyskin+1)
write(10,816)time1,temp1,temp2,temp3,temp4,
ldtb
816 format(5e11.5,1x,e11.5)
817 format(7f11.5)
818 format(i10,3e11.5)
close(10)
open(25,access='append',file=datafile3)
rhumid=pv(kyskin+1)/psat(kyskin+1)
write(25,816)time1,sweat(1),xmavai(1),xmflux(1),
1elw(kyskin+1),rhumid
close(25)
c write(*,*)"iter,temp(K),temp(C),dens,humid,xmdot"
open(11,access='append',file=datafile2)
write(*,*)"iter,sweat,mass left on skin"
write(*,*)iter,sweat(1),xmavai(1)
write(*,*)"original sweat,sweat evaporated,sweat left"
xsleft=xmavai(1)
swteb=xmflux(1)*dt*ycv(kyskin+1)
swtbeg=fold(3,15)+sweat(1)*dt
write(*,822)skmass,swteb,xsleft
print *,'source used (total), truesrc'
print *,xmflux(1),xmdot(kyskin),xmdotlw(kyskin),truesrc
c write(*,*)"j,temp,elw,rhumid,xmflux(1),xmdotlw,egam"
822 format(3e15.4)

c print *,'j,temp,rhov(j),rhumid,xmdot(j),ebw(j),xmlw(j),elw(j)'
print *,'j,temp,rhov(j),rhumid,xmdot(j),ebw(j),pv(j),pgam(j)'
c print *,'j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)'
jend=m1
jstrt=kyskin-1
do 897 j=jstrt,jend
temp=t(j)-273.15
rhumid=pv(j)/(psat(j)+1.D-50)
flux=xmdot(j)
if(j.eq.kyskin+1)then
flux=xmflux(1)
end if
c write(*,815)j,temp,rhov(j),rhumid,flux,ebw(j)
c 1,xmdotlw(j),elw(j)
write(*,815)j,temp,rhov(j),rhumid,flux,ebw(j)
1,pv(j),pgam(j)
c write(11,818)j,temp,rhumid,xmdot(j)
897 continue
c print *,'j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)'
do 898 j=kyskin-1,m1
temp=t(j)-273.15
rhumid=pv(j)/(psat(j)+1.D-50)
if(j.eq.kyskin+1)then
c write(*,815)j,egam(j),ebw(j),elw(j),xmflux(1),xmdotlw(j),xmf(j)

```



```

        end if
c      write(*,815)j,egam(j),ebw(j),elw(j),xmdot(j),xmdotlw(j),xmf(j)
c      write(11,818)j,temp,rhumid,xmdot(j)
898 continue
      close(11)
815 format(1x,i7,1f9.3,e9.3,5e9.3)
      iwrite=0
      endif
      iwrite=iwrite+1
811 format(1e10.3,1x,e10.5,1x,e10.5,4e10.3)
812 format (5e11.4)
899 RETURN

```

C*****

```

C
      ENTRY SAVEPC

      RETURN

```

C*****

```

C
      ENTRY SAVE2

      RETURN

```

```

      entry THERMO
c      if(ibus)print *,'entry thermo'
      do 1021 j=kyskin,m1
c      xmfr=xmf(j)/(1.d0-xmf(j))
c      rhov(j)=(xmfr*(patm*xma/(rgas*t(j))))
c      1/(1.d0+xmfr*(xma/xmw))

      PSAT(J)=614.3d0*DEXP(17.06d0*((T(J)-273.15d0)/
1(T(J)-40.25d0)))
      PV(J)=RHOV(J)*RGAS*T(J)/XMW
      PA(J)=PATM-PV(J)
      pgam(j)=pa(j)+pv(j)
      RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
1021 continue
      return

      entry pressure
      do 1026 j=kyskin,m1
      Pgam(J)=PATM
      PV(J)=RHOV(J)*RGAS*T(J)/XMW
      pa(j)=pgam(j)-pv(j)
      RHOAIR(J)=PA(J)*XMA/(RGAS*T(J))
1026 continue
      return

      entry source
      if(ibus)print *,'entry source'

```

c*****properties for the two layers

do 1932 j=kyskin+1,kyair

tau(j)=tau1

eds(j)=eds1

rf(j)=rf1

al(j)=al1

dsolid(j)=dsolid1

1932 continue

do 1933 j=kyair+1,m2

tau(j)=tau2

eds(j)=eds2

rf(j)=rf2

al(j)=al2

dsolid(j)=dsolid2

1933 continue

C*****FIBER SOLID-STATE SOURCE TERM

DO 1031 J=kyskin+1,M2

RHUMID=PV(J)/PSAT(J)

rhlim=0.95D0

if(rhumid.gt.rhlim)then

rhumid=rhlim

end if

EBW(J)=(.578d0*RF(J)*EDS(J)*RHODS/RHOW)*RHUMID*

1(1.d0/(.321d0+RHUMID)+1.d0/(1.262d0-RHUMID))

XMDOT(J)=(FOLD(J,7)-EBW(J))*RHOW/DT

1031 continue

do 1032 j=kyskin+1,m2

CSKIN=RHOW*EBW(J)/(EBW(J)+EDS(J))

CINSIDE=RHOW*FOLD(J,7)/(FOLD(J,7)+EDS(J))

REQ=EBW(J)*EDS(J)*RHODS/RHOW

RINST=FOLD(J,7)*EDS(J)*RHODS/RHOW

XMDOT(J)=DSOLID(j)*RHODS*(RINST-REQ)/(aL(j)*aL(j))

xmdot(j)=relax(8)*xmdot(j)+(1.d0-relax(8))*fold(j,8)

EBW(J)=FOLD(J,7)-XMDOT(J)*DT/RHOW

1032 CONTINUE

DO 1042 J=kyskin+1,m2

c goto 1742

c xmdot(j)=0.d0

xmdotlw(j)=0.d0

c if(j.eq.kyskin)goto 1742

pvvapor=psat(j)

rhovsat=pvvapor*xmw/(rgas*t(j))

if(rhov(j).gt.rhovsat)THEN

fog=elw(j)

xmdotlw(j)=fold(j,9)*(rhovsat-rhov(j))/dt

xmdotlw(j)=egam(j)*(rhovsat-rhov(j))/dt

```

xmdotlw(j)=relax(17)*xmdotlw(j)+(1.d0-relax(17))*fold(j,17)
elw(j)=fog-(xmdotlw(j)*dt/rhow)
end if

if(rhov(j).lt.rhovsat.and.ELW(J).GT.0.D0)THEN
fog=elw(j)
xmdotlw(j)=egam(j)*(rhovsat-rhov(j))/dt
xmdotlw(j)=relax(17)*xmdotlw(j)+(1.d0-relax(17))*fold(j,17)
elw(j)=fog-(xmdotlw(j)*dt/rhow)
if(elw(j).lt.0.d0)then
elw(j)=0.d0
xmdotlw(j)=fog*rhow/dt
end if
end if

1742 egam(j)=1.d0-eds(j)-ebw(j)-elw(j)

jswitch=kyskin+1
if(j.eq.jswitch)then
c*****use saturation properties at skin surface temperature
pvvapor=psat(j-1)
rhovsat=pvvapor*xmw/(rgas*t(j-1))
c*****use saturation properties at 1st interface temperature
pvvapor=psat(j)
rhovsat=pvvapor*xmw/(rgas*t(j))
xmflux(1)=0.d0
xmold=fold(3,15)
xmswt=sweat(1)*dt
skmass=xmold+xmswt
c*****use vapor properties at skin temperature
DAIR=(2.20d-5)*((fold(j,6)/273.16d0)**1.75d0)
deff=dair*fold(j,9)/tau(j)
dforce=rhovsat-rhov(j)
if(dforce.gt.0.d0.and.elw(j).eq.0.d0)then
xmflux(1)=(1.d0/ycv(j))*deff*(rhovsat-rhov(j))/(ycv(j)/2.d0)
c    xmflux(1)=egam(j)*dforce/dt
xmaxflx=(skmass/dt)*(1.d0/ycv(j))
xmflux(1)=dmin1(xmflux(1),xmaxflx)
xmleft=skmass-(xmflux(1)*dt*ycv(j))
xmavai(1)=xmleft
end if
c    end if
if(dforce.le.0.d0)then
xmflux(1)=0.d0
xmavai(1)=skmass
end if
end if
1042 continue
return

9999 stop
end

```