

NBSIR 83-2635

A Computer Algorithm for Estimating Infiltration and Inter-Room Air Flows



U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Building Physics Division
Washington, DC 20234

November 1982

Issued February 1983

Sponsored by
National Bureau of Standards
and
Building Energy Sciences Branch
Office of Building Energy Research and Development
U.S. Department of Energy
Washington, DC 20545

1210

AIC
US 26

NBSIR 83-2635

**A COMPUTER ALGORITHM FOR
ESTIMATING INFILTRATION AND
INTER-ROOM AIR FLOWS**

George N. Walton

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Building Physics Division
Washington, DC 20234

November 1982

Issued February 1983

Sponsored by
National Bureau of Standards
and
Building Energy Sciences Branch
Office of Building Energy Research and Development
U.S. Department of Energy
Washington, DC 20545



U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, *Secretary*
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, *Director*

ABSTRACT

This report discusses the extension of an infiltration predicting technique to the prediction of inter-room air movements. The air flow through openings is computed from the ASHRAE crack method together with a mass balance in each room. Simultaneous solution of the mass balances in all rooms having both large and small openings is accomplished by a slightly modified Newton's method. A simple theory for two-way flow through large openings is developed from consideration of density differences caused by different temperatures in adjoining rooms. The technique is verified by comparison to published experimental results. The results indicate that the simple model provides reasonable results for complex two way flows through openings. The model is as accurate as the available data, that is, about $\pm 20\%$. The air flow algorithm allows infiltration and forced air flows to interact with the doorway flows to provide a more general simulation capability.

Key words: building energy analysis; building heat transfer; computer modeling; convection; infiltration; ventilation.

PREFACE

This report is one of a series documenting NBS research and analysis efforts in developing energy and cost data to support the Department of Energy/National Bureau of Standards Measurements Program. It was prepared by the Thermal Analysis Group, Building Physics Division, Center for Building Technology, National Engineering Laboratory, National Bureau of Standards (NBS). This work was jointly sponsored by (NBS). The development of multi-room air flow modeling was supported by DoE/NBS Task Order A008 under Interagency Agreement No. E-77-A-01-6010. This report describes a computer program which was written as part of an effort to develop a comprehensive modeling technique for predicting the simultaneous transfer of air, moisture, and heat in and through multi-room buildings.

The author wishes to acknowledge the support and direction given by Dr. T. Kusuda, Thermal Analysis Group Leader.

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT	iii
PREFACE	iv
1. INTRODUCTION	1
1.1 BACKGROUND	1
1.2 OBJECTIVES	
1.3 SCOPE	
2. CALCULATION METHODS	3
2.1 THE AIR FLOW EQUATION	3
2.2 SOLUTION OF THE FLOW EQUATIONS	3
2.3 A THEORY OF FLOW THROUGH LARGE OPENINGS	6
3. VALIDATION OF THE LARGE OPENING MODEL	8
3.1 FLOW THROUGH OPENINGS IN VERTICAL PARTITIONS	8
3.2 FLOW THROUGH OPENINGS IN HORIZONTAL PARTITIONS	8
4. CONCLUSIONS AND RECOMMENDATIONS	10
5. REFERENCES	11
FIGURES	12
APPENDIS A: LISTING OF THE AIR FLOW TEST PROGRAM	17
APPENDIX B: SAMPLE INPUT FOR THE AIR FLOW TEST PROGRAM	27

LIST OF FIGURES

	<u>Page</u>
Figure 1. Orifice Flow Meter Coefficients	12
Figure 2. Schematic of opening in a vertical partition	13
Figure 3. Two-room model for determining natural convection transfer through the doorway	14
Figure 4. Comparison of experimental results for convection through openings in vertical partitions	15
Figure 5. Experimental results for air flow through a square opening in a horizontal partition	16

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)		1. PUBLICATION OR REPORT NO. NBSIR 83-2635	2. Performing Organ. Report No.	3. Publication Date February 1983
4. TITLE AND SUBTITLE A Computer Algorithm for Estimating Infiltration and Inter-Room Air Flows				
5. AUTHOR(S) George N. Walton				
6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions) NATIONAL BUREAU OF STANDARDS DEPARTMENT OF COMMERCE WASHINGTON, D.C. 20234			7. Contract/Grant No.	8. Type of Report & Period Covered
9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP) National Bureau of Standards U.S. Department of Energy, Building Energy Sciences Branch Office of Building Energy Research and Development Washington, DC 20545				
10. SUPPLEMENTARY NOTES <input type="checkbox"/> Document describes a computer program; SF-185, FIPS Software Summary, is attached.				
11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here) This report discusses the extension of an infiltration predicting technique to the prediction of inter-room air movements. The air flow through openings is computed from the ASHRAE crack method together with a mass balance in each room. Simultaneous solution of the mass balances in all rooms having both large and small openings is accomplished by a slightly modified Newton's method. A simple theory for two-way flow through large openings is developed from consideration of density differences caused by different temperatures in adjoining rooms. The technique is verified by comparison to published experimental results. The results indicate that the simple model provides reasonable results for complex two way flows through openings. The model is as accurate as the available data, that is, about +20%. The air flow algorithm allows infiltration and forced air flows to interact with the doorway flows to provide a more general simulation capability.				
12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons) Building energy analysis; building heat transfer; computer modeling; convection; infiltration; ventilation.				
13. AVAILABILITY <input checked="" type="checkbox"/> Unlimited <input type="checkbox"/> For Official Distribution. Do Not Release to NTIS <input type="checkbox"/> Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402. <input checked="" type="checkbox"/> Order From National Technical Information Service (NTIS), Springfield, VA. 22161			14. NO. OF PRINTED PAGES 35	15. Price \$8.50

1. INTRODUCTION

1.1 BACKGROUND

Although numerous building thermal modeling techniques and computer programs, for example NBSLD [1], BLAST [2], and DOE-2 [3], exist throughout the United States, none of the existing techniques/programs handle the following processes simultaneously:

- envelope heat transfer
- envelope air leakage
- envelope solar heat gain
- room-to-room heat transfer
- room-to-room air and moisture transfer
- intra-room air movement
- energy consumption by the heating/cooling equipment
- indoor comfort
- water vapor condensation and contaminant migration.

Existing models are virtually all single-room models where dynamic coupling between the heated and non-heated spaces and/or the cooled and non-cooled spaces are ignored.

Comprehensive multi-room building simulation capabilities will be needed in the coming years for the following reasons:

1. Intra-room convection plays a significant role not only for the transfer of heat from the interior surfaces to the room air but also for the thermal comfort of the occupants. Yet existing computational technology for predicting the room temperature stratification and room air motion is very inadequate.
2. Passive solar design techniques are expected to be used to a large extent in new building design. Proper design is going to depend on understanding and being able to predict natural energy flows within the buildings.
3. Proper control of the air flow in the central air system requires accurate knowledge of room-by-room energy demand.
4. Moisture and contaminant distribution throughout a building must be capable of being predicted in order to insure adequate designs as the emphasis on "tighter" buildings increases.
5. Until such capability exists, effective design of internal partitioning with respect to natural ventilation can only be done by empirical techniques.

In response to these needs, a research-oriented computer program has been developed to allow the detailed study of simultaneous air, heat, and moisture transfer in and through a building with complex internal architecture. This program is called the Thermal Analysis Research Program (TARP). Documentation [4] for the program will soon be published. Primary emphasis in the development

of TARP has been on air transfer because this is a major shortcoming of present techniques and because it is basic to further developments in moisture and contaminant analysis. A previous report [5] describes initial results in the development of TARP and listed five significant areas for further research:

1. Calculation of air flows through large openings in reasonable computation time,
2. Calculation of two-way flows between rooms,
3. Accurate evaluation of the wind induced pressure distribution around the envelope of the building,
4. Calculation of the effects of room air stratification, and
5. Availability of data for estimating the opening areas in the envelopes of buildings.

This report will address the first two areas.

1.2 OBJECTIVE

The overall objective of this research effort is to develop a comprehensive modeling technique for predicting simultaneous transfer of air, moisture, and heat in and through multi-room buildings. The objective of this particular study is the development of a method for predicting air flows through large openings between rooms.

1.3 SCOPE

Section 2 of this report discusses the extension of an infiltration predicting technique to the prediction of inter-room air movements. The technique is verified by comparison to published experimental results in section 3. Conclusions and recommendations are given in section 4.

2. CALCULATION METHODS

2.1 THE AIR FLOW EQUATION

The TARP air flow algorithm is described in detail in reference [5] where the program was called the Multi-Room Loads Program (MRLP). The air flow algorithm is based on the equation [6]

$$F = K * (\Delta P)^X \quad (1)$$

where

F = flow rate (kg/s)
K = a constant
 ΔP = pressure difference across an opening (Pa)
X = flow exponent

Pressure differences arise from wind, air density differences, and system induced flows. By applying equation (1) to all openings in the building envelope and all openings between rooms and requiring a mass balance in each room, a set of simultaneous non-linear algebraic equations is created which can be solved for the zone pressures and the air flow through each opening.

An estimate for the value of K was made by referring to the orifice equation:

$$F = C * A * \rho * \sqrt{2 * \Delta P / \rho} \quad (2)$$

where

C = flow coefficient
A = observed opening area (m^2)
 ρ = air density (kg/m^3)

When the opening is small, C equals 0.6 for a wide range of Reynolds numbers as shown in figure 1 [7]. TARP assumes this value of C as a default. TARP allows a variable flow exponent (instead of the 0.5 of the orifice equation) because building pressurization measurements typically correlate to equation (1) when X equals about 0.65 [5].

2.2 SOLUTION OF THE FLOW EQUATIONS

The development of a solution technique for the air flow equations has been particularly troublesome. Efforts have focused on two techniques which are described by Conte and DeBoor [8]. The first technique is the classic Newton's method. The mass balance requirement may be expressed as

$$\sum F_i = 0 \quad (3)$$

for every room. The flows, F_i , are summed over all openings, i, in the enclosing surfaces of the room. In Newton's method, successive values of room pressure, P_n , for each room, n, are calculated by

$$P_n^{(k+1)} = P_n^{(k)} - C_n \quad (4)$$

where k is the iteration number and C is computed from the matrix relationship

$$[J] * [C] = [B] \quad (5)$$

where [B] is a column matrix each element given by

$$B_n = \sum F_i \quad (6)$$

and J is the square (i.e., N by N for a building of N rooms) Jacobian matrix. The values of the diagonal elements of the Jacobian matrix are given by

$$J_{n,n} = \sum_i \frac{\partial F_i}{\partial P_n} \quad (7)$$

for all openings, i, into room n. The values of the other elements are given by

$$J_{n,m} = \sum_i \frac{\partial F_i}{\partial P_m} \quad (8)$$

for all openings, i, between room m and room n. Iteration proceeds until the net air flow into every room, B_n , is sufficiently close to zero. These partial derivatives are very easy to compute:

$$\frac{\partial F_i}{\partial P_n} = X_i * F_i / \Delta P \quad (9)$$

$$\frac{\partial F_i}{\partial P_m} = X_i * F_i / \Delta P \quad (\text{inter-room surface}) \quad (10)$$

or

$$\frac{\partial F_i}{\partial P_m} = 0 \quad (\text{envelope surface}) \quad (11)$$

Note that the term $(\Delta P)^{X-1}$ has been eliminated from these expressions, thus allowing evaluation of the derivatives by a simple division rather than a time consuming exponential. Also note that as ΔP approaches zero, the derivative is undefined.

The second method calls for successively approximating each room pressure according to

$$P_n^{(k+1)} = P_n^{(k)} - \Sigma F_i / \Sigma \frac{\partial F_i}{\partial P_n} \quad (12)$$

where k is again the iteration number. This method is quite simple and requires less storage space than Newton's method because it does not use the Jacobian matrix.

Initial tests of these two methods showed that, although Newton's method was fastest for most test cases, it occasionally would not converge. The second method converged for the original test cases and was chosen for the initial version of TARP [5]. It was then found that this method converged very slowly when the openings between rooms were much larger than the openings in the building envelope which is a very common condition. This problem led to a re-examination of the Newton's method. A simple four room test case was found to usually be quadratically convergent (about 4 iterations) except for a few cases where it converged very slowly (about 30 iterations). In those cases it was found that successive iterations were over-correcting. That is, they were successively far above and then far below the correct solution because successive corrections were of about the same magnitude but opposite sign. Convergence could be achieved by reducing the size of the pressure correction by about one half. Since the Newton's method is normally rapidly convergent, it is also necessary to reduce the size of the correction only when over-correcting occurs. The reason for occasional slow convergence of the Newton's method has not been found. It can occur with nothing more than a change in wind direction from what was a quadratically convergent case. It often occurs when the wind and stack pressures are about equal. Convergence is always fastest when the flow exponent for inter-room openings is near one. The experimental studies below indicate, however, that the exponent should be one half.

The air flow algorithm was incorporated into a test program which allowed various solution techniques to be studied without revising TARP, which is very large. A listing of the test program is attached as appendix A. Details of the solution algorithm are discussed in that listing.

The Newton's method requires the simultaneous solution of matrix equation (5) at each iteration. Several techniques were considered for that solution. The first choice, and the one ultimately chosen, was Gauss elimination. It has the disadvantage that solution time is proportional to the cube of the number of equations, N, when N is large. The number of equations is equal to the number of rooms. The Cholesky method [9] is somewhat faster at large N, but is was found to be sensitive to computer truncation errors. A Gauss-Seidel iteration was also tried since it has solution time proportional to the square of N. However, this iteration also failed for large openings in much the same way that equation (12) did. The Newton's method was tested for larger numbers of rooms. It was found that the number of iterations increased with the number of rooms. Thus, many room require both longer iterations and more iterations. The number of iterations was also found to increase with the size of the inter-room openings, but it did not increase as dramatically as it had done with equation (12). Therefore, the Newton's method is most appropriate for a small

number of rooms with large execution time penalties paid for simultaneously solving many rooms.

TARP uses an hourly heat balance in estimating dynamic room energy requirements. This heat balance is solved iteratively, and at each iteration a quasi-steady solution of the air flow equations is obtained by the Newton's method described above. Techniques used to reduce the number of heat balance iterations contribute to the overall efficiency of TARP and are described in [5].

2.3 A THEORY OF FLOW THROUGH LARGE OPENINGS

Equation (1) permits air to flow only one direction through an opening. Large openings, such as doorways, may have two way flow as the stack effect between two rooms may cause a positive ΔP at the bottom of the doorway and a negative ΔP at the top (or vice versa). A theoretical estimate of the stack induced air flow through a large opening in a vertical partition (a doorway) is given by Brown and Solvason [9]. The following discussion shows that the TARP method is equivalent. Figure 2 shows a cross section of a rectangular opening of height H and width W in a vertical partition which separates two rooms at temperatures T_1 and T_w . The absolute pressure at the centerline ($z = 0$) is everywhere equal. The pressure difference caused by stack effect at height z is

$$P_1 - P_2 = (\rho_1 - \rho_2) * g * z \quad (13)$$

The volumetric flow through an infinitesimal area is given by

$$dQ = C * W * dz * \sqrt{2 * \Delta P / \rho} \quad (14)$$

which can be integrated to give the flow through the top half of the opening

$$Q = \int_{Z=0}^{Z=H/2} dQ = C/3 * W * \sqrt{g * \Delta\rho / \rho} * H^3 \quad (15)$$

The coefficient of thermal expansion is $\rho = -\Delta\rho / (\rho * \Delta T)$ where ρ is the average density. For computational simplicity TARP uses the density of the incoming fluid instead, but this cannot cause a significant error at normal temperatures. Other definitions are:

$$\text{heat transfer rate: } \dot{q} = Q * \rho * c * (T_1 - T_1) \quad (16a)$$

$$\text{heat transfer coefficient: } h = \dot{q} / [W * H * (T_1 - T_2)] \quad (16b)$$

$$\text{Nusselt number: } Nu = h * H / k \quad (16c)$$

$$\text{Prandtl number: } Pr = c * \mu / k \quad (16d)$$

$$\text{Grashof number: } Gr = \rho^2 * g * \beta * (T_1 - T_2) * H^3 / \mu \quad (16e)$$

The expressions can be substituted into equation (15) to give

$$\text{Nu} = C/3 * \text{Pr} * \sqrt{\text{Gr}} \quad (17)$$

This simplified analysis has neglected viscous effects and effects of an air velocity parallel to the partition. According to Brown and Solvason the viscous effects reduce the air flow through thick partitions and an air velocity may either increase or reduce the air flow.

TARP can handle the two-way air flow through a doorway by dividing the door into an upper and a lower opening. It is easy to determine the heights of the two openings which cause a stack effect giving the same volumetric flow as the Brown and Solvason model. These turn out to be $13/18 * H$ for the upper opening and $5/18 * H$ for the lower one.

Brown also studied openings in horizontal partitions [10]. He developed a theoretical expression for convection through such openings:

$$\text{Nu} = (0.29 \text{ to } 0.35) * \text{Pr} * \sqrt{\text{Gr}} \quad (18)$$

In this case the Nusselt and Grashof numbers are based on the thickness of the partition. This thickness is the vertical space available for the development of a stack effect.

3. VALIDATION OF THE LARGE OPENING MODEL

3.1 FLOW THROUGH OPENINGS IN VERTICAL PARTITIONS

Weber and Kearney [11] give a correlation for the two way flow through a doorway based on temperature measurements in the doorway

$$\text{Nu} = 0.26 * \text{Pr} * \sqrt{\text{Gr}} \quad (19)$$

and another based on average room temperatures

$$\text{Nu} = 0.3 * \text{Pr} * \sqrt{\text{Gr}} \quad (20)$$

These correlations are based on similitude experimental studies of the room shown in figure 3 and tests on a full scale structure with a similar configuration. Weber and Kearney estimated that they should be dependable to within 20 percent. They also compare their results to two other studies (figure 4) including Brown and Solvason's. Comparing equations (19) and (17) gives a value of 0.78 for the flow coefficient, C. Equation (20), which uses average room temperatures, gives a value of 0.90 for the flow coefficient. This seems unreasonably large and is apparently due to the uneven temperature distributions which occur in real rooms, especially above the door openings. A study of a doorway between a small room and a much larger, high heat loss, two-story room did not agree well with the correlation (20).

The TARP model dividing the doorway into halves was used for a wide range of parameters and the results compared to equation (19). The inter-room mass and energy flows were computed for five values of ΔT (2, 4, 6, 8, and 10 C) each at five doorway heights (1.0, 1.5, 2.0, 2.5, and 3.0 meters). A flow coefficient of .78, a flow exponent of 0.5, and stack heights at 5/18ths and 13/18ths of the doorway height give TARP computed mass and energy flows which agree with equation (19) to within 0.1 percent for all cases.

3.2 FLOW THROUGH OPENINGS IN HORIZONTAL PARTITIONS

Figure 5 shows the experimental results (H = thickness; L = width of square opening) of Brown's study of openings in horizontal openings [10]. In this configuration there is a significant frictional effect in the thicker partitions. Brown incorporated this into a single equation:

$$\text{Nu} = 0.0546 * \text{Pr} * \text{Gr}^{.55} * (\text{L}/\text{H})^{.33} \quad (21)$$

For a TARP evaluation it is better to consider the thickness effect as a modifier to the stack height. A TARP model of an opening in a horizontal partition would again divide it into two openings, one a distance Z above the center of the partition and one and equal distance below it. Then, for C = 0.78, the values of Z for different H/L would be:

H/L	Z
.0833	.168*H
.167	.120*H
.333	.093*H
.667	.074*H

These almost exactly duplicate the Brown results.

There are still several questions about this model. The effects beyond the studied ranges of Gr and H/L are not known. The model predicts that the flow should go to zero as H approaches zero! There should be some flow. It is possible that two separate openings would behave differently than a single opening of equivalent area because one-way flow could develop in each. These questions indicate room for further experimental work. In addition, a TARP model would have to allow for no flow between rooms when the upper is warmer than the lower.

4. CONCLUSIONS AND RECOMMENDATIONS

The results indicate that the simple TARP model provides reasonable results for complex two way flows through openings. The model is as accurate as the available data, that is, about +20 percent. The TARP air flow algorithm allows infiltration and forced air flows to interact with the doorway flows to provide a more general simulation capability. Although more work must be done for other building configurations, the close match between the TARP algorithm and the experimental correlations is encouraging.

Because of the rapid increase of calculation time with the number of rooms simulated, it is recommended that large buildings with many rooms be simulated by dividing the building into groups of closely coupled rooms. First, treat each group of rooms as a single room to solve for the infiltration through the building envelope and air flows between the groups of rooms. Then, while treating those air flows as constant gains or losses to the appropriate rooms, solve for the air flows between the individual rooms in each group.

Further study is needed on the simulation of room air stratification both for its direct effects on comfort and energy requirements as well as for its effect on inter-room air movement. Study is also needed on the calculation of the wind pressure on the building envelope. Then detailed validation should be performed with carefully selected full scale tests. After successful validation, it will still be necessary to develop air openings data for typical construction components and building techniques for the analysis and design of buildings.

5. REFERENCES

- [1] Kusuda, T., "NBSLD, the Computer Program for Heating and Cooling Loads in Buildings," NBS Building Science Series 69, July 1976.
- [2] Hittle, D. C., "The Building Loads Analysis and Systems Thermodynamics (BLAST) Program, Version 2.0: Users Manual", U.S. Army Construction Engineering Research Laboratory Report TR-E-153, June 1979.
- [3] "DOE-2 Users Guide, Version 2.1", Lawrence Berkeley Laboratory Report LBL-8698, May 1980.
- [4] Walton, G. N., "Thermal Analysis Research Program Reference Manual", National Bureau of Standards, in review.
- [5] Walton, G. N., "Calculation of Inter-Room Air Movement for Multi-Room Building Energy Analysis", National Bureau of Standards Interagency Report (NBSIR) 81-2404, November 1981.
- [6] ASHRAE Handbook - 1977 Fundamentals, ASHRAE, 1981, p. 21.4.
- [7] ASHRAE Handbook - 1981 Fundamentals, ASHRAE, 1981, p. 4.13, Figure 18.
- [8] Conte, S. D. and De Boor, C., Elementary Numerical Analysis, an Algorithmic Approach, McGraw-Hill, 1972, pp. 84-88.
- [9] Brown, W. G., and Solvason, K. R., "Natural Convection Through Rectangular Openings in Partitions - 1: Vertical Partitions", Int. Journal of Heat and Mass Transfer, Vol 5, 1962, pp. 859-867.
- [10] Brown, W. G., "Natural Convection Through Rectangular Openings in Partitions - 2: Horizontal Partitions", International Journal of Heat and Mass Transfer, Vol 5, 1962, pp. 869-881.
- [11] Weber, D. D., and Kearney, R. J., "Natural Convective Heat Transfer Through an Aperture in Passive Solar Heated Buildings", 5th National Passive Solar Conference, 1980, pp. 1037-1041.
- [12] Weber, D. D., "Similitude Modeling of Natural Convection Heat Transfer Through an Aperture in Passive Solar Heated Buildings", Los Alamos Scientific Laboratory Report LA-8385-T, June 1981, 114 p.

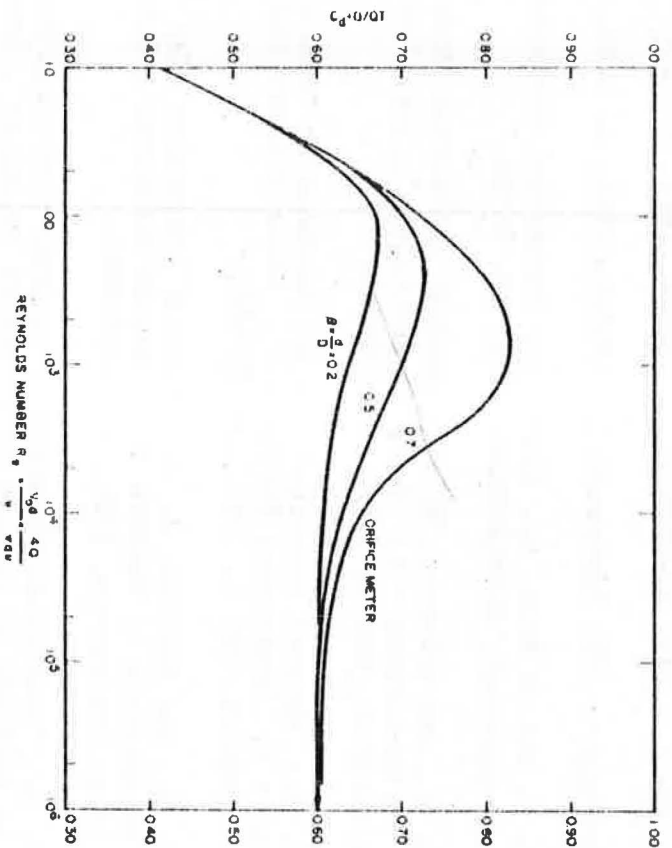


Figure 1. Orifice flow meter coefficients [6]

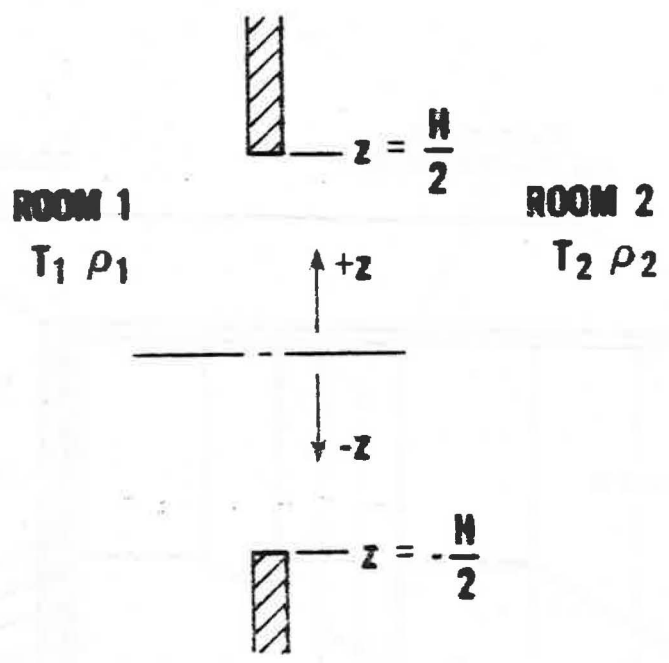


Figure 2. Schematic of opening in a vertical partition

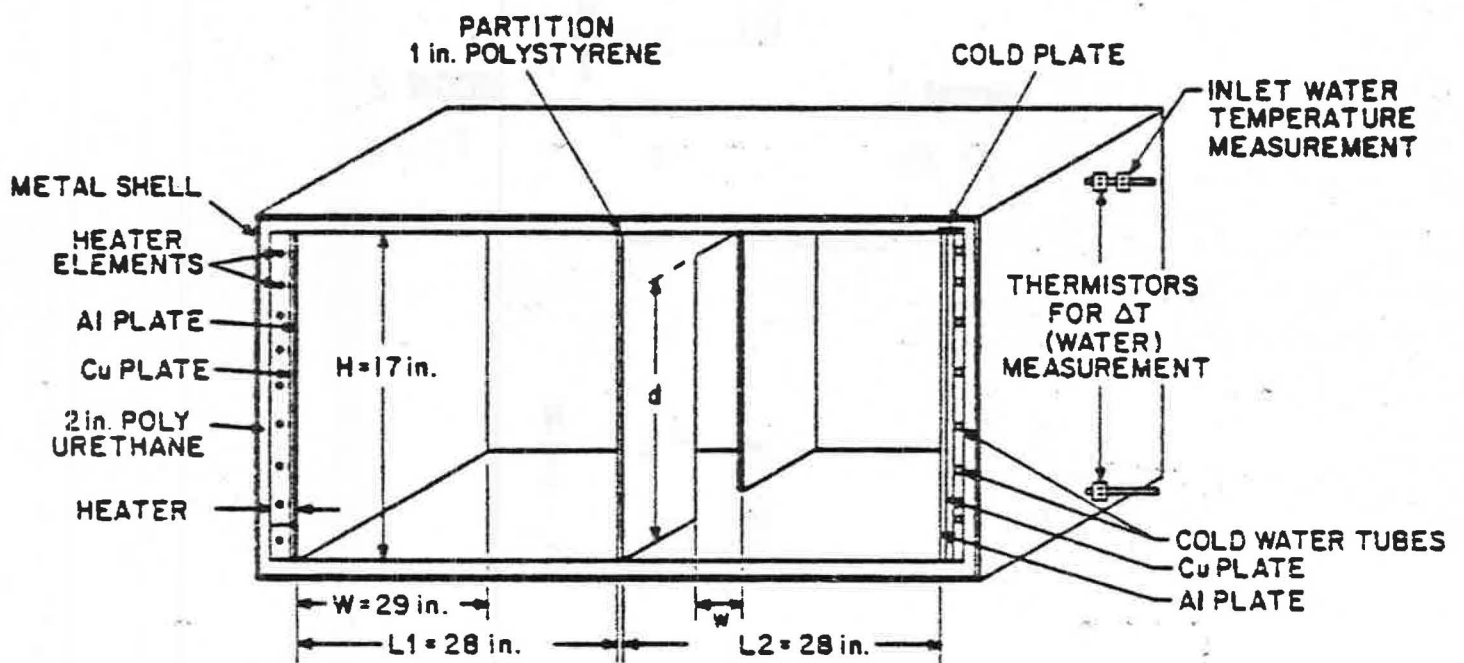


Figure 3. Two-room model for determining natural convection transfer through the doorway [11]

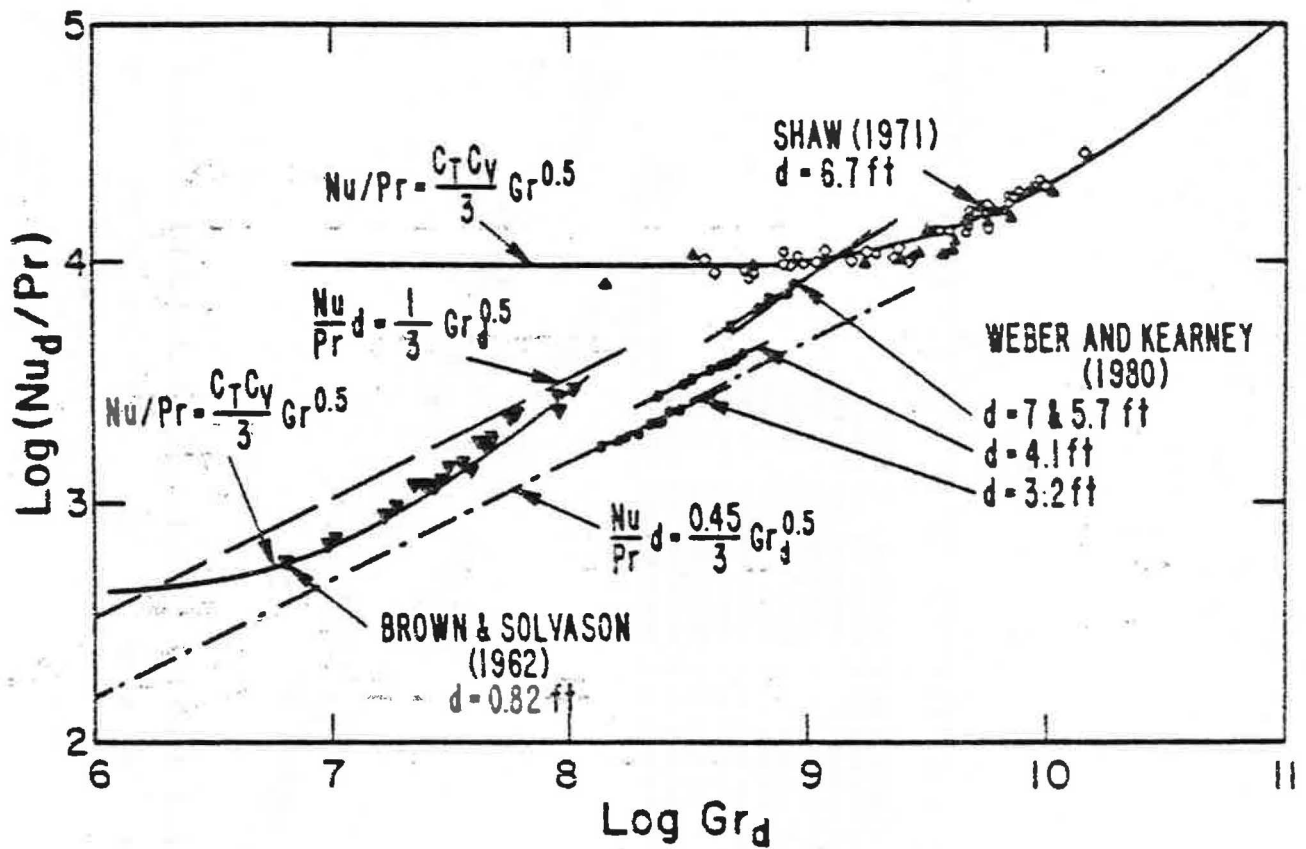


Figure 4. Comparison of experimental results for convection through openings in vertical partitions [12]

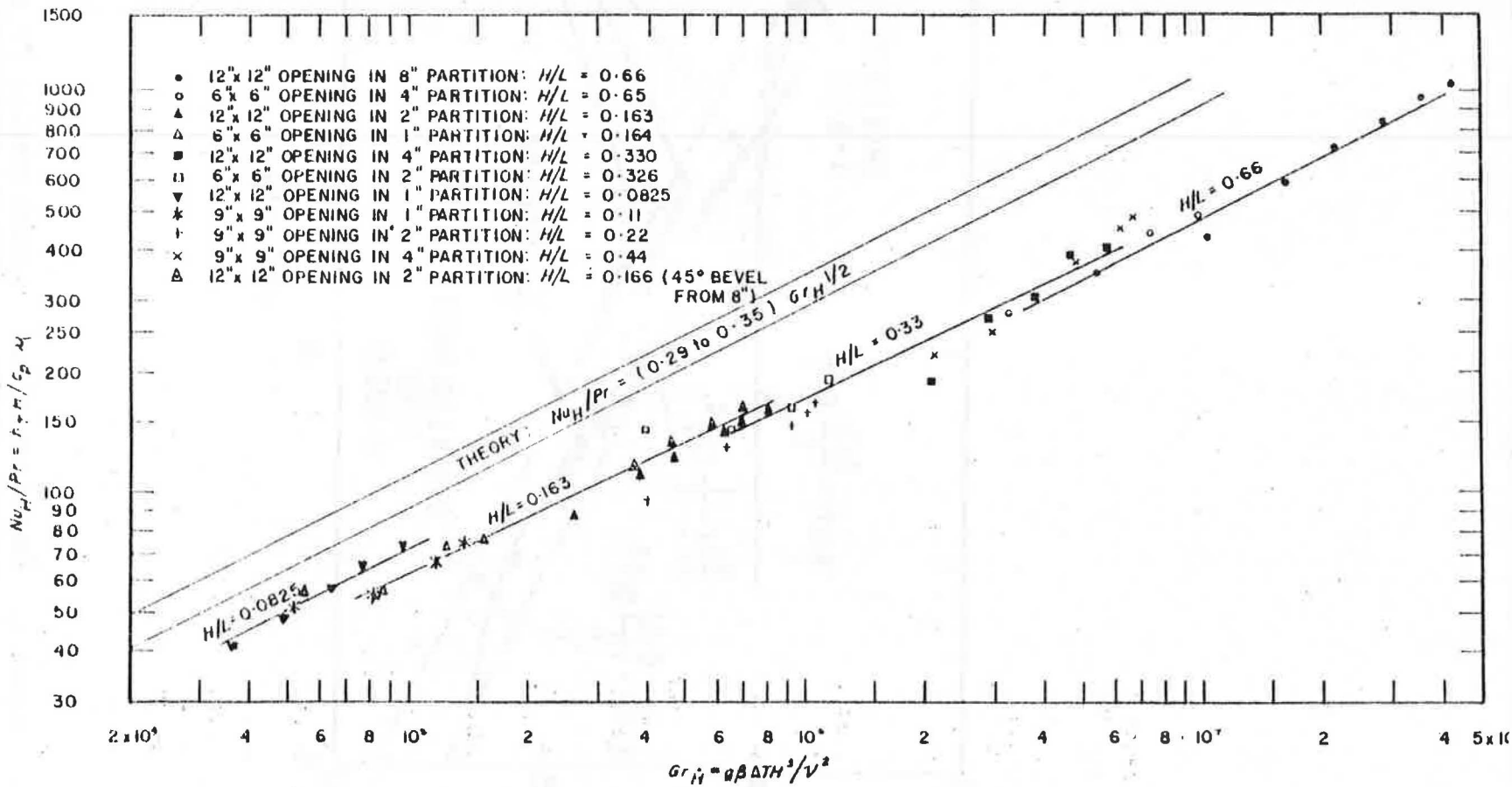


Figure 5. Experimental results for air flow through a square opening in a horizontal partition [10]

APPENDIX A: LISTING OF AIR FLOW TEST PROGRAM

The subroutines and variables used in this program were prepared for easy inclusion in TARP. Several features could have been simplified or improved if the only goal had been to develop a stand-alone air flow analysis computer program.

Descriptions of variables in common:

ACNVG1 relative air flow convergence.
ACNVG2 minimum air flow (kg/s).
ACNVG3 minimum pressure difference (Pa).
AFSPTR air flow opening pointers (near side and far side room numbers).
AIRDEN ambient air density (kg/m³).
AMAXIT maximum air flow mass balance iterations.
CPAIR specific heat of air (kWh/kg C).
DIR wind direction (degrees).
DTR conversion factor: degrees * DTR = radians.
FAHS air flow from the air handling system (m³/s).
FAREA effective flow area (m²).
FAZM azimuth angle of surface (degrees).
FEXP air flow exponent.
FRACT fraction of computed correction (to speed convergence).
HCOUNT count of heat balance iterations.
LIST printed output control flag (0 = least detail, 2 = most).
MAXAFS parameter: maximum number of air flow openings.
MAXZON parameter: maximum number of zones.
MCPM inward mass flow rate time specific heat (W/C)
MCPTM inward energy flow of moving air (W).
NAFS number of air flow openings.
NZON number of zones (rooms).
OBP ambient barometric pressure (Pa).
ODB ambient dry bulb temperature (C).
PS stack pressure (Pa).
PW wind pressure (Pa).
PZ zone (room) air pressures (Pa). [zone 0 = ambient]
SPD wind speed (m/s)
SQRTDZ square root of the zone air density ($\sqrt{\text{kg/m}^3}$) . [zone 0 = ambient]
STDTIM hour counter (standard time).
SUMAF sum of (unsigned) zone air flows (kg/s).
TZ zone air temperature (C). [zone 0 = ambient]
ZS height of opening (m).
ZT maximum height of surface(m).
ZZ height of zone (m). [zone 0 = ambient]

Listing of subroutines:

The driver program, MAIN, is structured to test convergence for various convergence limits and ambient conditions for a single building configuration. Other driver programs were written to test air flows through openings for

comparison to the experimental results. This routine uses NAMELIST input for the control variables. A sample input deck is shown in Appendix B.

```

1      PROGRAM MAIN
2      C
3      COMMON /ENVT/ ODB(1),OBP(1),SPD(1),DIR(1),AIRDEN,CPAIR,DTR,STDTIM,
4      -          LIST,HCOUNT,ACNVG1,ACNVG2,ACNVG3,AMAXIT,FRACT,NAFS,NZON
5      C
6      INTEGER  STDTIM, HCOUNT, AMAXIT, NAFS, NZON
7      REAL    ODB, OBP, SPD, DIR, AIRDEN, CPAIR, DTR,
8      -          ACNVG1, ACNVG2, ACNVG3, FRACT
9      C
10     NAMELIST /CONTROL/ AMAXIT,ACNVG1,ACNVG2,ACNVG3,ODB,OBP,SPD,
11     -          DIR,LIST
12     C
13     CALL INITAIR
14     10 CONTINUE
15     READ CONTROL
16     IF(AMAXIT.LE.0) STOP'END OF RUN'
17     PRINT CONTROL
18     CALL AIRMOV
19     GO TO 10
20     END

```

Subroutine INITAIR initializes most of the variables for the air flow simulation. In particular, it includes the statements that read the user data for the zone (line 56) and opening (line 74) descriptions. Wind and stack pressures are set to zero (lines 91, 92). Wind pressure (PW) remains zero for all inter-room openings.

```

1      SUBROUTINE INITAIR
2      C
3      REAL    Z, T, F
4      C
5      INTEGER MAXAFS
6      PARAMETER (MAXAFS=128)
7      C
8      INTEGER MAXZON
9      PARAMETER (MAXZON=36)
10     C
11     COMMON /AFSL/ FAREA(MAXAFS),FEXP(MAXAFS),ZS(MAXAFS),ZT(MAXAFS),
12     -          PW(MAXAFS),PS(MAXAFS),FAZM(MAXAFS),AFSPTR(2,MAXAFS)
13     C
14     INTEGER  AFSPTR
15     REAL    FAREA, FEXP, ZS, PW, PS, FAZM, ZT
16     C
17     COMMON /ENVT/ ODB(1),OBP(1),SPD(1),DIR(1),AIRDEN,CPAIR,DTR,STDTIM,
18     -          LIST,HCOUNT,ACNVG1,ACNVG2,ACNVG3,AMAXIT,FRACT,NAFS,NZON
19     C
20     INTEGER  STDTIM, HCOUNT, AMAXIT, NAFS, NZON
21     REAL    ODB, OBP, SPD, DIR, AIRDEN, CPAIR, DTR,

```

```

22 - ACNMG1, ACNMG2, ACNMG3, FRACT
23 C
24 COMMON /ZONL/ MCPM(MAXZON),MCPTM(MAXZON),TZ(0:MAXZON),
25 - ZZ(0:MAXZON),SQRTDZ(0:MAXZON),PZ(0:MAXZON),
26 - FAHS(MAXZON),SUMAF(MAXZON)
27 C
28 REAL TZ, ZZ, SQRTDZ, PZ, FAHS, MCPM, MCPTM, SUMAF
29 C
30 C INITIALIZE VARIABLES.
31 HCOUNT=1
32 STDTIM=1
33 AMAXIT=20
34 FRACT=.55
35 LIST=1
36 NAFS=0
37 NZON=0
38 DTR=0.0174533
39 ACNMG1=.0001
40 ACNMG2=.000005
41 ACNMG3=.00001
42 CPAIR=1004.
43 ODB(1)=0.
44 OBP(1)=100000.
45 SPD(1)=5.
46 DIR(1)=270.
47 SQRT2=SQRT(2.0)
48 WRITE(*,101)
49 101 FORMAT('1',8X,'N',6X,'ZZ',6X,'TZ',4X,'FAHS')
50 10 CONTINUE
51 C READ ZONE DATA:
52 C HEIGHT, TEMP, SYSTEM FLOW
53 READ *, Z,T,F
54 IF (Z.LT.0.0) GO TO 20
55 NZON=NZON+1
56 WRITE(*,102) NZON,Z,T,F
57 102 FORMAT(' ZON:',I4,3F8.2)
58 ZZ(NZON)=Z
59 TZ(NZON)=T
60 FAHS(NZON)=F
61 GO TO 10
62 20 CONTINUE
63 WRITE(*,103)
64 103 FORMAT('0',8X,'I N M',7X,'A',7X,'X',7X,'C',5X,'AZM',
65 - 6X,'ZS',6X,'ZT')
66 30 CONTINUE
67 C READ OPENING DATA:
68 C NEAR SIDE ZONE, FAR SIDE ZONE,
69 C AREA, EXPONENT, FLOW COEFFICIENT,
70 C AZIMUTH, HEIGHT, SURFACE HEIGHT
71 READ *, N,M,A,X,C,AZ,Z1,Z2
72 IF (N.LE.0) GO TO 40

```

```

73      NAFS=NAFS+1
74      WRITE(*,104) NAFS,N,M,A,X,C,AZ,Z1,Z2
75 104   FORMAT(' AFS:',3I4,3F8.4,3F8.2)
76      AFSPTR(1,NAFS)=N
77      AFSPTR(2,NAFS)=M
78      FAREA(NAFS)=SQRT2*C*A
79      FEXP(NAFS)=X
80      FAZM(NAFS)=AZ
81      ZS(NAFS)=Z1
82      ZT(NAFS)=Z2
83      GO TO 30
84      40 CONTINUE
85  C      COMPUTE VARIABLES.
86      TZ(0)=ODB(1)
87      AIRDEN=0.0034838*OBP(1)/(ODB(1)+273.15)
88      ZZ(0)=0.0
89      PZ(0)=0.0
90      DO 50 I=1,NAFS
91          PS(I)=0.0
92          PW(I)=0.0
93      50 CONTINUE
94      RETURN
95      END

```

Subroutine AIRMOV computes the room air densities (lines 31-33), the wind and stack pressures for each opening (lines 35-52), calls for solution of the mass balance (line 54), Computes the mass and energy flows into each room (lines 56-76), and reports the room and opening air flows (lines 74-90).

Descriptions of local variables:

I index for openings.
N room number.
M adjacent room number (0 = ambient).
V wind speed modified for height (m/s).
W wind pressure at given height (Pa).
Y relative surface to wind direction (degrees).
X wind pressure direction modifier.
DP pressure difference across opening (Pa).
MCP mass flow rate times specific heat of air flow into room (W/C).

```

1      SUBROUTINE AIRMOV
2      C
3          REAL MCP
4      C
5          INTEGER MAXAFS
6      PARAMETER (MAXAFS=128)
7      C
8          INTEGER MAXZON
9      PARAMETER (MAXZON=36)
10     C
11     COMMON /AFSL/ FAREA(MAXAFS),FEXP(MAXAFS),ZS(MAXAFS),ZT(MAXAFS),
12     -          PW(MAXAFS),PS(MAXAFS),FAZM(MAXAFS),AFSPTR(2,MAXAFS)
13     C
14     INTEGER AFSPTR
15     REAL    FAREA, FEXP, ZS, PW, PS, FAZM, ZT
16     C
17     COMMON /ENVT/ ODB(1),OBP(1),SPD(1),DIR(1),AIRDEN,CPAIR,DTR,STDTIM,
18     -          LIST,HCOUNT,ACNVG1,ACNVG2,ACNVG3,AMAXIT,FRACT,NAFS,NZON
19     C
20     INTEGER STDTIM, HCOUNT, AMAXIT, NAFS, NZON
21     REAL    ODB, OBP, SPD, DIR, AIRDEN, CPAIR, DTR,
22     -          ACNVG1, ACNVG2, ACNVG3, FRACT
23     C
24     COMMON /ZONL/ MCPM(MAXZON),MCPTM(MAXZON),TZ(0:MAXZON),
25     -          ZZ(0:MAXZON),SQRTDZ(0:MAXZON),PZ(0:MAXZON),
26     -          FAHS(MAXZON),SUMAF(MAXZON)
27     C
28     REAL    TZ, ZZ, SQRTDZ, PZ, FAHS, MCPM, MCPTM, SUMAF
29     C
30     C          COMPUTE ZONE AIR DENSITIES.
31     DO 10 N=0,NZON
32         SQRTDZ(N)=SQRT(0.0034838*OBP(STDTIM)/(TZ(N)+273.15))
33     10 CONTINUE
34     C          COMPUTE CONSTANT DELTA-P.
35     DO 20 I=1,NAFS
36         N=AFSPTR(1,I)
37         M=AFSPTR(2,I)
38         IF(M.EQ.0 .AND. HCOUNT.EQ.1) THEN
39     C          WIND PRESSURE.
40             V=SPD(STDTIM)*(0.1*ZT(I))**0.143
41             W=0.5*AIRDEN*V*V
42             Y=AMAX1(FAZM(I),DIR(STDTIM))-AMIN1(FAZM(I),DIR(STDTIM))
43             IF(Y.GT.180.) Y=360.-Y
44             IF(Y.LE.90.) X=0.75-1.05/90.*Y
45             IF(Y.GT.90.) X=0.15/90.*Y-0.45
46             PW(I)=X*W
47         END IF
48     C          STACK PRESSURE.
49         PS(I)=9.80*((ZZ(M)-ZS(I))*SQRTDZ(M)*SQRTDZ(M)-
50     -          (ZZ(N)-ZS(I))*SQRTDZ(N)*SQRTDZ(N))
51     IF(LIST.GE.1) PRINT *, 'CONSTS: ',I,PS(I),PW(I)

```

```

52      20  CONTINUE
53      C      COMPUTE ZONE PRESSURES TO CONVERGENCE.
54      CALL SOLVZP
55      C      COMPUTE ZONE FLOWS.
56      DO 30 N=1,NZON
57          MCPM(N)=0.0
58          MCPTM(N)=0.0
59      30  CONTINUE
60      DO 40 I=1,NAFS
61          N=AFSPTR(1,I)
62          M=AFSPTR(2,I)
63          DP=PS(I)+PW(I)+PZ(M)-PZ(N)
64          IF(DP.GT.0.0) THEN
65              MCP=CPAIR*FAREA(I)*SQRTDZ(M)*DP**FEXP(I)
66              MCPM(N)=MCPM(N)+MCP
67              MCPTM(N)=MCPTM(N)+MCP*TZ(M)
68          ELSE IF(M.GT.0) THEN
69              MCP=CPAIR*FAREA(I)*SQRTDZ(N)*(-DP)**FEXP(I)
70              MCPM(M)=MCPM(M)+MCP
71              MCPTM(M)=MCPTM(M)+MCP*TZ(N)
72          END IF
73      40  CONTINUE
74      DO 50 N=1,NZON
75          IF(LIST.GE.1) PRINT *, 'MCPM: ',N,PZ(N),MCPM(N)
76      50  CONTINUE
77      C      IF(LIST.GE.1) PRINT FLOWS.
78      DO 60 I=1,NAFS
79          N=AFSPTR(1,I)
80          M=AFSPTR(2,I)
81          F=0.0
82          DP=PS(I)+PW(I)+PZ(M)-PZ(N)
83          IF(DP.LT.0.0) THEN
84              F=-FAREA(I)*SQRTDZ(N)*(-DP)**FEXP(I)
85              IF(LIST.GE.1) PRINT *, 'FLOW: ',I,M,FAREA(I),SQRTDZ(N),DP,F
86          ELSE IF(DP.GT.0.0) THEN
87              F=FAREA(I)*SQRTDZ(M)*DP**FEXP(I)
88              IF(LIST.GE.1) PRINT *, 'FLOW: ',I,M,FAREA(I),SQRTDZ(M),DP,F
89          END IF
90      60  CONTINUE
91      RETURN
92      END

```

Subroutine SOLVZP computes the room air pressures. It consists of an initialization section (lines 27-54) and an iteration section (lines 56-116). The initialization is done by computing the room air pressures using flow exponents of 1. This produces a set of linear equations which are solved by Gauss elimination (line 50). In TARP, initialization is done only on the first heat balance iteration of each hour (line 32); otherwise, it is better to use the most recently computed room pressures. The components of the Jacobian matrix (AA) and the net room air flows are computed between lines 68 and 95. The test at line 72 prevents division by zero at line 79 and does not otherwise effect the solution. The test at line 91 accounts for rooms where the net air flow is too small to effect the room energy requirement and avoids a possible indefinite result (0/0) at line 92. The primary convergence test at line 92 compares relative flows. The Newton's method matrix is solved by gauss elimination at line 96. The TCOUNT variable prevents endless iteration in case of failure to converge (line 99). The simple improvement for successive over-corrections is in lines 103 through 106.

Descriptions of local variables:

I index for openings.
 N room number.
 M adjacent room number (0 = ambient).
 DP pressure difference across opening (Pa).
 F mass flow rate through the opening (kg/s).
 DF derivative of the flow rate.
 AA Jacobian matrix.
 BB sum of air flows into room (kg/s).
 CC Newton correction to the room pressures (Pa).
 DD Prior iteration corrections (Pa).

```

1      SUBROUTINE SOLVZP
2          INTEGER MAXAFS
3      PARAMETER (MAXAFS=128)
4      C
5          INTEGER MAXZON
6      PARAMETER (MAXZON=36)
7      C
8      COMMON /AFSL/ FAREA(MAXAFS),FEXP(MAXAFS),ZS(MAXAFS),ZT(MAXAFS),
9      -            PW(MAXAFS),PS(MAXAFS),FAZM(MAXAFS),AFSPTR(2,MAXAFS)
10     C
11     INTEGER AFSPTR
12     REAL    FAREA, FEXP, ZS, PW, PS, FAZM, ZT
13     C
14     COMMON /ENVT/ ODB(1),OBP(1),SPD(1),DIR(1),AIRDEN,CPAIR,DTR,STDTIM,
15     -            LIST,HCOUNT,ACNVG1,ACNVG2,ACNVG3,AMAXIT,FRACT,NAFS,NZON
16     C
17     INTEGER STDTIM, HCOUNT, AMAXIT, NAFS, NZON
18     REAL    ODB, OBP, SPD, DIR, AIRDEN, CPAIR, DTR,
19     -            ACNVG1, ACNVG2, ACNVG3, FRACT
20     C
21     COMMON /ZONL/ MCPM(MAXZON),MCPTM(MAXZON),TZ(0:MAXZON),

```

```

22 -          ZZ(0:MAXZON),SQRTDZ(0:MAXZON),PZ(0:MAXZON),
23 -          FAHS(MAXZON),SUMAF(MAXZON)
24 C
25     REAL    TZ, ZZ, SQRTDZ, PZ, FAHS, MCPM, MCPTM, SUMAF
26 C
27     INTEGER TCOUNT
28     LOGICAL CNVG
29     REAL    AA(MAXZON,MAXZON),BB(MAXZON),CC(MAXZON),DD(MAXZON)
30 IF(LIST.GE.1) PRINT *, 'INITIALIZATION'
31 TCOUNT=0
32 IF(HCOUNT.EQ.1) GO TO 30
33 DO 10 N=1,NZON
34     DD(N)=1.0
35     BB(N)=FAHS(N)
36     DO 10 M=1,NZON
37 10     AA(N,M)=0.0
38     DO 20 I=1,NAFS
39     N=AFSPTR(1,I)
40     M=AFSPTR(2,I)
41     AA(N,N)=AA(N,N)-FAREA(I)
42     BB(N)=BB(N)-FAREA(I)*(PS(I)+PW(I))
43     IF(M.LE.0) GO TO 20
44     AA(M,M)=AA(M,M)-FAREA(I)
45     AA(N,M)=AA(N,M)+FAREA(I)
46     AA(M,N)=AA(M,N)+FAREA(I)
47     BB(M)=BB(M)+FAREA(I)*(PS(I)+PW(I))
48 20     CONTINUE
49 IF(LIST.GE.2) CALL DUMPAB(AA,BB,NZON,MAXZON)
50 CALL GAUSSY(AA,BB,CC,NZON,MAXZON)
51 DO 25 N=1,NZON
52     PZ(N)=CC(N)
53     IF(LIST.GE.1) PRINT *, 'PZ: ',N,PZ(N)
54 25     CONTINUE
55 C          NEWTON ITERATION.
56 30 CONTINUE
57     TCOUNT=TCOUNT+1
58     IF(LIST.GE.1) PRINT *, 'BEGIN ITERATION ',TCOUNT
59     CNVG=.FALSE.
60     DO 40 N=1,NZON
61     BB(N)=FAHS(N)
62     SUMAF(N)=0.0
63     DO 40 M=1,NZON
64     AA(M,N)=0.0
65 40     CONTINUE
66 C          EVALUATE FUNCTIONS AND
67 C          PARTIAL DERIVATIVES.
68     DO 50 I=1,NAFS
69     N=AFSPTR(1,I)
70     M=AFSPTR(2,I)
71     DP=PZ(M)-PZ(N)+PS(I)+PW(I)
72     IF(ABS(DP).LT.ACNVG3*

```

```

73 - (ABS(PZ(M))+ABS(PZ(N))+ABS(PS(I))+ABS(PW(I)))) GO TO 50
74 IF(DP.LT.0.0) THEN
75 F=-FAREA(I)*SQRTDZ(N)*(-DP)**FEXP(I)
76 ELSE
77 F=FAREA(I)*SQRTDZ(M)*DP**FEXP(I)
78 END IF
79 DF=F**FEXP(I)/DP
80 BB(N)=BB(N)+F
81 SUMAF(N)=SUMAF(N)+ABS(F)
82 AA(N,N)=AA(N,N)-DF
83 IF(M.LE.0) GO TO 50
84 BB(M)=BB(M)-F
85 SUMAF(M)=SUMAF(M)+ABS(F)
86 AA(M,M)=AA(M,M)-DF
87 AA(N,M)=AA(N,M)+DF
88 AA(M,N)=AA(M,N)+DF
89 50 CONTINUE
90 DO 60 N=1,NZON
91 IF(ABS(BB(N)).LE.ACNVG2) GO TO 60
92 IF(ABS(BB(N)/SUMAF(N)).GT.ACNVG1) GO TO 70
93 60 CONTINUE
94 CNVG=.TRUE.
95 70 CONTINUE
96 IF(LIST.GE.2) CALL DUMPAB(AA,BB,NZON,MAXZON)
97 C CHECK CONVERGENCE.
98 IF(CNVG) GO TO 999
99 IF(TCOUNT.GT.AMAXIT) STOP 'ITERATIONS'
100 C SOLVE AA * CC = BB.
101 CALL GAUSSY(AA,BB,CC,NZON,MAXZON)
102 C IMPROVE CONVERGENCE.
103 DO 80 N=1,NZON
104 IF(CC(N)/DD(N).LE.-0.5) CC(N)=CC(N)*FRACT
105 IF(ABS(CC(N)).GT.ACNVG3) DD(N)=CC(N)
106 80 CONTINUE
107 C REVISE PZ.
108 DO 90 N=1,NZON
109 PZ(N)=PZ(N)-CC(N)
110 IF(LIST.GE.1) PRINT *, 'REVIS: ',N,DD(N),CC(N),PZ(N),BB(N)
111 - ,SUMAF(N)
112 90 CONTINUE
113 GO TO 30
114 C
115 999 CONTINUE
116 PRINT *, 'ITERATIONS ',TCOUNT
117 RETURN
118 END

```


Subroutine GAUSSY is a Gauss elimination routine for computing the correction terms in Newton's method. The elimination is done in lines 7 through 12. There is no pivoting because the matrix is diagonally dominant and row exchanges would never occur. Lines 14 through 18 allow for an indeterminate solution which occurs if, for example, we are solving for the air flow between two rooms but have no openings from the rooms to ambient. The back substitution is concluded in lines 19 through 23.

```

1      SUBROUTINE GAUSSY(A,B,X,N,MAX)
2      C
3          REAL      A(MAX,MAX), B(MAX), X(MAX)
4      C
5      C          GAUSS ELIMINATION.
6      C          NO PIVOTING.
7          DO 10 K=1,N-1
8              DO 10 I=K+1,N
9                  D=A(I,K)/A(K,K)
10                 B(I)=B(I)-B(K)*D
11                 DO 10 J=K+1,N
12                 A(I,J)=A(I,J)-A(K,J)*D
13      C          BACK SUBSTITUTION.
14         IF(ABS(A(N,N)).LT.1.E-12) THEN
15             X(N)=0.0
16         ELSE
17             X(N)=B(N)/A(N,N)
18         END IF
19         DO 30 I=N-1,1,-1
20             D=0.0
21             DO 20 J=I+1,N
22             20     D=D+A(I,J)*X(J)
23             30     X(I)=(B(I)-D)/A(I,I)
24      C
25         RETURN
26         END

```

Subroutine DUMPAB will dump the contents of the Jacobian matrix (A) and the net room air flows (B) when detailed output has been requested (LIST=2).

```

1      SUBROUTINE DUMPAB(A,B,N,MAX)
2      C
3          REAL      A(MAX,MAX), B(MAX)
4      C
5          DO 10 I=1,N
6          10     WRITE(*,101) I,(A(I,J),J=1,N),B(I)
7          101   FORMAT(I4,11E11.4)
8          RETURN
9          END

```

APPENDIX B: SAMPLE INPUT FOR THE AIR FLOW TEST PROGRAM

The following sample input deck describes a nine room, single story building as shown in the plan below. An opening is simulated on every wall. The first nine lines give the room height, temperature, and system air flow (read at line 53 of INITAIR) for each room. Room input is terminated at the line with a negative height. The next 24 lines describe the openings in the envelope and partition walls (read at line 71 of INITAIR). Each line gives the room number (room numbers must be sequential from 1), the number of the room on the opposite side of the wall (0 = ambient), the opening area, the flow exponent, the flow coefficient, the azimuth angle of the wall, the height of the opening, and the height of the wall. Openings in partitions are described only once. The openings information is terminated by the negative room number. The next line gives the convergence, wind, and output control data (NAMELIST input read at line 15 of MAIN). Simulation is terminated when the permitted air flow iterations (AMAXIT) are zero.

```

1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
1.5 20.0 0.0
-1.0 20. 0.0
1 0 .02 .66 .60 0. 1.5 3.0
1 0 .02 .66 .60 90. 1.5 3.0
1 2 .16 .50 .60 180. 1.5 3.0
1 4 .16 .50 .60 270. 1.5 3.0
2 0 .02 .66 .60 90. 1.5 3.0
2 3 .16 .66 .60 180. 1.5 3.0
2 9 .04 .50 .60 270. 1.5 3.0
3 0 .02 .66 .60 90. 1.5 3.0
3 0 .02 .66 .60 180. 1.5 3.0
3 5 .16 .50 .60 270. 1.5 3.0
4 0 .02 .66 .60 0. 1.5 3.0
4 9 .04 .50 .60 180. 1.5 3.0
4 6 .16 .50 .60 270. 1.5 3.0
5 0 .02 .66 .60 180. 1.5 3.0
5 8 .16 .50 .60 270. 1.5 3.0
5 9 .04 .50 .60 0. 1.5 3.0
6 0 .02 .66 .60 0. 1.5 3.0
6 7 .16 .50 .60 180. 1.5 3.0
6 0 .02 .66 .60 270. 1.5 3.0
7 9 .04 .50 .60 90. 1.5 3.0
7 8 .16 .50 .60 180. 1.5 3.0
7 0 .02 .66 .60 270. 1.5 3.0

```

↑
N
O
R
T
H
|

6	4	1
7	9	2
8	5	3

8 0 .02 .66 .60 180. 1.5 3.0
8 0 .02 .66 .60 270. 1.5 3.0
-1 0 .02 .66 .60 270. 4.5 6.0

AMAXIT=20,ACNVG1=.01,ACNVG2=.00001,ACNVG3=.00001,SPD=5.000,DIR=0.,LIST=1 /
AMAXIT=0 /