APPENDIX A

Environmental System Performance Program (ESP-r)

In this appendix, the approach employed in ESP-r to simulate energy flow within buildings is described. This is done by employing a simple building only (i.e. no air flow or plant simulation) problem.

A.1 Introduction

The environmental system performance program (ESP-r) is a tool for the transient simulation of heat and fluid flow within combined building/plant systems with control imposed. The structure of ESP-r is shown in Figure A.1. Within the ESP-r environment, a simulation problem is defined by the Project Manager which creates a set of data files whose names and locations are saved in a single system configuration file. By defining the system configuration file name to the Simulator, it will represent the problem by its equivalent network of time dependent thermal resistances and capacitances subjected to dynamic potential differences. By performing a simulation, the Simulator creates a results file which is analysed by the Results Analyser module.

The simulation of a problem within the Simulator is performed in a three stages process: discretisation of the problem to be simulated, derivation of the simulation equation for the nodal system, and simultaneous solution of the derived characteristic equations. In order to show how a problem is simulated within the ESP-r simulation environment, the simple two zone building shown in Figure A.2 will be used. For simplicity, no doors or windows are defined, and all the surfaces are made of the same composite construction. For generality, a two layer construction is assumed.

Since this thesis is concerned with better building fabric modelling, only building energy simulation is considered in this appendix. The theories employed by ESP-r to represent plant simulation and fluid flow are detailed elsewhere (Clarke, 1985; Hensen, 1991; Aasem, 1993; Negrao, 1995).

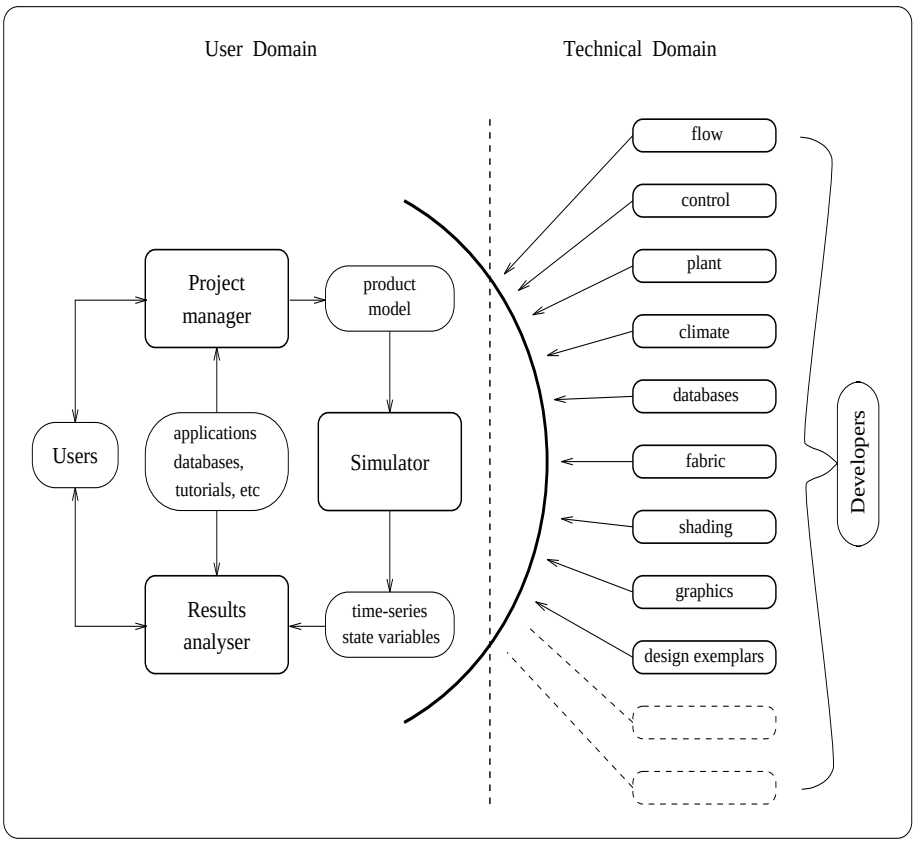

Figure A.1 The ESP-r System (from Clarke, 1994).

A.2 Discretisation

The default ESP-r space discretisation approach is shown in Figure A.3. Each layer of material is represented by two surface nodes (each representing 1/4 of the layer's thermal capacitance) and one central node (representing 1/2 of the layer's thermal capacitance). The two surface nodes at the layers' interface are combined into one heterogeneous control volume represented by a central node.

Each inter-constructional node has two heat conduction connections. However, construction surface nodes have only one conduction connection. Depending on the boundary conditions, the other connections for the construction surface nodes are defined. For generality, convective and radiative boundaries are assumed at both construction surfaces. Furthermore, the building external boundary variables are defined by a climate data file.

The zone air space is represented by one control volume (one node). Within each zone, the air node is in direct thermal contact with the constructions' internal surface nodes. Additionally, the air nodes may be in thermal contact with each other via an air flow network, or with a plant system.

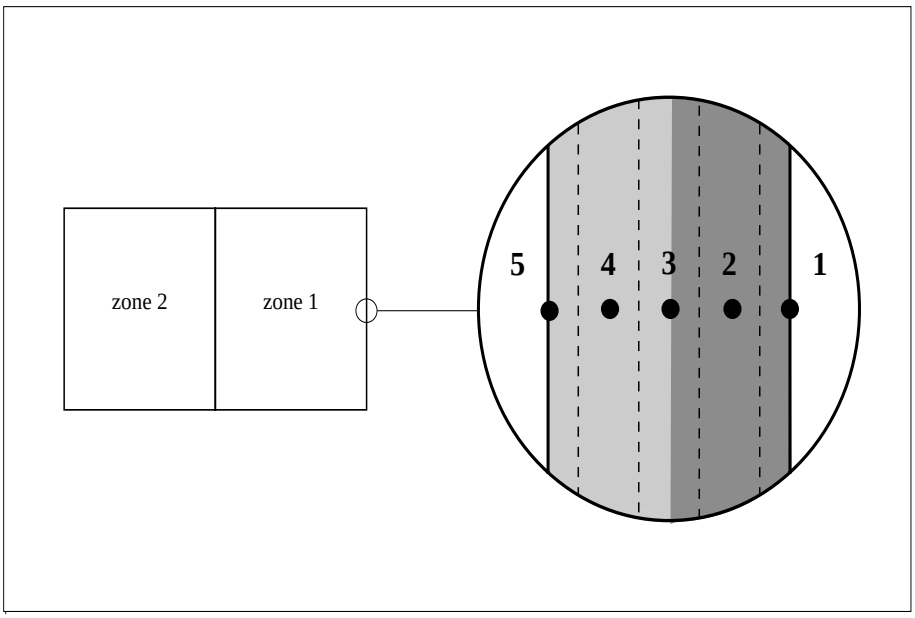

Figure A.2 A two zone simulation problem.

Accordingly, the equivalent nodal scheme for the simulation problem is shown in Figure A.3.

A.3 System matrix generation

The governing partial differential equation of heat conduction is presented in Chapter 2 as

$$\rho\, C_p\, \frac{\partial T(\vec{r},t)}{\partial t} = \nabla\, .\, [\, \lambda\, \nabla T(\vec{r},t)\,] + g(\vec{r},t) \qquad \text{(A.1)}$$

For 1D heat conduction through a multi-layered construction, equation (A.1) can be discretised by using the control volume approach, producing

$$a_i^{n+1}\, T_i^{n+1} + a_{i-1}^{n+1}\, T_{i-1}^{n+1} + a_{i+1}^{n+1}\, T_{i+1}^{n+1} + a_g^{n+1}\, g_i^{n+1} =$$

$$a_i^n\, T_i^n + a_{i-1}^n\, T_{i-1}^n + a_{i+1}^n\, T_{i+1}^n + a_g^n\, g_i^n \qquad \text{(A.2)}$$

Where,

$$a_j^{n+1} = \frac{-\,\gamma\, A\, \lambda_{j\to i}\, \Delta t}{\Delta X_{j\to i}\, \rho_o\, c_p\, V} \quad , \qquad j = 1\, ,\ 2\, .$$

$$a_j^n = \frac{(\,1-\gamma\,)\, A\, \lambda_{j\to i}^n\, \Delta t}{\Delta X_{j\to i}\, \rho_o\, c_p\, V} \quad , \qquad j = 1\, ,\ 2\, .$$

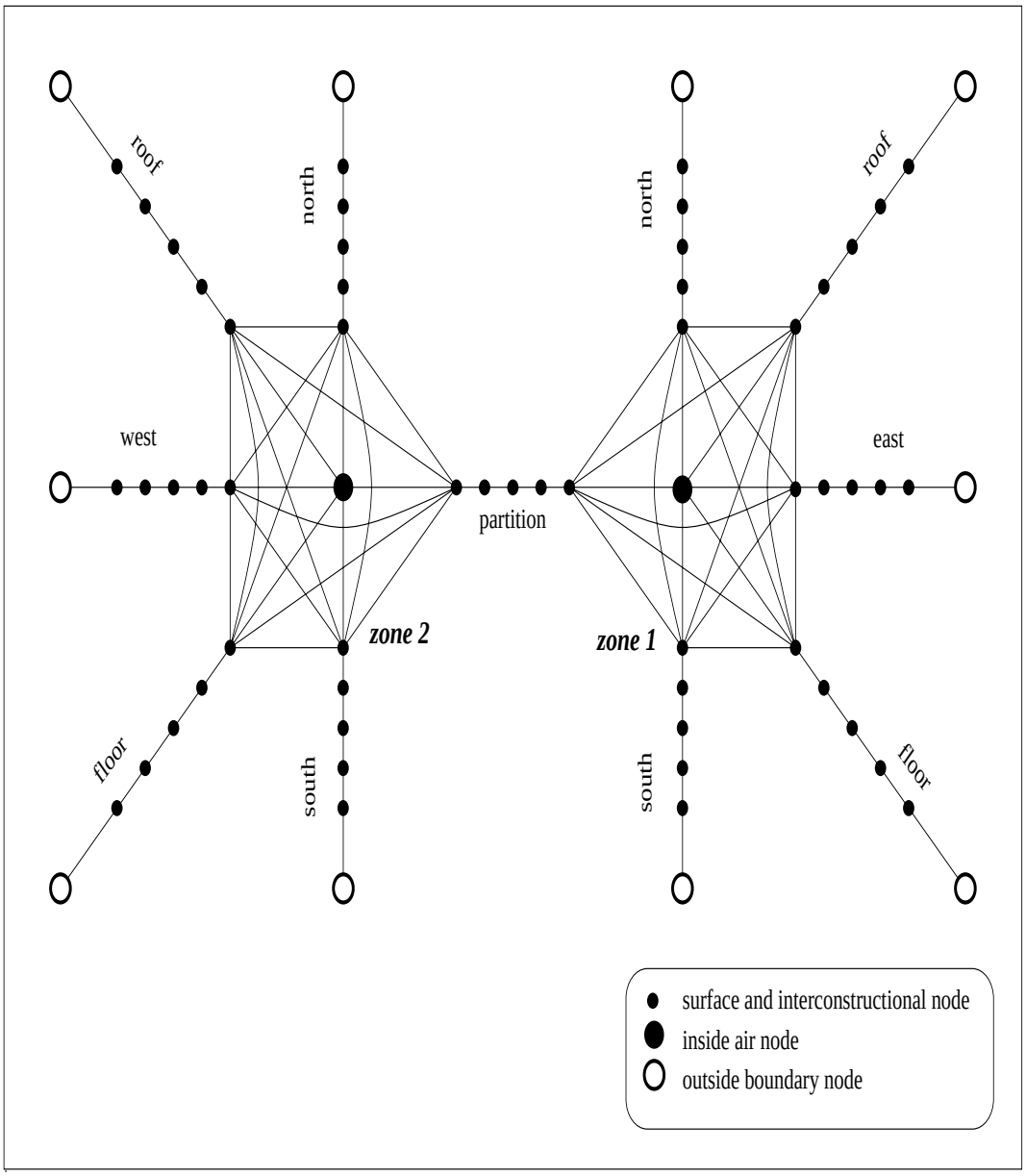

Figure A.3 Simulation problem's nodal scheme.

$$a_i^k = 1.0 - a_{i-1}^k - a_{i+1}^k$$

$$a_g^{n+1} = \frac{-\gamma\ \Delta t}{\rho_o\ c_p}$$

$$a_g^n = \frac{(1-\gamma)\ \Delta t}{\rho_o\ c_p}$$

$$g = g_s + g_p$$

where, g_s is the shortwave energy absorption (for transparent materials), and g_p is the energy absorption by plant interaction (all measured in W). For a boundary (internal or external) node, the convective heat transfer path is numerically represented by modifying the a coefficients for the convective connection to

$$a_j^{n+1} = -\gamma\ A\ h_{j\to i}^{n+1}$$

$$a_j^{n} = (\ 1-\gamma\ )\ A\ h_{j\to i}^{n}$$

where, h is the convective heat transfer coefficient. The same modification is valid for air gap surface nodes (if explicit air gap modelling is required). Furthermore, surface nodes may have radiative heat transfer coefficients for which the heat generation term g must be modified to

$$g = g_s + g_p + g_l + g_r$$

where, g_l is the longwave energy exchange with the surrounding, and g_r is the radiant energy from casual sources. In general other energy terms can be added to account for other complexities such as phase change. All the energy terms are linearised in ESP-r by adopting one time step in arrears approach. An exception is for longwave energy exchange g_l for internal surface nodes, which is estimated simultaneously with the solution process. However, the longwave energy exchange calculation is based on a radiative heat transfer coefficient h_r which is calculated one time step in arrears. That is, the net longwave radiation gained by surface j is estimated by

$$(\ g_l\ )_j^{n+1} = \sum_{\substack{k=1 \\ k\neq j}}^{N} (\ h_r\ )_{k\to j}^{n}\ A_j\ (\ T_k^{n+1} - T_j^{n+1}\ )$$

where N is the number o fsurfaces in the zone.

A.4 System matrix solution

For the 1D heat conduction domain with the enclosed air volume represented by one node, the entire example problem is represented by 57 nodes and so there will be 57 simultaneous equations each having a number of cross- and self-coupling terms evaluated at the present and future time-rows of the active time step within the simulation process. The matrix notation of the corresponding equation set can be written as

$$\boldsymbol{A}\,\boldsymbol{T}^{n+1} = \boldsymbol{B}\,\boldsymbol{T}^{n} + \boldsymbol{C} \qquad \text{(A.3)}$$

Where, $\boldsymbol{A}$ and $\boldsymbol{B}$ are the future and present time coefficient matrices respectively, $\boldsymbol{T}$ is the temperatures and plant flux column matrix, and $\boldsymbol{C}$ is the boundary conditions column matrix. Since the right hand side of equation (A.3) is known at each time step, equation (A.3) can be re-written as

$$\boldsymbol{A}\,\boldsymbol{T}^{n+1} = \boldsymbol{Z} \tag{A.4}$$

The coefficient matrix $\boldsymbol{A}$ for the simulation problem defined above is shown in Figure A.4. The $\boldsymbol{A}$ matrix is sparse and its sparseness increases as the number of nodes increase. Accordingly, sparse storage and solution techniques were adopted in ESP-r. The solution technique is based on partitioning the problem into zones, with partitions modelled in both zones. In addition, this solution approach requires defining the present and future boundaries for each zone. For the unknown boundary variables, the latest available data are used for defining unknown boundaries.

Therefore, the example problem system matrix is divided into two zone matrices. Figure A.5 shows the zone (1) coefficient matrix. By using the Gaussian direct solution method, this set of equations is solved simultaneously. This is done by first performing the forward reduction process. At the end of that process, an equation is generated with two unknowns: the sensor (or air node for free float control) temperature and actuator plant flux. This equation is solved according to the active control algorithm. Then the remaining temperatures are determined by backward substitution.

The sparse storage technique is not only based on building partitioning into zones, but also it is based on partitioning the zone matrix into construction matrices and one matrix for internal surface nodes and air node. For zone (1), the augmented matrix is stored as shown in Figure A.6. Each node within a construction, except the internal surface node, requires 5 storage locations so that there are two locations for cross-coupling, one for self-coupling, one for plant and one for known (i.e. present and boundary) coefficients.

The required number of storage locations for an internal surface node is equal to the number of constructions (NC) plus four, because (NC-1) locations are required for longwave radiation with other internal surface nodes, one location for the self-coupling term, the convective term, the plant term, the known term and the conduction term. The air node requires (NC+3) locations because NC locations are required for convective terms, and one location for the self-coupling term, the plant term and the known term.

References

Aasem, E O1993. , *Practical simulation of buildings and air-conditioning systems in the transient domain*, Glasgow. PhD thesis University of Strathclyde

Clarke, J. A. 1985. *Energy simulation in building design*, Adam Hilger Ltd, Bristol,UK.

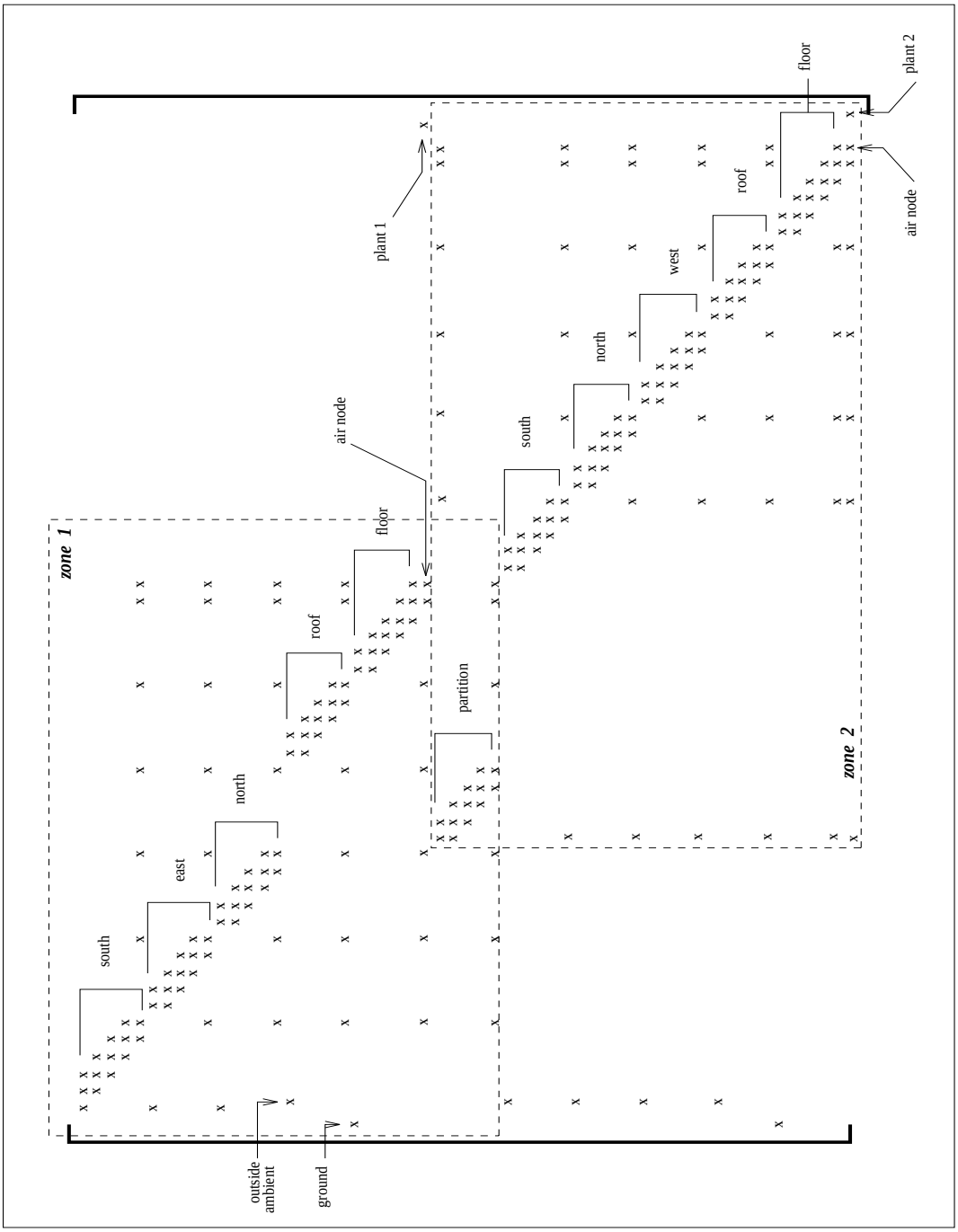

Figure A.4 System matrix.

Hensen, J.L.M. 1991. ‘‘On the thermal interaction of building structure and heating and ventilating system,’’ Ph.D. Thesis, Eindhoven University of Technology.

Negrao, C 1995. ‘‘Conflation of Computational Fluid Dynamics and Building Thermal Simulation,’’ *Phd thesis, University of Strathclyde.*

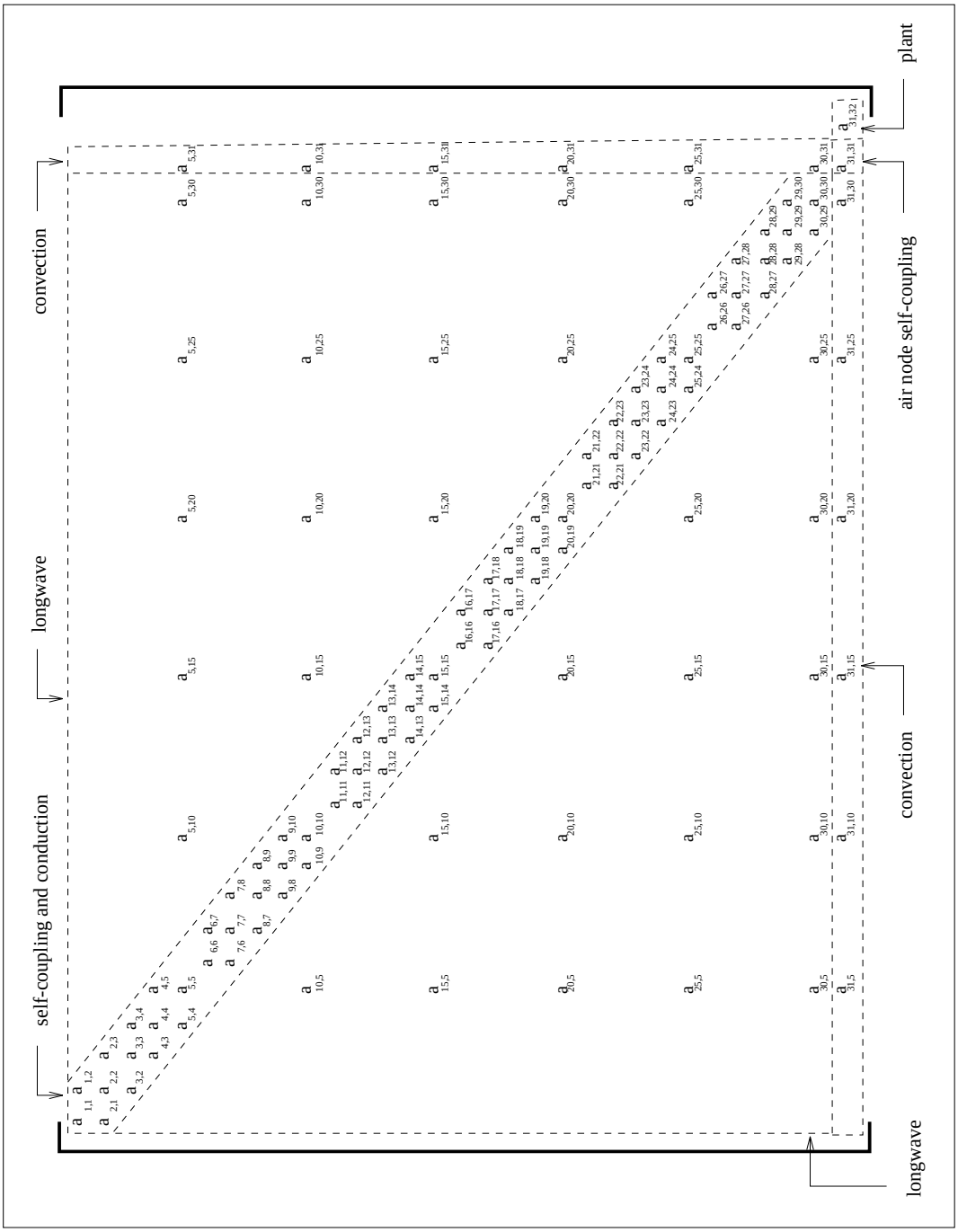

Figure A.5 Zone matrix.

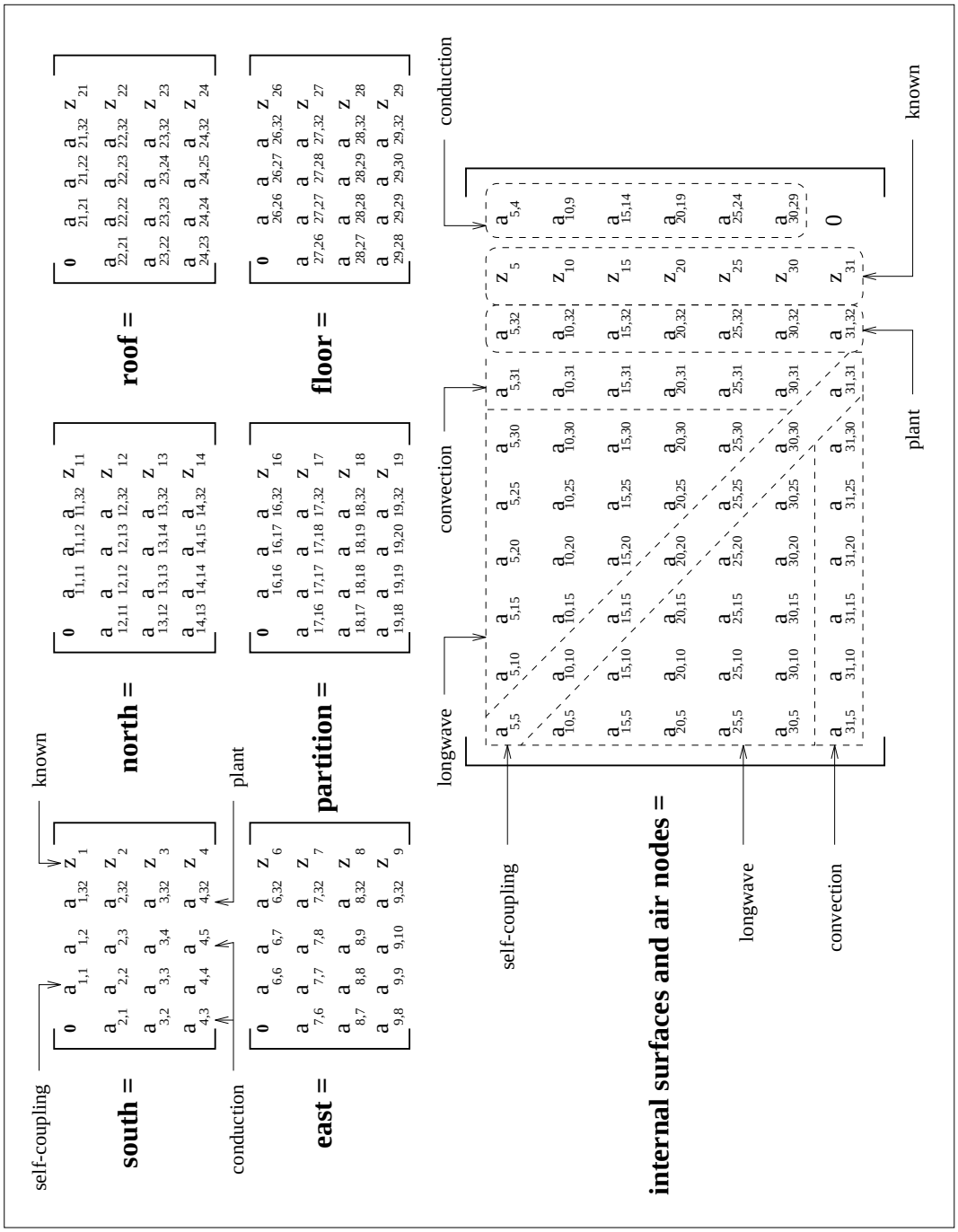

Figure A.6 Sparse storage.

APPENDIX B

Separation of Variables Solution Method

In this appendix, the method of separation of variables is used to solve two types of homogeneous linear diffusion problems. This appendix is referred to from Chapters 4 and 5. Although temperature is the dependent variable in these equations, the required solution for the equivalent moisture transport problem is generated by replacing temperature by partial vapour pressure and thermal diffusivity by hygroscopic diffusivity.

B.1 Problem (1)

For one-dimensional homogeneous linear diffusion through a homogeneous slab ($0 \le x \le d$), the mathematical model is defined by the following partial differential equation with its associated boundary and initial conditions.

$$\frac{\partial^2 T(x,t)}{\partial x^2} = \frac{1}{\alpha}\,\frac{\partial T(x,t)}{\partial t} \quad , \quad 0 < x < d \;\; , \;\; t > 0 \tag{B.1a}$$

$$T = T(x,0) \quad , \quad 0 < x < d \;\; , \;\; t = 0 \tag{B.1b}$$

$$\frac{\partial T}{\partial x} = 0 \quad , \quad x = 0 \;\; , \;\; t > 0 \tag{B.1c}$$

$$\lambda\,\frac{\partial T}{\partial x} + h\,T = 0 \;\; , \quad x = d \;\; , \;\; t > 0 \tag{B.1d}$$

The exact analytical solution can be determined by the method of separation of variables. First, the dependent variable $T(x,t)$ is separated into space and time dependent functions, that is:

$$T(x,t) = X(x)\,\Gamma(t) = X\,\Gamma \tag{B.2}$$

By substituting equation (B.2) into equation (B.1), we get

$$\frac{1}{X}\frac{d^2X}{dx^2} = \frac{1}{\alpha\,\Gamma}\frac{d\Gamma}{dt} \quad , \quad 0 < x < d \;\; , \;\; t > 0 \tag{B.3a}$$

$$T = T(x,0) \quad , \quad 0 < x < d \;\; , \;\; t = 0 \tag{B.3b}$$

$$\frac{dX}{dx} = 0 \quad , \quad x = 0 \;\; , \;\; t > 0 \tag{B.3c}$$

$$\lambda\,\frac{dX}{dx} + h\,X = 0 \;\; , \quad x = d \;\; , \;\; t > 0 \tag{B.3d}$$

Since the left-hand side of equation (B.3a) is a function of space only and its right-hand side is a function of time only, both sides should be equal to a constant (say ω^2, zero or $-\omega^2$). However, ω^2 and zero are not acceptable since they cannot describe the expected physical behaviour. Therefore, equation (B.3a) becomes

$$\frac{1}{X}\frac{d^2X}{dx^2} = \frac{1}{\alpha\,\Gamma}\frac{d\Gamma}{dt} = -\,\omega^2 \tag{B.4a}$$

Which implies that

$$\frac{d^2X}{dx^2} + \omega^2\,X = 0 \;\; , \quad 0 < x < d \tag{B.4b}$$

$$\frac{d\Gamma}{dt} + \alpha\omega^2\,\Gamma = 0 \;\; , \quad t > 0 \tag{B.4c}$$

Equations (B.4b) and (B.4c) are both linear differential equations with constant coefficients. Therefore, their solution are respectively given as

$$X = a_1\cos(\omega x) + a_2\sin(\omega x) \tag{B.5a}$$

$$\Gamma = a_3\,e^{-\alpha\omega^2 t} \tag{B.5b}$$

where a_1, a_2 and a_3 are arbitrary constants. Applying equation (B.5) into equation (B.2) yields

$$T(x,t) = [\;b_1\cos(\omega x) + b_2\sin(\omega x)\;]\;e^{-\alpha\omega^2 t} \tag{B.6a}$$

where, $b_1 = a_1a_3$ and $b_2 = a_2a_3$. Therefore, equation (B.5) can be rewritten as

$$X = b_1\cos(\omega x) + b_2\sin(\omega x) \tag{B.6b}$$

$$\Gamma = e^{-\alpha\omega^2 t} \tag{B.6c}$$

Substituting the boundary condition at $x = 0$ (i.e. equation (B.3c)) into equation (B.6b), produces

$$b_2 = 0$$

Therefore, equation (B.6b) becomes

$$X = b_1 \cos(\omega x) \tag{B.7}$$

By applying the second boundary condition (i.e. equation (B.3d)) into equation (B.7), we get

$$\omega \tan(\omega d) = \frac{h}{\lambda} \tag{B.8}$$

which has an infinite series of solutions, ω_m. The general solution of the problem can be found by adding together the product solutions corresponding to each eigenvalue ω_m (i.e. linear superposition). Therefore

$$T(x,t) = \sum_{m=1}^{\infty} c_m \; X(\omega_m, x) e^{-\alpha \omega_m^2 t} \tag{B.9}$$

where, c_m are constant coefficients which contain the b_1 value of equation (B.7) and $X(\omega_m, x)$ are the eigenfunctions defined by equation (B.7). In order to determine the values of the coefficients c_m the initial condition is applied to equation (B.9) as

$$T(x,0) = \sum_{m=1}^{\infty} c_m \; X(\omega_m, x) \tag{B.10}$$

Operating $(\int_0^d X(\omega_n, x)\, dx)$ on both sides of equation (B.10), we get

$$\int_0^d X(\omega_n, x)\, T(x,0)\, dx = \int_0^d X(\omega_n, x) \sum_{m=1}^{\infty} c_m \; X(\omega_m, x)\, dx \tag{B.11}$$

It is clear that equation (B.4b), with its boundary conditions (3c) and (3d), is a special case of the *Sturm-Liouville* problem (refer to Appendix B.4). Hence the eigenfunctions $X(\omega_m, x)$ are orthogonal with respect to the weight function $w(x) = 1$. Therefore

$$\int_0^d X(\omega_n, x)\; X(\omega_m, x)\, dx = -\begin{bmatrix} 0 & m \neq n \\ N(\omega_m) & m = n \end{bmatrix} \tag{B.12}$$

where

$$N(\omega_m) = \int_0^d [X(\omega_m, x)]^2 \, dx = \int_0^d \cos^2(\omega_m x)\, dx =$$

$$\frac{1}{2}\left[\frac{\sin(\omega_m d)\cos(\omega_m d)}{\omega_m} + d\right] \tag{B.13}$$

Therefore, from equation (B.11) it can be concluded that

$$c_m = \frac{1}{N(\omega_m)} \int_0^d X(\omega_m, x)\, T(x,0)\, dx \tag{B.14}$$

and

$$T(x,t) = \sum_{m=1}^{\infty}\left[\frac{e^{-\alpha\omega_m^2 t}}{N(\omega_m)}\, X(\omega_m, x) \int_0^d X(\omega_m, x)\, T(x,0)\, dx\right] \tag{B.15}$$

For the special case where $T(x,0) = T_0 =$ constant, equation (B.15) becomes

$$T(x,t) = 2T_0 \sum_{m=1}^{\infty} \frac{e^{-\alpha\omega_m^2 t}\cos(\omega_m x)\sin(\omega_m d)}{\omega_m\left[\frac{\sin(\omega_m d)\cos(\omega_m d)}{\omega_m} + d\right]} \tag{B.16}$$

where, ω_m are the roots of equation (B.8) and

$$\int_0^d X(\omega_m, x)\, T(x,0)\, dx = \int_0^d \cos(\omega_m x)\, T_0\, dx = \frac{T_0}{\omega_m}\sin(\omega_m d)$$

B.2 Problem (2)

The second problem is similar to the first one except that the boundary condition at $x = d$ is set to zero instead of a convective boundary. Therefore, the problem is defined by

$$\frac{\partial^2 T(x,t)}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T(x,t)}{\partial t} \quad , \quad 0 < x < d \;,\; t > 0 \tag{B.17a}$$

$$T = T(x,0) \quad , \quad 0 < x < d \;,\; t = 0 \tag{B.17b}$$

$$\frac{\partial T}{\partial x} = 0 \quad , \quad x = 0 \;,\; t > 0 \tag{B.17c}$$

$$T(x,t) = 0 \;, \quad x = d \;,\; t > 0 \tag{B.17d}$$

By following the same procedure used for solving the first problem, the temperature distribution for this problem is achieved as follows.

By applying the separation of variables method, we get

$$T(x,t) = [\ b_1 \cos(\omega x) + b_2 \sin(\omega x)\ ]\ e^{-\alpha\omega^2 t} \tag{B.18a}$$

$$X = b_1 \cos(\omega x) + b_2 \sin(\omega x) \tag{B.18b}$$

$$\Gamma = e^{-\alpha\omega^2 t} \tag{B.18c}$$

By applying the boundary condition at $x = 0$, equation (B.18b) becomes

$$X = b_1 \cos(\omega x) \tag{B.19}$$

By applying the boundary condition at $x = d$, we obtain

$$\cos(\omega d) = 0 \tag{B.20}$$

Therefore, we can write

$$N(\omega_m) = \int_0^d [X(\omega_m, x)]^2\ dx = \int_0^d \cos^2(\omega_m x)\ dx =$$

$$\frac{1}{2}\left[\ \frac{\sin(\omega_m d)\cos(\omega_m d)}{\omega_m} + d\ \right] = \frac{d}{2} \tag{B.21}$$

and

$$T(x,t) = \sum_{m=1}^{\infty} \frac{e^{-\alpha\omega_m^2 t}}{N(\omega_m)}\ X(\omega_m, x) \int_0^d X(\omega_m, x)\ T(x,0)\ dx \tag{B.22}$$

For the special case where $T(x,0) = T_0 =$ constant, equation (B.22) becomes

$$T(x,t) = \frac{2T_0}{d} \sum_{m=1}^{\infty} \frac{e^{-\alpha\omega_m^2 t}\ \cos(\omega_m x)\ \sin(\omega_m d)}{\omega_m} \tag{B.23}$$

where, ω_m are the roots of

$$\cos(\omega_m d) = 0$$

B.3 Problem (3)

The third problem is similar to the second one except that the boundary condition at $x = 0$ is set to zero instead of an adiabatic boundary. Therefore, the problem is defined by

$$\frac{\partial^2 T(x,t)}{\partial x^2} = \frac{1}{\alpha}\,\frac{\partial T(x,t)}{\partial t} \quad , \quad 0 < x < d \;\; , \;\; t > 0 \tag{B.24a}$$

$$T = T(x,0) \quad , \quad 0 < x < d \;\; , \;\; t = 0 \tag{B.24b}$$

$$T(x,t) = 0 \quad , \quad x = 0 \;\; , \;\; t > 0 \tag{B.24c}$$

$$T(x,t) = 0 \;\; , \quad x = d \;\; , \;\; t > 0 \tag{B.24d}$$

by following the same procedure used for solving the first problem, the temperature distribution for this problem is achieved as follows.

By applying separation of variables method, we get

$$T(x,t) = [\; b_1 \cos(\omega x) + b_2 \sin(\omega x) \;]\; e^{-\alpha\omega^2 t} \tag{B.25a}$$

$$X = b_1 \cos(\omega x) + b_2 \sin(\omega x) \tag{B.25b}$$

$$\Gamma = e^{-\alpha\omega^2 t} \tag{B.25c}$$

By applying the boundary condition at $x = 0$, equation (B.25b) becomes

$$X = b_2 \sin(\omega x) \tag{B.26}$$

By applying the boundary condition at $x = d$, we obtain

$$\sin(\omega d) = 0 \tag{B.27}$$

Therefore, we can write

$$N(\omega_m) = \int_0^d [X(\omega_m, x)]^2 \; dx = \int_0^d \sin^2(\omega_m x) \; dx =$$

$$\frac{1}{2}\left[\; d - \frac{\sin(\omega_m d)\,\cos(\omega_m d)}{\omega_m} \;\right] = \frac{d}{2} \tag{B.28}$$

and

$$T(x,t) = \sum_{m=1}^{\infty} \frac{e^{-\alpha\omega_m^2 t}}{N(\omega_m)}\; X(\omega_m, x) \int_0^d X(\omega_m, x)\; T(x,0)\; dx \tag{B.29}$$

For the special case where $T(x,0) = T_0 =$ constant, equation (B.29) becomes

$$T(x,t) = \frac{2T_0}{d} \sum_{m=1}^{\infty} \frac{e^{-\alpha\omega_m^2 t} \sin(\omega_m x) \left[1 - \cos(\omega_m d) \right]}{\omega_m} \tag{B.30}$$

where, ω_m are the roots of

$$\sin(\omega_m d) = 0$$

B.4 Sturm - Liouville problem

The Sturm - Liouville problem is a homogeneous boundary value problem which can be expressed as

$$\frac{d}{dx}\left[p(x) \frac{dX}{dx} \right] + \left[q(x) + \omega\, w(x) \right] X = 0 \qquad in \quad a < x < b$$

$$a_1 \frac{dX}{dx} + a_2 X = 0 \qquad at \quad x = a$$

$$b_1 \frac{dX}{dx} + b_2 X = 0 \qquad at \quad x = b$$

where $p(x)$ and $w(x)$ are continuous on the closed interval $a \le x \le b$, and $q(x)$ is continuous at least over the open interval $a < x < b$. The constants a_1, a_2, b_1 and b_2 are real, a_1 and a_2 are not both zero, and b_1 and b_2 are not both zero.

The eigenfunctions of the Sturm - Liouville problem are orthogonal with respect to the weighting function $w(x)$ over the interval (a,b), that is

$$\int_a^b w(x)\, X_m\, X_n\, dx = 0 \quad \text{for} \quad n \neq m$$

APPENDIX C

Transformation of Coordinates

Within the multi-D gridding process, each building component is treated in isolation in order to allow variable gridding resolution and the future importation of building components gridded by other software. In addition, building component local coordinates were transformed to global coordinates before gridding in an attempt to reduce round off errors. This coordinate transformation is established in two steps, translation and rotation. The transformation of edges and corners as well as connection surfaces are based on the surfaces transformation. Therefore, only surfaces are transformed based on the following procedure.

C.1 Translation

First, the origin of the local coordinate system is defined. The first vertex in the surface vertex list is used as the local origin. Then, the surface vertices are translated so that the local coordinates origin is moved to the global origin as shown in Figure C.1. In matrix notation, the translation of a vertex (x, y, z) to its new location (x', y', z') is formulated as

$$[\ x'\ y'\ z'\ ] = [\ x\ y\ z\ 1\ ] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -x_o & -y_o & -z_o \end{bmatrix} \tag{C.1}$$

Where, (x_o, y_o, z_o) are the old local origin coordinates.

C.2 Rotation

The local axes should be defined first. This is done in the following order (refer to Figure C.2):

- the x-axis is defined to be from the first to the second vertex.
- a dummy z-axis is defined from the first to the last vertex.

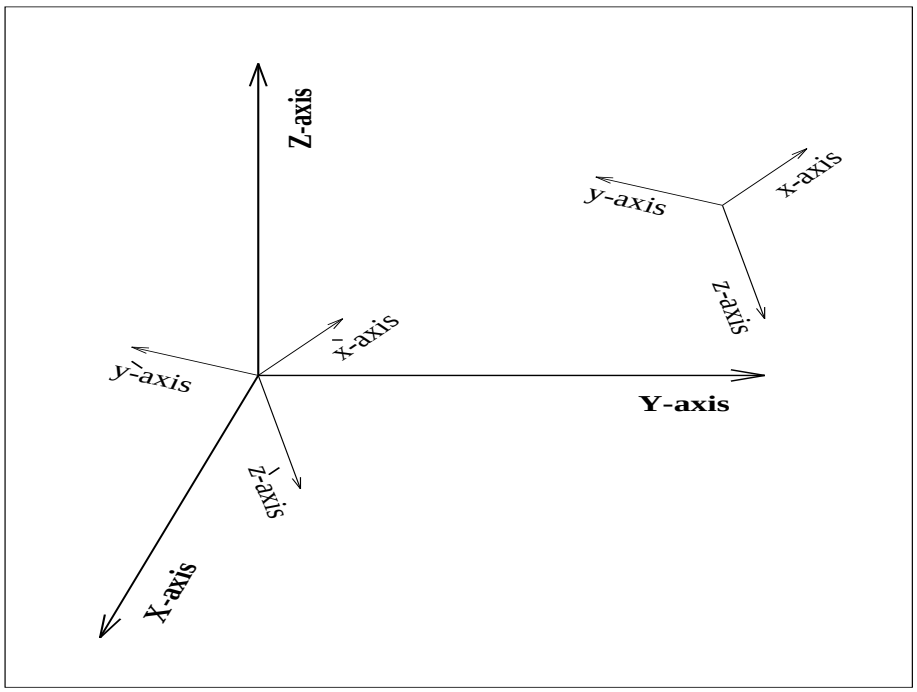

Figure C.1 Axis translation.

- the y-axis is determined from the cross product of the dummy z-axis and the x-axis.
- the z-axis is defined from the cross product of the x-axis and the y-axis.

These steps require defining the local axes by their direction cosines l, m and n which are given by

$$l = \frac{(x_2 - x_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}} \tag{C.2}$$

$$m = \frac{(y_2 - y_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}} \tag{C.3}$$

$$n = \frac{(z_2 - z_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}} \tag{C.4}$$

The rotation (refer to Figure C.3) is then perform according to

$$[\, x'' \; y'' \; z'' \,] = [\, x' \; y' \; z' \,] \begin{bmatrix} l_x & m_x & n_x \\ l_y & m_y & n_y \\ l_z & m_z & n_z \end{bmatrix} \tag{C.5}$$

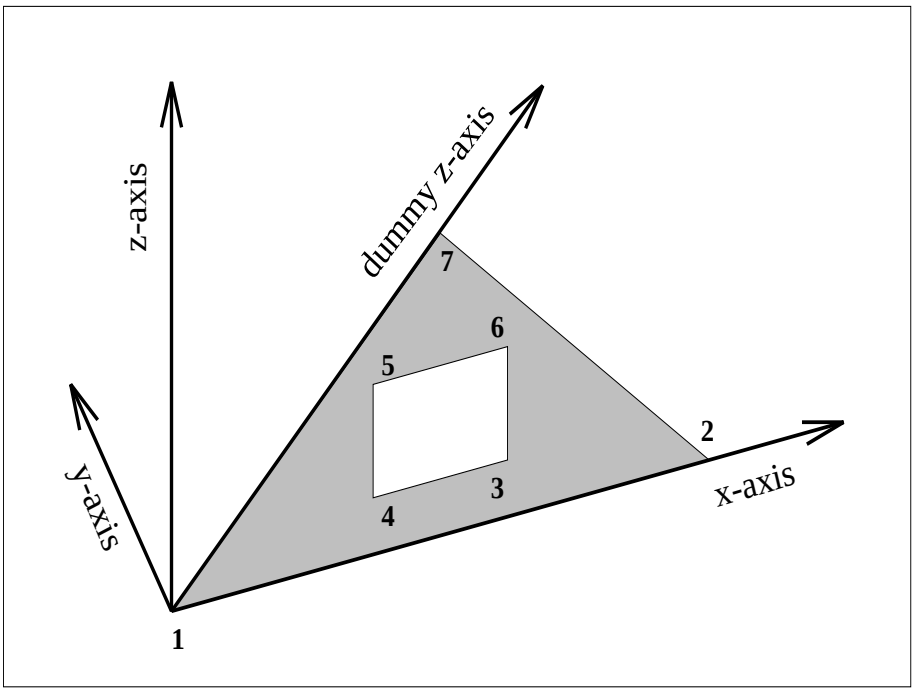

Figure C.2 Arbitrary surface.

Finally, the axes may require another translation process in order to assure that all vertex coordinates are positive.

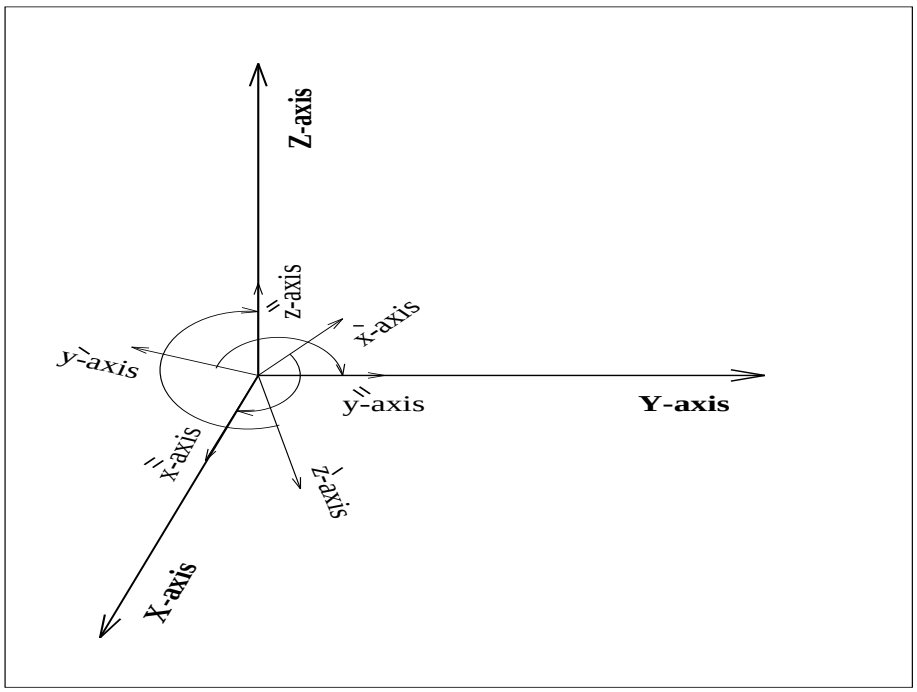

Figure C.3 Axis rotation.

APPENDIX D

PHOENICS

D.1 Introduction

PHOENICS is a computer code for simulating combined heat and fluid flow. It consists of a pre-processor called SATELLITE, a solver called EARTH, and two graphical post-processors called PHOTON and AUTOPLOT. In addition, PHOENICS includes a module GUIDE which consists of a browsing guide.

Usually, the problem to be simulated is described in an instruction file called Q1. The user instructions Q1 file is transformed into a data file by the SATELLITE. This data file is then read by the EARTH which produces two output files: an ASCII text file called RESULT which the user can read, and data file called PHI which can be read by the two graphical post-processors PHOTON and AUTOPLOT.

PHOENICS was used for inter-model validation work in Chapter 4. It was used to model 1D and 3D heat transfer through a homogeneous corner. The Q1 files for these problems are given below.

D.2 Homogeneous wall

For the 1D case, a homogeneous wall of thickness $0.1\ m$ was simulated. The thermophysical properties of the wall are shown in Table (D.1). The wall is subjected to a convective boundary at both sides. The outside ambient is set to $20\ ^oC$, while the inside temperature changes at time $t = 0$ from $20\ ^oC$ to $0\ ^oC$. The initial temperature everywhere is $20\ ^oC$.

In order to approximate massless boundary nodes at the boundaries, the space discretisation is defined by NREGX and IREGX, where (0.0001, 0.0998, 0.0001) are the space steps in metres. The simulation period is 86400 seconds divided into 24 equal time steps as defined by GRD-PWR(T,24,86400.,1.).

Table D.1 Thermophysical properties for the homogeneous wall

property name	symbol	value	units
conductivity	λ	0.96	$W/m\ K$
density	ρ	2000.00	kg/m^3
heat capacity	C_p	650.00	$J/kg\ K$

Q1 file

```
 TALK=T;RUN( 1, 1);VDU=X11-TERM
   GROUP 1. Run title and other preliminaries
REAL(KON,CP,HTC)
KON=0.96
CP=650.
HTC=10.
TEXT(3D transient room corner model)
   GROUP 2. Transience; time-step specification
STEADY=F
GRDPWR(T,24,86400.,1.)
   GROUP 3. X-direction grid specification
NREGX=3
IREGX=1;GRDPWR(X,1,.0001,1.)
IREGX=2;GRDPWR(X,1,.0998,1.)
IREGX=3;GRDPWR(X,1,.0001,1.)
   GROUP 4. Y-direction grid specification
GRDPWR(Y,1,2.,1.)
   GROUP 5. Z-direction grid specification
GRDPWR(Z,1,2.,1.)
   GROUP 6. Body-fitted coordinates or grid distortion
   GROUP 7. Variables stored, solved & named
SOLVE(H1)
   GROUP 8. Terms (in differential equations) & devices
TERMS(H1,N,N,Y,Y,N,N)
   GROUP 9. Properties of the medium (or media)
RHO1=2000.
ENUL=1.
PRNDTL(H1)=CP*ENUL*RHO1/KON
```

```
   GROUP 10. Inter-phase-transfer processes and properties
   GROUP 11. Initialisation of variable or porosity fields
   ** Define the initial temperature to be ,T=20.
FIINIT(H1)=20.0*CP
   GROUP 12. Patchwise adjustment of terms (in differential equation
   GROUP 13. Boundary conditions and special sources
   ** Uniform ambient temperature at inside east surface.
PATCH(INEAST,EAST,3,3,1,1,1,1,1,LSTEP)
COVAL(INEAST,H1,HTC/CP,0.0)
   ** Uniform ambient temperature at outside west surface.
PATCH(OUTWEST,WEST,1,1,1,1,1,1,1,LSTEP)
COVAL(OUTWEST,H1,HTC/CP,20.0*CP)
   GROUP 14. Downstream pressure for PARAB=.TRUE.
   GROUP 15. Termination of sweeps
LSWEEP=20
   GROUP 16. Termination of iterations
   GROUP 17. Under-relaxation devices
   GROUP 18. Limits on variables or increments to them
   GROUP 19. Data communicated by satellite to GROUND
   GROUP 20. Preliminary print-out
ECHO=F
   GROUP 21. Print-out of variables
   GROUP 22. Spot-value print-out
   GROUP 23. Field print-out and plot control
   GROUP 24. Dumps for restarts
STOP
```

D.3 Homogeneous corner

The corner shown in Figure D.1 is at the intersection of three walls similar to the homogeneous wall described above. The same boundary and initial conditions of the homogeneous wall problem are applied here also.

Q1 file

```
 TALK=T;RUN( 1, 1);VDU=X11-TERM
   GROUP 1. Run title and other preliminaries
REAL(KON,CP,HTC)
KON=0.96
CP=650.
```

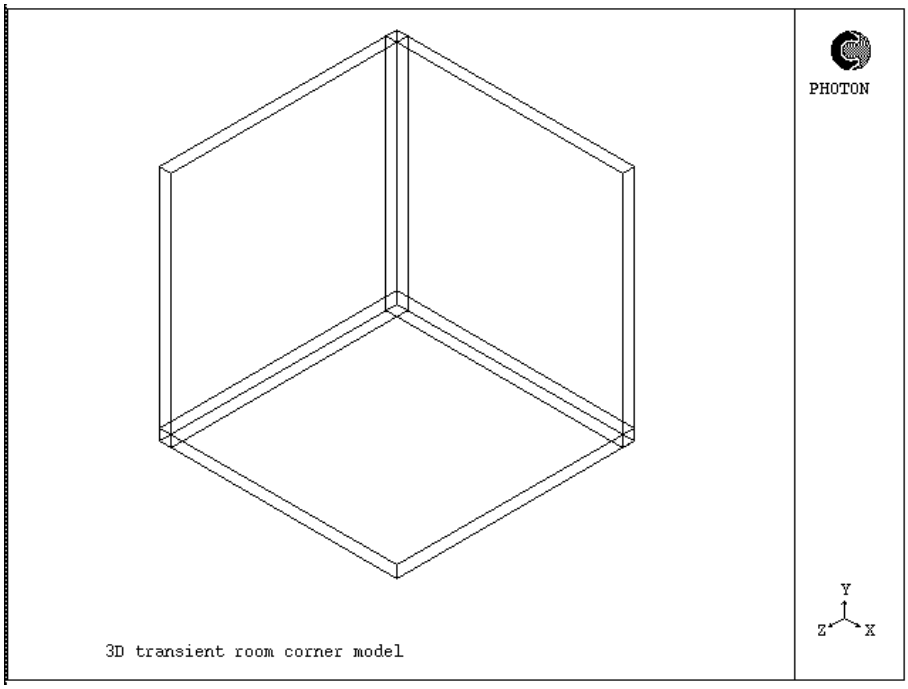

Figure D.1 Corner gridding.

HTC=10.

TEXT(3D transient room corner model)

GROUP 2. Transience; time-step specification

STEADY=F

GRDPWR(T,24,86400.,1.)

GROUP 3. X-direction grid specification

NREGX=4

IREGX=1;GRDPWR(X,1,.0001,1.)

IREGX=2;GRDPWR(X,1,.0998,1.)

IREGX=3;GRDPWR(X,1,.0001,1.)

IREGX=4;GRDPWR(X,1,2.,1.)

GROUP 4. Y-direction grid specification

NREGY=4

IREGY=1;GRDPWR(Y,1,.0001,1.)

IREGY=2;GRDPWR(Y,1,.0998,1.)

IREGY=3;GRDPWR(Y,1,.0001,1.)

IREGY=4;GRDPWR(Y,1,2.,1.)

GROUP 5. Z-direction grid specification

```
NREGZ=4
IREGZ=1;GRDPWR(Z,1,.0001,1.)
IREGZ=2;GRDPWR(Z,1,.0998,1.)
IREGZ=3;GRDPWR(Z,1,.0001,1.)
IREGZ=4;GRDPWR(Z,1,2.,1.)
   GROUP 6. Body-fitted coordinates or grid distortion
   GROUP 7. Variables stored, solved & named
SOLVE(H1)
   GROUP 8. Terms (in differential equations) & devices
TERMS(H1,N,N,Y,Y,N,N)
   GROUP 9. Properties of the medium (or media)
RHO1=2000.
ENUL=1.
PRNDTL(H1)=CP*ENUL*RHO1/KON
   GROUP 10. Inter-phase-transfer processes and properties
   GROUP 11. Initialisation of variable or porosity fields
   ** Exclude the interior block from the calculations.
CONPOR(INSID,0.0,CELL,4,4,4,4,4,4)
   ** Define the initial temperature to be ,T=20.
FIINIT(H1)=20.0*CP
   GROUP 12. Patchwise adjustment of terms (in differential equation
   GROUP 13. Boundary conditions and special sources
   ** Uniform ambient temperature at inside east surface.
PATCH(INEAST,EAST,3,3,4,4,4,4,1,LSTEP)
COVAL(INEAST,H1,HTC/CP,0.0)
   ** Uniform ambient temperature at inside high surface.
PATCH(INHIGH,HIGH,4,4,4,4,3,3,1,LSTEP)
COVAL(INHIGH,H1,HTC/CP,0.0)
   ** Uniform ambient temperature at inside north surface.
PATCH(INNORTH,NORTH,4,4,3,3,4,4,1,LSTEP)
COVAL(INNORTH,H1,HTC/CP,0.0)
   ** Uniform ambient temperature at outside west surface.
PATCH(OUTWEST,WEST,1,1,1,4,1,4,1,LSTEP)
COVAL(OUTWEST,H1,HTC/CP,20.0*CP)
   ** Uniform ambient temperature at outside low surface.
PATCH(OUTLOW,LOW,1,4,1,4,1,1,1,LSTEP)
COVAL(OUTLOW,H1,HTC/CP,20.0*CP)
```

```
   ** Uniform ambient temperature at outside south surface.
PATCH(OUTSOUTH,SOUTH,1,4,1,1,1,4,1,LSTEP)
COVAL(OUTSOUTH,H1,HTC/CP,20.0*CP)
   GROUP 14. Downstream pressure for PARAB=.TRUE.
   GROUP 15. Termination of sweeps
LSWEEP=20
   GROUP 16. Termination of iterations
   GROUP 17. Under-relaxation devices
   GROUP 18. Limits on variables or increments to them
   GROUP 19. Data communicated by satellite to GROUND
   GROUP 20. Preliminary print-out
ECHO=F
   GROUP 21. Print-out of variables
   GROUP 22. Spot-value print-out
   GROUP 23. Field print-out and plot control
   GROUP 24. Dumps for restarts
STOP
```

APPENDIX E

EUROKOBRA Thermal Bridge Database

A number of printed atlases are available which give general guidance on the problems that can result from, and methods for avoiding, thermal bridges in specific construction types. These are inflexible so it may be difficult to decide how closely a specific construction matches the atlas and it will not be possible to investigate the effect of modifications. It is also difficult to obtain quantitative information on energy loss or surface temperatures that are likely to lead to condensation risk.

At the other extreme, sophisticated finite element software, such as ANSYS (Swanson Analysis Systems, 1991), that allows detailed modelling of 3-D heat flow and temperature distribution through complex components has been available for many years. These are complex to use and need considerable input from experienced users before structures can be specified or modified. It is also possible for plausible answers to emerge although their accuracy may not be clear.

More recently a Committee of the European Standards (CEN) Organisation, TC89 WG1, has produced draft European standards on the calculation of thermal bridges. These specify the input meshes, boundary conditions and two types of output:

1. The Linear Thermal Transmittance or κ factor; this is the heat loss per metre length through a structural element containing a thermal bridge minus the heat loss that would occur if the thermal bridge was not present.

2. The temperature factor of the surface,

$$f = \frac{T_s - T_o}{T_i - T_o} \tag{E.1}$$

where T_s is the internal surface temperature, T_i is the internal air temperature and T_o is the external air temperature. This gives an estimate of the ‘quality’ of the thermal bridge independently of the imposed boundary conditions and can be used as an index of the likelihood of condensation.

KOBRA is a program developed by the Belgian company Physibel to interrogate a database of 2-D thermal bridge details, provide quantitative information on the linear thermal transmittances and temperature factors, and allow the effect of modifications to be rapidly investigated.

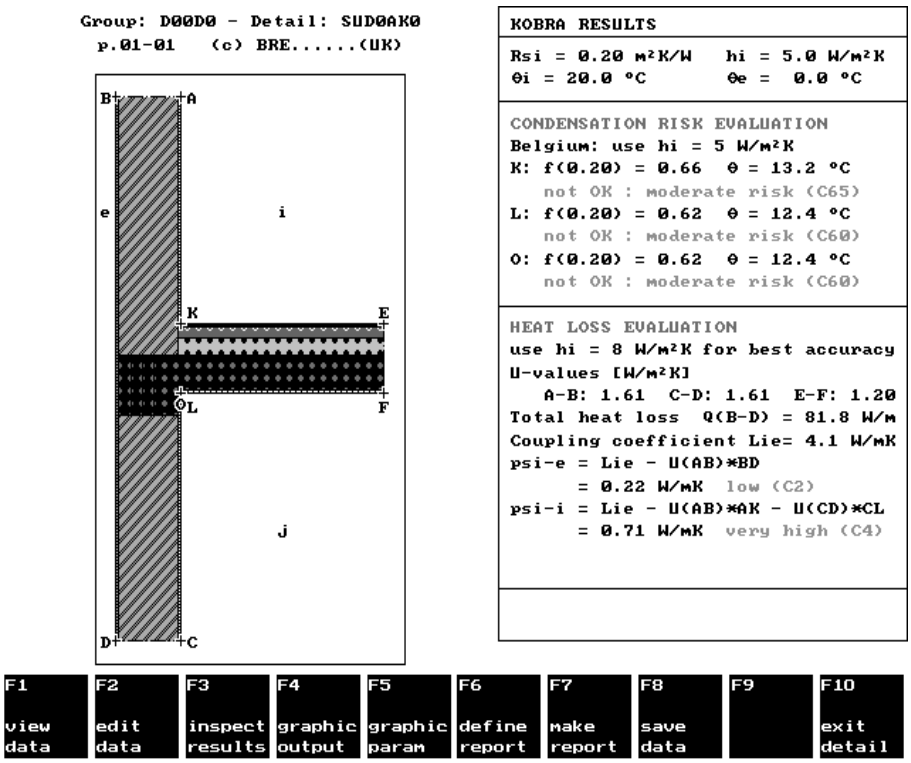

Figure E.1 Typical output from KOBRA

The EUROKOBRA database, which can be interrogated by KOBRA, was developed under an EC SAVE project by eight participating countries, led by the Belgian Building Research Institute. The database, or atlas, contains the 2-D geometry of the details including the thermal conductivities of each of the materials; a rectangular grid for the calculation is also specified. The left hand side of Figure E.1 shows an example. Within the given topology of the detail, the user can vary the boundary conditions, the horizontal or vertical size of each element and the conductivities of each material. The outputs are calculated within a few seconds on a standard PC. They include, as shown on the right hand side of Figure E.1, the temperature factors at key points and warnings on the risks of condensation. Colour pictures of the temperature and heat flow distributions through the section can also be displayed.

The EUROKOBRA database, currently containing about 1000 details from typical European buildings, is now available. Further specialised atlases covering details such as window frames and steel-framed buildings are in preparation, and consideration is being given to extending the

capabilities of the system to 3-D.

References

Swanson Analysis Systems1991. , *General Index to ANSYS Documentation*, Houston.

APPENDIX F

Modified 1D Conduction Simulation

This modified 1D conduction approach within a whole building energy simulation program was demonstrated in a major European research programme called PASSYS (Vandaele and Wouters, 1994). This programme, in which 10 European countries participated, involved the establishment of a series of test cells throughout Europe. One of the aims of the project was to use the test cells to gather high quality data sets for use in the empirical validation of dynamic simulation programs. Figure F.1 shows the test cell structure.

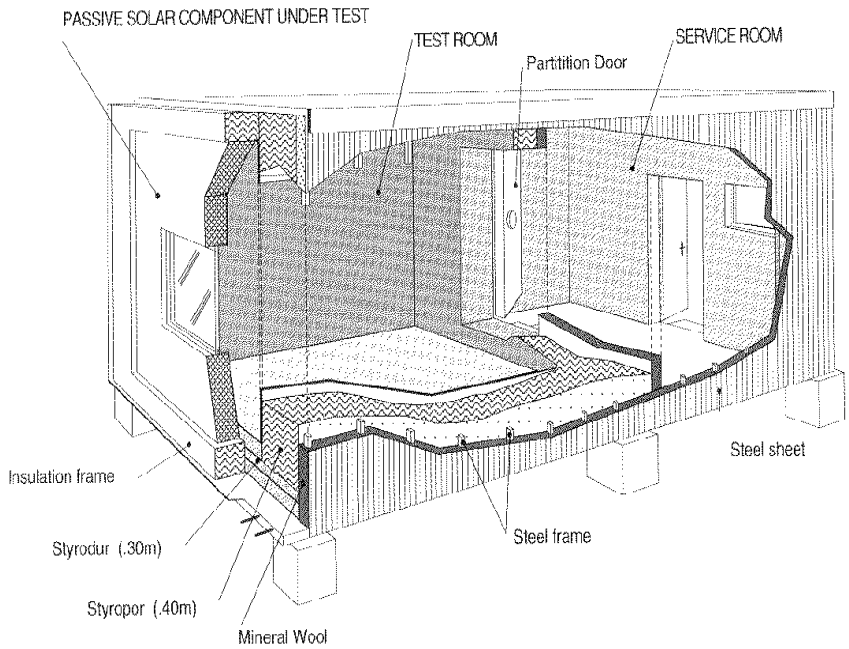

Figure F.1 PASSYS Test Cell

A major difficulty encountered in the work was the relatively large 2-D and 3-D conduction losses in the test cells resulting from the thick walls ($0.52m$) of the cells. In the case of the test cell fitted with the insulated calibration wall on the south facade, it was estimated that the heat losses were 48% higher than those estimated from a simple 1-D heat loss calculation based on internal dimensions. The idea of the test cell construction was to have highly insulating constructions on the floor, ceiling and all walls except the south wall. Different wall components could be mounted on the south wall as these would typically have much higher heat losses than the rest of the insulated cell and experimental uncertainty of the south wall performance (the focus of the work) would be minimised. The dynamic simulation program ESP-r which was used within PASSYS, in common with other comparable simulation programs, had only 1-D conduction capability and was therefore unable to model explicitly the edge heat losses through the test cell envelope.

Within PASSYS, the edge loss problem was addressed with the use of extra "edge constructions" within the model which attempted to account for the heat transfer in the edges of the test cell. The material conductivity of these constructions were obtained with the use of 2-D and 3-D steady state analyses using the program Trisco (Standaert, 1989). A similar procedure using 2-D steady-state analyses was used in the recent IEA Annex 21 empirical validation exercise (Lomas et al, 1994).

The technique employed aimed to keep the same internal geometry of the test cells but to change the properties of the construction lying within 0.4m of each edge. Beyond this distance from the edge, the 3-D analyses showed that the conduction was essentially 1-D. The resulting model of the test room is shown in Figure F.2.

The conductivities of the edge constructions were then modified to take account of the extra edge losses, i.e. those obtained from the difference between the 3-D and 1-D steady-state analysis for the particular edge or corner. To allow for the changing area through the thickness of the cell, use was made of the "developed area" (see Figure F.3 for an example). Each edge construction was subdivided into a number of strips. The inside layer corresponded to the actual material in the test cell, the other layers had artificial conductivity values, increasing in value from inside to outside so that:

- the conductivity was proportional to the developed area of the layer
- the overall conduction losses were in agreement with the steady state analyses.

With regard to capacity of these edge constructions, the capacity of the edge region was sized to approximately match the time taken for the effect of a step heat pulse to propagate from the internal surface to the external surface of the edge (as the primary interest was in the dynamic response resulting from internal changes in temperature).

The method and the results are written up fully in (Jensen, 1993). In summary, the results of comparing predicted data from a model with these edge constructions with measured data for the calibration-walled test cell showed that:

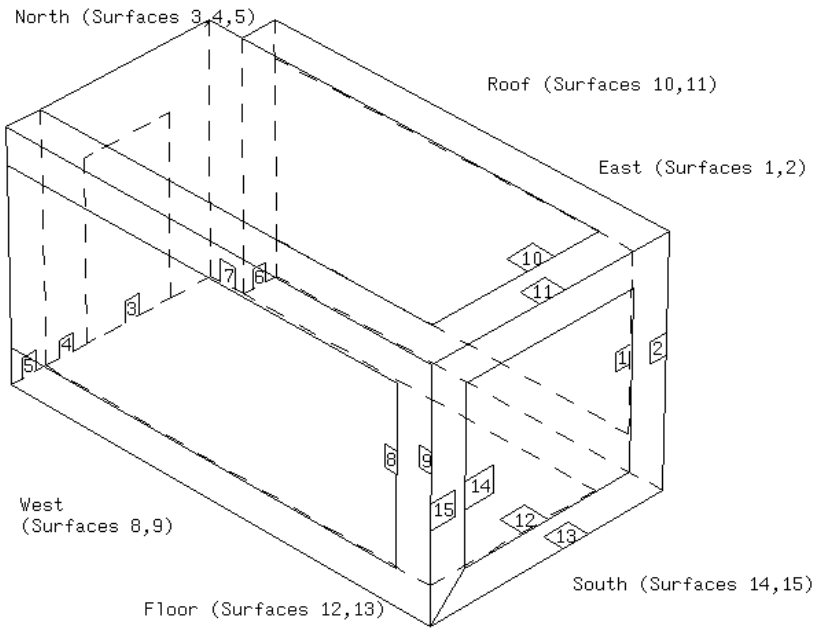

Figure F.2 Test Cell Model with Edge Constructions

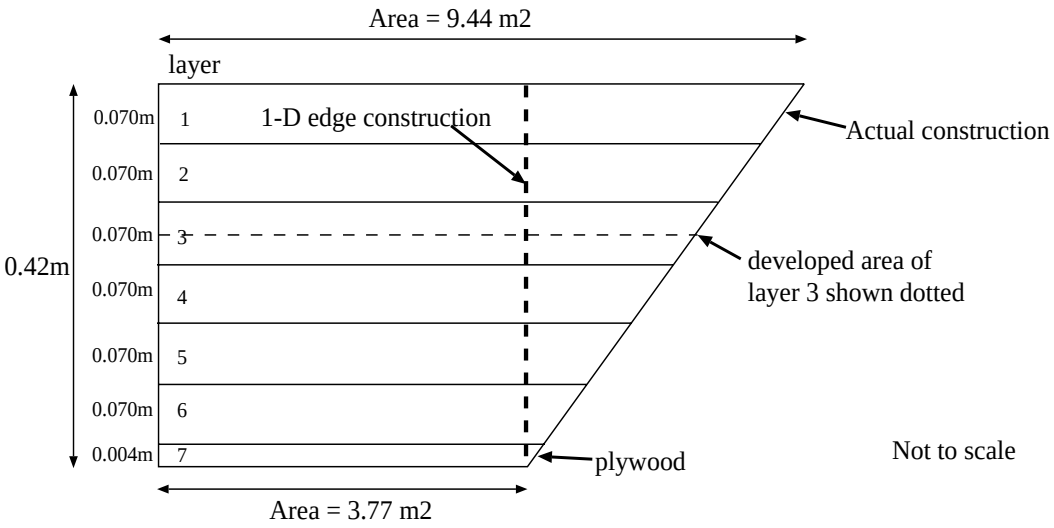

Figure F.3 Geometry of the Developed Area

- the steady state model predictions were in reasonable agreement with the measured data when measurement and prediction uncertainties, in particular with respect to internal

convection and temperature-dependent conductivities, were taken into account

- The dynamic response was poor.

Figure F.4 shows the measured results for a calibration-walled test cell, together with 1-D simulation results based on internal cell dimensions, and the results after adding edge constructions ("modified 1-D").

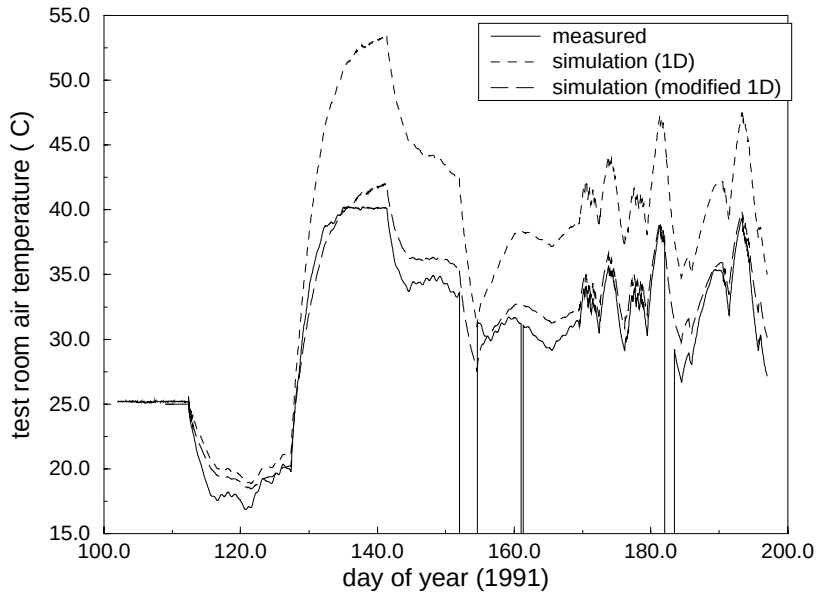

Figure F.4 Measured, 1-D and Modified 1-D Air Temperatures

Within PASSYS, the results from the calibration experiments were used to calibrate the ESP-r model of the test cell. In particular, the capacity derived from the results of the application of system identification techniques was used to adjust the capacities of the edge constructions to obtain satisfactory agreement for the calibration-walled test cell (Clarke et al 1993). This calibrated model was then used for further comparisons of different test walls.

References

Clarke, J A, P A Strachan, and C Pernot 1993. ‘‘An Approach to the Calibration of Building Energy Simulation Models,’’ *ASHRAE Transactions*, vol. 99, no. 2, pp. 917-27.

Jensen, S O1993. , *The PASSYS Project: Subgroup Model Validation and Development*, Brussels. Final Report, Part I and II 1986-1992. Commission of the European Comunities. DGX1I,EUR

15115 EN.

Lomas, K, H Eppel, C Martin, and D Bloomfield 1994. ‘‘Empirical Validation of Thermal Building Simulation Programs using Test Room Data,’’ *IEA Annex 21 (Calculation of Energy and Environmental Performance of Buildings)*.

Standaert, P1989. , *TRISCO, Version 3.2, Computer Program to Calculate Three-Dimensional Steady-State Heat Transfer in Objects described in a Rectangular Grid using the Energy Balance Technique*, Physibel C.V, Maldagem, Belgium.

Vandaele, L and P Wouters1994. , *The PASSYS Services: Summary Report*. European Commission Publication No EUR 15113 EN