A REFERENCE MODEL FOR GROUND COUPLED HEAT TRANSFER

Michael Crowley
Dublin Institute of Technology, Dublin, Ireland

Department of Building Services Engineering, DIT Bolton Street, Dublin 1, Ireland
E-mail: michael.crowley@dit.ie

ABSTRACT

A recently completed extension to IEA BESTEST includes further work on tests suitable for the validation of ground coupled heat transfer modules within building energy simulation software. The model described here forms part of this work. A finite difference model was prepared and applied to ground coupled heat transfer in the environs of a building for a range of geometries and boundary conditions. The model was verified by demonstrating close agreement with an analytical solution and with two independent models; it is therefore fit for purpose. A fundamental modelling limitation, applying to ground coupled heat transfer, emerged from the work.

INTRODUCTION

Recently, successive reductions in building fabric and infiltration heat losses have increased the relative importance of heat loss to ground. Its magnitude depends on a wide variety of parameters, including climate, presence of slab and/or perimeter insulation and the ground heat transfer model used for the calculation. A well-built house is so energy efficient above ground that the ground-coupled heat losses can account for 30%–50% of the total heat loss (Deru, 2003). A recent review of simulations carried out for houses located in Colorado, including both uninsulated and insulated slab-on-grade test cases, indicates 25%–45% of the heating load is caused by heat transfer through the floor (Neymark et al., 2008). In order to estimate the desired precision for a floor heat loss calculation, one might take a figure of 25% for the heat loss through an insulated floor and a further 10%–25% each for the losses through the external walls, the windows, the roof and infiltration. Assuming each is estimated with an uncertainty of ±10%, the propagated error in the total heat loss would be ±4.6%. This is calculated using Equation 1 in which δa is the uncertainty in the quantity *a*.

$$\delta(a+b+\ldots) = \left((\delta a)^2 + (\delta b)^2 + \ldots\right)^{1/2} \qquad (1)$$

Repeating this calculation with an uninsulated floor heat loss of 50% and a loss of 5%–20% through each of the other four elements would lead to a propagated error of ±5.7%. If a 5%–6% error in total heat loss is considered acceptable, then a tolerance of ±10% is called for in each of the component calculations.

Because of the complexity of the problem, model predictions for uninsulated slab-on-grade heat transfer, for example, can differ by between 25% and 60% depending on the type of ground heat transfer model used. This discrepancy has motivated further validation efforts in an extension to IEA BESTEST (Neymark et al., 2008) of which the work described here forms part. The novel validation methodology prescribed in the extended BESTEST calls for putative reference models/programs to be gauged first against a known exact solution to a ground coupled heat transfer problem, and then to agree closely with the predictions of other proposed reference models when all are applied to progressively more complex and realistic test problems for which analytical solutions do not exist. The closely agreeing reference solutions produced in this manner can then be used to test the performance of more general whole-building models in this demanding aspect. Empirical validation is rarely considered in this context chiefly because of the lengthy thermal time constants associated with large volumes of soil. However, the Heat Diffusion Equation, used in the present model, is known to produce accurate results for a homogeneous solid.

Test Problems

Seventeen test cases (GC10a to GC80c) were formulated, the details of which are to be found in Neymark et al., 2008. Geometrical aspects of a typical case are set out in Figure 1. A slab-in-grade exchanges heat with soil and with a space which is otherwise bounded by adiabatic surfaces. Vertical soil boundary conditions are typically considered to be adiabatic and the deep ground horizontal soil boundary to be isothermal. Soil and slab top surfaces are driven by a variety of boundary conditions including isothermal, convective and periodic. Radiant heat is not considered. Parametric variations include dimensions of soil volume considered, floor slab area and aspect ratio, slab and soil thermal conductivities, interior and exterior surface heat transfer coefficients. The analytical case, GC10a, is exceptional in that it was originally derived for a semi-infinite solid, that is, for infinite E and F in Figure 1.

Model performance was gauge mainly on floor heat loss, either cumulative or instantaneous. The hour of occurance of peak heat loss was used to check for phase differences between model outputs. Some surface temperatures were also requested.

Software

The software package used for this work was MATLAB 7.0.4.365 (R14) SP2. It is a general purpose mathematical package and is independent of building energy simulation. A series of ground coupled heat transfer programs was prepared specifically for this task using the high-level language within MATLAB which includes an extensive catalog of built-in functions. Functions for interpolation and the solution of linear algebraic equations (LAE) and ordinary differential equations (ODE) were of most use for the present work. A large part of the work of these functions is matrix processing and MATLAB is considered 'state-of-the-art' for matrix computation – incorporating, as it does, the LAPACK and BLAS libraries (The MathWorks, Inc., 2007).

Each of MATLAB's built-in ODE solvers forms an estimate of the error for the proposed time step and, if necessary, varies the length of the step to keep the error close to the specified tolerance. At each step, the error e in each nodal temperature T satisfies $|e| \le \max(\text{RelTol} \times |T|, \text{AbsTol})$. A relative tolerance (RelTol) of 10^{-5} and an absolute tolerance (AbsTol) of 10^{-8} were used here. Fixed time steps, if large, can lead to excessive error where the solution changes rapidly and, if small, can be unnecessarily accurate (slow) for a large part of the interval of integration. MATLAB does not offer fixed step size as an option.

MATLAB includes a range of direct and iterative solvers for linear algebraic equations. Direct methods were used exclusively here because they are generally more accurate and more reliable than iterative solvers (Heath, 2002). Direct solvers – all based on Gaussian elimination – produce exact solutions in the absence of rounding error. Since MATLAB calculates in double precision, results can therefore be expected to be many orders of magnitude more accurate than the specified precision for this project unless the matrix of coefficients is ill-conditioned. MATLAB tests for this circumstance and issues a warning if necessary. All of the foregoing concerns rounding error. Truncation error is, of course, separately present and is reduced by decreasing the space increment.

The test problems are very large – approaching 10^6 nodes/equations for the required precision – and this would usually necessitate the use of iterative methods on the PC available for the work (HP Compaq D330; Pentium 4, 2.8 GHz; 512 MB RAM). In order to use direct solution methods operating entirely within fast memory (RAM), two steps were taken. Memory was increased to 4 GB, of which 1 GB was reserved for the operating system. Also, the problems were programmed taking maximum advantage of symmetry. Three geometric symmetries are present in most of the problems and consequently only one-eighth of the problem domain need be considered. The system matrix is not fully symmetric and so no further efficiencies of this type are available. The matrix is, however, very sparse and this allowed two further efficiencies: (i) MATLAB can be programmed to store just the nonzero elements of a matrix, thus freeing up RAM for other uses and (ii) MATLAB's direct solvers take advantage of sparsity in that they do minimal wasteful processing of zero valued matrix elements – as well as minimizing 'fill-in' during the elimination stage.

SIMULATION

Modelling Assumptions

The modelling assumptions and inputs used were generally as described in the Test Specification (Neymark et al., 2008) and summerised in the Introduction above with the following exceptions.

Discrete hourly ambient temperature data was provided as part of a weather file for the harmonic test cases, that is, for those cases driven by a periodically varying outdoor temperature – a daily cosine modulating a yearly cosine in this case. These data are normally interpolated within building energy models that produce output at less than hourly intervals. Some building energy models use linear interpolation, others use smoother functions. The programs prepared here calculated ambient temperature as a smooth function of time down to the limits of double precision accuracy (approx. 10^{-15} seconds). This is close to perfect interpolation and may be considered the ideal to be aspired to in building energy models. Including a simpler interpolation method (or none) may bias the test toward those using that specific, less than perfect, method. As a check, hourly integrated floor heat flow calculations were carried out for a small sample of ambient temperatures from the weather file (1 June, hours 1-24) using (i) the hourly temperatures as provided, i.e. no interpolation and (ii) the smooth cosine/cosine function, i.e. perfect interpolation. The maximum difference was 0.8%, occurring at hour 4. The inclusion of even a simple interpolation routine would be expected to reduce this difference.

The Test Specification assumes that dynamic models (DM) will be used for the steady state cases and that these models will be run to steady state to obtain the desired results. Since the boundary conditions do not change over time, these cases are more appropriately modelled using algebraic equations, i.e. steady state models (SSM), which require just one application of an appropriate solver. SSMs are much quicker than DMs, in this case many orders of magnitude quicker, because the test cases are large (many nodes), moderately stiff and require exceptionally long integration intervals. A stiff system is one with a great range of characteristic time scales – thermal time constants in this context.

Modelling Approach

The models prepared here are three-dimensional finite difference approximations to the Heat Diffusion Equation.

$$\frac{1}{\alpha}\cdot\frac{\partial T}{\partial t}=\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2} \qquad (2)$$

The SSMs include a second-order centered difference approximation to the space derivatives and the DMs include, in addition, a third-order forward difference representation of the time derivative. As such, they are all capable of converging (Fletcher, 1991) onto their respective exact solutions provided the issues of stiffness, stability, matrix condition – and even the accumulation of the minute rounding error over some of the lengthy integration intervals to be discussed below – are dealt with correctly; and, of course, assuming sufficient computing power is available. The specific solvers employed were as follows: (i) a sparse LU decomposition method of the analyse/factorise/solve variety (Davis, 2004) was applied to the LAEs arising in the SSMs, (ii) an explicit numerical method (Bogacki and Shampine, 1989) was used to solve the ODEs presented in the DMs; the problems were not stiff enough to justify the costly matrix processing associated with implicit methods.

All the programs used here include mesh spacings that are independently variable in the x-direction (which, due to symmetry, includes the y-direction for problems with square perimeter boundary walls) and the z-direction. The spacings are denser near to the boundaries and a different spacing factor is used inside (fxL0) and outside (fxF0) the perimeter, and a third spacing factor (fz0) for the z-direction. Grid spacings increase by approximately these factors as the distance from the perimeter boundary increases; the program adjusts the factors slightly so that a whole number of grid spacings fits within the available dimension (L, F or E). The initial grid spacings in the vicinity of the perimeter are dx0 in the x- (and y-) directions and dz0 in the z-direction. Two of the test cases (GC45b and GC45c) have rectangular rather than square perimeter boundaries and so further grid spacing parameters are required for the y-direction. The number of space increments within the boundary wall dimension (W) is the same in both directions for these latter cases (Figure 2).

In broad terms, the values of these parameters, together with E and F defining the extent of the modelled soil volume, were varied to minimize the error for Case GC10a within the constraints of the available computing power. The final steady state heat loss through the floor slab was noted and compared with the analytical solution (Delsante and Stokes, 1983). They differed by 0.032%. Most subsequent models, both SSMs and DMs, shared this same mesh density defined by dx0, dz0, fxL0, fxF0 and fz0; the exceptions being GC45b and GC45c discussed above and the very large slab case, GC50b. DMs were evolved from SSMs by including mass and replacing the LAE solver with an ODE solver. DMs were initially tested by running them to quasi-steady state and comparing the averaged outcome with that of a SSM for the same problem. The problems were designed so that these two results should be identical. They were found to differ typically by 0.005% or less. Further tests of SSMs and DMs are detailed in the sub-section describing confirmatory test runs below.

Aditionally, the output of the several participating models were compared for each of the cases. A number of modelling and programming errors were thus highlighted and eliminated. The final range of disagreement among simulation results was 24% for the whole group and 4.2% for the sub-set described as reference models (TRNSYS, FLUENT and MATLAB).

Modelling options

Since bespoke programs were prepared specifically for this task, the question of options does not arise. All aspects were modelled as specified insofar as the finite difference domain allows. The three types of boundary condition (isothermal, adiabatic and convective), for example, were explicitly and separately modelled. Convective surface coefficients behaved well, even at large values. Radiation and moisture were excluded from the Test Specification as an idealization to simplify diagnostics, and consequently were not included in the MATLAB model.

Modelling difficulties

No difficulties were experienced with the use of the Test Specification or the values specified therein. The major modelling difficulty arose out of the nature of the test problems themselves. Following the construction of the first DM, a first principles method (Crowley and Hashmi, 2000) was used to establish the pre-conditioning period for the system. The pre-conditioning period is the simulation time required to allow the temperatures of all nodes to converge to values which are no longer affected by their arbitrarily chosen initial values. This is identical to (i) the longest thermal time constant for the system and (ii) the time necessary to reach a quasi-steady state. It was needed to determine the required interval of integration. It was estimated by calculating the time taken for the slowest transient solution of the nodal equation set to decay to 0.1% of its initial value. The period found was 28.25 years – an exceptionally long one in the context of building energy simulation.

Scaled runs were next carried out to estimate the real time required for a 30-40 year interval of integration and it was found to be 1-2 months on the available PC described above. Because of the number of test runs and confirming runs required, this would have taken the work well beyond the time scale for the project. Initially it was decided to increase the grid spacings in the DMs by a factor of five so that 40 years of simulated time could be processed overnight. However, this led to a 2% discrepancy between the averaged output of the DM, GC40a, and the result for its steady state counterpart, GC30a. This was greater than the difference for TRNSYS and FLUENT (< 0.6% variation) so, instead, four more PCs of similar specification were equipped to work on the project and a 10-year integration interval was investigated (using the original high mesh density). It was found that the results for 10 year runs were well within the principal error tolerance specified, that is, that there be at most 0.1% variation in floor heat loss between the last hour of each of the final two simulated years. This finding does not appear probable in the light of the very long pre-conditioning period but the apparent contradiction can be understood as follows: (i) performance is judged on floor heat flow rate which involves just a small fraction of the total number of nodal temperatures solved for in each simulation run and (ii) the initial conditions used here for each dynamic test run were just the solution temperatures for the corresponding steady state run. Since the average of the harmonically varying ambient temperature used in the DMs is the same as the outdoor temperature used in the SSMs, this meant that the nodes furthest from the surface – and therefore slowest to respond – were already close to their quasi-steady temperatures.

Ten-year integration intervals were thus adopted for the DMs and, as expected, most cases required 1-2 weeks processing time, with some taking as long as 12 weeks to complete. The SSMs, on the other hand, ran for 45-70 seconds typically for the same number of equations – upwards of 112,000. There are two reasons for the great difference in run times: (i) the DMs require more computation because of the long integration intervals and the stiffness of the equation system, and (ii) within MATLAB, LAE solvers are compiled whereas ODE solvers are interpreted. One of the DMs was subsequently written in FORTRAN and this, together with other improvements, led to a speed-up factor of approximately 750.

Confirmatory test runs

The Test Specification calls for supplementary test runs to confirm that the principal result for each of the cases is close to exact. This is demonstrated by varying problem parameters and confirming minimal sensitivity of results to these changes. The prescribed changes are as follows:

(i) Reduce the integration interval by one year (does not apply to SSMs).

(ii) Reduce the volume of soil modelled (applies to case GC10a only).

(iii) Reduce the mesh density (i.e. reduce the number of nodes used).

(iv) Increase the error tolerance or convergence tolerance used in the solver (does not apply to LAE solver used in SSMs here).

The specified maximum allowable variation in floor slab heat loss is 0.1% for any of these changes. A 5% change in the above parameters was, where possible, used in the confirmatory runs here, i.e. 50 times the permitted change in floor heat loss. For (ii) and (iii) the number of nodes was reduced by 5% – by reducing E and F simultaneously in the case of (ii). For (iv) the tolerance variables RelTol and AbsTol were each increased by 5%. Because the dynamic runs took so long, sensitivity tests (iii) and (iv) were usually carried out together. For the same reason, not all of the cases were tested in this way; the chosen cases are representative problem types.

The test results are presented in Table 1. The optimal runs are identified and the parameters varied in each of the confirmatory runs are highlighted. In every case, the change in floor slab heat loss was much less than the permitted 0.1%.

Modelling errors/improvements

During the project a number of error/improvements were identified by comparing output with the other two reference models. A programming error became apparent when convective boundary conditions were first introduced. MATLAB’s floor heat loss predictions for several cases were initially 3% to 5% higher than those of TRNSYS and FLUENT. An error in the modelling of convection within the edge loss routine was quickly located and corrected; the edge loss subprogram models floor heat loss just inside the perimeter boundary. A further minor interpolation improvement was later included in this same edge loss routine. A three-dimensional interpolation function was initially used to produce a set of sub-surface temperatures requested in the Test Specification. The results disagreed with TRNSYS and FLUENT. Use of a one-dimensional interpolation function – the required interpolation points being collinear with the sub-surface nodes – resulted in better agreement. The nature of the problem was not established with certainty. MATLAB’s three-dimensional function interpolates using 64 neighboring points. It may be utilizing unrepresentative nodal temperatures considering the small scale of the perimeter boundary and the large temperature gradients in its vicinity.

RESULTS

The results produced by MATLAB are in good agreement with those of TRNSYS and FLUENT. The final range of disagreement among simulation results for these three models is 4.2%. All three programs are within 0.3% of the known exact solution for case GC10a. Each of them under-predicts the floor heat loss for this case, with MATLAB producing the highest heat flow of the three. For most other cases MATLAB predicts slightly higher heat flow rates also – typically about 2% higher. This may be attributable to the denser mesh used in MATLAB but this has not been confirmed.

CONCLUSION & RECOMMENDATIONS

It can therefore be concluded that the model presented here is fit for purpose.

All of the test cases included here are concerned with steady or quasi-steady state solutions – even for the dynamic problems. Dynamic performance is not severely tested, for example, by step changes in the load. To do this one needs analytical solutions for dynamic cases but these are generally not available. Almost as useful are what can be termed 'converged solutions' (Crowley, 2006). These can be generated to arbitrary accuracy for dynamic problems of realistic scale and complexity.

The very lengthy thermal time constants discussed above for ground coupled heat transfer problems such as slab in/on grade – also applying to basements, earth tube cooling systems and buried coils – have serious consequences for the HVAC design and simulation communities when this heat transfer mode is a significant fraction of the total thermal load. Normally one discards the initial output of a simulation run, that is, the output calculated during the pre-conditioning period, which is typically of the order of weeks. In doing this, we are recognizing that the initial transient response of the building is not accessible to us without knowledge of the initial temperature conditions throughout the building. If, however, the pre-conditioning period is decades in duration, as is the case when heat is transferred to/from the ground, then temperature predictions are inaccurate for a significant fraction of the building life unless initial conditions are directly measured – using bore-holes for example. Standard sub-surface temperature profiles are of little use on most sites, which will have been disturbed by site excavation during the construction phase and probably by road works, pipe laying and construction of adjacent buildings prior to that.

ACKNOWLEDGEMENTS

The author wishes to acknowledge the cooperation of the International Energy Agency (IEA) Testing and Validation of Building Energy Simulation Tools Experts Group (IEA SHC Task 34 / ECBCS Annex 43) and in particular the leaders of Subtask A: Joel Neymark, J. Neymark & Associates and Ron Judkoff, National Renewable Energy Laboratory, US.

The financial support of the Dublin Institute of Technology is gratefully acknowledged.

NOMENCLATURE

t	time (s)
T	nodal temperature (K)
x, y, z	space coordinates (m)
α	thermal diffusivity (m^2/s)
δa	absolute uncertainty in a
DM	dynamic model
LAE	linear algebraic equation
ODE	ordinary differential equation
SSM	steady state model

REFERENCES

Bogacki, P. and Shampine, L.F., 1989. A 3(2) pair of Runge-Kutta formulas. *Appl. Math. Lett.*, 2 (4), 321-325.

Crowley, M.E., 2006. Validation of solution methods for building energy simulation. Proceedings of *eSim 2006: 4th Biennial Building Performance Simulation Conference*, University of Toronto, Canada, 173-180.

Crowley, M.E., Hashmi, M.S.J., 2000. Improved direct solver for building energy simulation. *Proc. CIBSE A: Building Serv. Eng. Res. Technol.*, 21 (3), 167-173.

Davis, T. A., 2004. Algorithm 832: UMFPACK V4.3---an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Software*, 30 (2), 196-199.

Delsante, A.E. and Stokes, A.N., 1983. Application of Fourier transforms to periodic heat flow into the ground under a building. *Int. J. Heat Mass Transfer*, 26 (1) 121-132.

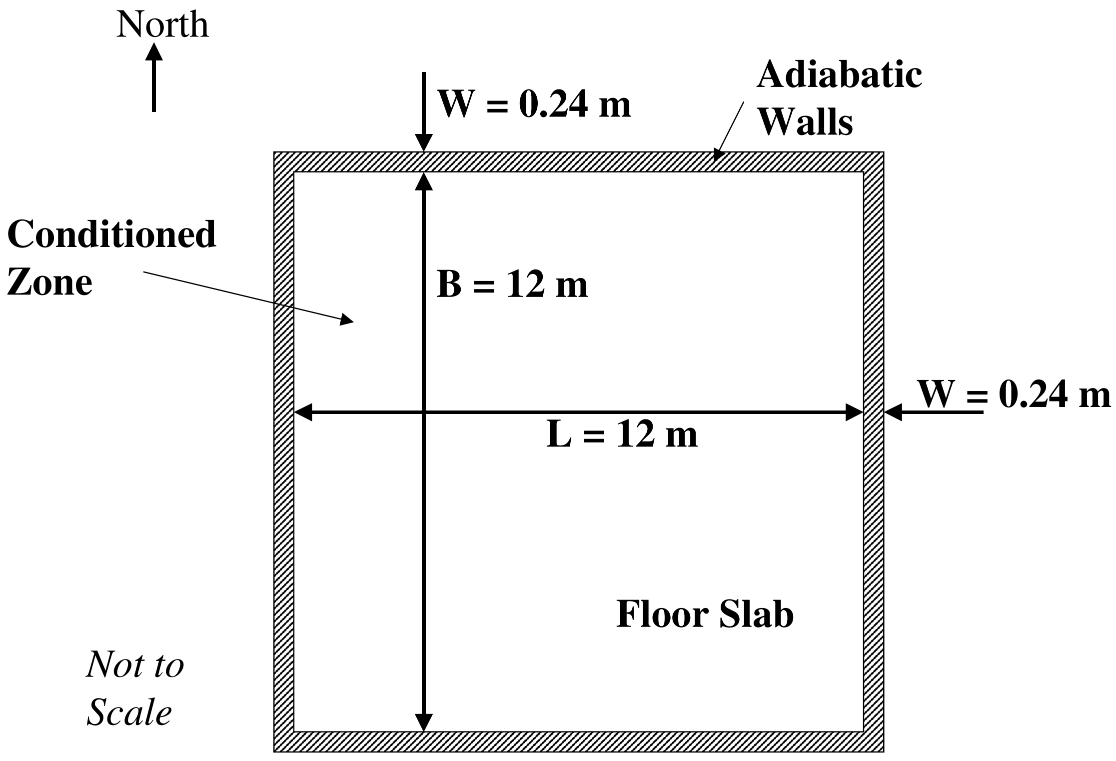
Deru, M., (2003). *A model for ground-coupled heat and moisture transfer from buildings*. Golden, Colorado: National Renewable Energy Laboratory [NREL/TP-550-33954]. [Available at http://www.nrel.gov/docs/fy03osti/33954.pdf]

Fletcher, C.A.J., 1991. *Computational techniques for fluid dynamics, Vol. 1.* (2nd ed.) Berlin: Springer-Verlag.

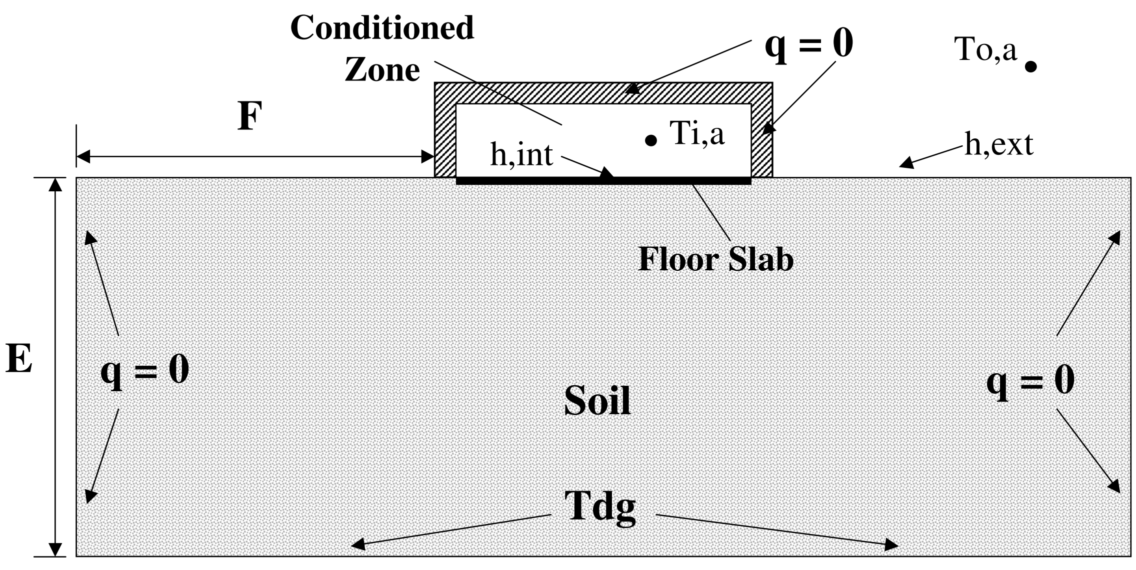
Heath, M.T., 2002. *Scientific computing; An introductory survey*. (2nd ed.) Boston: McGraw-Hill.

Neymark, J., Judkoff, R., Beausoleil-Morrison, I., Crowley, M., Deru, M., Henninger, R., Nakhi, A., Ribberink, H., Thornton, J., Wijsman, A., Witte, M. (2008). *IEA BESTEST In-Depth Diagnostic Cases for Ground Coupled Heat Transfer Related to Slab-On-Grade Construction*. Golden, Colorado: National Renewable Energy Laboratory [NREL/TP-550-43388]. In Conjunction with International Energy Agency Solar, Heating and Cooling Programme Task 34 / Energy Conservation in Buildings and Community Systems Annex 43. [Available at http://www.nrel.gov/docs/fy08osti/43388.pdf]

The MathWorks, Inc., 2007. *Getting started with MATLAB*. Massachusetts: The MathWorks, Inc.



Plan view



Sectional elevation

Figure 1 Geometry of a typical test problem (Neymark et al., 2008).

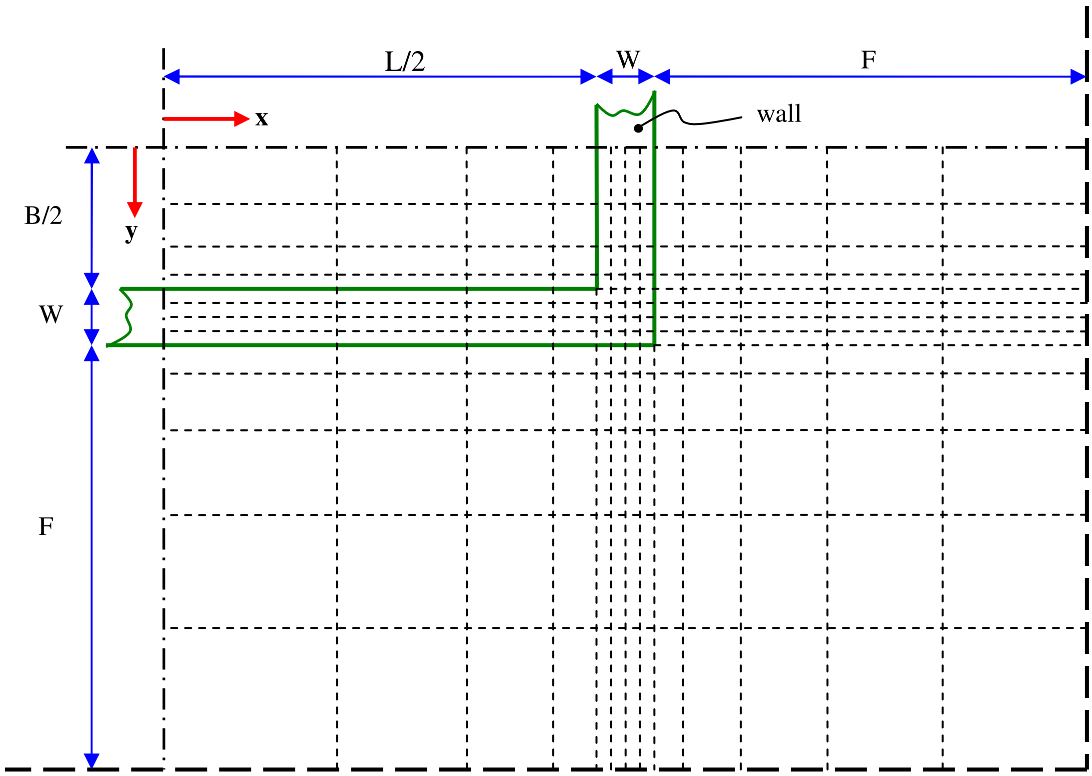


Figure 2 Mesh pattern for a test problem that includes two axes of symmetry.

Table 1
Optimal and confirmatory runs for selected cases

Case	F (m)	E (m)	dx0 (m)	dz0 (m)	fxL0	fxF0	fz0	Number of nodes	Floor slab heat loss	Comments
GC10a*	150	300	0.020	0.012	1.150	1.495	1.150	892,552	2431.82 W	Optimal run.
GC10a	120	240	0.020	0.012	1.150	1.495	1.150	849,120	2431.60 W	0.009% change.
GC10a	150	300	0.020	0.012	1.155	1.5015	1.155	849,120	2431.54 W	0.012% change.
GC30a	20	30	0.015	0.0090	1.135	1.4755	1.135	840,840	2694.99 W	Optimal run.
GC30a	20	30	0.016	0.0096	1.135	1.4755	1.135	798,720	2694.87 W	0.004% change.
GC40a	20	30	0.015	0.0090	1.135	1.4755	1.135	840,840	23,608.9 kWh/y	Optimal run, (10 year run).
GC40a	20	30	0.015	0.0090	1.135	1.4755	1.135	840,840	23,609.1 kWh/y	0.001% change, (9 year run).
GC40a[#]	20	30	0.016	0.0096	1.135	1.4755	1.135	798,720	23,607.8 kWh/y	0.005% change, (10 year run).
GC45b	15	15	0.020	0.012	1.135	1.4755	1.135	410,000	33,483.5 kWh/y	Optimal run, (10 year run).
GC45b	15	15	0.020	0.012	1.135	1.4755	1.135	410,000	33,483.5 kWh/y	0.000% change, (9 year run).
GC45b[#]	15	15	0.020	0.012	1.138	1.4794	1.138	384,160	33,479.2 kWh/y	0.013% change, (10 year run).
GC65b	15	15	0.015	0.0090	1.135	1.4755	1.135	715,520	2003.66 W	Optimal run.
GC65b	15	15	0.015	0.0090	1.140	1.4820	1.140	677,376	2003.50 W	0.008% change.
GC80b	15	15	0.015	0.0090	1.135	1.4755	1.135	715,520	6151.43 kWh/y	Optimal run, (10 year run).
GC80b	15	15	0.015	0.0090	1.135	1.4755	1.135	715,520	6151.53 kWh/y	0.002% change, (9 year run).
GC80b[#]	15	15	0.015	0.0090	1.140	1.4820	1.140	677,376	6150.86 kWh/y	0.009% change, (10 year run).

*Analytical solution is 2432.60 W. [#]Error tolerance for ODE solver was increased by 5%.