Software tools for the technical evaluation of design alternatives

T. Mayer
University of Strathclyde (United Kingdom)

INTRODUCTION

Designing buildings which 'work' - economically, socially and technically - remains the central challenge for architects. This paper is concerned with the state of development of software tools for the evaluation of the technical issues which are relevant at the conceptual stages, as opposed to the detailed stages, of design decision-making.

The technical efficiency of building is of enormous economic importance. The capital investment in building in Europe represents some 12% of the Gross Domestic Product; this capital investment is exceeded by an order of magnitude, however, by the operating costs of buildings over their life span. In turn, these operating costs are exceeded - again by an order of magnitude - by the costs associated with the (human) operations which go on within the building, and on which the design of the building has some impact [Mayer, 1986].

THERMAL MODELS

Let us take, by way of example, those sub systems within buildings which contribute to the control of the thermal environment. In many building types in Northern Europe, the engineering services account for over half the capital cost of the building - ie 6% of the GDP. Much more importantly, the annual energy consumption in these buildings accounts for over half of all of the energy delivered annually in Europe. According to the estimates of the UK Department of Energy, up to 50% of this energy consumption could be saved by more energy conscious building design.

What is less predictable, but economically even more significant, is the loss of efficiency in the operations taking place within buildings which are environmentally unsuitable for their purpose: the social consequences in the UK of the rise in energy prices over the last decade - in terms of fuel poverty and the demolition of hoses unfit for human habitation - are truly tragic.

In the early days of CAAD when many of us were committed (- as some of us still are -) to the development of integrated computer systems appropriate to the early conceptual stages of design decision making, there were rare opportunities to present prototype systems to practicing architects. It was s safe bet, when such an opportunity arose, that the first questioner would query why the program attempted to calculate heat-loss - 'surely a matter for the consultant and not for the designer'.

In this abrogation of technical responsibility, the architect found a ready accomplice in the services engineer. After the design was complete, the engineer would, with appropriate 'factors-of-safety', multiply the area of the building envelope by a 'U' value, add the product of air-change-rate by building volume, double it, and choose a boiler twice that size. Little wonder that there is 50% saving to be made on those buildings which are yet to be prematurely demolished!

Regrettably, the conspiracy against the building owner/user did not end there. Those of us in CAAD sanctified the architects ignorance and the engineers naivety by <u>encoding</u> rules-of-thumb. Early energy programs had all the characteristics of engine-driven cylinder lawn mowers: ie the application of great power to s mechanism specifically designed to be handcranked!

When it dawned on the Royal Institute of British Architects simultaneously and traumatically that:

a. the energy behaviour of buildings may be significant, and that
b. computers might have some modest role to play, if not in design at least in the low grade service activities of engineering consultants,

there was an unholy rush to offer, commercially, a calculator based 'energy-model'. Little or no account was taken of long-standing research and development or of the growing body of results from attempts to validate increasingly sophisticated energy models.

As shown by Clarke in his book Energy Simulation in Building Design [Clarke, 1985], the energy flowpaths in buildings are truly complex (Figure 1); factors to be considered include:

- transient conduction through the building envelope
- time-dependent sensible and latent heat gains
- infiltration, natural and controlled ventilation and air movement
- shortwave solar radiation
- longwave radiation exchange
- shading of opaque and translucent surfaces
- insolation of internal surfaces
- time-varying convection
- effects of moisture

Clarke identifies four categories of energy model - steady state, simple dynamic, response function and numeric; each is concerned, at its own level, to satisfy the first and second laws of thermodynamics but, as the level of sophistication of the method falls, many of the active flowpaths are ignored and the method becomes indicative rather than deterministic. The categories are summarised as follows:

- steady state: these methods (the RIBA Calculator is an example) have no mechanism for the accurate inclusion of the effects of solar gain, casual gain, longwave radiation exchanges, plant operation, etc. and typically address only fabric heat loss.
- simple dynamic: these methods are mostly based on regression techniques applied to the results of multiple parametric runs of more powerful modelling systems
- response function: by careful specification of system boundary conditions, these methods solve the partial differential heat equation as a means of modelling the dynamic response of the building
- numeric: finite difference methods allow the tracking of all relevant energy and mass flowpaths in the building and the simultaneous solution of the complete set of energy equations. The finite difference model ESP [Clarke, 1982] which is becoming accepted as a European standard is typical of this category of approach.

Not surprisingly, different approaches yield different results. In an exercise to determine the appropriateness or otherwise of building energy regulations introduced to the UK in 1979, the RIBA Calculator and ESP were both applied to a hypothetical but entirely typical building [ABACUS and VALTOS, 1979]. Figure 2 compares the results, which could not be more at

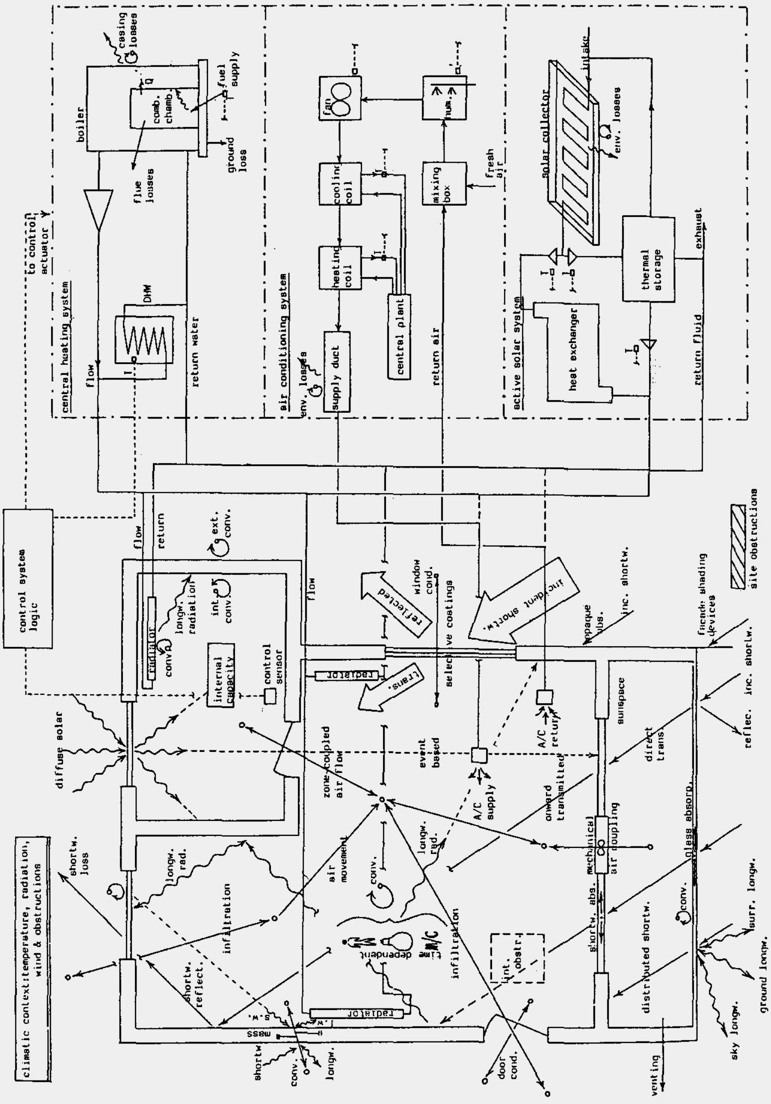

Figure 1 Energy Flowpaths in a Building

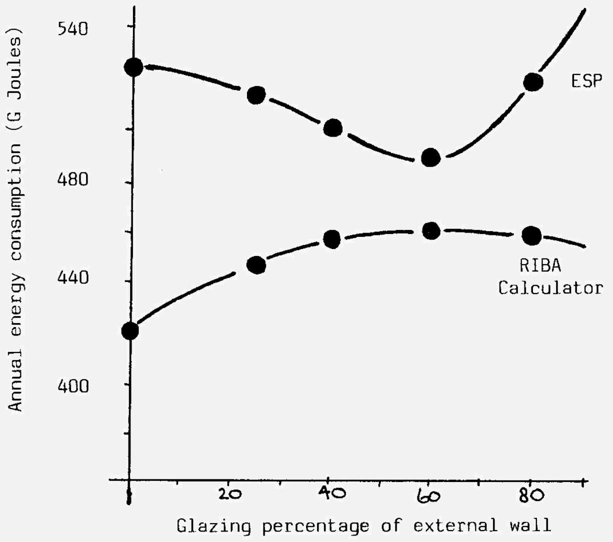

Figure 2 Comparison of ESP and RIBA Calculator

odds: ESP shows a minimum energy consumption at 60% glazing, the RIBA Calculator shows a maximum energy consumption at 60% glazing; to achieve a minimum consumption of energy, the user of the RIBA Calculator would have gone for an un-glazed building! The energy regulations, in this particular case, would in fact have restricted the architect to a maximum of 25% glazing.

But where does the truth lie? This question opens up the vexed issues of model validation and of the trade-off between accuracy and ease of use. Before these are considered, it is perhaps appropriate to generalise the discussion to other technical issues in building design.

LIGHTING MODELS

Lighting in buildings is now receiving considerable attention. Clearly the interaction of light and heat energy is central to serious study of the concept of integrated environmental design, and particularly to the contribution which fenestration and automatic switching can make to energy conscious buildings.

According to Grant [Grant, 1987] the methods of calculating the reflected components of the light are what determine the sophistication and accuracy of the models. The most basic types rely on the split flux method. This assumes that light reflected and inter-reflected inside a room will behave am if the room were the interior of a sphere and all surfaces were perfect diffusers (ie Lambertian). None of these assumptions are correct and so errors must inevitably be introduced by a simplification of this nature. Any model written around this principle is unable to represent accurately an enclosure geometry that is non-orthogonal. The illumination at any point is considered as the sum of the illuminance from all sources incident on that point.

To progress beyond this stage the ability to assess the inter-reflection of direct illuminance enables further realisation of the potential of the system. This level of model, eg SUPERLITE [LBL, 1985] is the most sophisticated of those to be found in widespread usage. However, at this level there is still an inherent assumption in that the surfaces are only capable of being defined as perfect diffusers, the direct and reflected components being estimated by numerically integrating surface illuminances in iterative loops. Internal reflection between surfaces is calculated from angle factors found by calculating the fraction of energy leaving an element of one surface which arrives at a second surface. It is a function of the relative orientation of the surfaces, their separating distance, intersurface obstructions and their reflective characteristics. The main problem in determining the angle factor using numerical methods is to ensure both reciprocity and conservation of energy. Most techniques stress either one or the other of these requirements by, for example, calculating only half the angle factors or by attempting to ensure that all the elemental areas are equal.

In most architectural environments there are a high proportion of surfaces that exhibit specular or off-specular properties. These are difficult to model unless a rigorous first principle approach is employed. This necessitates the deployment of a means of recursively tracking multiple reflections within an enclosure, eg DIM [ABACUS, 1987]. The recursive technique involves tracing a ray from reflection to reflection until the intensity of the reflected ray falls below some threshold value or some arbitrary number of inter-reflections is exceeded. Alternatively a simultaneous or matrix method may be employed involving the derivation of a matrix of coefficients which express the relationship between the surfaces. It is possible to establish a set of illuminocity balance equations of the form AL=E which can be solved simultaneously using standard matrix inversion techniques.

THE PROBLEMS

The parallels between methods for technically evaluating the thermal environment and the lighting environment in buildings are echoed in the other aspects of technical performance. Evaluation of the acoustic environment, for instance, can be carried out at various levels of rigour which almost exactly parallel the categories of model appropriate to heating and lighting. Even in structural analysis we see the equivalent of the 'steady-state' model and the 'finite difference' (or in this case 'finite element') model.

Whichever aspect of the technical behaviour of the building is being evaluated, the designer is faced with the problem of choosing between relatively easy access to a simplistic model or relatively difficult access to a sophisticated model. The plea from architects for simple design tools is understandable but worrying; why, if we can model the physics of the phenomena, should we degrade the process to a 'rule-of-thumb' which, like the RIBA Calculator, may give wholly erroneous guidance?

On the other hand, if sophisticated evaluation systems are to be accepted and used by designers, they will have to be available on affordable machines, easy to use, and operate at a technical level appropriate to the designer's knowledge. Particular difficulties faced by the user relate to

- data input: the cheer quantity of data required to describe/manipulate the building present a major problem. Not only is the gathering of this data a time consuming task but frequently the data has not yet been specified, as in the case at the early design stages. Also, due to the complex inter-relationships, ensuring the integrity of the data can demand very high levels of understanding of the underlying simulation principle.
  This creates two problems for the user. Firstly, if the data requested is not available, no help is provided to generate a sensible default. Secondly, without s good knowledge of the simulation mechanism, the importance, and hence the required accuracy, of an individual piece of data is very difficult to judge.
  As well as the question of 'what', there is also the problem of 'how' to input such a large quantity of highly inter-related data. The

various factors associated with data acquisition, together with the users' often idiosyncratic conceptualisation of its interrelationships, tend to conflict with the rigid question/answer style of input common to many of todays progress.

Control: generally, control of the appraisal does not require much sophisticated user interaction. At this stage the major difficulty faced by the user is the selection of the simulation parameters to produce a sufficient quality and quantity of output to allow a worthwhile appraisal of the building. For the novice, a lack of understanding of the implications of the selections being made can lead to confusion, or even erroneous deductions, due to the inadequacy of the output date.

- Output: it is here that the requirements of the novice and expert differ most. The expert will be trying to detect patterns in and relationships between different building parameters, in order to build up a picture of the dominant energy flowpaths. To do this, all the data generated by the simulation has to be available and capable of being displayed in juxtaposition with any other data. The novice, on the other hand, merely wishes a concise summary of the building performance, preferably in terms of those variables most meaningful to himself and him client. Unfortunately, due to the primitive nature of the system's output, the novice may experience difficulty relating poor performance to the design decisions that caused the problem, or, indeed, even to the possible design modifications that could improve this performance.

There are also problems for the software developers:

- The software structure is often extremely inflexible and unyielding. The program will have been conceived in a now outdated machine environment; this means that the structure is monolithic, imposing extreme management and updating difficulties.

- The software structure is often inelegant, with the application knowledge inextricably bound to the source code. It is therefore extremely difficult to upgrade individual algorithms within the simulation mince these may require the detailed knowledge of data structures, internal memory and the side effects of one change on the rest of the software package.

THE WAY FORWARD

There is growing recognition of the need for a radically new approach to the design, development and maintenance of the software for technical evaluation within CAAD systems. In the USA and in the UK very substantial funding has been allocated to the development of an Intelligent Front End and a socalled KERNEL System for the evaluation of the energy behaviour of buildings; it is anticipated that the IFE and the Kernel concept could be extended to all areas of technical concern in building design.

An appropriate Intelligent Front End for software to simulate some aspect of the technical performance of buildings might be structured as in Figure 3. This envisages the system as a suite of independent modules communicating via a central corking memory, ie a sort of 'blackboard'. The modules of such an IFE are:

- Blackboard: this module is the communciations centre of the system. The other modules examine the blackboard for information they can use, and post their results back on it. This scheme facilitates multiple use of information, eg one user statement posted by the dialogue handler may be useful simultaneously for the user model, knowledge base, planner and building model.
- Knowledge Base: the crucial area is the knowledge base, upon which everything else hangs, and the design/implementation of this constitutes the major part of the required development. It is envisaged that it would be implemented in Prolog.

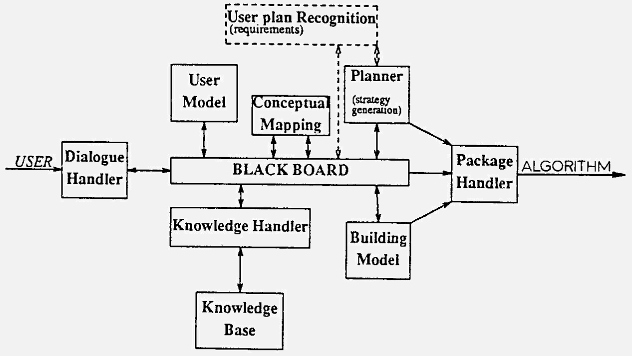

Figure 3 Outline of System Architecture

- Dialogue Handler: this would deal with the flexible, and extensible, command language, which would be designed to give the user the opportunity to volunteer information, abort or redicrect the systems line of inquiry and give 'I don't know' replies. A state transition network would be used to provide maximum flexibility and to facilitate dialogue control by the user model.
- User Model: the user would be classified in one of a small number of categories. For example, an 'architect' would not be asked to provide information about the plant control strategy, whereas an 'energy modeller' would be expected to provide information about the timestep to be used for the simulation. The decision will be based on a user database, initialised by querying the user, and modified should the dialogue appear to be breaking down.
- Plan Recognition: the user would be required to state his/her objectives in the predefined terms that the planner requires, eg overheating analysis, solar gain analysis, etc. However, should the dialogue be extended to allow natural language, then some form of plan recognition would become essential, and therefore the system, and in particular the planner, will be constructed such that an appropriate module can be added.
- Planning: in order to avoid major modifications to existing software, planning would initially be restricted to selection of one out of a series of predefined processes. For example, if the user indicates that he/she wishes to investigate the possibility of overheating, the planner would initiate the relevant simulation and display the necessary output results in a separate window.
- Building Model: this module would depend rather critically on the design of the knowledge base module. At the moment, it is envisaged as a fairly straightforward database, containing the building's geometry, construction, occupancy, etc, as input by the user or as supplied by the knowledge base.
- Back-end Handling: initially this will simply extract the data from the building model and planner, creating the data and control files to drive the simulation in a 'batch' mode.

The Kernel is an object-oriented system which can be used by model builders to construct programs of <u>any</u> architecture, for any purpose, from primitive objects. The system would allow different individuals end different organisations to operate within their own field of competence while having ready access to the developments of others.

The possible elements of the Kernel System are:

- A template which is constructed to define the objects and interconnections which will comprise the proposed model
- This is passed to the harness which then constructs the program, outputing it in the form of a multi-object model, expressed in source or executable code, and located in one or more files
- Objects exist to hold the working primitives which are manipulated by the Kernel
- An object dictionary holds object definitions including a description of its data input and output
- A management program exists to control the entry of objects and their dictionary entry

Quite clearly, the implementation of an appropriate Intelligent Front End for existing models and the development of subsequent generations of models based on the Kernel requires sustained intellectual effort, significant sponsor funding and a deal of patience on the part of designers. But surely it will be worth it!

ACKNOWLEDGEMENTS

The R&D effort over the last two decades which has informed the views expressed in this paper have been supported in large measure by the UK Science and Engineering Research Council and by DGV of the Commission of European Communities. Most of the ideas for the Intelligent Front End and the Kernel System have been generated by Joe Clarke of ABACUS and Damian MacRandal of the Rutherford Appleton Laboratory and I am grateful for the opportunity to give these ideas an audience at CAAD Futures.

REFERENCES

ABACUS. Working document on the inter-reflective lighting model DIM, ABACUS, University of Strathclyde, (1987).

ABACUS and I/ALTOS. 'Deemed to Satisfy', Architects Journal 24 October (1979).

Clarke, J. A. ESP User Manual ABACUS, University of Strathclyde, (1982).

Clarke, J. A. Energy Simulation in Building Design Adam Huger, Bristol, (1985).

Grant, M. Report to PASTOR Daylight Group ABACUS, University of Strathclyde, (1987).

Lawrence Berkeley Laboratory. SUPERLITE 1.0 Program Description Summary Lawrence Berkeley Laboratory, (1985).

Maver, T. W. 'Social Impacts of Computer Aided Architectural Design', Design Studies Vol 7, No 4, (October 1986).