ISLEC®
A SELF-EMERGING DYNAMIC SIMULATION MODEL

Dr L Jankovic
InteSys Ltd
University of Central England
Birmingham, UK

ABSTRACT

InteSys Limited has developed a new dynamic simulation model for buildings, which emerges by itself as it receives the data or signals from the building under consideration. The model exists in two forms Software written for PC Windows environment; and Hardware in the forra of a silicon microchip. This work is an extension of the previous work by Jankovic (1991 -1993). The software has been developed in C" and runs under Windows 3.1 or above. The paper describes the operation of this software.

INTRODUCTION

ISLEC® stands for 'Intelligent Self Learning Electronic Chip', of which this paper describes the software equivalent. The software provides a dynamic simulation and modelling capability for complex systems. The models emerge by themselves as the software receives data or signals from the dynamic complex system under consideration.

ISLEC® was conceived in 1990 for the purpose of modelling and simulation of dynamic heat transfer in buildings. The model was improved through several major versions, until it became clear that its application was not limited to buildings but covered many other dynamic systems, such as modelling of road traffic, telecommunications traffic, electricity and gas demand, and others.

The development of ISLEC® started as the after-work activity of a University Research Fellow, and subsequently received a SMART Award from the UK Department of Trade and Industry. At this point, the University of Central England also became interested and provided matched funding to the SMART Award. Subsequently, InteSys Ltd was born and the ISLEC® microchip developed.

The ISLEC® method was created as a result of frustration arising from the operation of conventional simulation models for buildings. These models were based on the principle of describing every component of the building by means of simultaneous differential equations and combining all components and their models into a unified (and complex) mathematical model. Such models were not easy to operate and run. Typically, it would take a week for an experienced operator to assemble the model of a building. Further time was then needed to run the model.

By contrast, ISLEC® abandons the conventional principles of Physics and considers dynamic systems as objects, or black boxes, where only inputs and outputs matter, not the Physics inside the black box. It took time to develop the model and to make it capable of emerging by itself from the data by extracting the essence of the dynamic system under consideration. The result was something totally different from conventional mathematical models: a set of relatively simple equations used in a unique way. ISLEC®'s computational non-intensity allows for its implementation onto a microchip.

This software replicates all functions of the ISLEC® microchip, making the underlying modelling method available for use by professionals on their PC's, as opposed to the microchip, which is available for use by manufacturers of electronic control systems.

Today, ISLEC® is considered a leading edge technology. It is a powerful and versatile modelling tool which requires minimum effort from the user, but, at the same time, an in-depth understanding of the modelled process and the driving forces which affect it. Therefore, ISLEC® will not perform miracles, but will act as a tool to broaden professional horizons and enable informed decisions to be made on the basis of the results it produces. It behaves like a violin which plays better, the more skilled the user becomes. The opportunity to generate revenues or savings based on the results gained from ISLEC® gives any organisation a competitive edge previously unavailable with conventional tools.

OBJECTIVE OF THE SOFTWARE

The software provides a dynamic simulation and modelling capability for complex systems. It delivers the necessary tool for the development of predictive control systems which can, for example, save up to 40 % of energy costs in buildings (Scartezzini 1991).

TECHNICAL BASIS

In general, a complex dynamic system driven by n independent variables can be described as

$$Y = f(X_1, X_2, ..., X_n) \quad (1)$$

where $X_1, X_2, ..., X_n$ are independent variables which are functions of time.

Over a moving range r scatter-plots of pairs of variables

$$\begin{array}{l} Y, X_1 \\ Y, X_2 \\ \vdots \\ Y, X_r \end{array} \quad (2)$$

generally appear to be in random relationships (Fig. 1). However, filtering of the scatter-plots (2) reveals non-linear relationships between the dependent variable and individual independent variables:

$$\begin{array}{l} Y_1 = f_1(X_1) \\ Y_2 = f_2(X_2) \\ \vdots \\ Y_n = f_n(X_n) \end{array} \quad (3)$$

The behaviour of the complex dynamic system can then be reconstructed as:

$$Y = Y_1 + Y_2 + ... + Y_n \quad (4)$$

or more generally

$$Y = Y_1 \times w_1 + Y_2 \times w_n + \quad + Y_n \times w_r \quad (5)$$

where $w_1, w_2, ..., w_n$ are optional weighting factors.

The dynamic functions $f_1, f_2, ..., f_n$ from equations (3) represent non-linear contributions of individual independent variables to the dependent variable Y, the system behaviour function.

The simplicity of this method results in its important advantage: the capability to adapt on-line, to learn and to forget as the process travels in the time domain spanning a moving range of r time steps.

OPERATION

Modelling and Simulation

This section describes the steps necessary to carry out modelling and simulation of a complex system.

Modelling is defined as the building of a logical machine which replicates the functions of the system under consideration in the form of a computer program.

Simulation is defined as a numerical experiment in which the model is subjected to dynamically changing conditions with the purpose of investigating its behaviour.

The dynamically changing conditions will be the description of the actual conditions affecting the actual system, supplied by the user to the model in the form of computer data.

Starting The ISLEC® Software Session

The ISLEC® software session is initiated by selecting and double clicking on the ISLEC4.1 icon from Windows Program Manager.

Loading of input data file

A typical modelling and simulation task will start with loading a data file by selecting the Input file option from the file pull down menu or by pressing the Load button, (Fig.2). If there is a previously loaded data file in the memory, the current preferences will be saved in an *.INI file automatically, before loading a new input data file.

Data Requirements

ISLEC® will model dynamic systems driven with up to 8 independent variables. These variables need to be supplied in ASCII form, in the time domain, one data point per time step. ISLEC® will also require the measured historic output of the dynamic system under consideration in order to assemble the model. A typical data file will therefore look as shown in Fig. 3. Each row in Fig. 3 contains all values for different variables during a single time step, and each column corresponds to a different dependent or independent variable. The maximum number of columns in the input data file is 9. The measured output can be in any column, which will be specified by the user in the Preferences dialogue box. Each variable, either dependent or independent, can be assigned scaling factors in terms of an offset and multiplication factor, through the Preferences dialogue box. The length of the input data file will be limited only by the capacity of the computer.

ISLEC® software will consider 2 different rows of the input data file as values of all variables during subsequent time steps. Therefore, there is no need to insert a column with information on time into the input data file.

Setting Up The Preferences

If a data file is being used for the first time with ISLEC® software, the user will most certainly not have an appropriate *.INI file which describes preferences for this data file. If the *.INI file does not exist, ISLEC® will warn the user and will assign a default preference parameters to the data file. The first step in this case would be to go into Preferences by pressing the Preference button or selecting the Preferences option from the Options pull-down menu.

In order to run a meaningful simulation, the user must set the Preferences to correspond with the actual data file if the default preferences were assigned. The user must therefore enter the columns in the data file, and the column number for measured variable (Fig.4). Subsequently, the user must enter time steps per natural cycle for each independent or dependent variable and the offset and multiplication factors.

Specifying The Driving Functions

The next step is to determine which driving functions will be used in the simulation. The driving functions can be specified from the driving functions dialogue box, which can be called from the Driving Functions button or from the Driving Functions option from the Options pull-down menu, as shown in Fig.5.

It should be noted that in the case when the measured value is in the last column of the input data file, then Driving function 1 will correspond to the first column in the input data file, Driving function 2 to the second column, and so on. In any other case, when the measured value is not in the last column of the input data file, but is, for example, in column number 3, the first 2 Driving functions will correspond to the first 2 columns of the input data file, but the third Driving function will correspond to the fourth column in the input data file, and so on. In the Preferences dialogue box the sequence of the time steps per cycle, offset factors and multiplication factors will correspond to the sequence of columns in the input data file, regardless of the column in which the measured value is.

Starting The Modelling And Simulation

The modelling and simulation can then be initiated with the Run option which can be accessed from the Run button or from the Run option in the File pull-down menu, (Fig.6). If no driving functions or consequences are selected for display, then the 2 curves which will always be displayed will be the measured value, shown in black, and the calculated value or the result, shown in grey, (Fig.6).

In the beginning of the simulation, the grey curve will not be displayed in the first few cycles during the initial learning period. After the initial learning period has been completed, the grey curve will start converging towards the black curve, the measured variable, and the accuracy of the simulation will be greater as it progresses further, (Fig.7)

Disengaging The Learning Mode

Once the simulation has started achieving reasonable results according to the user's judgement, it can proceed in two ways. The learning can either be kept continuously on, in which case the simulation result will represent the prediction of one cycle ahead of the the current time, (Fig.8). Alternatively, the learning mode can be switched off, using the Learn button or the Learning option from the Options pull-down Menu.

At the time step when learning was switched off, a vertical line will be drawn, (Fig.9). By switching the learning off, the measured value will be totally disregarded and the model will be driven entirely by the independent variables or the driving functions of the modelled system, (Fig.10).

Should the user wish to switch the learning mode off at a particular time, it is advisable to enter the ISLEC® software into step-by-step mode by pressing the Auto button. The user can then drive the simulation step-by-step by pressing any key on the keyboard until the desired time is reached. Having reached the desired time step and having switched the learning mode off, the user can again engage the automatic operation of the software by switching the same button, which will now be called NoAuto, (Fig.9).

After the learning mode has been switched off, the simulation will become less and less accurate the further it progresses from the point at which the learning was switched off. The user can engage or disengage the learning mode an infinite number of times, according to the requirements of a particular task.

Suppressing The Driving Functions

When learning is switched off the user can also decide to suppress the contribution of any of the driving functions to the calculated output, again, according to the requirements of the specific modelling task being carried out. That can be done by calling up the driving functions dialogue box by pressing the Driv button or selecting the Driving Functions option from the Options pull-down menu (Fig.11). The Suppression can be engaged and disengaged an infinite number of times during the simulation while the learning mode is switched off.

Comparing The Results Of Different Simulations

At the end of the simulation, mean error and standard deviation will be displayed in the lower right hand corner, (Fig.8, Fig.10, and Fig.12). These can be used to compare the results of different simulations. To do this, the user must record the errors manually. Errors are defined as absolute discrepancies between the measured and predicted variable. They can either be displayed in the units of the measured variable or in percentage terms depending on the selection made by the user in the Preferences dialogue box.

Portability Of The Model

At the end of simulation, the user can store the dynamic model created by ISLEC® software in a separate file by selecting the M Out (Model Out) button, (Fig.12).

Once the model file has been saved, the user can apply a new input data file to the existing model. This can be done by selecting the model file name to be loaded from the disk, by disengaging the Reset on Run option in the Preferences dialogue box, (Fig.13), and by loading a new data file with the same order of columns as the old data file.

If Reset on Run has been disengaged, then ISLEC® software will not start creating the new dynamic model when the simulation starts. Instead the software will use an existing model loaded from a *.MOD file or the model which is already in the memory.

A model stored in a data file can be loaded by pressing M In button (Model In), (Fig.14). If this operation is successful, a message in the lower left corner of the screen will appear saying Model Loaded. If there are any problems with loading, a warning message will be displayed in the lower left corner.

At this stage, the user can proceed with the simulation with a totally new data file by pressing the Load button or selecting the Input File option from the File pull-down Menu. The simulation can then be initiated using the Run button or Run option from the File pull-down Menu. The beginning of such new simulation with a new data file and the model loaded from an *.MOD file may show larger discrepancies between the measured and the calculated output if the new data file starts with significantly different values of measured variable in comparison with the values at the end of the previous data file, (Fig.15).

In this way, the effect of various scenarios of driving functions on the output of the model can be investigated by choosing different data files in which the desired driving function scenarios have been stored. These driving functions can simply be 'what if' scenarios, prepared totally synthetically by the user. The effects of different 'what if' scenarios can be compared by recording and comparing the errors of simulation displayed in the lower right corner of the screen.

It is very important to note that the models are portable only between data files with the same preferences. Application of a single model to data files with different preferences will cause unpredictable operation of the software.

Recording Of Detailed Results Of Simulations

Detailed results of simulations are recorded automatically in a file with the default name the same as the input data file and with the default extension "OUT". The user can change the name and extension for the output file in the Preferences dialogue box.

Output files can be very large and the user must do proper housekeeping in order to maintain the required files and discard unwanted files.

On exiting the ISLEC® software session, the current preferences will not be saved in an *.INI file automatically, and it is the user's responsibility to ensure that the preferences are saved if they are wanted.

Exiting ISLEC® Software Session

On completion of a modelling and simulation session, the user can exit ISLEC® software by selecting Exit option from the File pull down menu.

ACKNOWLEDGEMENTS

Financial support from the UK Department of Trade and Industry and the University of Central England in Birmingham for development of the ISLEC® microchip is gratefully acknowledged.

REFERENCES

Jankovic, L. (1993). "Performance of On-Line Adaptive Algorithms for Building Modelling and Control". Proceedings of CLIMA 2000 Conference, 1-3 November 1993, London. CIBSE.

Jankovic, L. (1993). "Steps Towards Greater System Intelligence". Proceedings of Intelligent Buildings Today and in the Future, 7th October 1993, Birmingham. UCE.

Jankovic, L. (1991) "Learning Models - A New Approach to Simulation". Proceedings of Building Simulation '91 Conference, Nice, France, 1991.

Jankovic, L. (1991) "Evaluation of Learning Models for Building Energy Simulation". In Proceedings of ISES Solar World Congress 1991, Denver, 1991.

Jankovic, L. (1991) "Building Simulation Models Which Learn", The Proceedings of CIBSE National Conference, Canterbury, 7-9 April 1991. CIBSE, London.

Scartezzini, J L (1991) Applying Stochastic Methods to Building Thermal Design and Control, 1989-1991, Lausanne, EPFL.

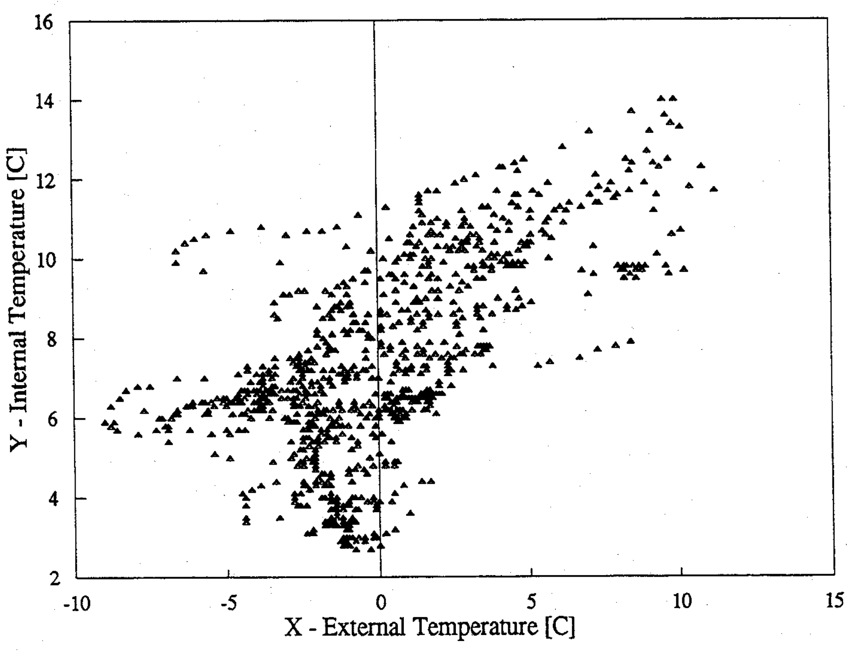

Figure 1 **Example of an XY scatter-plot**

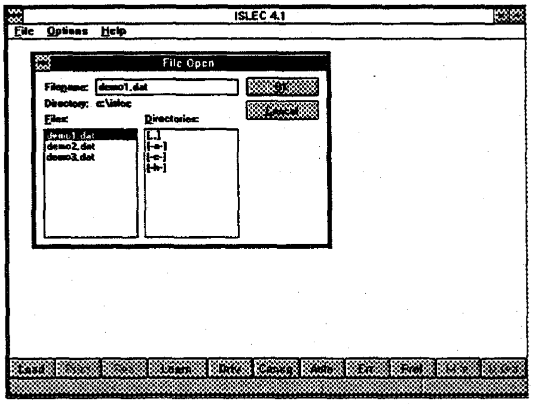

Figure 2 **Loading of Data File**

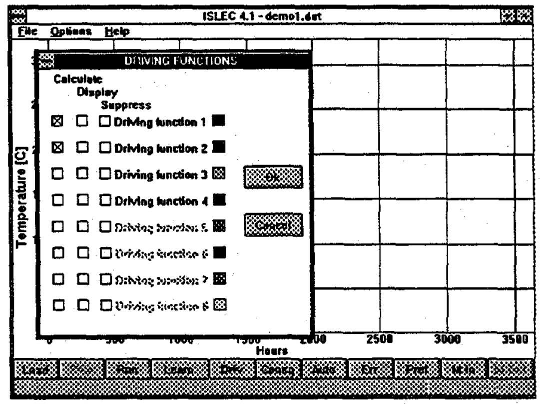

Figure 5 **Activation of the Driving Functions**

variable1	variable2	....	variable9
variable1	variable2	....	variable9
:	:	:	:
variable1	variable2	....	variable9

Figure 3 **Data File Format**

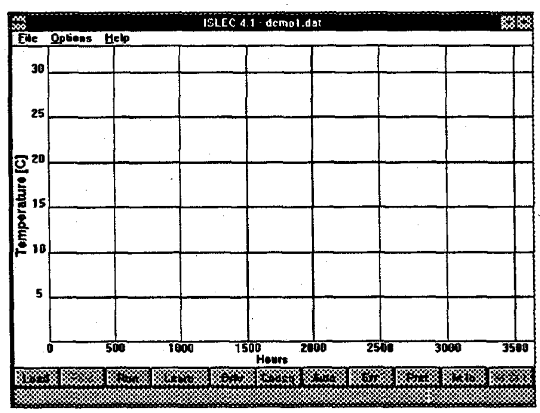

Figure 6 **Display before start of simulation**

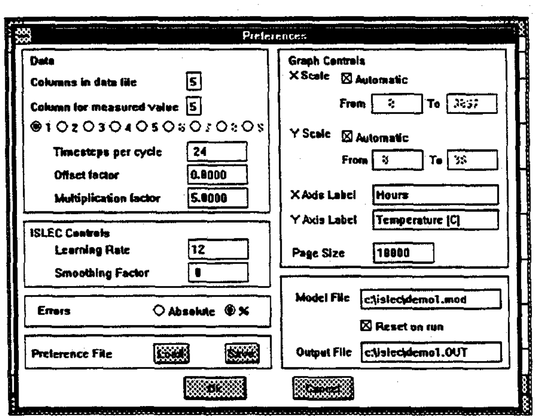

Figure 4 **Preferences**

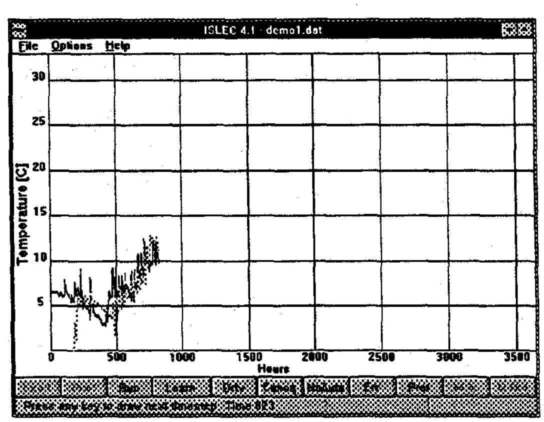

Figure 7 **Simulation after initial learning period**

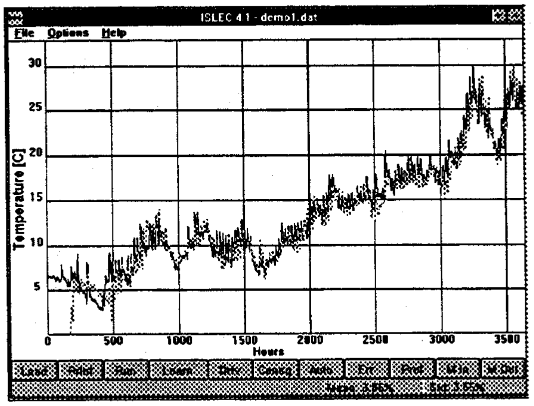

Figure 8 **Simulation in continuous learning mode**

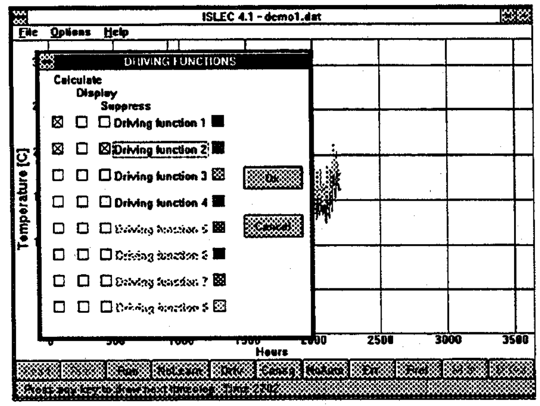

Figure 11 **Suppression of Driving Function**

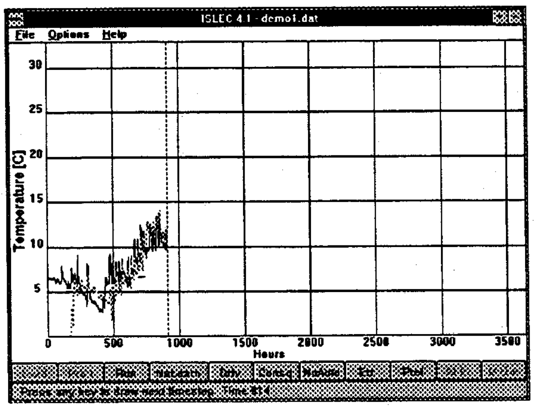

Figure 9 **Vertical line indicates the timestep when learning mode was switched off**

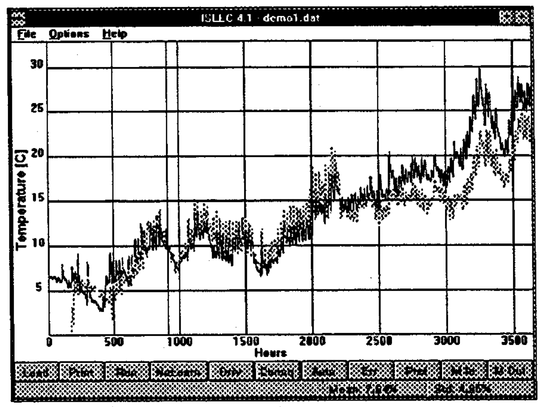

Figure 12 **End of Simulation with one Driving Function Suppressed**

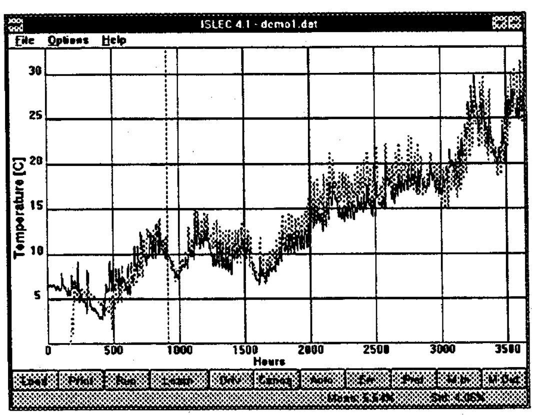

Figure 10 **Simulation in Forecasting Mode**

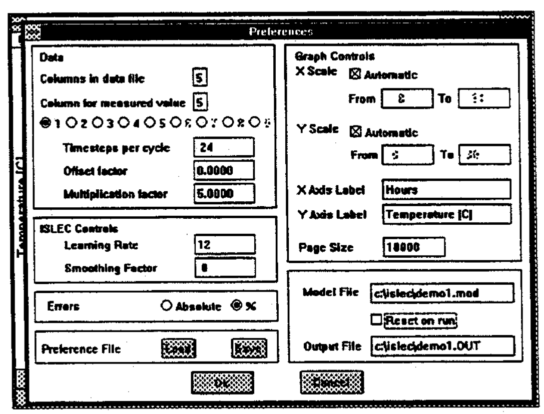

Figure 13 **Disengaging of Reset on Run**

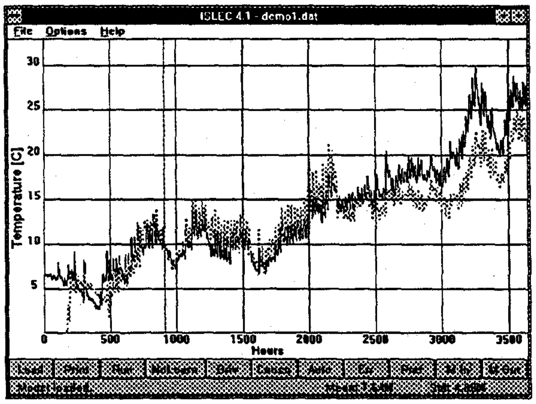

Figure 14 **Loading a Stored Model**

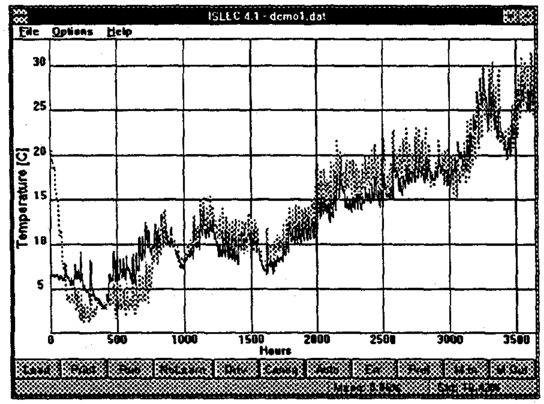

Figure 15 **Simulation with new data file and model loaded from MOD file**