HTD-Vol. 344, National Heat Transfer Conference
Volume 6
ASME 1997

HEAT TRANSFER 1.0
AN EDUCATIONAL SOFTWARE FOR HEAT CONDUCTION TEACHING

**Clovis R. Maliska, Marcus V. Filgueiras dos Reis,
Clovis R. Maliska Jr. and Axel Dihlmann**

Mechanical Engineering Department
Computational Fluid Dynamics Laboratory
Federal University of Santa Catarina
88040-900 - Florianópolis - SC - Brasil
maliska@sinmec.ufsc.br

ABSTRACT

This paper presents an educational software to be used as an auxiliary tool for heat conduction teaching in undergraduate engineering courses. It is an effort in putting together available numerical and computational techniques for helping heat transfer teaching. The main idea is to use the software for stimulating the physical reasoning in heat conduction rather than as a tool for the solution of heat conduction problems. Since the software can be used by the instructor in many different ways, the paper describes selected examples trying to demonstrate how the software can be helpful in motivating the student to explore and understand the physics of heat conduction. The kernel of the software solves two-dimensional transient heat conduction problems with variable thermal properties.

INTRODUCTION

Engineering Education is a field of study which is nowadays receiving an enormous attention. Two key questions are behind this move. First of all, it is the strategy of using the new technologies in computer tools and multimedia for improving learning. The other aspect is to search for teaching methodologies to keep students motivated in their field of study.

The present paper falls in the former category, and aims to offer the students learning tools which stimulates the physical reasoning in heat transfer. The idea is to offer more than a software to help them perform their heat transfer calculations, but a tool which allows to formulate problems, speculate about its physical behavior and explore the change of the physics with the variation of the significant parameters of the problem.

To accomplish this, a package for solving transient two-dimensional heat conduction problems was developed. It is written in C++ object oriented programming running on Windows 95/NT. The interface was constructed using the Borland's OWL (Object Windows Library) library. The equations are solved numerically using a finite volume method with boundary-fitted coordinates, allowing geometries other than the classical Cartesian and cylindrical shapes to be used. The package permits the solution of very interesting problems. Two of those examples are shown later in this paper.

The software is to be used as an auxiliary tool during classroom teaching for the investigation, learning and "visualization" of the heat conduction phenomenon. The purpose of this work is to describe the software and to show its capabilities. Some of the techniques and tools used to create the software may not be known by the students. Therefore, this paper is not a compulsory reading to students that may use the software.

THE MATHEMATICAL MODEL EMBODIED IN THE SOFTWARE

The software is able to solve two-dimensional transient heat conduction problems with heat generation and variable properties. It also allows to solve problems in geometries other than Cartesian and polar. This permits the instructor to go further, if desired, in analyzing more complex heat conduction problems. Additionally, since numerical methods in heat conduction is a topic itself in a heat conduction course, the software can be used for this purpose, exposing the students to a numerical technique, nowadays largely employed for solving engineering problems.

It must be put clear to the students that the software solves a mathematical model given by the following partial differential equation

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right)+q''' = \rho c_p \frac{\partial T}{\partial t} \tag{1}$$

with boundary conditions of prescribed temperature, prescribed heat flux and convection. The above equation is solved using a boundary-fitted finite-volume method, which allows complex geometries to be solved, as shown in Fig. 1, where the discretization used is also shown. The students should recognize that Eq.(1) could be solved using analytical methods. For the purpose of the package, it does not matter how the equation is solved. However, using a numerical method one broads the range of problems to be studied. The numerical method employed is now briefly presented.

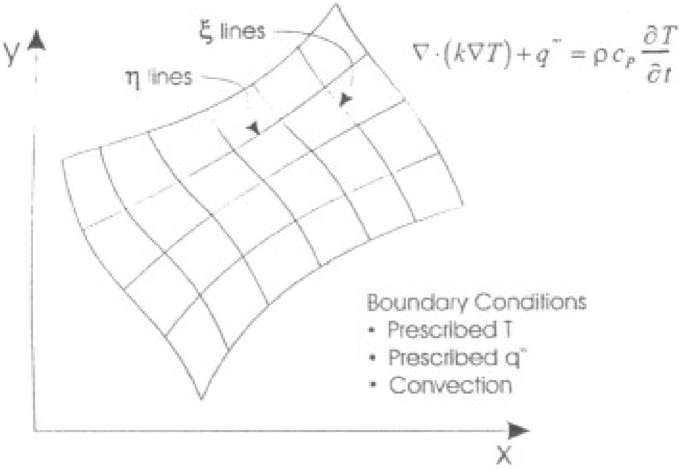


Fig. 1 - Typical geometry used in *Heat Transfer 1.0*

The Numerical Method

The numerical method employed is the usual finite-volume method, whereby the partial differential equation is integrated in each elemental control volume originating an approximate equation for each control volume. It must be clear that the numerical method can be kept "hidden" to the student. However, if desired, more details about it can be found in the help under the title "Basic Numerical Methods.

If one wants to numerically solve heat conduction problems in geometries other than the usual orthogonal ones, it is necessary to discretize the domain in arbitrary control volumes (Thompson, 1986), as shown in Fig. 1, and integrate Eq. (1) over the arbitrary elements in the new system. The integration of the equation, following the basic rules of the numerical methodology, gives a linear systems of the type

$$[A]\,[T] = [B] \tag{2}$$

which can be solved by many existing methods for solving systems of algebraic equations. The solution of Eq. (2) gives the temperature for each control volume. The temperature field can be used in several ways to explain the conduction heat transfer phenomena.

THE SOFTWARE HEAT TRANSFER 1.0

General

The *Heat Transfer 1.0* is an educational software which takes advantage of the modern computational resources to provide an exciting and intelligent learning environment. This section is devoted to explain the functionality of the software. Even considering that *Heat Transfer 1.0* is to be used with the theoretical heat conduction background given by the instructor, it is included in the help the basics of heat conduction with some solved problems. How to use the software is also shown through appropriate interfaces to be described.

The software is based on the construction of an organized structure of the internal data and on its powerful graphical user interface. The data organization and the user interface were based on steps that describe the solution of a heat conduction problem. These steps are the definition of the geometry and boundary conditions, the setting of the numerical parameters, the solution of the problem and the visualization of the results.

Technical Information

The software runs on Windows95/NT, and all its numerical kernel, was written in C++ using the object oriented programming approach (OOP). Its graphical interface was developed using the Borland's OWL (Object Windows Library) library.

Interface Objects Organization

The software takes advantage of the Windows95/NT platform, using the best facilities offered by the system, namely the friendly graphical user interface and its intuitive easy-to-use capability.

The definition of the domain, the setting of the parameters, the simulation (solution of the problem) and the visualization of the results are accessed by menus. The procedures frequently used have a correspondent shortcut on the toolbar (Fig. 2) available in the software main window, shown in Fig. 3, which depicts the graphical interface environment of *Heat Transfer 1.0*. The temperature profile across the domain can be seen in the insert of Fig. 3. The main window also shows the color map used in the visualization (here prepared in black and white) of the temperature fields or any other scalar quantity. It is also seen that it is possible, using a slider, to select plots of temperature along the coordinate lines. Fig 4 shows the structure and all related objects of *Heat Transfer 1.0*

Fig. 2 - *Heat Transfer 1.0 toolbar zoom*

Interface Objects Description

It is now provided a brief explanation of all interface object modules and its potentialities, with the main goal of furnishing a deeper view of the software capabilities.

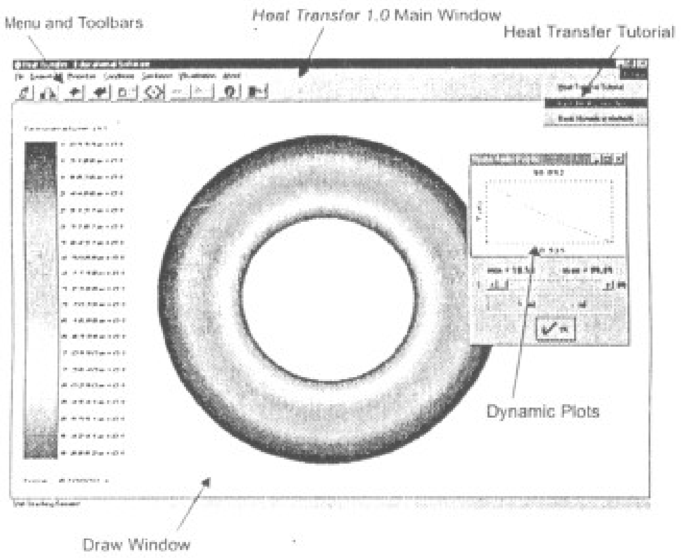


Fig. 3 - *Heat Transfer 1.0 main window*

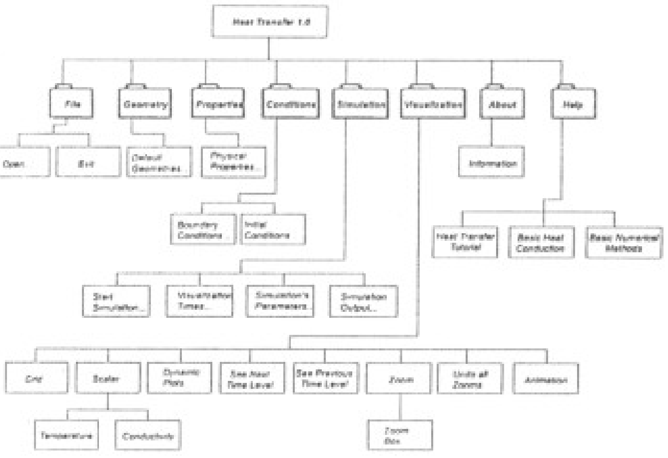


Fig. 4 - Structure of *Heat Transfer 1.0*

File: This module allows the user to import from other programs the geometry and its grid, to be used in *Heat Transfer 1.0*. There is a specified file format to be obeyed.

Geometries: This is a module responsible for the domain definition and its grid. In this module (Default Geometries), the user has the possibility to generate three different types of domains [rectangular, radial, mixed (rectangular + radial)] changing its characteristics dimensions (such as geometrical dimensions and number of volumes). Just by clicking on the desired geometry and giving the parameters which defines it, the user will have its domain created. If other geometries, than the default ones, are required, the user can generate them in any grid generator and employ the module file to use the grid in *Heat Transfer 1.0*

After the definition of the geometry and its discretization (grid), the user needs to define which will be the problem to be solved. This operation is executed by setting the physical properties, the initial and boundary conditions and the simulation parameters. These modules are now presented.

Properties: In this module the user defines the physical properties of the medium, such as thermal conductivity, density, specific heat and sources and sinks of heat.

Variable properties are handled by setting their values for groups of cells. Therefore, heat transfer in heterogeneous domains can be simulated. Adding this to the possibility of prescribing mixed boundary conditions, the *Heat Transfer 1.0* can cover a large class of problems, allowing the student to speculate and check his (her) physical feelings .

Boundary Conditions: In defining boundary conditions, the user can have prescribed temperature (a fixed value or a sinusoidal distribution $T(x) = To + sin(\pi x)/L$), prescribed heat flux or convection. These conditions are applied in each boundary (north, south, east and west), and they can be of different type at the same boundary. For example, the user can prescribe, at the north face, a convection boundary condition mixed with prescribed temperature.

If a transient problem is being solved, there is the need of specifying the initial conditions (initial temperature field). If the interest is only the steady state solution, the initial conditions works as initial guess for the iterative procedure.

Simulation: After the definition of the boundary conditions and of the physical properties of the problem, the user has the possibility of manipulating the numerical parameters of the simulator embodied in *Heat Transfer 1.0*. Among them the user can modify the maximum number of iterations (in space and time), the value of the maximum error allowed in the solution for each time level, and the criterion for defining when the steady state has been reached. The user can also define the time levels he wants to have the results printed or visualized. During the simulation the software gives information about errors, iteration number (in space and time) and other details of the solution.

Visualization: A good visualization helps the student to have an overview of the problem solution. This module deals specifically with these post-processing operations.

This is done using available visualization tools which consist in the presentation of a color interpolation of the temperature (or other scalar, like thermal conductivity) field. In addition, the software also offers a visualization of the results in dynamic plots of the temperature profiles along the coordinate lines. It is possible to visualize the results in any specified time and to compose the color fields, or the plots, for having an animation of the transient process. Just by clicking *Animation* in the menu, all the color interpolated temperature fields are shown, allowing the student to "see" the movement of the heat front. The available zoom allows a more detailed analysis of the temperature distribution in selected parts of the domain.

About: This module brings specific information about the program, such as the developer team, institution and current version.

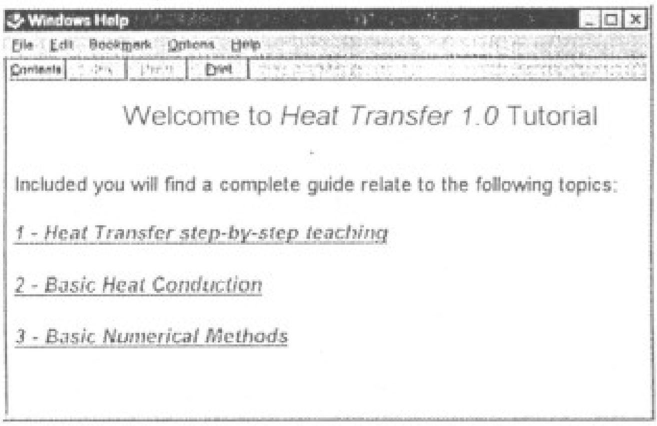
Tutorial: Finally, the software contains a help guide, the *Heat Transfer 1.0* Tutorial,, divided in three parts, as shown in Fig. 5.

***Heat Transfer 1.0* step-by-step teaching,** contains all basic information about the use of the software.

Basic Heat Conduction, contains information about the heat conduction phenomena and a large group of examples which can be solved analytically, to be used for comparing with the results obtained by the software. It contains also some suggested problems for analysis.

Basic Numerical Methods, contains basic notions about the numerical methods used in the solution of the heat conduction problems, emphasizing the finite volume methodology and the other techniques used in implementing the program. This part is used only if the student wants to know how the conduction problem is solved.

The help guide was built using the Windows Help Format, which allows the user to navigate through it and to use it simultaneously to the program.



Welcome to *Heat Transfer 1.0* Tutorial

Included you will find a complete guide relate to the following topics:

1 - Heat Transfer step-by-step teaching

2 - Basic Heat Conduction

3 - Basic Numerical Methods

Fig. 5 - Windows help window

Internal Objects Description

As previously mentioned, all the software internal data was organized by using the philosophy of the object oriented programming with the C++ language. The main components that constitute the numerical simulator, as well as all components of the visual interface, were separated into functions, creating a set of objects which, at the end, creates a library. Figure 6 illustrates the internal objects organization used in the construction of the algorithm.

EXAMPLES ON HOW TO USE THE SOFTWARE

The software herein described can be used by the instructor according to his (her) course program. The following examples just aim to illustrate how it can be used allowing the students to take active role when studying important topics of the heat transfer course program.

Thermal resistance calculations - Understanding the limits of the electrical circuit analogy

Let us suppose that the instructor is now teaching the students the flow of heat in a 1D situation, no heat generation, constant thermal conductivity and steady-state. In this case the electrical resistance analogy applies. It is always stressed to the students that this analogy applies ONLY TO ONE-DIMENSIONAL SITUATIONS. Following, the students learn that they can apply the electrical circuit analogy to composite walls, if the flow of heat is 1D, calculating the equivalent resistance of the composite wall, as shown in Fig. 7. For this situation it is easy to recognize that the problem is 1D.

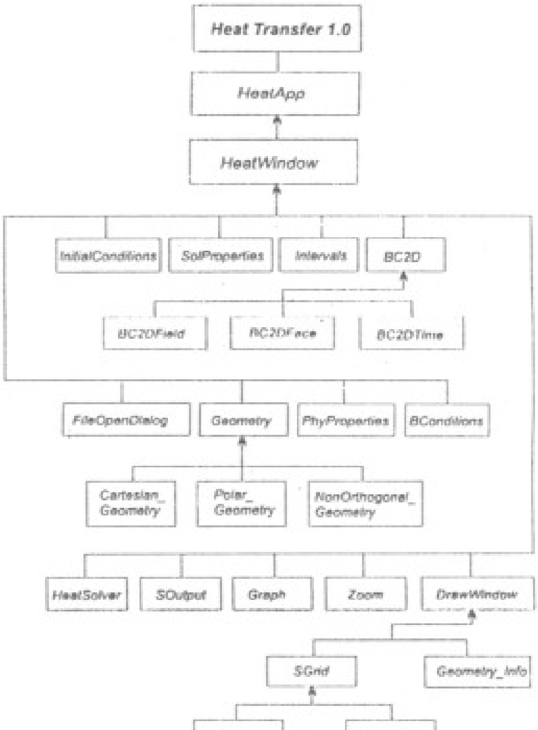


Fig. 6 - *Heat Transfer 1.0* internal objects organization

An important question appears when it is told to the students that they can apply the electrical analogy for situations where different materials are associated in parallel, as shown in Fig. 8. In this case, two different circuits can be used, denoted by the letters (a) and (b).

At this point it is interesting to show the students the reason why the 1D characteristic of the heat flow may be lost when the association is in parallel. To reach this goal, the problem shown in Fig. 9 (the physical model of the problem depicted in Fig. 8) is solved for different values of k_2 and k_3, pointing out that the ratio between k_2 and k_3 dictates how the flow of heat deviates from the 1D situation. The student must realize that when k_2 is greater than k_3, the heat "finds easier" to reach the outer wall going through material 2, since it offers less thermal resistance.

When k_2 is different from k_3 the problem is two-dimensional and can be solved using the software. At this point the instructor must warns the students that sufficient grid resolution and appropriate convergence criterion needs to be applied to the numerical solution, in order to have a good solution of the mathematical model given by Eq. (1).

Considering that the numerical solution obtained with the software *Heat Transfer 1.0* is an "exact" solution to the 2D problem, one can compare this solution with the 1D solutions obtained using the two electrical analogy arrangements. Fig. 10 shows the error when the 1D model (electrical resistance analogy) is used with the two procedures. The student can realize that for k_2 equal to k_3 the error is zero, since the problem reduces to a 1D problem. It must be remembered that the 2D problem, in this paper, was not solved using the dimensionless parameters of the problem and, therefore, the error shown in Fig. 10 can not be applied to others parameter combination.

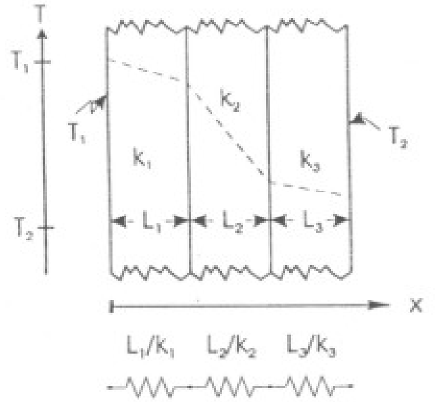


Fig. 7 - Composite wall. Series circuit analogy

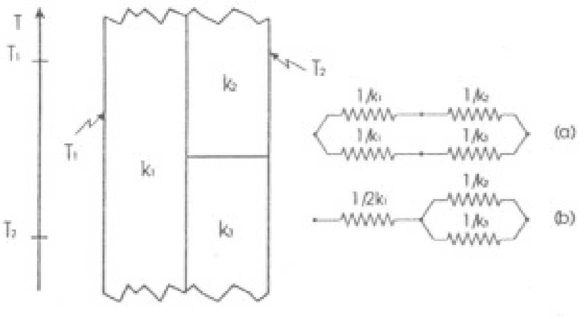


Fig. 8 - Composite wall. Parallel circuit analogy

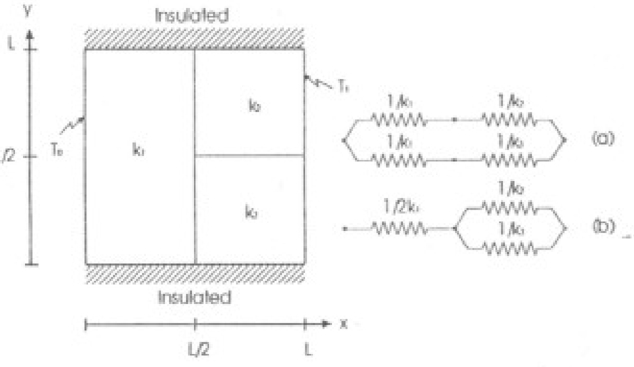


Fig. 9 - 2D problem when K_2 is different from K_3

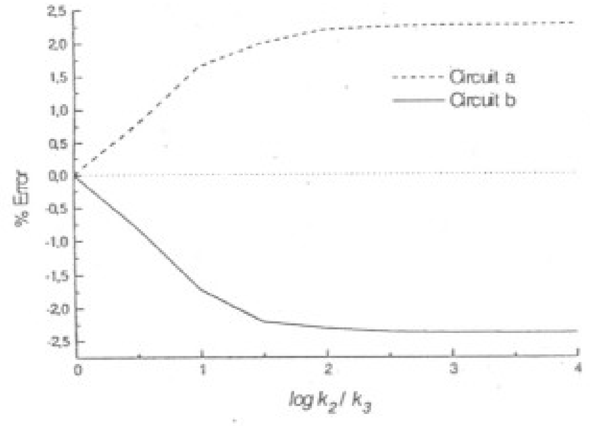


Fig. 10 - Error when using 1D model to calculate the thermal resistance

It can be seen also that the "exact" result lies between the two 1D results, one of them overestimating and the other one underestimating the heat flux. The percentage error in the heat flux is calculated comparing the results of the 1D model with the "exact".

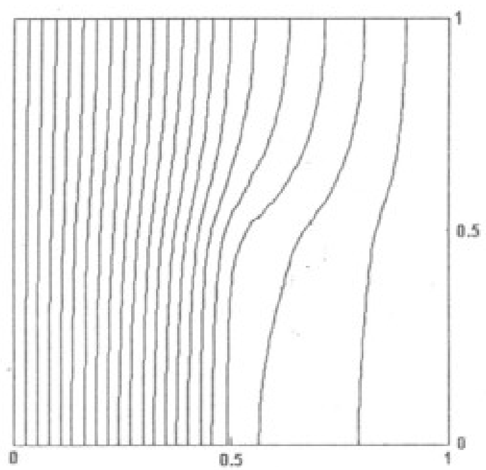


Fig. 11 - Isotherms of the 2D problem

Finally, the student can appreciate in Fig. 11 how the isotherms deviate from its 1D situation. This picture clearly demonstrate the 2D region caused by the difference in k_2 and k_3. The student now can change the dimensions of the domain, for example making it taller, and appreciating that the 2D region will become less important in the whole domain.

It is expected that analyses of this type, that can be done in the classroom by the teacher, using a computer and a data show, will promote a better understanding of the phenomena and put clear the limitations of using a 1D electrical analogy.

Transient flow in a Slab - Understanding the meaning of the Lumped Capacity analysis an the Bi less than 0.1 limit

When studying transient heat conduction, the lumped capacity analysis plays an important role, since many engineering problems fits in this category, specially when dealing with temperature sensors and heating and cooling of small pieces.

To illustrate this concept, Fig. 12a depicts a transient problem where the temperature changes only with time, while in Fig. 12b the temperature depends on time and on the spatial coordinate. The problem sketched in Fig. 12a. is said to fit in the lumped capacity analysis because there is no "significant" spatial temperature variations. The key point here is in fact to define what "significant" means. Therefore, it must be clear to the student that the use of the lumped capacity analysis depends on the required accuracy of the solution, and there is no an absolute boundary defining when the problem can be attacked using this model. It must be recognized that the classical Bi = hL/k less than 0.1 limit comes from the analysis of a 1D transient flow in a slab where it is accepted that the changes in spatial temperature are up to 5% (Incropera and De Witt, 1990). It is possible that in some other engineering application the admissible variation be only 2.5%, what will require a Biot number limit even less than 0.1.

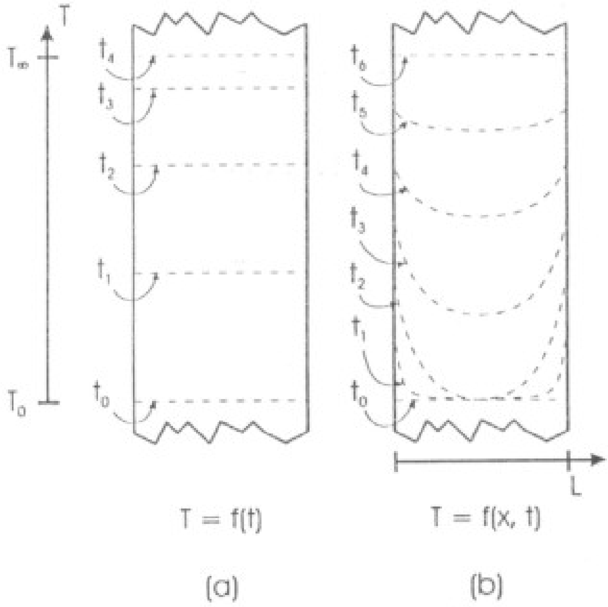


Fig. 12 - 1D transient problem.
Lumped analysis (a) and spatial effects (b)

The student can exercise this point solving a 1D transient problem in a slab, using the *Heat Transfer 1.0* software, defining the acceptable spatial temperature variation according to

$$\frac{T_f - T_\infty}{T_c - T_\infty} \tag{3}$$

where T_f and T_c are the surface and center temperature of the slab, respectively, as shown in Fig. 13.

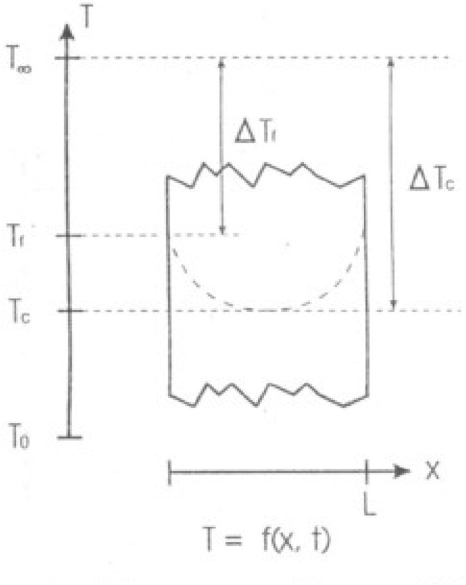
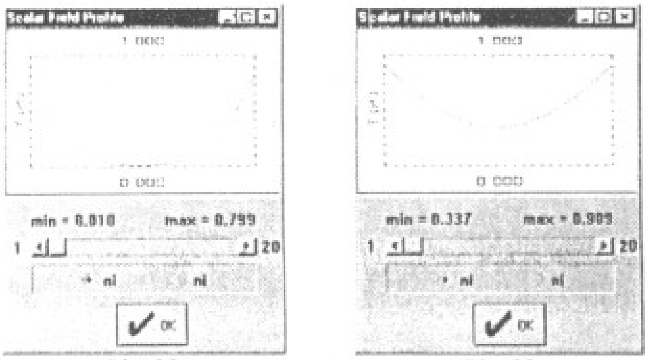


Fig. 13 - Spatial temperature difference definition

Using the software *Heat Transfer 1.0*, the student can solve the transient heat conduction problem in a slab suddenly put in contact with an ambient at T_∞ with a heat convection coefficient h.

Figures 14, 15 and 16 shows the result of the problem for Biot numbers of 25, 1 and 0.01, for three different times. It can be seen in this figure that for Biot equal to 25 one has considerable spatial variation of the temperature. In the other hand, for Biot number equal to 0.01, the temperature variation is only with time. For Biot number equal to 1 one has an intermediate situation. Therefore, using the temperature plots (the values of the temperature can be printed) the student can, based on a chosen spatial temperature difference, to find which Biot number will satisfy the admissible temperature variation inside the domain. They (the students) can also check what is the spatial temperature difference resulting of using the classical Biot = 0.1.

In addition, with the results of this problem the heat front moving from the surface to the center of the slab can be "seen" using the visualization facilities. Also the spatial temperature plots, for each time, can be composed and the animation of the transient can be performed.

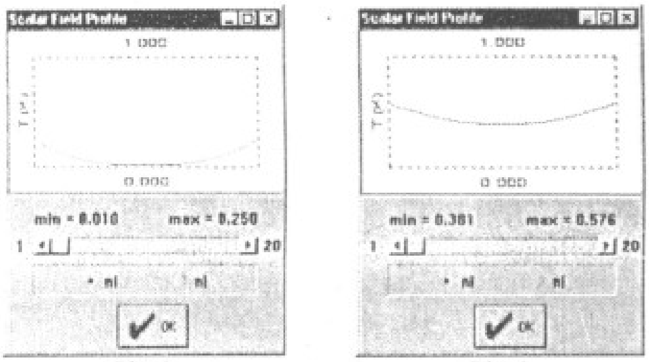


min = 0.010 max = 0.799

min = 0.337 max = 0.909

Bi = 25 Bi = 25

Fig. 14 - Transient heating of a slab for Biot equal to 25

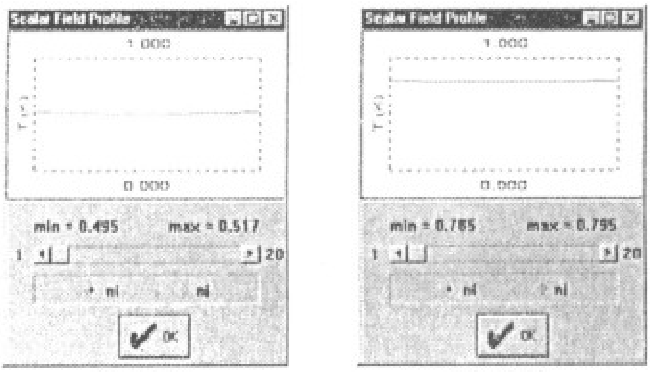


min = 0.010 max = 0.250

min = 0.381 max = 0.576

Bi = 1 Bi = 1

Fig. 15 - Transient heating of a slab for Biot equal to 1



min = 0.495 max = 0.517

min = 0.785 max = 0.795

Bi = 0.01 Bi = 0.01

Fig. 16 - Transient heating of a slab for Biot equal to 0.01

CONCLUSIONS

This work presented the software *Heat Transfer 1.0*, an auxiliary tool for heat conduction teaching. The software is highly versatile, with powerful graphical user interface and visualization facilities. It can be used during classroom, solving well selected problems, to help students to deeper their understanding about the physics of heat conduction. It can be also used at home by the students working in their assignments and, associated to a heat conduction text, it is an strong tool for individual heat conduction studies.

Due to its user-friendly interface, it requires little effort to run the software. This is of utmost importance when it is needed to introduce the software to a large group of students. The software is now being used in a undergraduate heat conduction course with 3 hours class per week in a mechanical engineering course.

References

Incropera, F.P. and De Witt, D.P., *Introduction to Heat Transfer*, John Wiley and Sons, Inc., 1990.

Thompson, J.F., *Numerical Grid Generation* , Elsevier Publishing Corporation, 1986