

Unusual Techniques Employed in Heat Transfer Programs

D. J. CAMPBELL† AND D. B. VOLLENWEIDER‡

A PROGRAM for the IBM 704 to solve general transient and steady state heat transfer problems is described. The computer program was developed by Evendale Computations Operation in cooperation with the Jet Engine Department, General Electric Company, Flight Propulsion Division. Jet Engine Department personnel who were instrumental in the formulation of the problem are Mr. William K. Koffel and Dr. J. M. Botje. Dr. James T. Anderson, of Michigan State University, acted as a consultant and played a central role in the development of the program. The method of solution permits the analysis of problems with arbitrary geometry and several combined modes of heat transfer. To facilitate the description and solution of extensive and complicated problems, many logical and computational techniques not usually applied to engineering calculations are incorporated in the program.

The program yields the temperature distribution at a maximum of 200 points in three-dimensions for homogeneous or composite bodies. Heat transfer by conduction, convection, surface flux, thermal radiation, both solid and gaseous, and internal heat generation is treated. One dimensional fluid flow in several channels can be incorporated. The thermal properties of the system may vary with temperature. Boundary conditions and heat transfer rates for convective exchanges as well as surface flux rates and mass flow rates may vary with time. Internal heat generation is given either as a function of time or of temperature.

THERMAL STORAGE	NON-SOURCE, SINK FLUID HEAT TRANSFER	CONDUCTION & CONVECTION	INTERNAL HEAT GENERATION
$\frac{C\rho V}{\Delta\tau} (T_0 - T_0^0) = cW(T_f - T_0) + \sum_{n=1}^6 \left[\frac{A_n(T_n - T_0)}{\frac{\Delta X_n}{K} + \frac{1}{h_c}} \right] + Q_0 V$			
SURFACE FLUX	SOLID RADIATION	GASEOUS RADIATION	
$+ A_s Q_f + \sum_{i=1}^3 \sum_{j=1}^3 \left[\sigma A_i F_{ij} (T_h^4 - T_0^4) \right] + \sum_{j=1}^3 \left[\sigma A_j G_j (\epsilon_g T_g^4 - \alpha_g T_0^4) \right]$			

Fig. 1—The general heat balance equation. T_0 indicates the temperature at the beginning of a time interval. Linearized versions of the radiation terms are actually programmed.

The method of solution for the temperature distribution is to divide the geometry into cells enclosing nodes and to write heat balance equations (in finite difference form) for each node. The general heat bal-

† Flight Propulsion Laboratory Department, General Electric Company, Cincinnati, Ohio.

ance equation that is written for each node is given in Fig. 1. Note that T' indicates temperatures at the beginning of a time interval. The radiation terms are linearized by writing them as a radiation coefficient times a first power temperature difference. The radiation coefficient is computed using the temperatures at the beginning of the time interval, while the linear temperature difference uses a given radiation source temperature. The implicit form of heat balance equation is used in preference to the explicit form, because the stability requirement for the explicit form places a limit on the time increment which is restrictive for large systems. The implicit form permits more freedom in the choice of time step. The accelerated Gauss-Seidel method is used to solve the system of heat balance equations for the n unknown temperatures. Gauss-Seidel is used in preference to other methods because the convergence criterion for the method is readily met for most applications and because there are a large number of zero elements in the coefficient matrix.

One of the most interesting features of the program is the method used for describing the geometry. Rectangular coordinate systems are not adequate to fully describe the variety and intricacy of shapes commonly encountered in applications of the program.

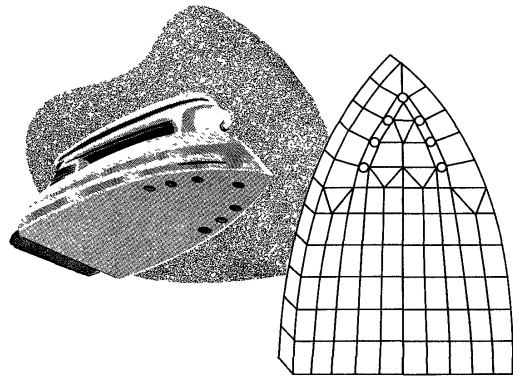


Fig. 2—As an example of the method of dividing an object into cells, the plate of a steam iron is shown. Note the irregular manner in which the cell division has been made.

Instead, the geometry is divided into cells of any shape or size, and the cells are given numbers to identify them. Fig. 2 shows how the plate of a steam iron might be divided into cells. At most six faces are defined for each cell, and the faces are labeled by numbers from one to six. Fig. 3 illustrates the numbering of faces of a cell. In order to write heat balance equations, it is necessary at each cell to give

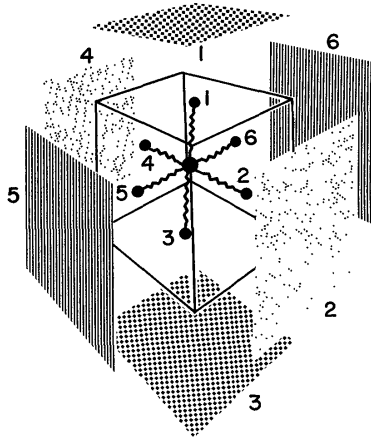


Fig. 3—Illustration of the convention for numbering the faces of a cell.

the adjacent cell, if any, for every face through which there is conduction. In the program, this is done by giving the number of the cell adjacent to each face. A convention with respect to adjoining cells makes it unnecessary to give the number of the face of the adjacent cell. The face common to two adjacent cells is, of course, assigned a number in each of the cells. As can be seen from Fig. 4, if a face is labeled "1" for one of the cells, then it must be labeled "3" for the other cell. Similarly, if the face is labeled "2" in one cell, it must be labeled "4" in the other, and if it is labeled face "5" in one cell, it must be labeled face "6" of the other.

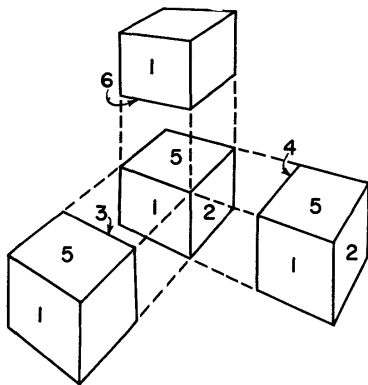


Fig. 4—Illustration of the face numbering convention for adjacent cells. The numbers assigned to a face of adjacent cells must be one of the pairs (1, 3), (2, 4) or (5, 6).

Although this method of describing geometry may seem awkward at first glance, it conveniently accommodates unusual patterns of cells and irregular surfaces. Fig. 5 shows some unusual shapes that may be accommodated with this method of describing geometry. It can be seen that cells need not have six faces and can be highly irregular in shape. Any face not needed for conduction exchanges may be used to accommodate special effects. For example, in Fig. 6, the two narrow cells exploded out of posi-

tion are used to increase the number of faces for the cell that occupies the position in the center of the drawing. Since the cells are assumed to have zero volume and zero thickness, there are no thermal storage terms and no thermal gradients across them. Therefore, they will have no effect on the heat balance except to balance the cell in the center against all of the surrounding cells. These nodes increase the effective number of faces of the central node from six to eleven.

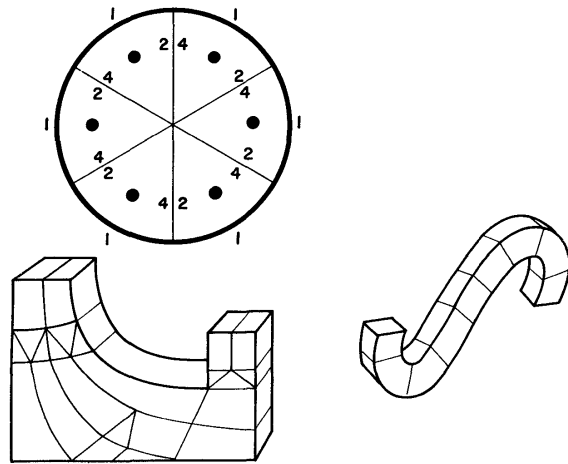


Fig. 5—Examples of irregular shapes that have been divided into cells.

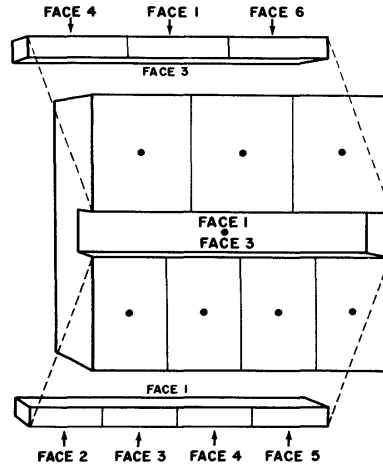


Fig. 6—Illustration of a method for increasing the effective number of faces of a cell. By including two 'zero volume' cells, the number of faces of the long, innermost cell is increased to eleven.

A symbol defining a table of thermal properties is given for each cell in the system. These properties are tabulated as functions of temperature. Given the geometry description and the material symbol just mentioned, sufficient information is available to assemble the conductive heat transfer equations. In describing the dimensions of a cell, a distinction is made between "regular" and "irregular" cells. Regular cells are rectangular parallelepipeds, or linear approximations thereof, and only height, width, and

depth of each cell are required as input. For irregular cells, distances from each face to the node point, face areas, and the cell volume must be specified. Physical parameters and boundary conditions are given as functions of time or of temperature. For example, tables of adiabatic wall temperature and fluid film coefficient are given, and the collection of tables is assigned a symbol. The symbol defining the applicable tables of functions is entered for the proper face of the cell which has a convective heat exchange. An analogous method is used to indicate other modes of heat transfer. That is, the functions are defined and identified by a symbol, and the symbols are given at each appropriate cell. Consequently, there are two basic types of information for a cell. One gives the dimensions of the cell and other geometric constants such as configuration factors for thermal radiation. Secondly, symbols are given to define the surrounding geometry, to indicate the modes of heat transfer, and to define the tables of values of the various parameters.

From the symbols describing the cell configuration and corresponding properties, the heat balance equations may be assembled for each time step. However, an assembly directly from symbols would require a large number of sorting procedures on the basis of node numbers and table symbols. To avoid searching for information designated by symbols more than once, a simple device is utilized in the program. Each symbol is replaced by the machine address of the initial memory location of the information defined by the symbol at the beginning of the program, before any temperature calculations are begun. The searches that are undertaken automatically check for completeness. If a table or node has not been given, the search for the location of that table must fail. When such a failure occurs, a comment is printed giving the symbol of the missing table or node.

COL.		10	13	16	19	22	25	28	31	34	37	40	43	46	49
		1n.c	1bc	3n.c	3bc	2n.c	2bc	4n.c	4bc	5n.c	5bc	6n.c	6bc	MAT	B.Cd
CELL NO. 9		101		2		13		503			A	202		CM	
		3	BC							16				DM	
					AA					31		66		Q	
CELL NO. 13		4	BD	36		102		9		63		7		DM	R

Fig. 7—An example of data describing the geometry of a problem. The connection between cells number 9 and 13 is illustrated to indicate the need for checking its consistency.

Further examination of the form of the input indicates that testing the cell connection information is highly desirable. The method of describing geometry requires that connection information for adjoining cells must be given for the designated face of each cell. This may be seen by examining Fig. 7, a facsimile of the cell information input form. The entries necessary to define a connection between cells 9 and

13 are indicated. If the connection is not indicated for both cells, erroneous heat balance equations will be generated, and the equations may yield results that appear correct but are in error in the vicinity of the inconsistency. Moreover, the volume of data required for large problems encourages error in input. Consequently, the program checks the consistency of cell connections and makes many other tests for completeness of input data to recognize and identify errors before the equations are assembled. The machine time required for these completeness and consistency tests is negligible when compared with overall computing time.

It can be seen that the method for describing arbitrary cell configurations does necessitate considerable pre-processing, searching, and testing of input data. If the geometry were described by the usual mesh or lattice point scheme, input for the network could be organized by three-dimensional arrays and could be assigned to ordered blocks of machine storage. There would be a simple arithmetic relationship between nodes and the corresponding memory location of information pertaining to nodes. However, if it is necessary to describe and analyze arbitrary patterns of cells enclosing node points, a simple arithmetic formula for computing the address of the machine location of information for another node from the address of a given node cannot in general be established.

Having presented some of the interesting aspects of the program, we would like to describe possible extensions of the techniques and methods employed. In the present program, input data is stored in fixed sections of memory. Consequently, there are specified limits for the maximum number of entries in a table, the number of tables of a given variety, and the maximum number of cells in the division of the problem. These restrictions do not allow the available memory space to be used efficiently for many problems. In a subsequent program, similar input is stored in consecutive memory locations which are computed as the data are read into memory. Since sections of data are not assigned fixed storage locations, an economical means for finding specified quantities is necessary to accomplish the processing and testing of input data. Fig. 8 illustrates this memory packing system. Each table begins with two symbols that are shown shaded in the figure. The first symbol identifies the table and defines its type. The second symbol indicates the length of the table. The next table begins in the first unused memory location after the preceding table. Thus, for each table there is information which can be used to locate the next table. By allocating storage as the input data are read into the machine, considerable flexibility in the type of problem acceptable to the program is achieved. There are no pre-assigned limits on the maximum number of such quantities as nodes

or physical parameters. Only the total amount of storage available for data limits the size and type of problem that can be handled by the program. An additional advantage to flexible input storage assignment is that the program can be run on a 704 with any memory capacity. In one program this idea was extended further by allowing tables of a particular type to have one of several different formats of varying length. In this case, the next table could begin at any one of several alternate locations. To determine the location of the next table, the symbols are defined in such a manner that the binary representation of these symbols are distinct from ordinary floating-point numbers. Thus, the table following a given table is found by searching each of several alternate locations for a word not in normal floating-point format, that is, for the symbol defining the new table.

MEMORY PACKING SYSTEM FOR FUNCTION TABLES

BCC	A	3					
BCC	B	4					
BCC	F	6					
CH	2	3					
CH	2	2					

Fig. 8—Illustration of memory packing system. The shaded areas indicate symbols defining the type and length of a table. Each table begins immediately after the preceding table.

Another extension of the transient heat transfer program is to assign an order to the operation on cells. At present, the heat balance equations are assembled and solved in an arbitrary order. Certain applications of the program to large classes of less general problems suggest that methods be incorporated for computing quantities required as input to the present program. A body with two horizontal fluid passages is illustrated in Fig. 9. In some cases it is desirable to make calculations in the order indicated by the arrows, regardless of the order of the data within the computer. For example, the computation of weight flow and fluid film coefficients from given inlet and outlet pressure for one-dimensional fluid channels could be programmed. If the computer were programmed to operate on the cells in each channel in the direction of flow, calculation of such quantities as fluid film coefficient could be incorporated easily. A straight-forward method for constructing an order of operation on node information is to recognize the desired order from the symbols describing the problem and to store with information

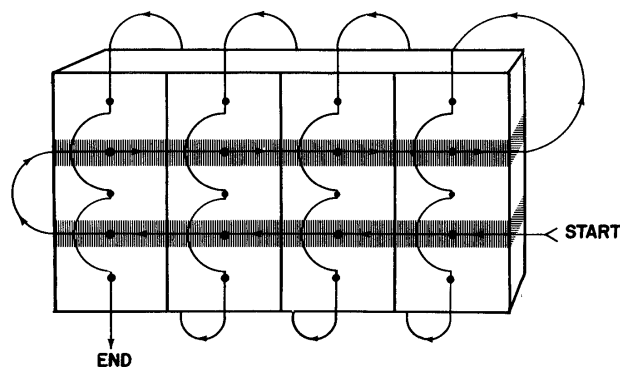


Fig. 9—Illustration of assigning an order to program operation on cells to accommodate specialized calculations or to increase the rate of convergence.

for one cell the address of the location of the next cell to be treated. In certain applications of the program, assigning an order to the operation on cells can facilitate the solution of the system of equations. If an order of rows of the matrix is known to yield faster solution, this order can be used in applying the Gauss-Seidel method. Moreover, ordering program operation on cells provides a simple means for utilizing the symmetry of many elements of the matrix. The conduction-convection terms between two connected cells are symmetrical, and considerable computing time is saved if these terms are computed once for each time step. One means for utilizing the symmetry of coefficients is to order the assembly of elements by increasing or decreasing machine address.

In conclusion, several interesting methods and computational techniques are incorporated in the transient heat transfer program to allow its application to a wide range of problems. The program has been used to analyze large problems incorporating unusual shapes and several combined modes of heat transfer. The machine time and consequent cost required for the solution of most problems is very reasonable. For example, typical problems with 150 nodes and 100 transient time steps are run in about ten minutes of 704 computer time. It is economical to use the heat transfer program to solve extensive problems and to obtain solutions that are considerably more adequate than those resulting from other methods. Consequently, extensions and further application of the ideas used in the present program are planned for the solution of other engineering problems.

DISCUSSION

J. Ricketts (AC Spark Plug): Where can detailed information concerning the heat transfer program be requested?

Mr. Campbell: Information can be obtained by writing to me. However, I am afraid that you cannot obtain the program, because a large part of the information is company priority. I would be glad to discuss this with you or tell you as much as I can, but unfortunately the usual restrictions are hampering this exchange of ideas.

By the way, let me introduce Mrs. Vollenweider who is an expert on a different part of the programs and is here to answer the questions that I can't.

Mr. Ricketts: How long would a large problem take?

Mrs. Vollenweider: Large problems have been worked in approximately three minutes of computing time. One of the assets of the program is its short running time.

L. Hellerman (IBM): Breaking a solid into cells suggests G. Kron's "tearing". Could you comment on this?

Mr. Campbell: Neither of us know enough about this.

Mr. Hellerman: Has there been any thought of using your method of geometric specification in the control of machine tools?

Mrs. Vollenweider: Yes, and we are investigating this now.

H. T. Middleton (IBM): Will this program be available through

SHARE?

Mrs. Vollenweider: Information on the solution methods we used is available. However, the actual program is not available except to other GE installations.

F. Engel (Westinghouse): What assumptions are made in arriving at the difference approximations for the conduction terms considering the irregular geometry permitted?

Mr. Campbell: The assumptions that are made in this case are, of course, the assumptions one usually makes for a finite different mesh. I believe it is clear that if you divide an object into cells in the same manner that a flux plot would break such an object into cells you have a better chance of getting reasonable answers. Our method does allow you to do this if you are so inclined.