CR-152048

Department of Aerospace Engineering

University of Cincinnati

(NASA-CR-152048) AN INVESTIGATION OF TEMPERATURE DISTRIBUTION IN COOLED GUIDE VANES (Cincinnati Univ.) 143 p HC A07/MF A01 CSCL 21E

N78-10099

Unclas

G3/07 52522

AN INVESTIGATION OF TEMPERATURE DISTRIBUTION IN COOLED GUIDE VANES

BY

R. KOTWAL, W. TABAKOFF AND A. HAMED

AN INVESTIGATION OF TEMPERATURE DISTRIBUTION IN COOLED GUIDE VANES

by

R. Kotwal, W. Tabakoff and A. Hamed

NAS 2-7850

1 October 1977

TABLE OF CONTENTS

SUMMARY

A numerical study to determine the temperature distribution in the guide vane blades of a radial inflow turbine is presented. A computer program has been developed which permits the temperature distribution to be calculated when this blade is cooled internally using a combination of impingement and film cooling techniques. The study is based on the use of the finite difference method in a two dimensional heat conduction problem. The results are then compared to determine the best cooling configuration for a certain coolant to primary mass flow ratio.

INTRODUCTION

The power output of a gas turbine is directly proportional to its inlet gas temperature. The inlet temperature of a gas turbine is controlled however by metallurgical limitations. Hence, effective cooling of the turbine blades would considerably improve the power output by allowing higher inlet gas temperatures. Thermal stresses can be minimized by reducing temperature gradients in the blade. This can be accomplished by matching the coolant distribution and internal heat transfer coefficients with the external heat flux distribution to achieve the desired temperature distribution. The cooling of a turbine blade therefore serves two purposes:

(a) Reducing the temperature level in the blade material.

(b) Equalizing the temperature differences in the blade section.

There have been several experimental and theoretical studies dealing with turbine cooling. A good discussion of the different cooling techniques, namely convection, transpiration and film cooling is given in Reference [1]. References [2,3,4,5] present a review of the present state of the art for the internal cooling of turbine nozzles in aircraft applications. Okapuu and Calvert [3] have presented design features of the turbine and the results of thermal and stress analyses of its components. Kuhl [6] reported temperature measurements obtained within the cooled and uncooled turbine blades. Hamed, Baskharone and Tabakoff [7] investigated the temperature distribution in the rotor of a radial inflow turbine. Tabakoff and Clevenger [8] carried out an experimental investigation of heat transfer characteristics for various configurations of air jets impinging on a concave surface representing the leading edge of a cooled blade. Metzger, Yamashita and Jenkins [9] have experimentally investigated the heat transfer characteristics between single lines of circular jets and concave cylindrical surfaces. Kercher and Tabakoff [10] studied the heat transfer by a square array of round jets impinging perpendicular to a flat surface. Chupp, Helms, McFadden and Brown [11] were concerned with impinging characteristics of a concave surface sized to match the leading edge of a typical turbine blade.

Experimental development work for blade cooling techniques is very expensive. It also does not provide a deeper insight into the local physical interrelations owing to the complex measuring techniques involved. Computational methods can therefore be developed to provide valuable information about the temperature distributions resulting from the different cooling techniques. In the present study a numerical method is developed to determine the temperature field in a radial turbine guide vane blade. This analysis is based on the finite difference method in the two dimensional conduction problem. A computer program has been developed for calculating the temperature distribution throughout the blade section. Some of the available experimental test results are drawn on to back up the theoretical procedure.

The blade profile used in the present investigation is shown in Figure 1. The computations are carried out to determine the resulting temperature distribution when this blade is cooled internally using a combination of impingement and film cooling techniques. The impingement tube configuration and the position of coolant discharge slot were varied to study their effects on the blade temperature distribution. The resulting cooling effectiveness and temperature distributions are reported.

ANALYSIS

A radial turbine guide vane blade does not usually experience high axial temperature gradients at its inlet. The heat conducted to the two guide vane back plates is also small. The use of a two dimensional solution to determine the blade temperature distribution is therefore justified.

Governing Equations

Under the steady state conditions and assuming that the conductivity of the material is constant, the differential equation governing the temperature distribution in the turbine blade is given by:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \qquad (1)$$

There are three different types of boundary conditions which may be assigned to the different blade surfaces. They are:

i) The boundary conditions on the gas side surface at points far from being affected by coolant discharge.

ii) The boundary conditions on the gas side surface at points downstream of the slot.

iii) The boundary conditions on the coolant side surface.

The general convective boundary condition is written as follows:

$$k \frac{\partial T}{\partial n} = h_e(T_e - T_s) \qquad (2)$$

where k is the thermal conductivity of the blade material.

$\frac{\partial T}{\partial n}$ is the temperature gradient normal to the surface.

h_e represents the local convective heat transfer coefficient evaluated at the reference stream temperature, T_e.

T_s is the local temperature on the blade surface.

The value of the reference stream temperature T_e, depends upon the type of the boundary point. In the case of a point on the hot gas surface which is not affected by the coolant discharge, the reference stream temperature takes the value of the free stream gas temperature, T_g. If the point happens to be on the coolant surface, then T_e assumes the value of the temperature of the coolant, T_c. On the other hand, if the point is downstream of the point of coolant discharge, the value assigned to T_e is the adiabatic wall temperature, T_{aw}.

Finite Difference Equations

In this section, the differential equation (1) and the boundary condition, equation (2), are recast into a finite difference form which will be solved using an iterative, successive over relaxation technique. Referring to Figure 2, for the mesh point $P(i,j)$, its neighboring points A and C are situated at ξ_1 and ξ_2 fractions of DX and points B and D are situated at δ_1 and δ_2 fractions of DY. The following finite-difference equation is obtained by expressing the temperatures at A, B, C and D in terms of the temperature at P:

$$\frac{2}{DX^2(\xi_1+\xi_2)}\left[\frac{T_{i-1,j}}{\xi_1} - T_{i,j}\left(\frac{1}{\xi_1}+\frac{1}{\xi_2}\right) + \frac{T_{i+1,j}}{\xi_2}\right]$$

$$+ \frac{2}{DY^2(\delta_1+\delta_2)}\left[\frac{T_{i,j-1}}{\delta_1} - T_{i,j}\left(\frac{1}{\delta_1}+\frac{1}{\delta_2}\right) + \frac{T_{i,j+1}}{\delta_2}\right] = 0 \quad (3)$$

The above equation can be written in the following more convenient form:

$$T_{i,j} = \frac{1}{E} \left[\frac{T_{i-1,j}}{\xi_1(\xi_1+\xi_2)} + \frac{T_{i+1,j}}{\xi_2(\xi_1+\xi_2)} + \left(\frac{DX}{DY}\right)^2 \left\{\frac{T_{i,j-1}}{\delta_1(\delta_1+\delta_2)} + \frac{T_{i,j+1}}{\delta_2(\delta_1+\delta_2)}\right\}\right] \qquad (4)$$

where

$$E = \left[\frac{1}{\xi_1\xi_2} + \left(\frac{DX}{DY}\right)^2 \frac{1}{\delta_1\delta_2}\right] \qquad (5)$$

For equally spaced grid lines the factors ξ_1, ξ_2, δ_1 and δ_2 will be equal to unity and the equations (4) and (5) will reduce to the following form:

$$T_{i,j} = \frac{1}{2(DX^2 + DY^2)} \left[DY^2(T_{i+1,j} + T_{i-1,j}) + DX^2(T_{i,j+1} + T_{i,j-1})\right] \qquad (6)$$

Referring to Figure 3, the convective boundary condition, represented by equation (2), is used to evaluate the temperature at point D on the blade surface. If NDN' is the normal at the boundary point D in Figure 3, then the finite-difference approximation of the normal temperature derivative at D is given by:

$$\frac{\partial T}{\partial n} = \frac{T_D - T_N}{DN} \qquad 7)$$

where T_D is the temperature at point D

T_N is the temperature at point N

DN is the distance between points D and N.

Equation (2) can therefore be written in the following finite-difference form:

$$\frac{T_D - T_N}{DN} = \frac{h_e}{k} (T_e - T_s) \tag{8}$$

or,

$$T_D = \frac{[T_N + \frac{h_e DN}{k} T_e]}{[1 + \frac{h_e}{k} DN]} \tag{9}$$

COOLING CONFIGURATION

The purpose of the present investigation is to predict the temperature distribution in a radial turbine guide vane blade, using different cooling techniques. The present chapter is intended to give the description of the various cooling configurations considered in the present study and their basic terminology.

The nozzle blade, due to its small size, presents a formidable design challange. The internal cooling passage configuration producing a satisfactory vane temperature distribution is one of considerable complexity. Various investigators have pointed out that transpiration cooling is the most effective means of cooling turbine blades. Unfortunately, such cooling system is very complicated from the mechanical point of view. One way to avoid these difficulties and in the meantime retain a high cooling efficiency, is offered by the simultaneous application of impingement and film cooling (Figure 4a). This is accomplished by placing a tube inside the hollow airfoil and orienting the tube such that a series of holes machined into the tube are opposite the inner surface of the blade wall. Cooling air introduced into the tube (Figure 4b) exits through the holes as a series of jets directed towards the inner surface. The cooling air can later be ejected through a slot near the blade trailing edge, to provide film cooling.

Figures 5A and 5B show schematically the four different impinging jet geometries which were investigated in the present study. A schematic representation of the three different slot geometries considered for coolant discharge is shown in Figure 6.

The impingement tubes are identified by a number (1, 2, 3 or 4) and the slot positions are identified by a letter (A, B or D). In a particular cooling configuration one of the coolant discharge slot geometries is used in conjunction with one of the impingement tubes. A particular cooling configuration will therefore be identified by a combination of a letter and a number. Table 1 gives a description of the seven cooling configurations investigated. In all cases the mass flow rate of coolant constituted 3% of the main flow rate.

Once the blade geometry and its cooling configuration are defined, the external and internal flow parameters can be determined according to the procedure described in Chapter 4.

TABLE 1

Slot Position	Impingement Tube No.	Cooling System	Description of the Cooling System
A	1	A1	The coolant is injected through a spanwise slot in the insert and is discharged on the lower surface.
A	2	A2	The coolant is injected through a spanwise slot in the insert and is discharged on the lower surface.
A	3	A3	The coolant is injected through a spanwise slot and an array of holes in the insert and is discharged on lower surface.
B	2	B2	The coolant is injected through a spanwise slot in the insert and is discharged on the lower surface near trailing edge.
B	3	B3	The coolant is injected through a spanwise slot and an array of holes in the insert and is discharged on the lower surface near trailing edge.
B	4	B4	The coolant is injected through a spanwise slot and an array of holes (opposite suction side) in the insert and is discharged on the lower surface near trailing edge.
C	3	C3	The coolant is injected through a spanwise slot and an array of holes in the insert and is discharged on the upper surface near trailing edge.

PRELIMINARY CALCULATIONS

In addition to the blade geometry, the values of the flow velocity, reference stream temperature, heat transfer coefficient over the entire blade surface, and the coolant mass flow distribution are required in the blade temperature computation. In the following the procedure for determining these parameters will be described.

For an essentially subsonic flow, the nozzle vane velocity distribution can be obtained using the computer program of Katsanis [12]. Knowing the velocity, the gas temperature can then be calculated using the energy equation. In the present analysis, the nozzle vane geometry and the main flow conditions are taken similar to the experimental turbine of Reference [5]. The flow channel between the two neighboring nozzle vanes is shown in Figure 7. The flow accelerates on both the pressure and suction surfaces, up to the throat. The dotted line represents the sonic line for the operating conditions under investigation. The flow experiences further acceleration on the blade suction side up to the trailing edge. The gas velocity, the gas temperature and the heat transfer coefficient distributions over the nozzle vane surface under investigation are shown in Figures 8, 9 and 10 respectively. This data was available in the experimental turbine study of Reference [5].

The heat transfer coefficients of Figure 10 were obtained without taking into consideration the effect of coolant discharge. It has been shown in References [14] and [15] that the heat transfer coefficient does not change appreciably in the presence of film cooling, for low coolant mass flow rates. Figure 11 presents the ratio of the heat transfer coefficient h, obtained with film cooling to the heat transfer coefficient h_o, without film cooling plotted over the dimensionless distance d/w downstream from the slot. This figure was plotted in Reference [14] using the data of Reference [15] and is reproduced here. The parameter on the curves is the ratio of the density ρ_c times the exit velocity u_c from the slot to the external mass velocity

$\rho_g u_g$ in the mainstream outside the boundary layer. It can be observed that starting with a distance $d/w = 22$, the heat transfer coefficient differs by less than 10 percent as long as the ratio $\rho_c u_c/\rho_g u_g$ is smaller than 1. In the present investigation the mass flow rate of coolant constituted 3% of the main flow rate and the ratio $\rho_c u_c/\rho_g u_g$ was less than 0.4. This justifies the use of the same values of the convective heat transfer coefficients of Figure 10, downstream of the discharge slot. At these locations, the adiabatic wall temperature, T_{aw}, was determined using the empirical relations of Reference [16].

$$\eta_F = \frac{T_{aw} - T_g}{T_s - T_g} \tag{10}$$

$$\eta_F = \frac{1.9\, Pr^{2/3}}{1 + 0.329\, Re_s^{-0.2} \frac{Cp_g}{Cp_c} \left(\frac{\mu_g}{\mu_c}\right)^{0.2} \left(\frac{\rho_g u_g d}{m_c}\right)^{0.8} \beta} \tag{11}$$

where $\beta = 1 + 5 \times 10^{-5}\, Re_s \frac{\mu_c}{\mu_g} \sin\alpha$

and η_F is the film cooling effectiveness,

T_{aw} is the adiabatic wall temperature,

T_g is the temperature of the hot gas,

T_s is the temperature on the blade surface,

C_p is the specific heat at constant pressure,

m_c is the coolant mass flow per unit length of the slot,

Pr is the Prandtl number,

$Re_s = m_c/\mu_c$ is the slot Reynolds number,

u is the velocity,

d is the distance of the point from the point of coolant discharge,

α is the ejection angle,

μ is the viscosity,

ρ is the density,

g subscript for the gas side parameter,

c subscript for the coolant side parameter.

The physical properties of the coolant such as ρ, Cp, μ depend on the film coolant temperature T_F. Initially this temperature is assumed to be constant and equal to the temperature of the coolant in the impingement tube, T_i. Subsequently it is corrected in the computer program and the temperature of the coolant which increases from leading edge to trailing edge is also obtained using the heat balance equations.

The convective heat transfer coefficient over the internal blade surface, depends on the geometry of the impingement tube as can be seen from Figures 12 through 15. Impingement tubes number one and two provide cooling only near the leading edge with the rest of the internal passage being mainly cooled by convection due to the coolant flow through the passage between the impingement tube and the blade inner surface. Impingement tubes number three and four with the arrays of round jets over the tube surface, in addition to the leading edge slot provide impingement cooling over a larger portion of the blade surface. It therefore results in higher values of film coefficient (h) over the intermediate portion of the internal cooling passage as shown in Figures 14 and 15. The experimental results of Reference [8] were used for the calculation of h at the leading edge in all cases. For impingement tubes number three and four, the experimental data of Reference [10] for jet arrays impingement in the coolant cross flow was used. The forced convective heat transfer coefficients due to the coolant flow either between impingement number two and the blade inner surface or between the two internal blade surfaces downstream of the tubes were calculated using the empirical relation for flow between parallel plates.

$$Nu = 0.014\ Re^{0.81} \tag{12}$$

The mean hydraulic diameter was used as the characteristic length in the calculation of Nusselt and Reynolds numbers. The above equation is based on the experimental work at Rolls Royce and gives a value slightly higher (up to 5%) than the traditional equation of Reference [19] which is

$$Nu = 0.023\ Pr^{0.4}\ Re^{0.8} \tag{13}$$

All the preliminary data necessary in the determination of the temperature distribution in the blade cross-section is now available. In the following chapter, the computer program for blade temperature computations and its input variables, will be described.

PROGRAM DESCRIPTION

In the following, a description of the computer program, its input and output and other information regarding its use will be given. The program listing is given in Appendix A.

Main Program and Subroutines

There are two separate main programs listed in Appendix A. The first program is concerned with the determination of temperature distribution throughout the blade section having coolant discharge on the blade outer lower surface (e.g., configurations A1, A2, A3, B2, B3, B4). The second main program does the same job for the blade section having coolant discharge on the blade outer upper surface (e.g., configuration C3). Many of the computational patterns are similar in the two main programs and it might seem that the two could be efficiently combined. However, the differences are fundamental enough so that the cause of clarity is best served by keeping them separate. One of the two main programs is used along with all the subroutines described later in this chapter, to obtain temperature distribution throughout the blade section.

The main program begins by calling subroutine DATA. This instruction reads in all of the input data required. The appropriate formats and description of the input are discussed in sections (5.2) and (5.3). As an initial guess the temperatures at all grid points are set to any convenient value. The program then calls subroutine TEMAD which calculates the adiabatic wall temperatures at points downstream from the slot using equation (11). At points downstream from the slot, the gas temperatures will be replaced by the corresponding adiabatic wall temperatures in the main program.

The Gauss-Seidel iteration procedure is used to solve equations (4) and (9), giving the better values of the temperatures at the grid points. The convective boundary condition (equation 9) is used to evaluate the temperature on the blade surface point such as D in Figure 3. Referring to this figure, it is necessary to determine the normal DN at point D, in order to use equation (9). Subroutine CUR finds coordinates of points F and C which are adjacent to point D. Subroutine SLO finds an equation of the curve $y = a + bx + cx^2$ passing through points F, D and C. The constants b and c in the parabolic equation are returned to the main program to get the slope of the tangent to the surface and hence slope of the normal to the curve at point D. If the adjacent grid points on the blade surface happen to be very close to each other, the value of the slope of the tangent obtained by subroutine SLO sometimes is not correct. In such cases, the correct value of the slope must be directly specified by inserting a card in the main program. The point N where the normal DN intersects one of the closest mesh lines inside the blade surface can thus be found. The temperature T_N at this point can be determined by linear interpolation using the temperatures at points A and P. Subroutine MESH calculates the temperature $T(i,j)$ at any interior mesh point P (see Figure 2). It determines ξ_1, ξ_2 and δ_1 and δ_2 of Figure 2 and knowing the temperatures at neighboring grid points proceeds to find the new temperature $T(i,j)$ using equation (4).

The temperature of the cooling stream increases along its flow path due to the heat exchange with the hot blade. The new coolant flow temperature computations are performed following the computations of new blade temperatures for each iteration. The heat balance equations which are used to determine the coolant temperature rise over an incremental length of the passage are derived in Appendix B. The local blade temperature and the local coolant temperature are interdependent. For initial set of iterations required to achieve the desired accuracy, the temperature of the film coolant T_f is assumed to be the same as the temperature of the coolant T_i in the impingement tube. The actual film coolant temperature T_f which is higher than T_i and which is determined using heat balance equations, and the physical properties of the coolant at this temperature are used in subroutine TEMAD to obtain new adiabatic wall temperatures which are closer to actual adiabatic wall temperatures. The new values of the film coolant temperature and adiabatic wall temperatures are then used in the boundary conditions of the next iteration. This process is repeated until the desired accuracy is achieved.

The accuracy of the convergence process is specified by a number ε. The iteration process is continued until the sum of the squares of differences in temperatures between two successive iterations is less than ε. Mathematically we can write,

$$\sum_{\text{all points}} (T_n - T_{n-1})^2 < \varepsilon \qquad (14)$$

where T_n is the temperature after n^{th} iteration,

T_{n-1} is the temperature after $(n-1)^{th}$ iteration,

and ε is the prescribed error limit.

One of the problems of Gauss-Seidel method of solving equations is that it is relatively slow to converge to the solution. The method of successive over-relaxation has been used to accelerate the convergence process. In this technique, T_n' the temperature to be used for $(n+1)^{th}$ iteration is obtained by using the following relation:

$$T_n' = \omega T_n + (1-\omega) T_{n-1} \qquad (15)$$

where T_{n-1} is temperature after $(n-1)^{th}$ iteration,

T_n is the temperature after n^{th} iteration,

and ω is the overrelaxation factor.

For $\omega = 0$ the new value of the temperature T_n', would be identical to the old value T_{n-1}, and hence no progress would be made. For $\omega = 1$ the new value of the temperature T_n', would be the same as T_n calculated in the Gauss-Seidel Procedure. Values of ω greater than unity would represent an overrelaxation or an extrapolation beyond the Gauss-Seidel value for temperature. It is this extrapolation process that can accelerate the convergence of the iteration process. Since the iteration process will not converge for values of ω greater than 2, the value of ω will be between 1 and 2.

After the convergence of the solution is achieved, temperatures at all the grid points are printed. The program then calls subroutine TEMP which determines the maximum and minimum temperatures in the blade along any I-line and finds the locations of all the temperatures lying between these two temperatures. Subroutine ISOTH determines isothermal line locations in the blade. The desired number of isotherms and their temperatures are specified by a DATA statement in a BLOCK DATA subprogram. Finally, the cooling effectiveness η_c is evaluated. This parameter facilitates the comparison of different cooling configurations and is defined as follows:

$$\eta_c = \frac{T_g - T_s}{T_g - T_c} \qquad (16)$$

where η_c is the cooling effectiveness,

T_g is the local temperature of the hot gas,

T_s is the local blade temperature obtained using particular type of cooling,

and T_c is the initial temperature of the coolant.

In the following sections the description of the input variables as well as the instructions for preparing the input data are given.

Input Variables

There are ten sets of input data. They are fed in the program in the following order. The variables appearing in these sets will be described in section (5.3).

1. Physical parameters of the grid system:

DX, DY, NXO, NXI	(2F5.3, 2I5)
NX, NY, N1, N2, N3, N4, N5, N6, N7, NP	(10I5)

2. Properties of the blade:

LB , XK, CP	(3F8.6)

3. Properties of the coolant:

PRNO, ALPHA, TCOLN, TEMPC	(4F8.2)
DENSC, CPC, VISCC, FLOC	(2F13.10, E16.10, F8.4)

4. Iteration checks for required convergence:

OME, SUMM, NTE, IMAX	(2F8.3, 2I5)

5. Blade outer and inner boundary point coordinates and their corresponding sequence numbers:

(Y1(I), I=1, NX1)	(12F6.3)
(IP(I), I=1, NX1)	(20I4)
(Y2(I), I=1, NX1)	(12F6.3)
(IP(I), I=1, NX1)	(20I4)
(X1(J), J=1, NY1)	(12F6.3)
(IP(J), J=1, NY1)	(20I4)
(X2(J), J=1, NY1)	(12F6.3)
(IP(J), J=1, NY1)	(20I4)
(Y3(I), I=N11, N21)	(12F6.3)
(IP(I), I=N11, N21)	(20I4)
(Y4(I), I=N11, N21)	(12F6.3)

(IP(I), I=N11, N21)	(20I4)
(X3(J), J=N31, N41)	(12F6.3)
(IP(J), J=N31, N41)	(20I4)
(X4(J), J=N31, N41)	(12F6.3)
(IP(J), J=N31, N41)	(20I4)

6. Gas temperatures and initial coolant temperatures:

(TG(I,1), I=1, NX1)	(10F8.2)
(TG(I,2), I=1, NX1)	(10F8.2)
(TG(I,3), I=N11, N21)	(10F8.2)
(TG(I,4), I=N11, N21)	(10F8.2)
(TGX(1,J), J=2, NY1)	(10F8.2)
(TGX(2,J), J=2, NY1)	(10F8.2)
(TGX(3,J), J=N31, N41)	(10F8.2)
(TGX(4,J), J=N31, N41)	(10F8.2)

7. Heat transfer coefficients on gas side and coolant side boundary surfaces:

(H(I,1), I=1, NX1)	(10F8.4)
(H(I,2), I=1, NX1)	(10F8.4)
(H(I,3), I=N11, N21)	(10F8.4)
(H(I,4), I=N11, N21)	(10F8.4)
(HX(1,J), J=2, NY1)	(10F8.4)
(HX(2,J), J=2, NY1)	(10F8.4)
(HX(3,J), J=N31, N41)	(10F8.4)
(HX(4,J), J=N31, N41)	(10F8.4)

8. Velocities at points where adiabatic wall temperatures are to be determined:

(VELG(1,a), I=N51, NX1)	(10F8.2)
(VELG(b,J), I=2, N6)	(10F8.2)

9. Coolant mass flow:

(MASC(I,3), I=N11, N51)	(10F8.4)
(MASC(I,4), I=N11, N51)	(10F8.4)
(DELM(I,3), I=N11, N5)	(10F8.4)
(DELM(I,4), I=N11, N5)	(10F8.4)

10. Surface distance between coordinate points on coolant side boundary surface:

(DELS(I,3), I=N11, N5)	(10F8.4)
(DELS(I,4), I=N11, N5)	(10F8.4)

Explanation of the Input Variables

There are two types of input variables, geometric and nongeometric. The geometric variables are shown in Figure 16. All input variables and other key variables used in the program are explained in this section.

DX — Grid spacing in the x direction (inches).

DY — Grid spacing in the y direction (inches).

NXO — Total number of mesh line intersections with the outer boundary.

NXI — Total number of mesh line intersections with the inner boundary.

NX — Total number of mesh lines in the x direction, if NX is greater than 30, the DIMENSION statements in the program need to be changed.

NY — Total number of mesh lines in the y direction, if NY is greater than 65, the DIMENSION statements in the program need to be changed.

I — Mesh line perpendicular to the x axis.

J — Mesh line perpendicular to the y axis.

N1, N2 — I mesh lines enclosing the blade inner boundary without intersecting it.

N3, N4	J mesh lines enclosing the blade inner boundary without intersecting it.
N5, N2	I mesh lines enclosing the coolant discharge slot on the blade outer surface, without intersecting it.
N3, N6	J mesh lines enclosing the coolant discharge slot on the blade outer surface, without intersecting it.
N1, N7	I mesh lines enclosing the impingement tube without intersecting it.
NP	Variable to identify the coolant discharge surface. In case of coolant discharge on outer lower surface, put NP = 0 and for coolant discharge on outer upper surface, put NP = 1.
LB	Blade height (inches).
XK	Thermal conductivity of the blade material (Btu/hr in °R).
CP	Specific heat of coolant at constant pressure (Btu/lbm °R).
PRNO	Prandtl number for coolant.
ALPHA	Coolant discharge angle (degrees).
TCOLN	Temperature of the coolant in the impingement tube (°R).
TEMPC	Temperature of the coolant for film cooling (°R).
DENSC	Density of the film coolant (lbm/ft^3).
CPC	Specific heat of film coolant at constant pressure (Btu/lbm °R).
VISCC	Viscosity of the film coolant (lbm/ft-sec).
FLOC	Mass flow of the coolant through the slot per unit length (lbm/ft-sec).
OME	Over-relaxation factor.
SUMM	Maximum allowable error (ε in equation 14).

NTE	Number of iterations after which a print out of the grid temperatures is required.
IMAX	Maximum number of iterations. After IMAX iterations the program terminates even if the required convergence is not achieved.
Y1(I)	Y coordinates of points on the outer lower surface intersected by I mesh lines (inches), (see Figure 16).
IP(I)	Sequence numbers corresponding to the above points (see Figure 16).
NX1	Equals to (NX - 1).
Y2(I), IP(I)	Y coordinates of points on the outer upper surface intersected by I mesh lines (inches), and corresponding sequence numbers (see Figure 16).
X1(J), IP(J)	X coordinates of points corresponding to the first intersection of J mesh lines with the outer surface, i.e., near outer surface (inches), and corresponding sequence numbers (see Figure 16).
NY1	Equals to (NY - 1).
X2(J), IP(J)	X coordinates of points corresponding to the second intersection of J lines with the outer surface, i.e., farther outer surface (inches) and corresponding sequence numbers (see Figure 16).
Y3(I), IP(I)	Y coordinates of points on the inner lower surface intersected by I mesh lines (inches) and corresponding sequence numbers.
N11	Equals to (N1 + 1).
N21	Equals to (N2 - 1).
Y4(I), IP(I)	Y coordinates of points on the inner upper surface intersected by I mesh lines (inches) and corresponding sequence numbers.

X3(J), IP(J)	X coordinates of points corresponding to the first intersection of J mesh lines with the inner surface (inches), and corresponding sequence numbers.
N31	Equals to (N3 + 1).
N41	Equals to (N4 - 1).
X4(J), IP(J)	X coordinates of points corresponding to the second intersection of J mesh lines with the inner surface (inches), and corresponding sequence numbers.
TG(I,a)	Gas temperature at I mesh line intersection point on the blade surface (a = 1,2,3,4), (°R).
TGX(b,J)	Gas temperature at J mesh line intersection point on the blade surface (b = 1,2,3,4), (°R).
H(I,a)	Heat transfer coefficient at I mesh line intersection point on the blade surface (Btu/hr in^2 °R).
HX(b,J)	Heat transfer coefficient at J mesh line intersection point on blade surface (Btu/hr in^2 °R).
VELG(I,a)	Velocity at I mesh line intersection point on the blade surface (ft/sec).
VELG(b,J)	Velocity at J mesh line intersection point on the blade surface (ft/sec).
N51	Equals to (N5 + 1).
MASC(I,a)	Total mass of coolant entering the fluid element at I mesh line, per unit time (see Figures 17 and 18), (lb/hr). For further description refer to Appendix B.
DELM(I,a)	Coolant mass added to the fluid element between I and I+1 mesh lines from the impingement tube, per unit time (see Figures 17 and 18), (lb/hr).
DELS(I,a)	Surface distance between consecutive points on inner surface intersected by I mesh lines (see Figures 17 and 18), (inches).

a=1	Corresponds to outer lower surface.
a=2	Corresponds to outer upper surface.
a=3	Corresponds to inner lower surface.
a=4	Corresponds to inner upper surface.
b=1	Corresponds to near outer surface.
b=2	Corresponds to farther outer surface.
b=3	Corresponds to near inner surface.
b=4	Corresponds to farther inner surface.
T(I,J)	Temperature at a grid point, (°R).
TB(I,a)	Blade temperature at I mesh line intersection point on the blade surface, (°R).
TBX(b,J)	Blade temperature at J mesh line intersection point on the blade surface, (°R).
NSCLT	Number of isotherms for which locations are to be determined.
TSCLT	Temperatures of the isotherms (NSCLT temperatures have been specified with the DATA statement in the BLOCK DATA subprogram. These can be changed if isothermal line locations for different set of temperatures are desired).
NNY	Dummy variable which counts the number of points determined on an isotherm using ISOTH subroutine. Initial values of this variable have been set to zero in the DATA statement.

Preparation of the Input Data

The input to the computer program to determine the temperature distribution in the guide vane blade consists of the sets of variables indicated in Sections (5.2) and (5.3). In the following, more information regarding the preparation of the input data will be given.

A rectangular grid network is used for the solution of the basic differential equation with specified boundary conditions. A typical mesh pattern is shown in Figure 16. The cross section of the blade is divided into grids of equal increments in both x and y directions. The mesh spacing in the x direction need not be the same as the one in the y direction. The mesh lines perpendicular to the x direction are referred to as the I lines while the mesh lines perpendicular to the y direction are referred to as the J lines. The grid network should be drawn such that there are not multiple intersection points, i.e., each grid line intersects the outer boundary at most twice and the inner boundary also at most twice. It is convenient to draw the grid network with $I = 1$ line tangential to the blade.

In order to read the coordinates of points on the blade boundary, each point is associated with a number for purpose of its identification. The numbering can start from any nodal point on the blade boundary but it is done in the following manner for convenience. For the blade outer surface the numbering starts from the node point where $I = 1$ line meets the outer blade surface and for the blade inner surface it starts from the node point near the leading edge of the blade inner surface (see Figure 16). The (I=1, J=1) point is treated as origin and the coordinates of points on the boundary are given with respect to this point. The boundary coordinates and their corresponding sequence numbers are now defined.

The next two sets of input data consist of gas temperatures, convective heat transfer coefficients and velocities which are calculated in Chapter 4. Note that the initial temperature of the coolant is assumed to be constant and equal to that in the impingement tube, T_i.

Finally, the coolant mass flow distribution may be obtained as follows. The total coolant mass flow entering the impingement tube per unit time and the impingement tube configuration are selected. As mentioned in Chapter 3, the impingement tube may have slots or rows of holes for impinging the coolant against the blade inner surface. Knowing the amount of coolant injected through each slot or hole, it is possible to calculate the mass of the coolant crossing any I mesh line. Figure 17 shows typical coolant flow path. The typical control volumes used to calculate the increment in coolant temperature are shown in Figures 18 and 19 (refer to Appendix B for further description of the control volumes). It can be noted that

$$\dot{m}(i) + \Delta\dot{m}(i) = \dot{m}(i+1) \qquad (17)$$

where $\dot{m}(i)$ = Coolant mass entering the control volume per unit time at I mesh line.

$\Delta\dot{m}(i)$ = Coolant mass added between I and I+1 mesh lines from the impingement tube per unit time.

$\dot{m}(i+1)$ = Coolant mass leaving the control volume at I+1 mesh line per unit time.

It is very unusual to prepare the input without any errors, the first time this program is run. Therefore, it is recommended that the first attempt for running the program should allow maximum of ten iterations. The resulting output should be checked carefully. Of particular interest are the values of the slopes of tangent to the blade boundary at all the boundary points. Any error in the blade geometry input will usually result in wild values of some of these derivatives. All other preliminary output should be checked to see that it is reasonable.

Output

The first part of the output consists of a printout of the list of input parameters for the purpose of reference and checking. Such data checks are useful in correcting keypunch mistakes on the input cards. The program then prints out the boundary point number, boundary point coordinates and the coordinates of two adjacent points. The coefficients a, b and c in the equation $y = a + bx + cx^2$ of the parabolic curve passing through these three points and the slope of the tangent to the curve at the boundary point are also printed. The error ε for every iteration is then printed. This allows monitoring of the convergence process. The grid temperatures as well as blade surface temperatures and gas temperatures are printed after NTE iterations and also after the convergence of the solution is achieved. In the output, the gas temperatures at points downstream from the slot are replaced by the adiabatic wall temperatures. The next part of the output is a printout of the locations along an I line, of blade temperatures which have values between the maximum and minimum temperatures along that I line. The isothermal line locations corresponding to the temperatures specified in the DATA statement of the BLOCK DATA subprogram are then printed. Finally, the values of cooling effectiveness at all surface points are printed. In Appendix A, a sample of the program output is presented. This output corresponds to the B3 cooling configuration of Figure 29.

6. RESULTS AND DISCUSSION

A computer program has been developed to predict the temperature distribution of the cooled guide vane blades. The results of the investigation are presented as plots of isothermal lines in the blade cross-section, and the cooling effectivness variation over the blade surfaces for the various cooling arrangements of Table 1. The geometry of the turbine nozzle blades used in this study is that of Reference [5] and under the same main flow operating conditions. The hot gas inlet stagnation temperature is 2760°R and the gas turbine flow rate is 4.9 lb/sec. The integral nozzle casting consists of 20 hollow vanes with inserts as shown in Figure 4A. The coolant mass flow rate in the nozzle vane was taken as 3% of the main flow rate per channel for all the cooling configurations that are investigated.

The temperature distribution and the cooling effectiveness corresponding to the A1 cooling configuration are shown in Figures 20 and 21 respectively. In this cooling arrangement efficient cooling of the leading edge inner surface is achieved by the normal impingement of all the coolant through the leading edge slot in the impingement tube number 1 of Figure 5A. The coolant is divided into two streams around the tube with a lower velocity resulting when they rejoin at the end of the tube. The coolant is finally discharged on the blade pressure surface. The blade temperatures are low near the leading edge inner surface and increase rapidly on both the pressure and suction surfaces away from the leading edge. This is expected because for a slot jet, the local heat transfer coefficient undergoes a rapid decay after its maximum value in the stagnation region. Higher metal temperatures are observed in the central portion of the blade section (surface distance $\Delta S_o/S_o$ between 20% and 40%). This is a result of the reduced coolant velocities in that region as explained earlier. The metal temperatures are low near the point of coolant discharge and increase at the points downstream of it. This is expected since the value of the film cooling effectiveness, η_F, expressed in equation (11), is higher near the point of coolant discharge and decreases towards the trailing edge.

Figure 22 shows the isothermal lines in the blade for the A2 cooling configuration. In this case, the vane has a closely conforming tube which provides impingement cooling to the leading edge and gives good convection cooling also, on account of high velocity flow along the side wall passages. The cooling effectiveness variation on the blade surface for this cooling configuration is shown in Figure 23. The temperature distribution in the blade and the cooling effectiveness variation for the A3 cooling configuration are shown in Figures 24 and 25 respectively. The impingement tube used in this case has a grid of 0.012 inch diameter holes and a spanwise slot in the leading edge. In such cooling arrangement, higher internal heat transfer coefficients are achieved in the leading edge as well as in the central portion of the cooling passage. As a consequence, lower blade temperatures result in both these regions for this particular cooling arrangement.

For the purpose of comparison, the cooling effectiveness variations for A1, A2 and A3 cooling configurations are shown together in Figure 26. It is evident that the best heat transfer performance is obtained using the round jet array, configuration A3. It is also clear that the temperatures are almost the same near the trailing edge region for the three different configurations. This could be expected as a result of the same coolant discharge position in all three cases.

The temperature distributions obtained so far are for a blade having the same coolant passage with different impingement tubes. In the following, the temperature distributions in the blade for various coolant discharge positions are presented. Theoretically, it is feasible to exhaust the cooling air back into the gas stream at any point on the vane surface. This is possible because the static pressure around the surface of the vane is everywhere lower than the stagnation pressure of the coolant, except at leading edge. However, when a tube insert is used as part of the cooling system, the coolant stagnation pressure is reduced and then it is only possible to discharge the cooling air back into the gas stream at points on the vane surfaces, which are close to trailing edge. The choice in practice is a compromise between the requirements of cooling, mechanical strength and good aerodynamic performance. The present analysis does not consider the aerodynamic effects of cooling air discharge on the turbine performance. The penalties in terms of turbine efficiency are minimized however for cooling air exhaust on the pressure surface near trailing edge.

The blade metal temperature distribution obtained for the B2 cooling configuration is shown in Figure 27. For this configuration the cooling air path near discharge location was chosen keeping in mind that the minimum thickness for casting is 0.02 to 0.025 inch. The cooling effectiveness on the blade surface corresponding to the above configuration is shown in Figure 28. The isothermal lines and the cooling effectiveness corresponding to B3 and C3 configurations are shown in Figures 29 through 32. The effect of changing the coolant discharge position is demonstrated in Figure 33 which combines the results of A3, B3 and C3 cooling configurations. It can be seen that the B3 cooling configuration shows a better heat transfer performance compared to the A3 and C3 configurations.

A closer examination of Figure 30 reveals that, generally the cooling effectiveness is higher on the pressure side as compared to the suction side when the array configuration impingement tube of configuration B3 is used. Another cooling configuration B4 was studied in order to improve the suction surface cooling. In this arrangement, the tube has a closely spaced grid of 0.012 inch diameter holes which provides impingement cooling on the suction surface, while the pressure surface is provided with convection cooling only. Figures 34 and 35 show the temperature and cooling effectiveness distributions corresponding to this configuration. The cooling effectiveness variations for B3 and B4 configurations are plotted in Figure 36 to provide a means of comparison. It is evident that more uniform temperature distribution is obtained by employing B4 configuration and that this configuration demonstrates best heat transfer performance in the leading edge area, central portion of the blade as well as in the trailing edge area. The computer time naturally depends on the particular blade geometry under consideration, as well as on the initial guesses of metal and coolant temperatures, but did not exceed 35 seconds in any of the cases reported here.

7. CONCLUSION

A useful numerical technique has been developed to predict the temperature field in the cooled guide vane blade of a radial inflow turbine. The computer program which has been developed can be used to predict the temperature distribution in the conventional nozzle guide vane with impingement cooling at the leading edge and film cooling near the trailing edge. The impingement tube can have any geometry but the coolant can be discharged through a single slot. This computer program can also be used to study the effects of changes in the coolant to primary mass flow ratio and initial temperature of the coolant on the temperature distribution in the blade. In the present investigation, the temperature distribution in the cooled guide vane blade was obtained using four different impingement tubes and three different discharge locations. The results show that simple convection cooling configurations are not particularly suitable for turbine vanes under the combination of high gas temperatures and pressures. Of the various cooling configurations investigated the case of impingement cooling of the leading edge, cooling of suction side by round jet array and cooling discharge near the trailing edge was found to give the best temperature distribution. According to the numerical results obtained the areas which need special attention if high local metal temperatures are to be avoided are:

(1) Leading edge where external heat transfer coefficients are high.

(2) The suction surface where external heat transfer coefficients are high after boundary layer transition from laminar to turbulent.

(3) Trailing edge where difficulties exist in getting internal passages with adequate surface area.

REFERENCES

1. Esgar, J.B., "Turbine Cooling, Its Limitations and Its Future," AGARD CP 73-71, Paper No. 14.

2. Halls, G.A., "Nozzle Guide Vane Cooling, The State of the Art," AGARD CP 73-71, Paper No. 25.

3. Okapuu, U., and Calvert, G.S., "An Experimental Cooled Radial Turbine," AGARD CP 73-71, Paper No. 10.

4. Calvert, G.S., and Okapuu, U., "Design and Evaluation of a High Temperature Radial Turbine," USAA Labs Technical Report 68-69, January 1969.

5. Calvert, G.S., Beck, S., and Okapuu, U., "Design and Experimental Evaluation of High Temperature Radial Turbine," USAA Labs. Technical Report 71-20, May 1971.

6. Kuhl, W., "Experimental Investigation on a Single-Stage Air Cooled Gas Turbine," AGARD CP 73-71, Paper No. 16.

7. Hamed, A., Baskharone, E., and Tabakoff, W., "Temperature Distribution Study in a Cooled Radial Inflow Turbine Rotor," AIAA Paper No. 76-44.

8. Tabakoff, W., and Clevenger, W., "Gas Turbine Blade Heat Transfer Augmentation by Impinging of Air Jets Having Various Configurations," Journal of Engineering for Power, Trans. ASME, January 1972.

9. Metzger, D.E., Yamashita, T., and Jenkins, C.W., "Impinging Cooling of Concave Surfaces With Lines of Circular Air Jets," Journal of Engineering for Power, Trans. ASME, July 1969.

10. Kercher, D.M., and Tabakoff, W., "Heat Transfer by a Square Array of Round Air Jets Impinging Perpendicular to a Flat Surface Including the Effect of Spent Air," Journal of Engineering for Power, Trans. ASME, January 1970.

11. Chupp, R.E., Helms, H.E., McFadden, P.W., and Brown, T.R., "Evaluation of Internal Heat Transfer Coefficients for Impingement Cooled Turbine Airfoils," AIAA Paper No. 68-564.

12. Katsanis, T., "Fortran Program for Calculating Transonic Velocities on a Blade-To-Blade Stream Surface of a Turbine," NASA TN D-5427, September 1969.

13. Herring, H.J., and Mellor, G.L., "A Computer Program to Calculate Incompressible Laminar and Turbulent Boundary Layer Development," NASA CR-1564, March 1970.

14. Eckert, E.R.G., "Film Cooling With Injection Through Holes," AGARD CP 73-71, Paper No. 17.

15. Hartnett, J.P., Birkebak, R.C., and Eckert, E.R.G., "Velocity Distributions, Effectiveness and Heat Transfer in Film Cooling of a Surface with a Pressure Gradient," International Developments in Heat Transfer, Part IV, Section A, Paper No. 81.

16. Goldstein, R.J., and Haji Sheikh, A., "Prediction of Film Cooling Effectiveness," Heat and Mass Transfer, Thermal Stress, Vol. I, Paper No. 125.

17. Hosny, W., and Tabakoff, W., "A Two-Dimensional Finite Difference Solution for the Temperature Distribution in a Radial Gas Turbine Guide Vane Blade," NASA CR 137633.

18. Myers, G.E., "Analytical Methods in Conduction Heat Transfer," McGraw-Hill Book Company, 1971.

19. McAdams, W.H., "Heat Transmission," McGraw-Hill Book Company, 1954.

LIST OF SYMBOLS

Symbol	Description
A,B,C	Letters to identify coolant discharge position (see table 1).
C_p	Specific heat at constant pressure.
d	Distance of a point from the point of coolant discharge (see figure 11).
DN	Normal distance (see figure 3).
DX	Grid spacing in the x direction (see figure 3).
DY	Grid spacing in the y direction (see figure 3).
h	Heat transfer coefficient.
I	Mesh line perpendicular to the x axis (see figure 16).
J	Mesh line perpendicular to the y axis (see figure 16).
k	Thermal conductivity.
LB	Length of the blade (equation b.1).
$\dot{m}$	Mass flow rate (see figure 18).
Nu	Nusselt number.
n	Normal direction (see equation 2).
Pr	Prandtl number.
Re	Reynolds number.
s	Distance along blade surface (see figure 8).
T	Temperature.
T(i,J)	Temperature at grid point (i,J), (see figure 2).
T_f	Film coolant temperature.
T_i	Temperature of the coolant in the impingement tube.
T_n	Temperature after n^{th} iteration (equation 14).

Symbol	Description
T_s	Temperature on the blade surface (equation 16).
TG	Temperature of the gas (equation b.1).
TB	Temperature of the blade (equation b.1).
u	Velocity of the stream.
w	Slot width.
ω	Over-relaxation factor (equation 15).
x,y	Coordinate directions (see figure 1).
η_c	Cooling effectiveness (equation 16).
η_F	Film cooling effectiveness (equation 10).
$\Delta\dot{m}$	Additional mass flow rate from the impingement tube (see figure 18).
ρ	Density.
μ	Viscosity.
α	Coolant discharge angle (equation 11).
ε	Error limit (equation 14).
$\xi_1, \xi_2, \delta_1, \delta_2$	Fractions (see figure 2).
1,2,3,4	Numbers to identify the impingement tube (see table 1).

Subscripts

aw	Adiabatic wall.
c	Coolant.
e	Reference stream.
g	Hot gas.

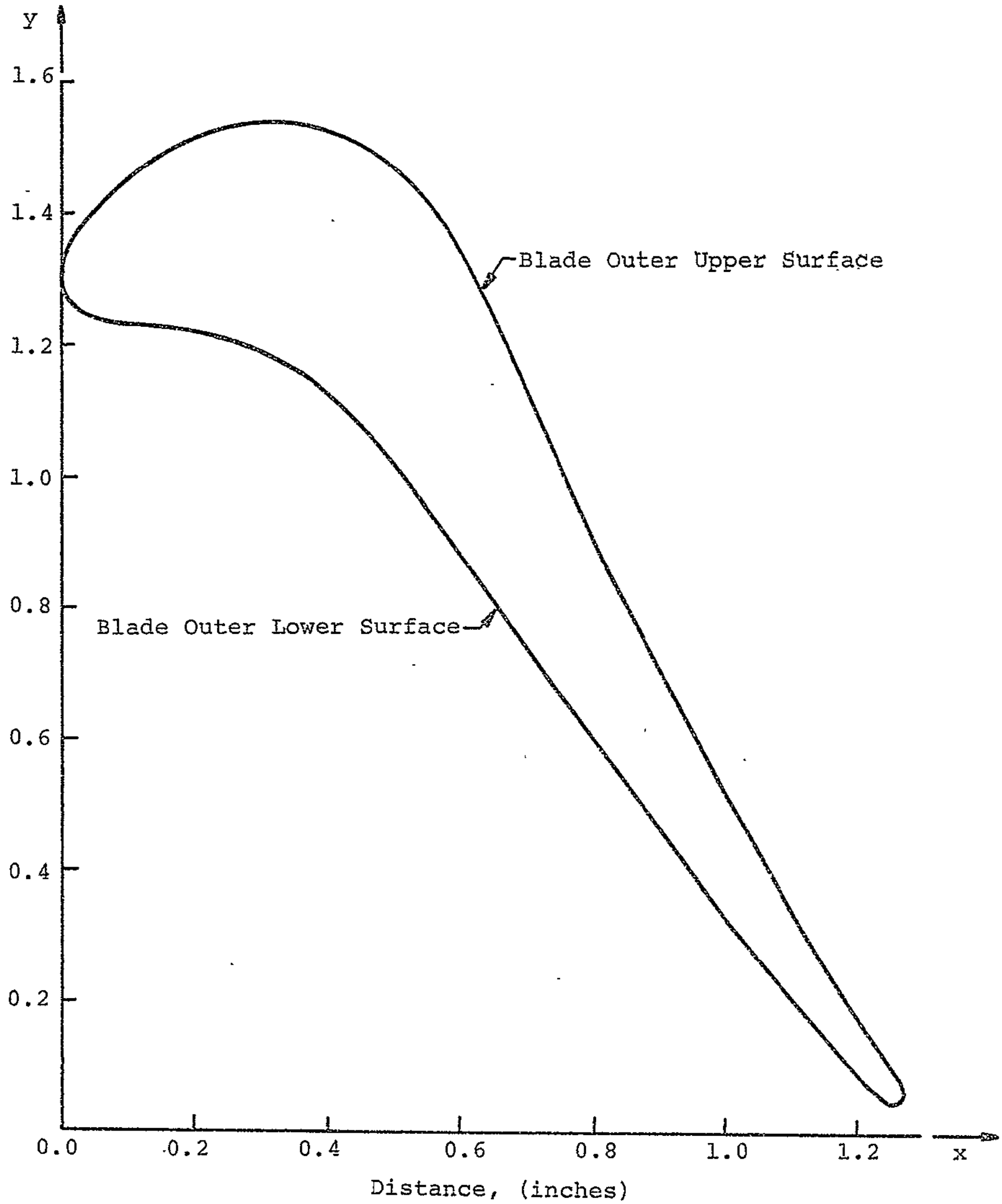


FIGURE 1. RADIAL TURBINE GUIDE VANE BLADE PROFILE.

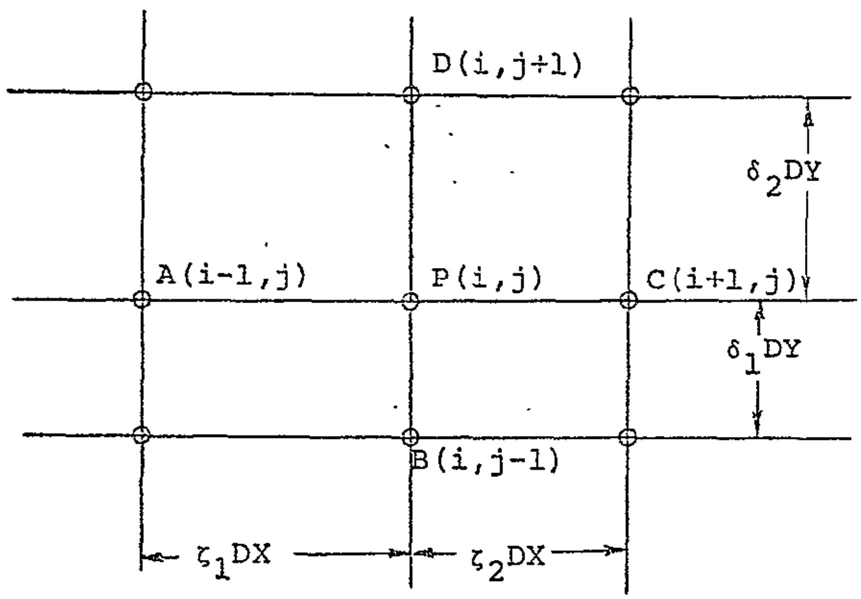


FIGURE 2. TYPICAL GRID SYSTEM WITH UNEQUAL SPACINGS.

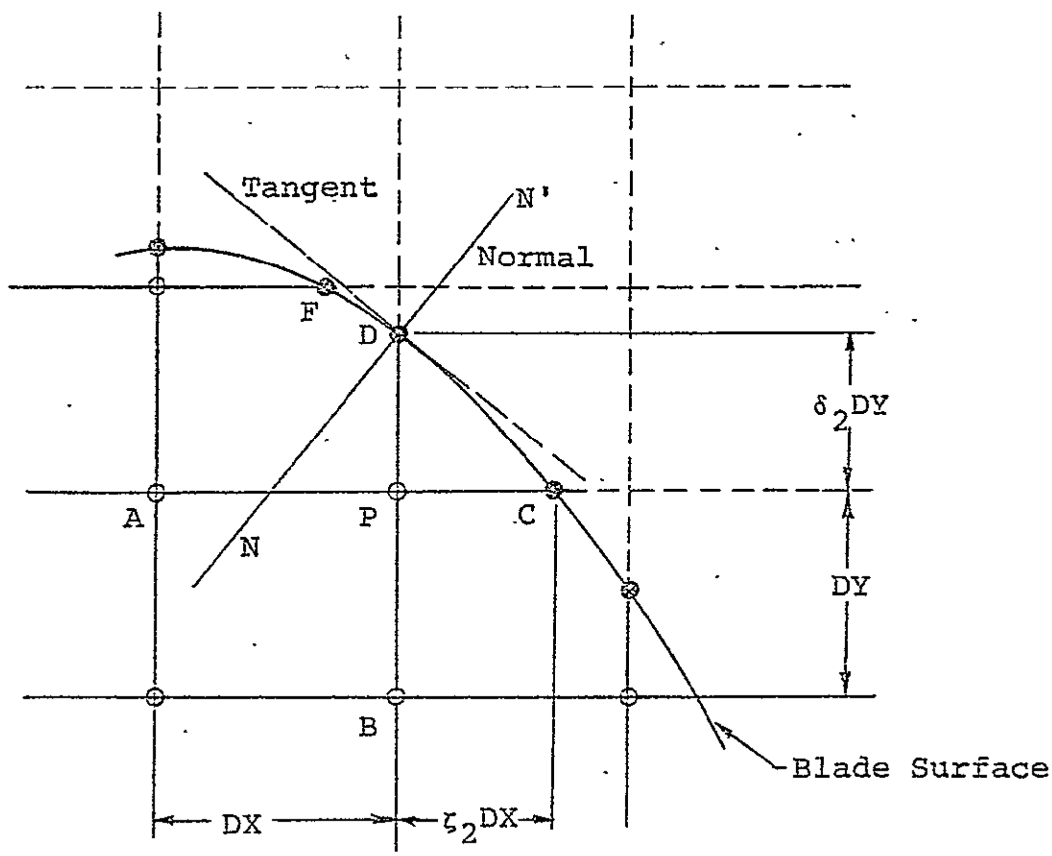


FIGURE 3. GRID POINTS NEAR BLADE SURFACE.

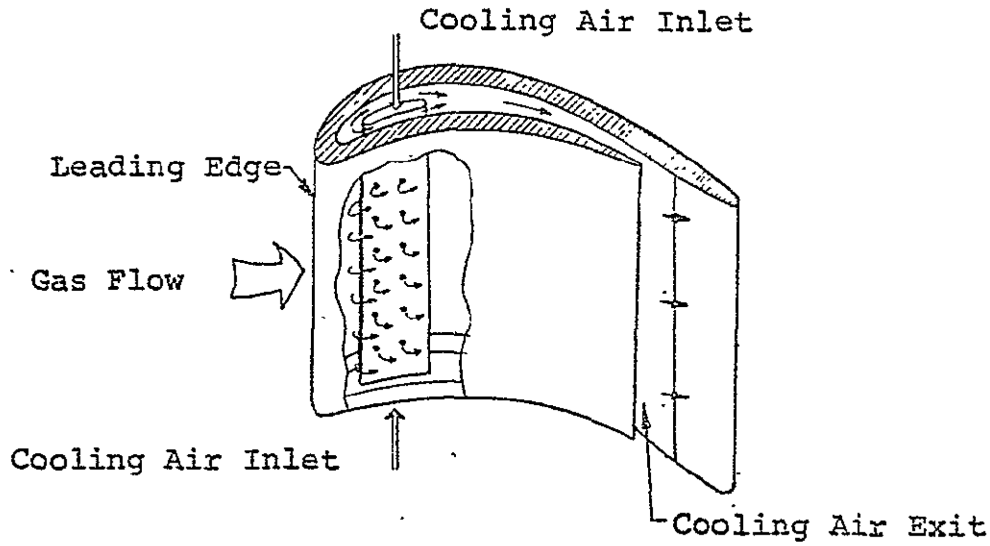


FIGURE 4A. SCHEMATIC OF IMPINGEMENT AND FILM COOLED AIRFOIL.

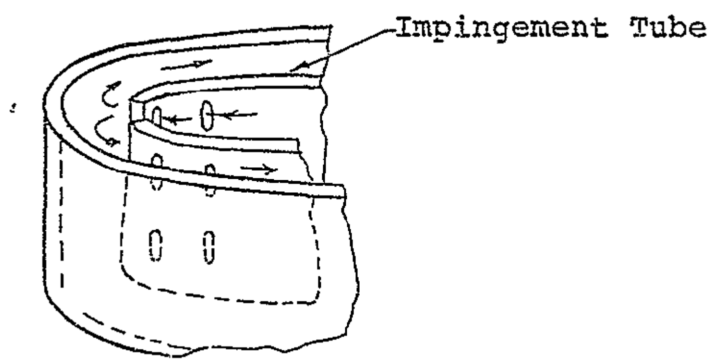
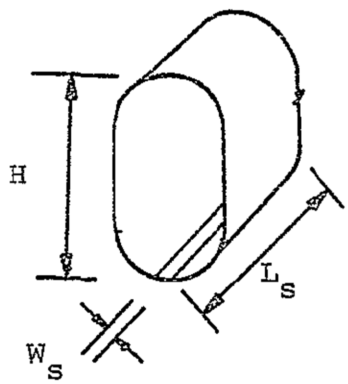


FIGURE 4B. SCHEMATIC OF IMPINGEMENT-COOLED AIRFOIL LEADING EDGE.



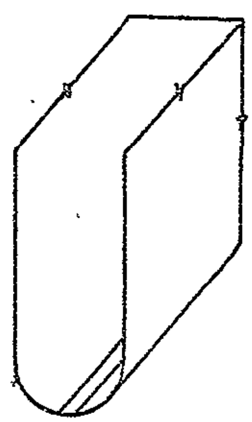
Width of Slot Jet = W_s = 0.02 in.

Length of Slot Jet = L_s = 0.339 in.

Height of Impingement Tube = H = 0.18 in.

3% Cooling Air Through Leading Edge Slot

Impingement Tube No. 1 (Slot Jet)



Width of Slot Jet = 0.02 in.

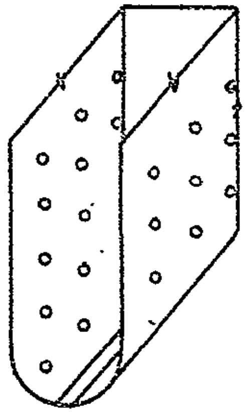
Length of Slot Jet = 0.339 in.

Height of Impingement Tube = 0.5 in.

3% Cooling Air Through Leading Edge Slot

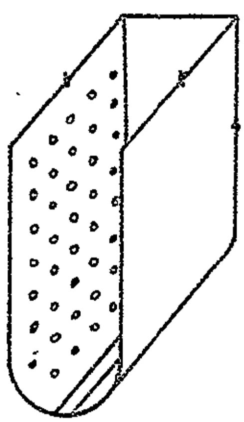
Impingement Tube No. 2 (Slot Jet)

FIGURE 5A. SCHEMATIC OF THE AIR JET IMPINGEMENT CONFIGURATIONS.



Width of Slot Jet = 0.01 in.
Length of Slot Jet = 0.339 in.
Height of Impingement Tube = 0.5 in.
Round Jet Diameter = 0.012 in.
Round Jet Spacing = 0.10 in.
1% Cooling Air Through Leading Edge Slot
1% Cooling Air Through Grid on Suction Side
1% Cooling Air Through Grid on Pressure Sid

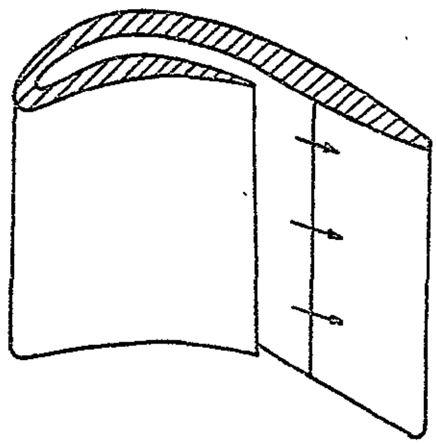
Impingement Tube No. 3 (Slot Jet and Array of Round Jets)



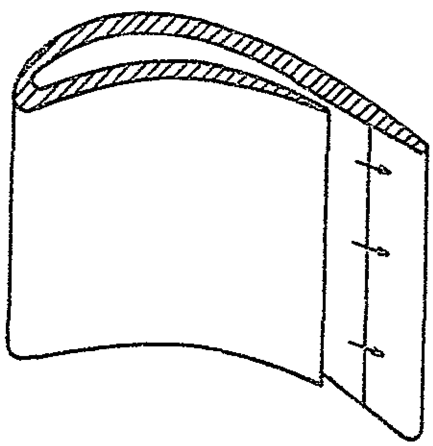
Width of Slot Jet = 0.01 in.
Length of Slot Jet = 0.339 in.
Height of Impingement Tube = 0.5 in.
Round Jet Diameter = 0.012 in.
Round Jet Spacing = 0.05 in.
1.5% Cooling Air Through Leading Edge Slot
of which 1% flows towards pressure side
1.5% Cooling Air Through Grid on Suction Si

Impingement Tube No. 4 (Slot Jet and Array of Round Jets)

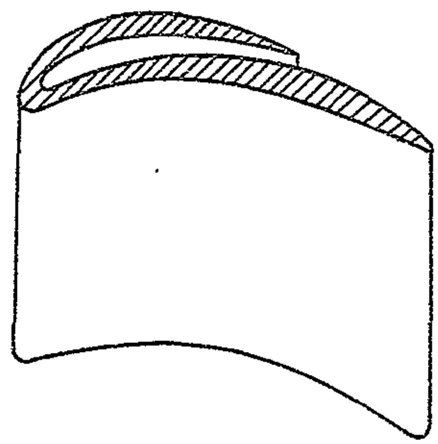
FIGURE 5B. SCHEMATIC OF THE AIR JET IMPINGEMENT CONFIGURATIONS.



(A) Coolant Discharge on Lower Surface



(B) Coolant Discharge on Lower Surface Near Trailing Edge.



(C) Coolant Discharge on Upper Surface

FIGURE 6. SCHEMATIC OF THE SLOT GEOMETRIES FOR COOLANT DISCHARGE.

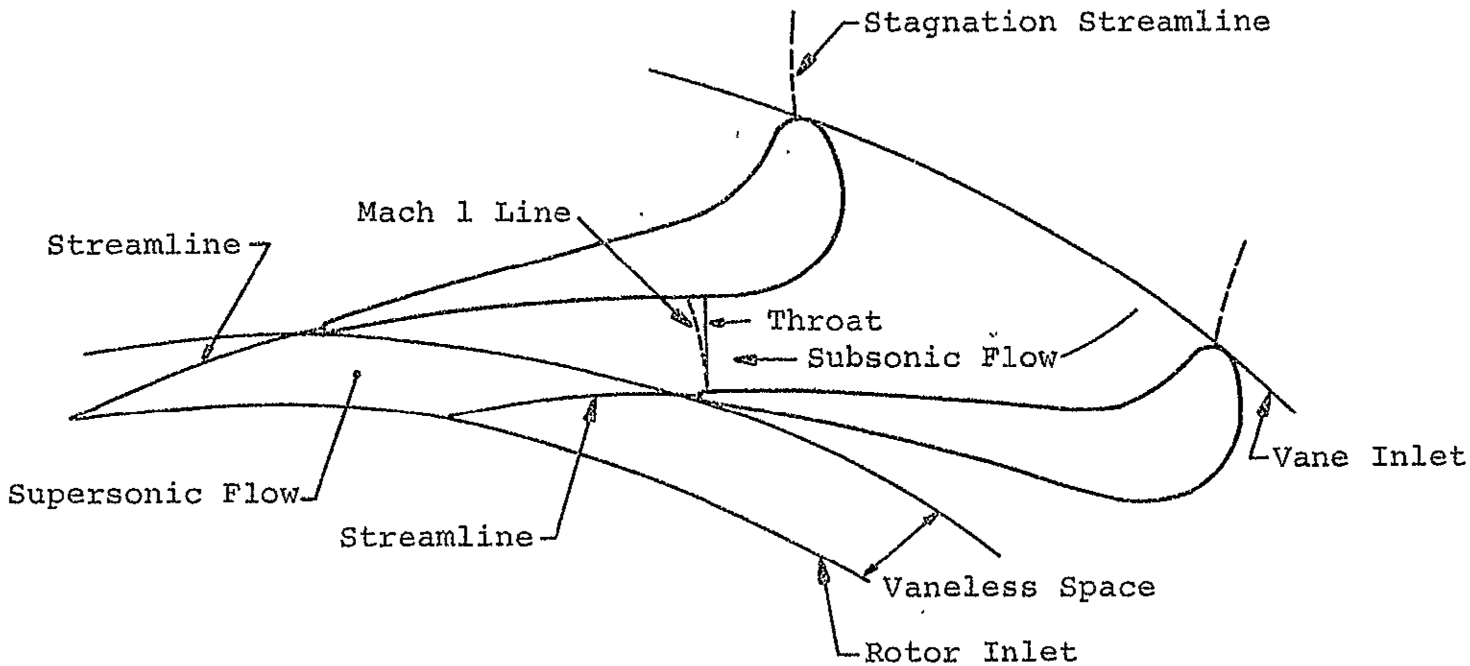


FIGURE 7. REGION OF THE FLOW THROUGH NOZZLE VANES (REFERENCE 4).

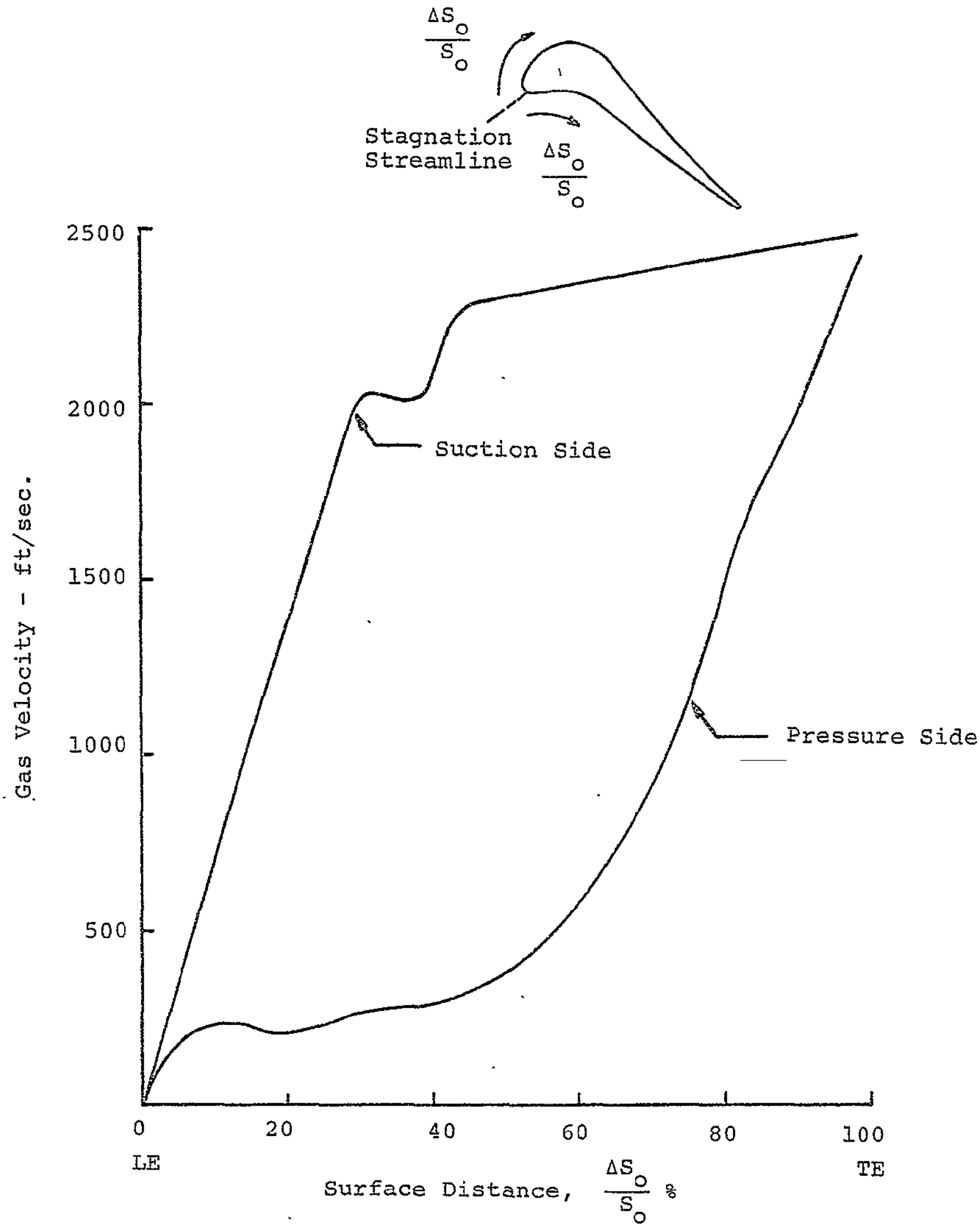


FIGURE 8. NOZZLE VANE VELOCITY DISTRIBUTION (REFERENCE 5).

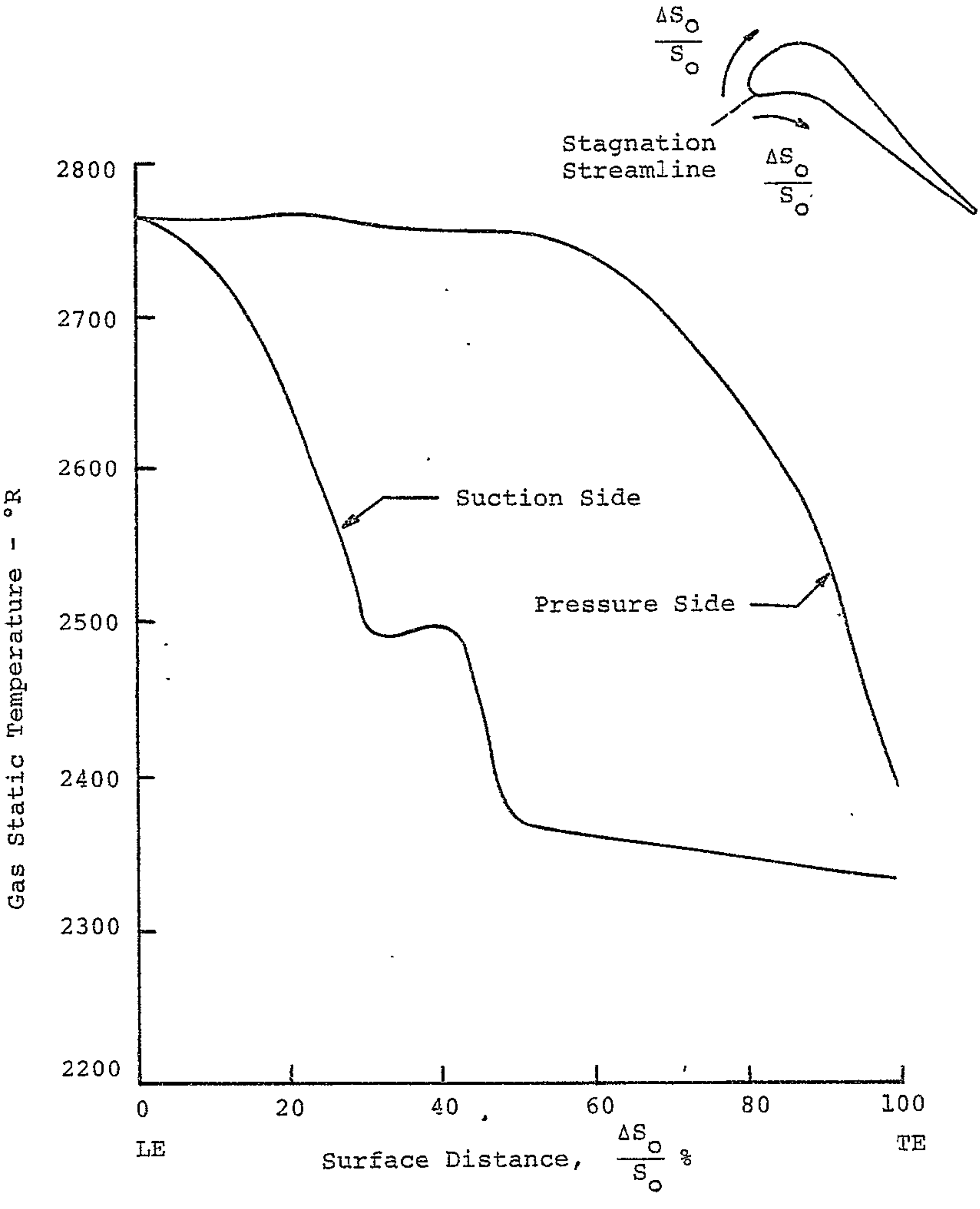


FIGURE 9. NOZZLE VANE GAS TEMPERATURE DISTRIBUTION (REFERENCE 5).

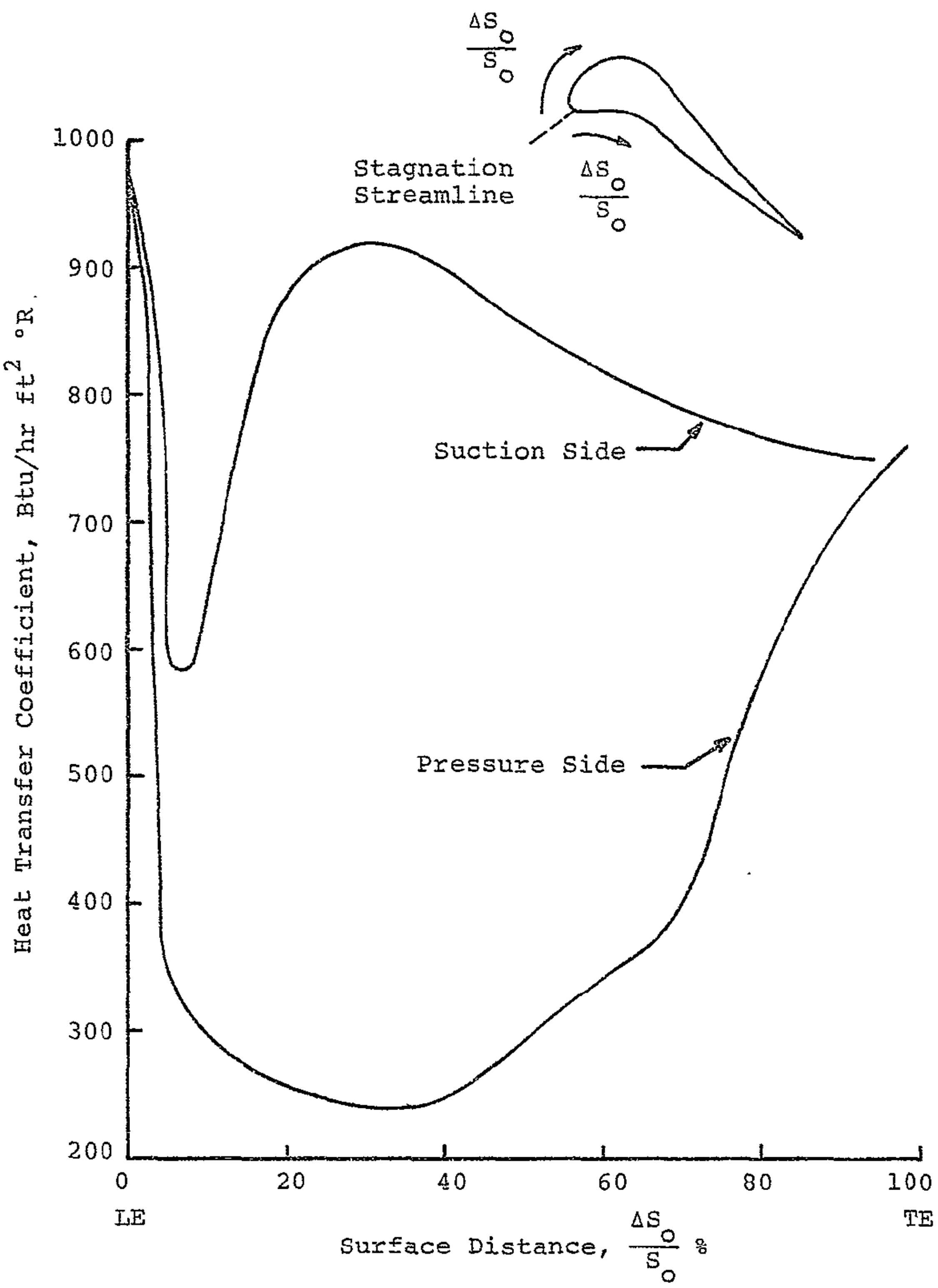


FIGURE 10. NOZZLE VANE HEAT TRANSFER COEFFICIENTS, GAS SIDE (REFERENCE 5).

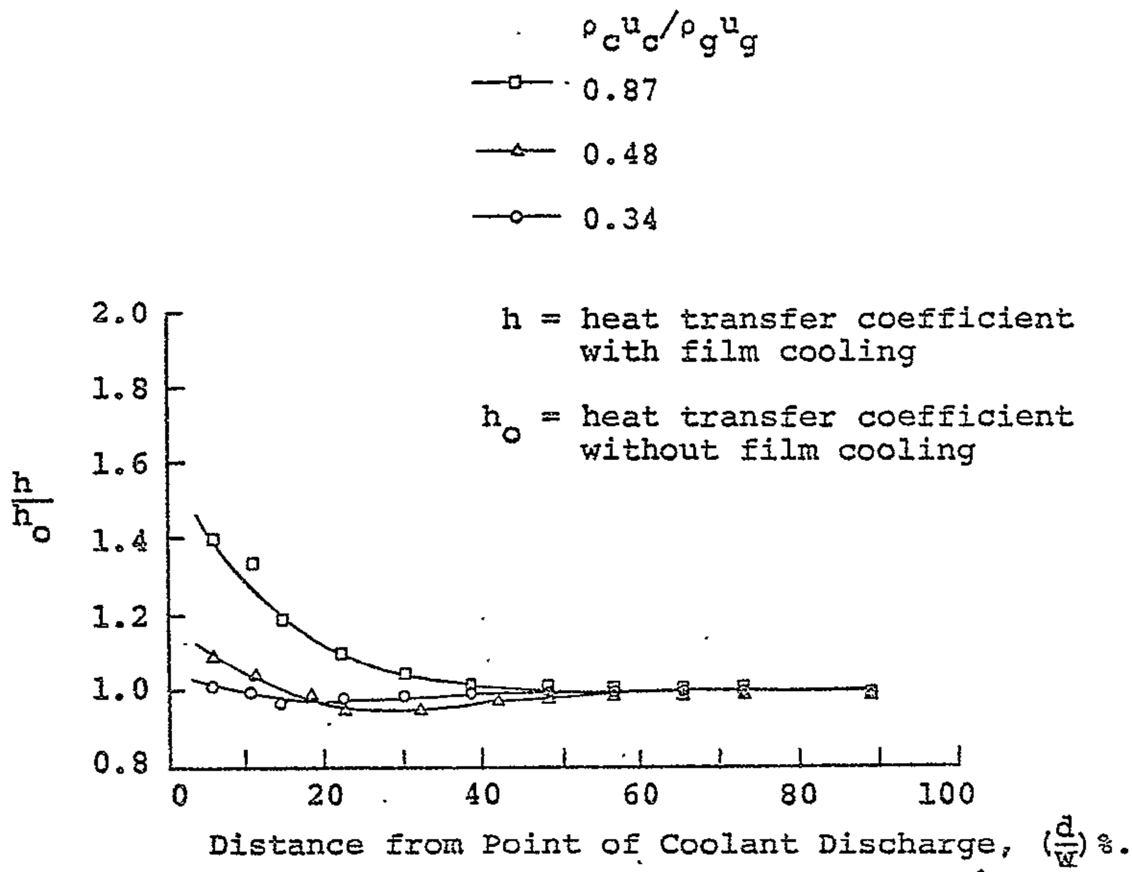


FIGURE 11. h/h_o AS A FUNCTION OF d/w (REFERENCE 15).

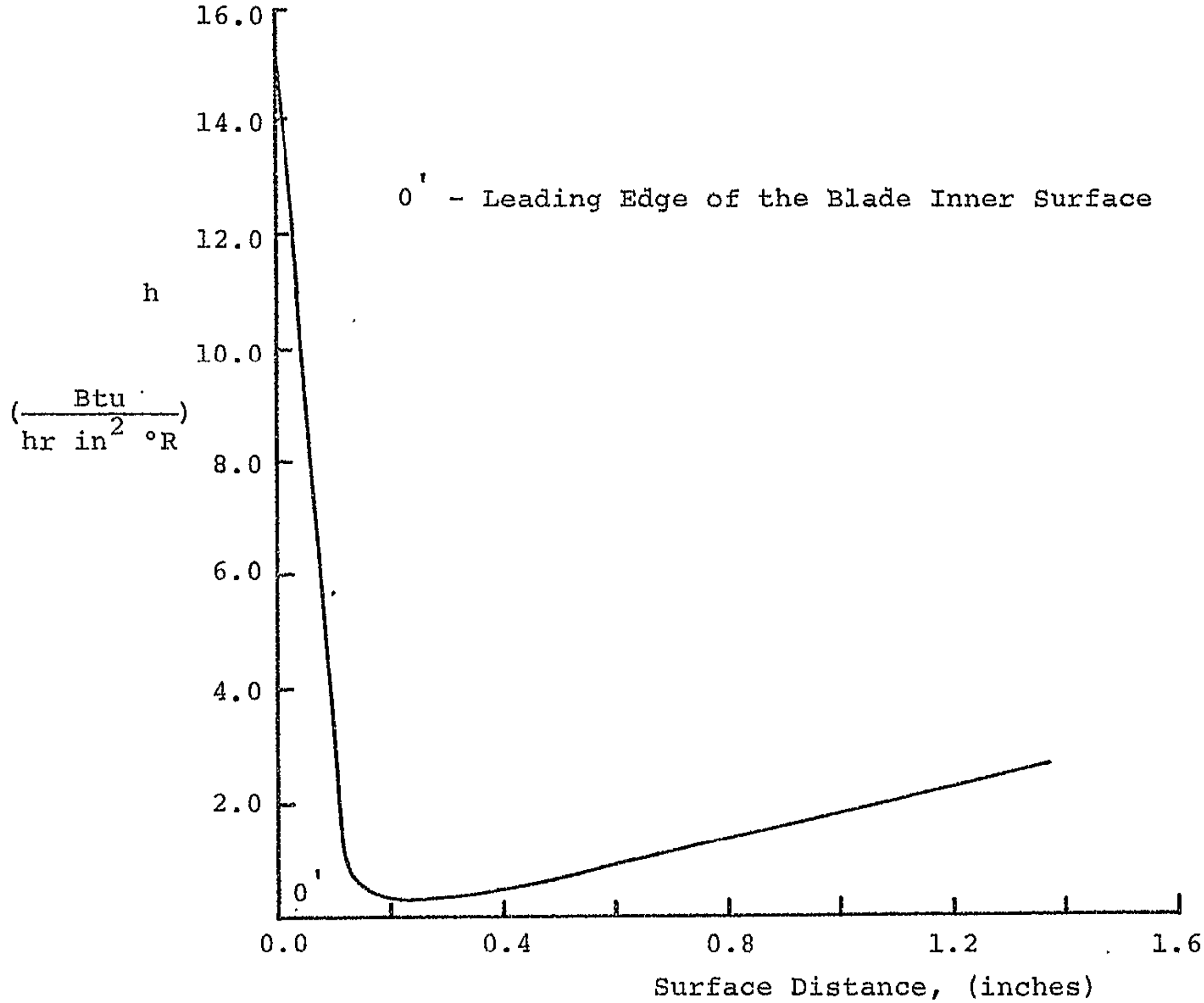


FIGURE 12. HEAT TRANSFER COEFFICIENT DISTRIBUTION ON INNER BOUNDARY WITH 3% COOLANT (CONFIGURATION A1).

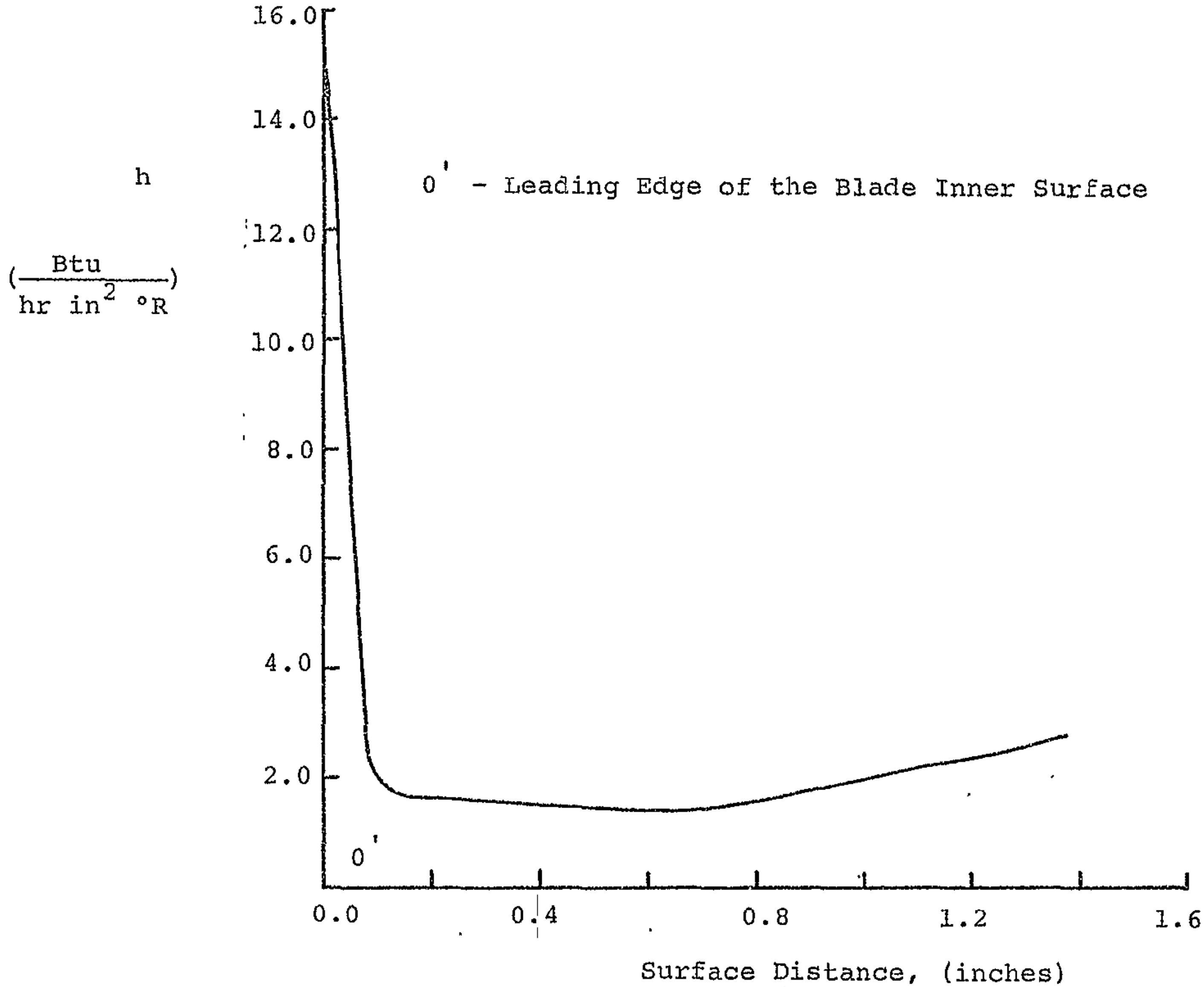


FIGURE 13. HEAT TRANSFER COEFFICIENT DISTRIBUTION ON INNER BOUNDARY WITH 3% COOLANT (CONFIGURATIONS A2, B2).

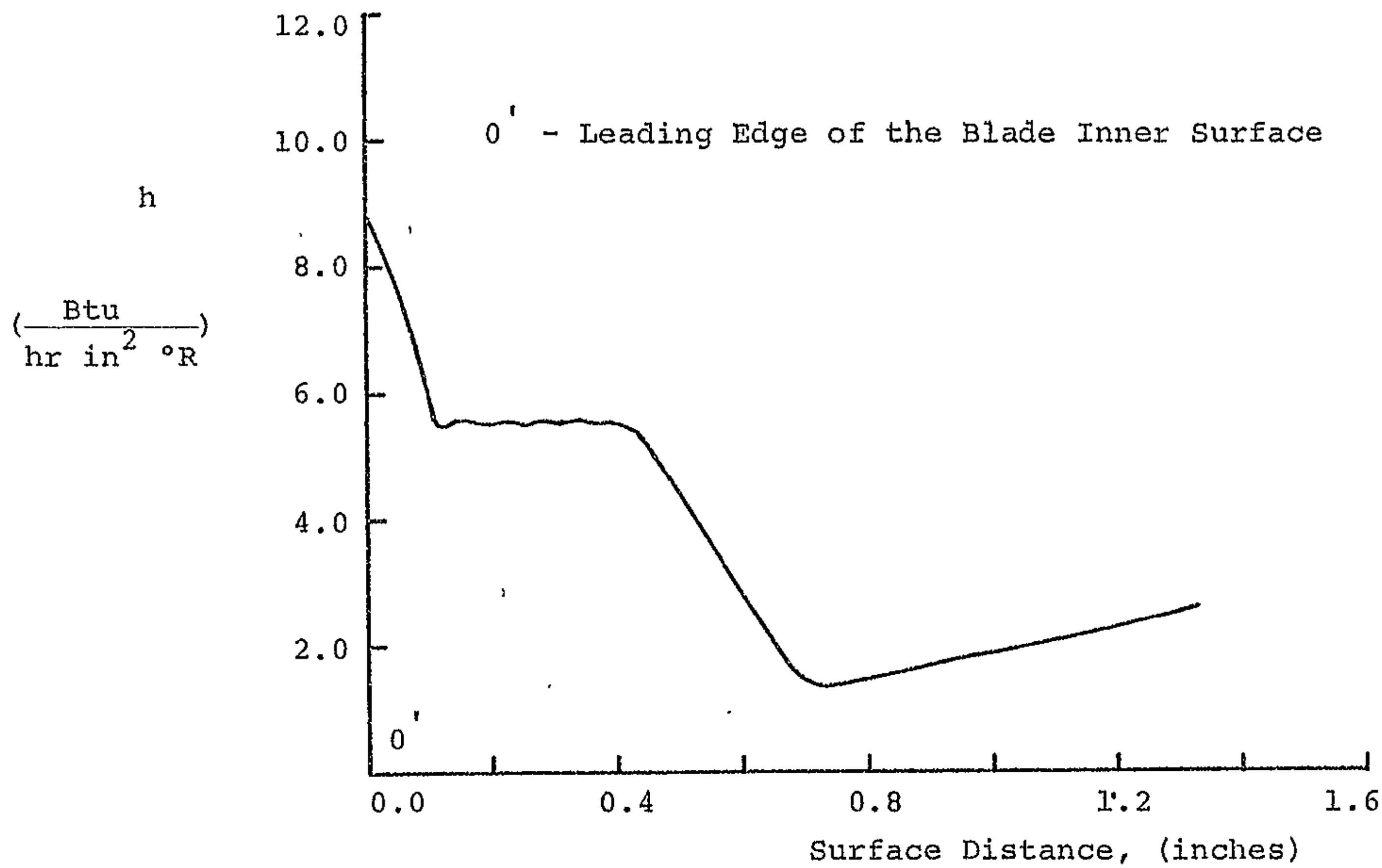


FIGURE 14. HEAT TRANSFER COEFFICIENT DISTRIBUTION ON INNER BOUNDARY WITH 3% COOLANT (CONFIGURATIONS A3, B3, C3).

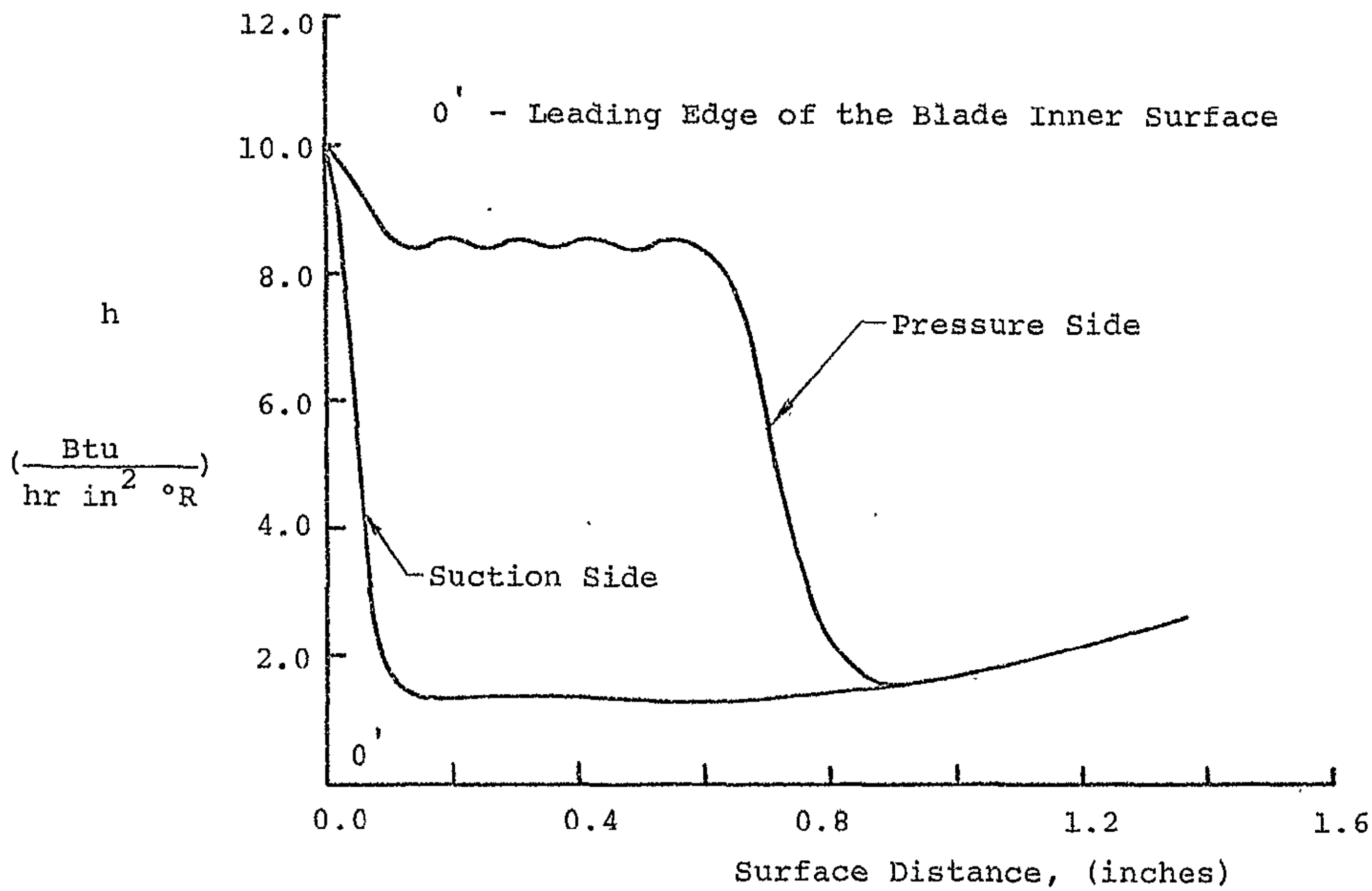


FIGURE 15. HEAT TRANSFER COEFFICIENT DISTRIBUTION ON INNER BOUNDARY WITH 3% COOLANT (CONFIGURATION B4).

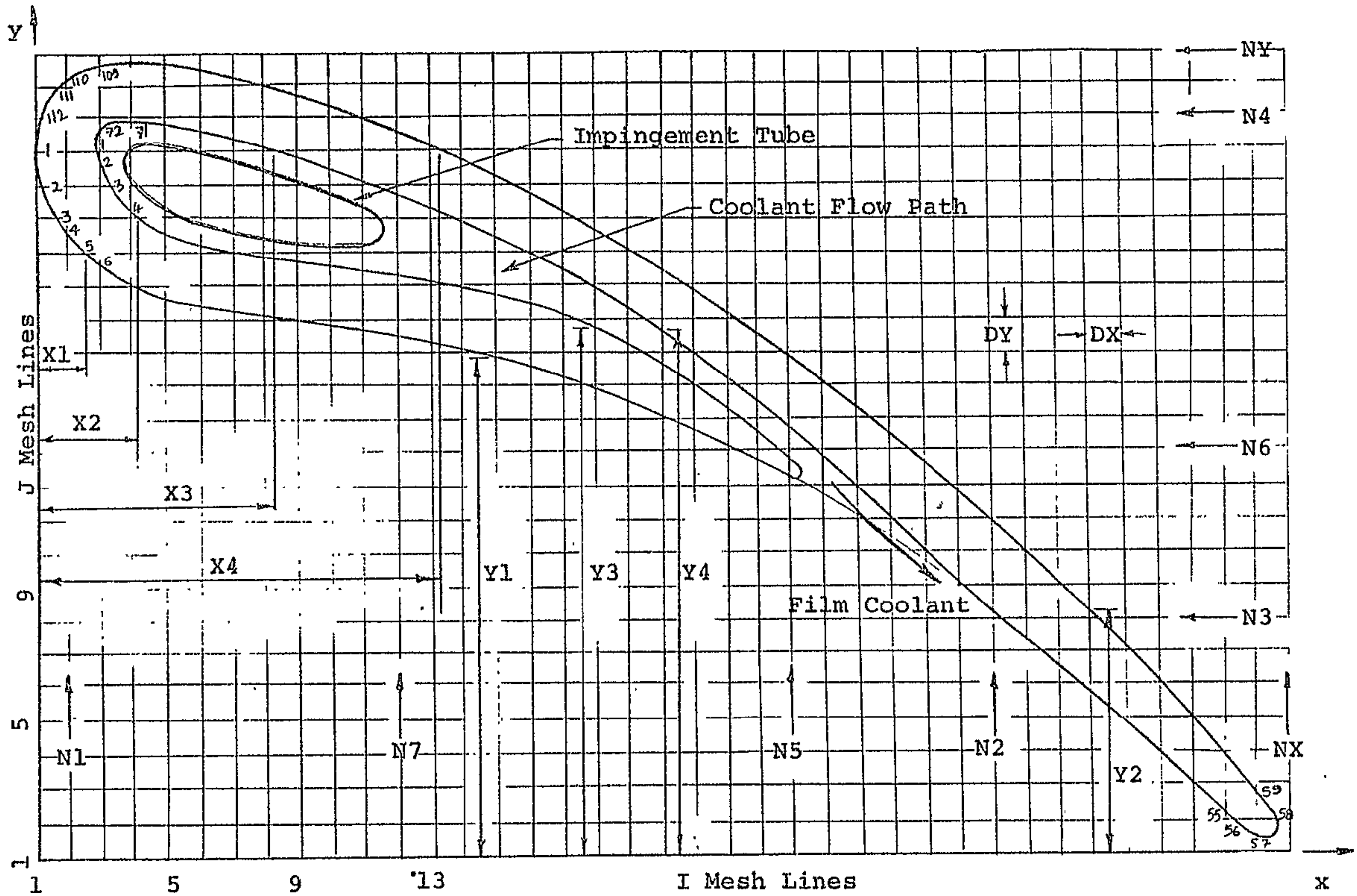


FIGURE 16. BLADE GRID NET WORK.

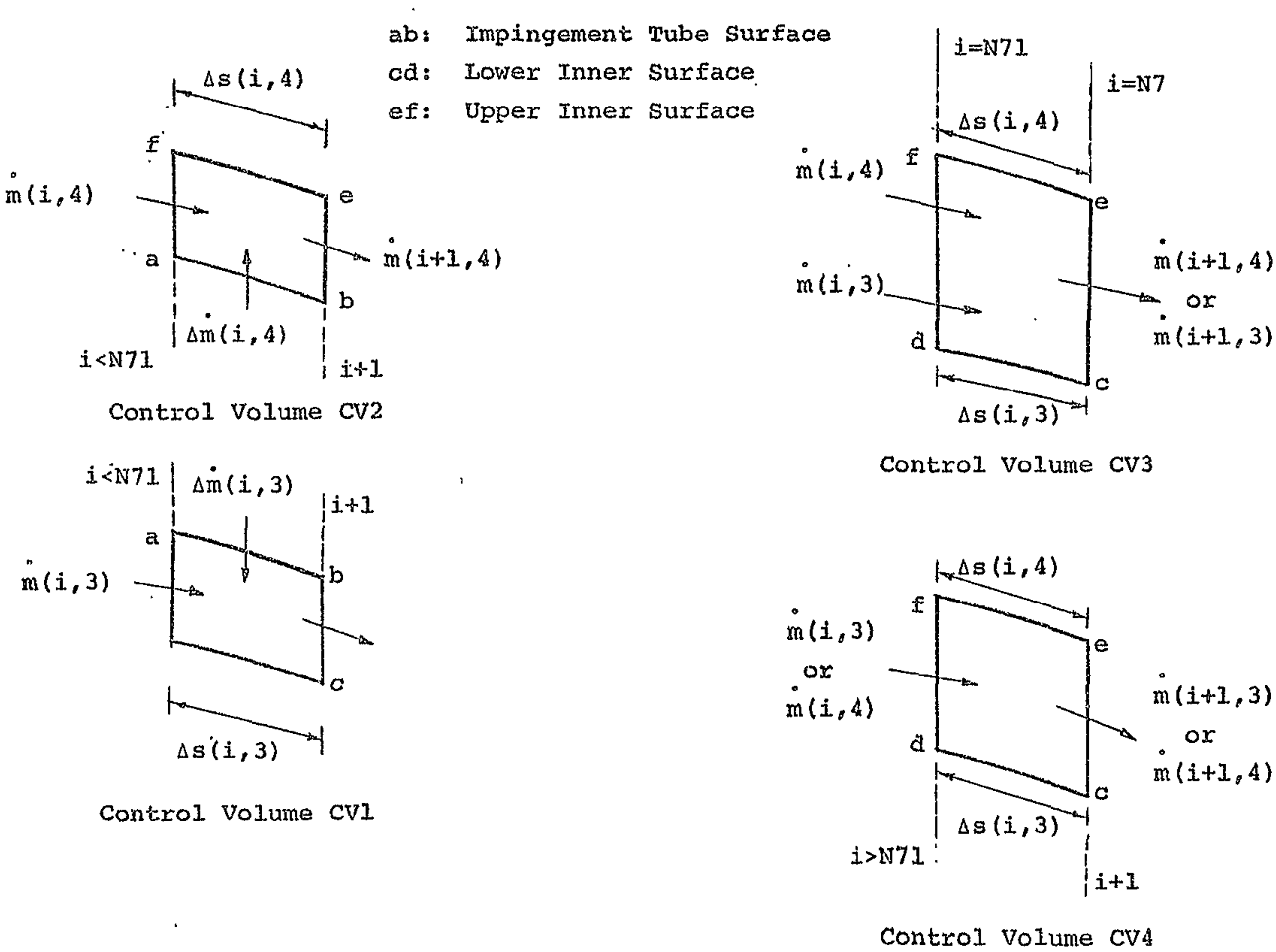
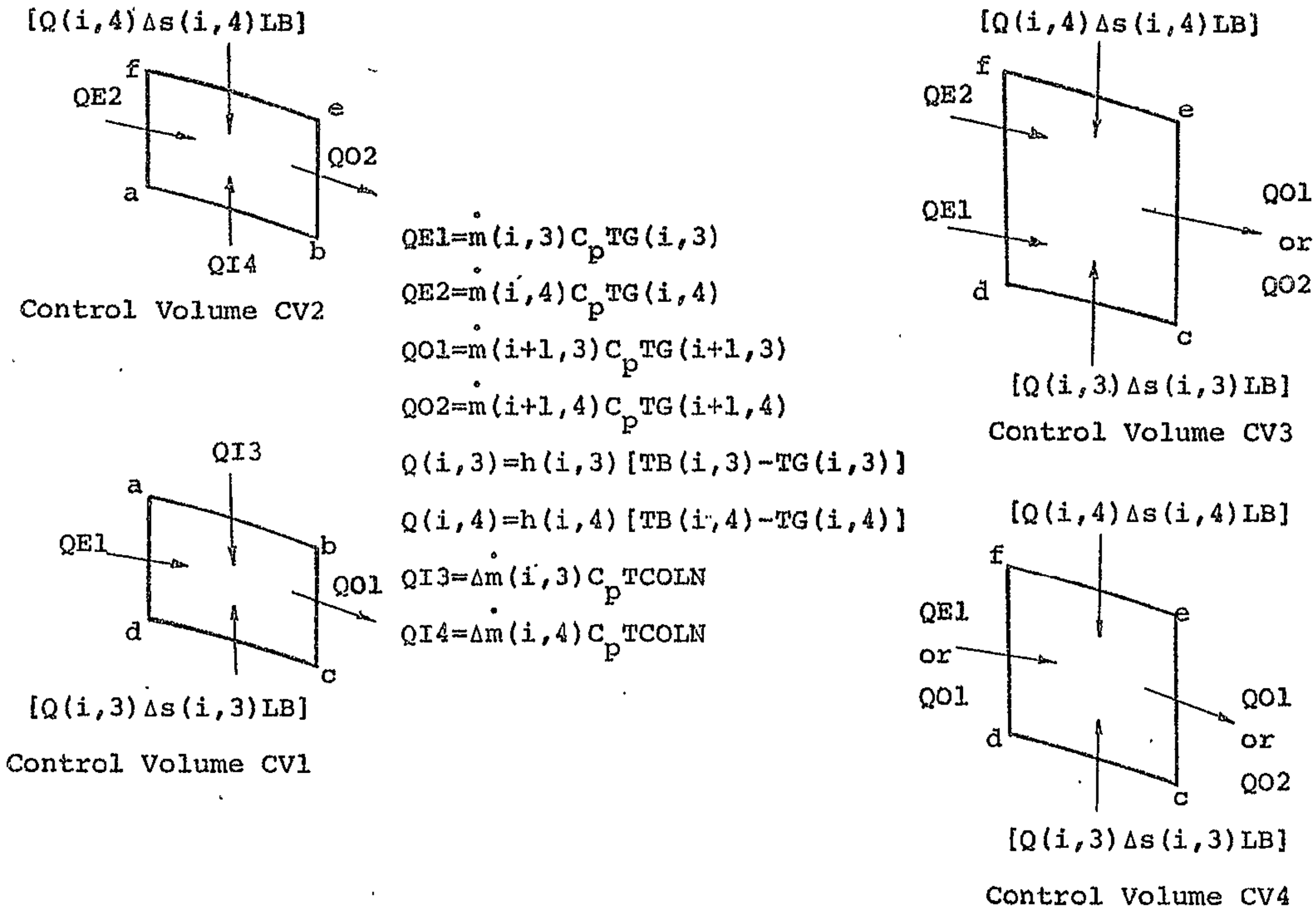


FIGURE 18. SKETCH ILLUSTRATING THE CONTROL VOLUMES FOR COOLANT MASS BALANCE.



$$QE1 = \dot{m}(i,3)\,C_p\,TG(i,3)$$

$$QE2 = \dot{m}(i,4)\,C_p\,TG(i,4)$$

$$QO1 = \dot{m}(i+1,3)\,C_p\,TG(i+1,3)$$

$$QO2 = \dot{m}(i+1,4)\,C_p\,TG(i+1,4)$$

$$Q(i,3) = h(i,3)\,[TB(i,3) - TG(i,3)]$$

$$Q(i,4) = h(i,4)\,[TB(i,4) - TG(i,4)]$$

$$QI3 = \Delta\dot{m}(i,3)\,C_p\,TCOLN$$

$$QI4 = \Delta\dot{m}(i,4)\,C_p\,TCOLN$$

FIGURE 19. SKETCH ILLUSTRATING THE CONTROL VOLUMES FOR DERIVING HEAT BALANCE EQUATIONS.

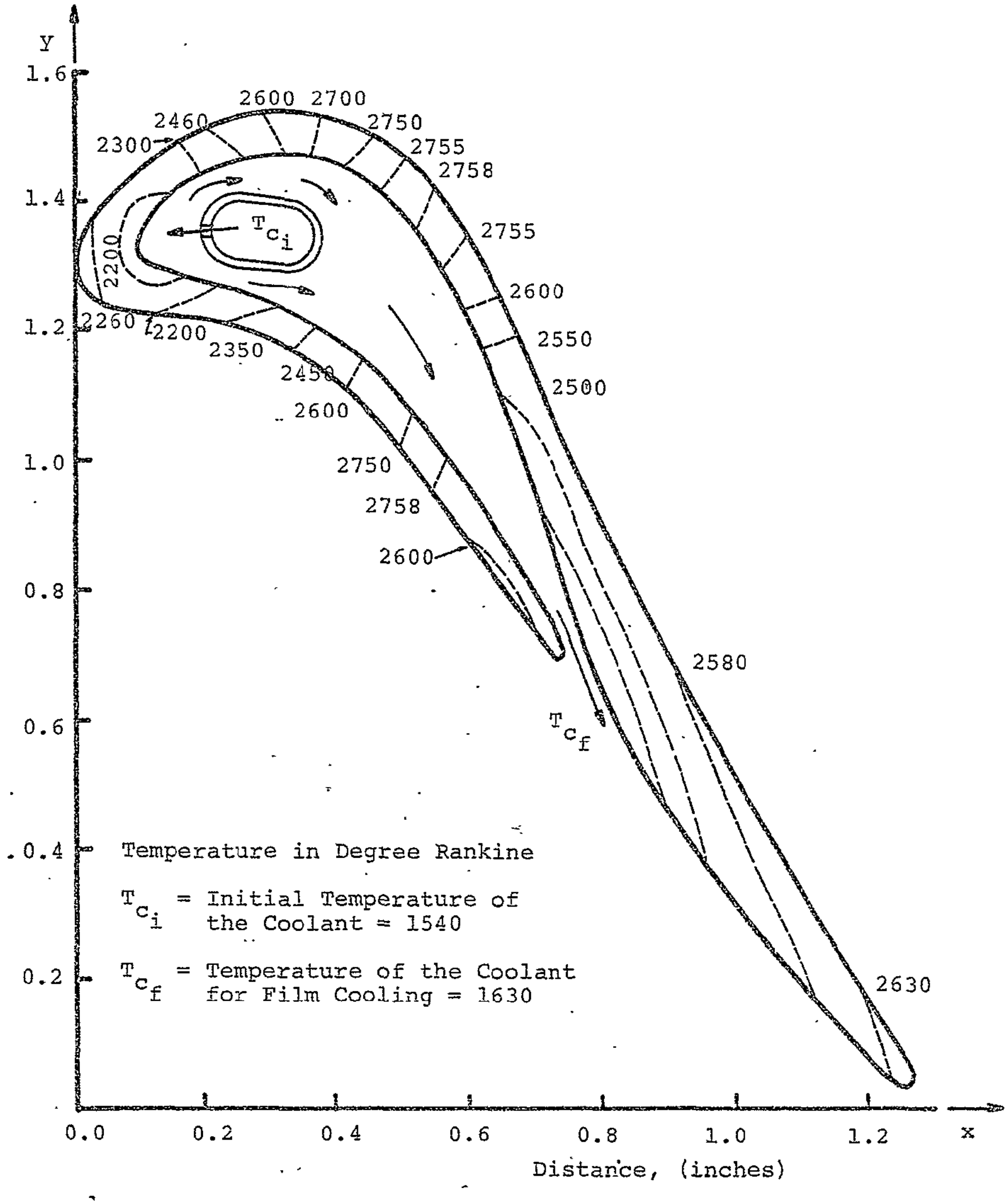


FIGURE 20. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION A1).

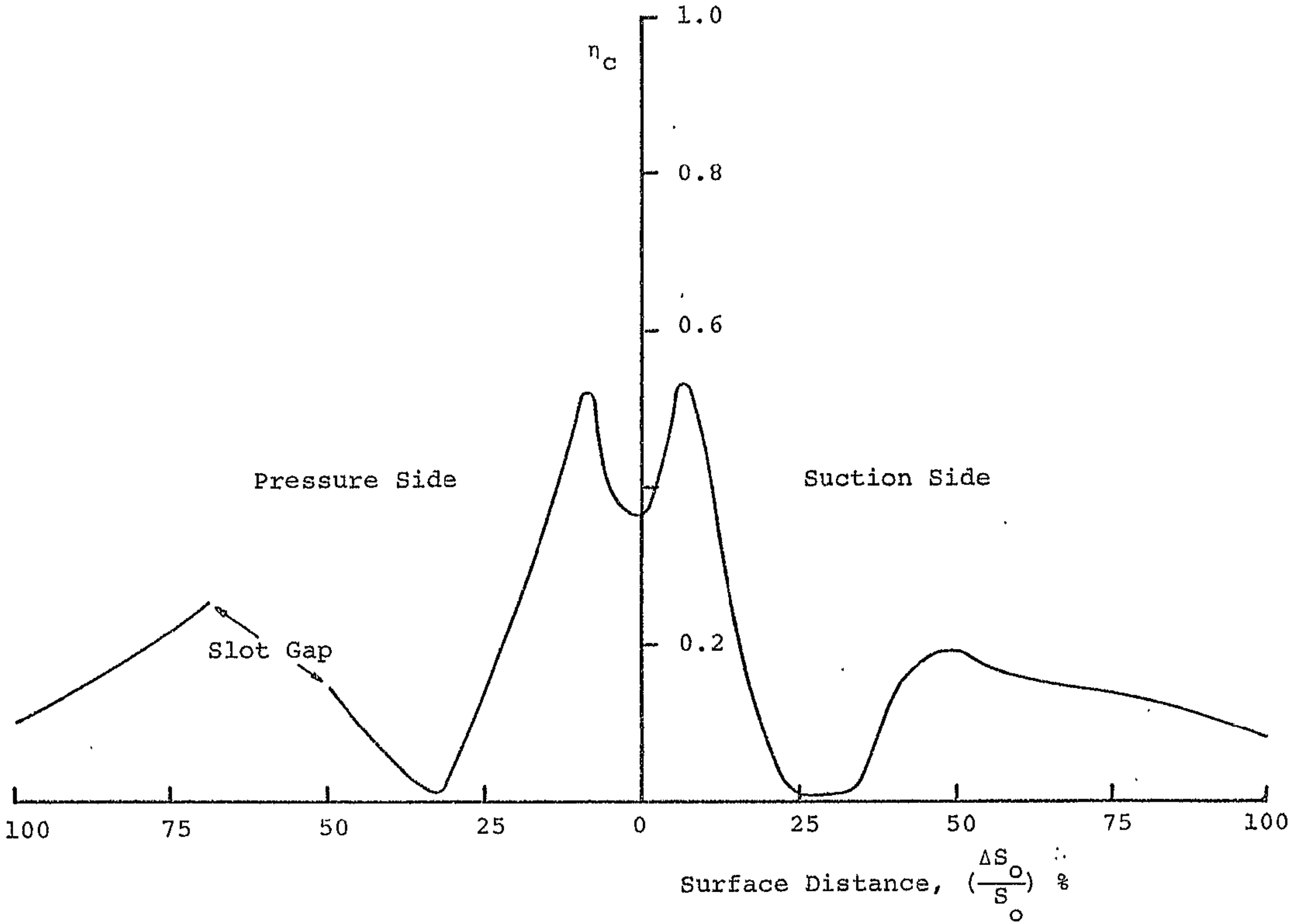


FIGURE 21. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION A1).

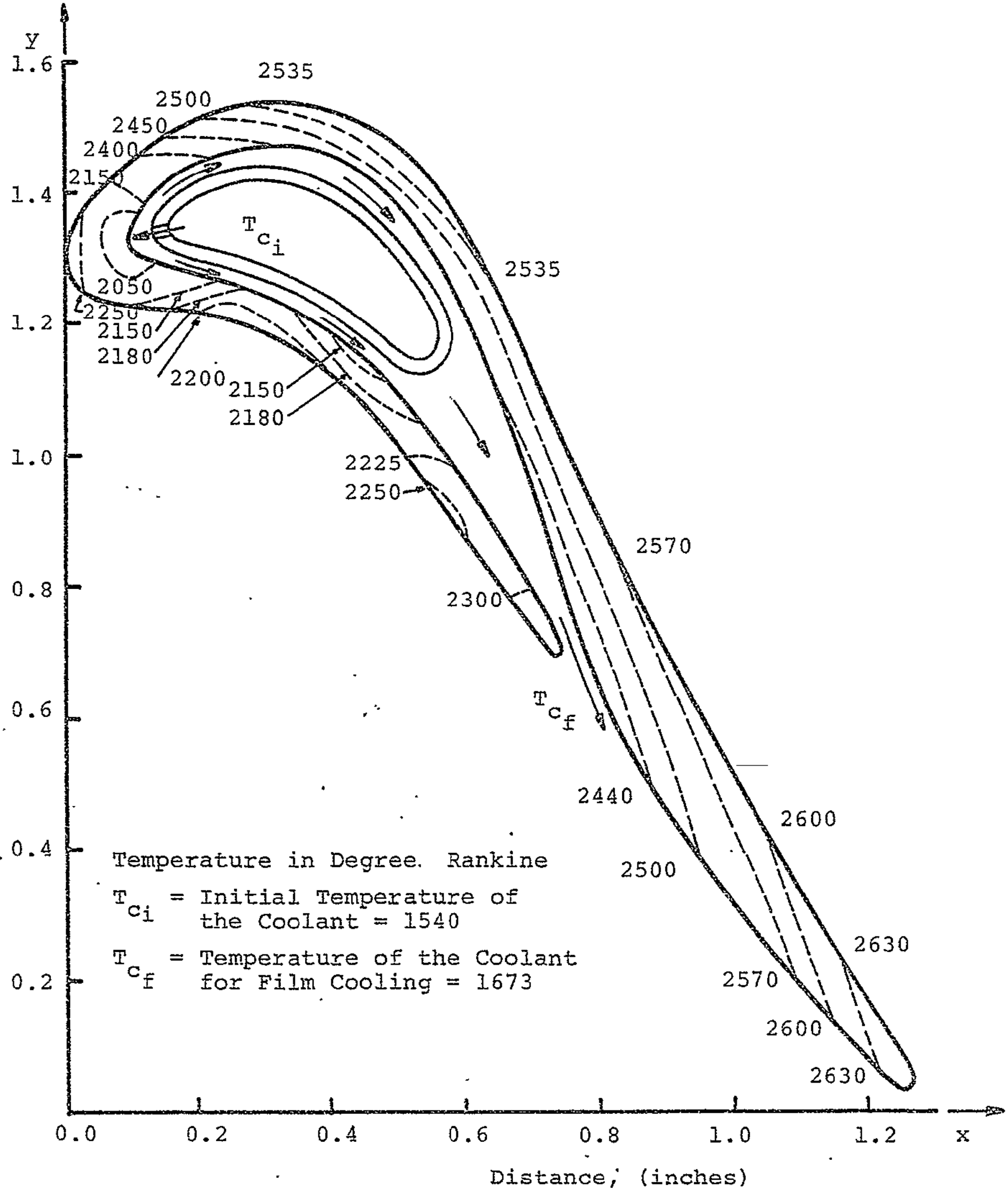


FIGURE 22. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION A2).

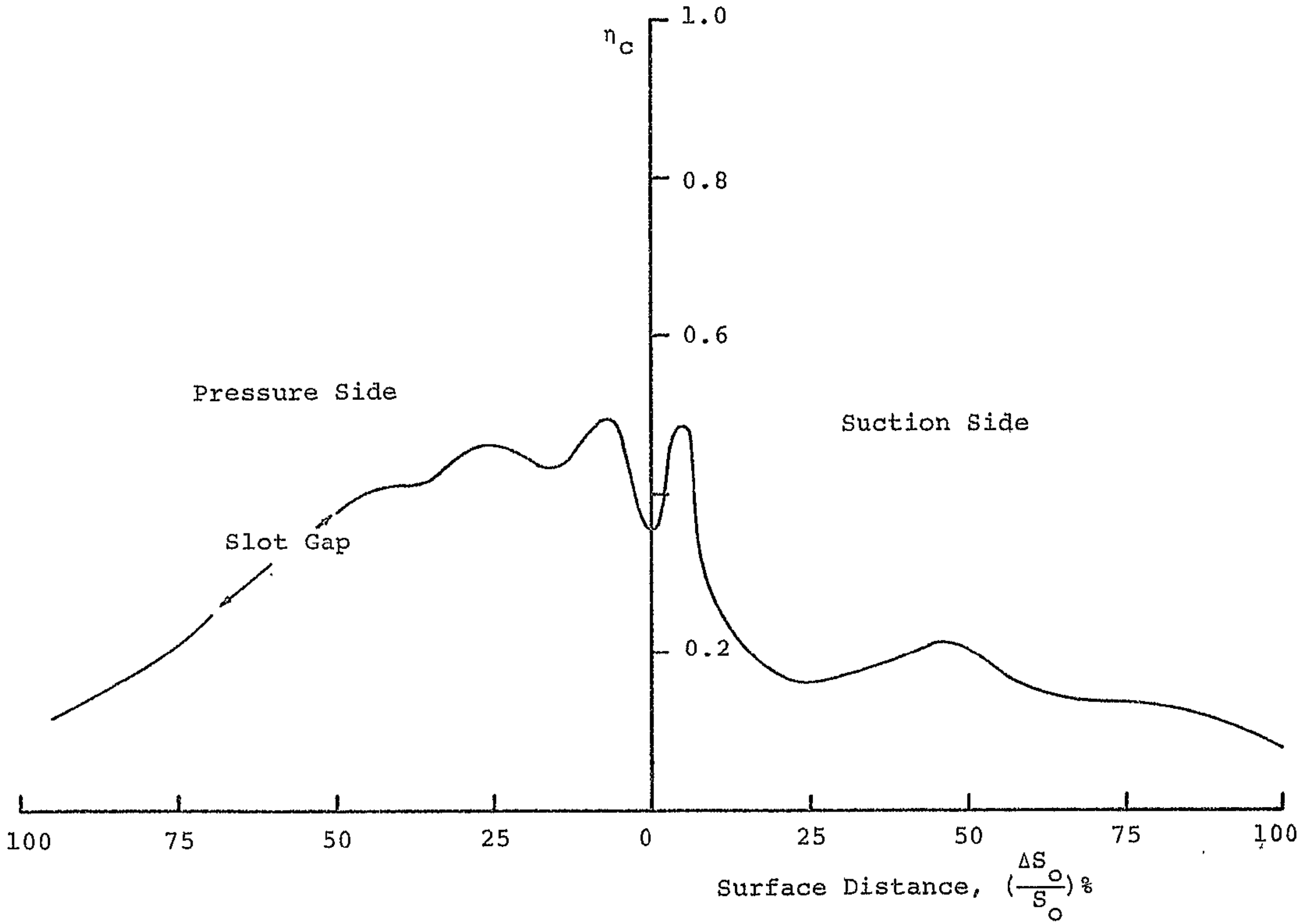


FIGURE 23. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION A2).

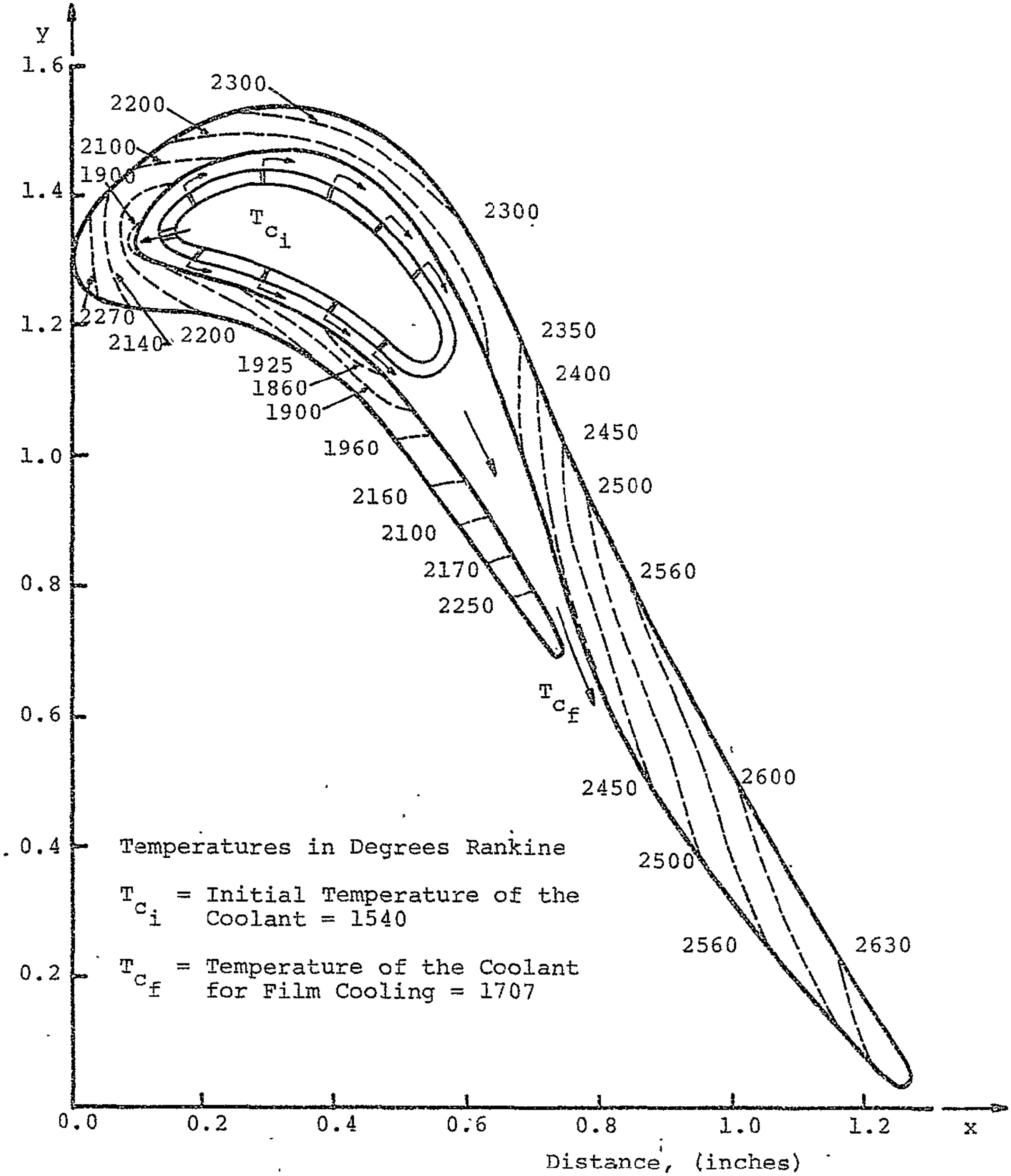


FIGURE 24. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION A3).

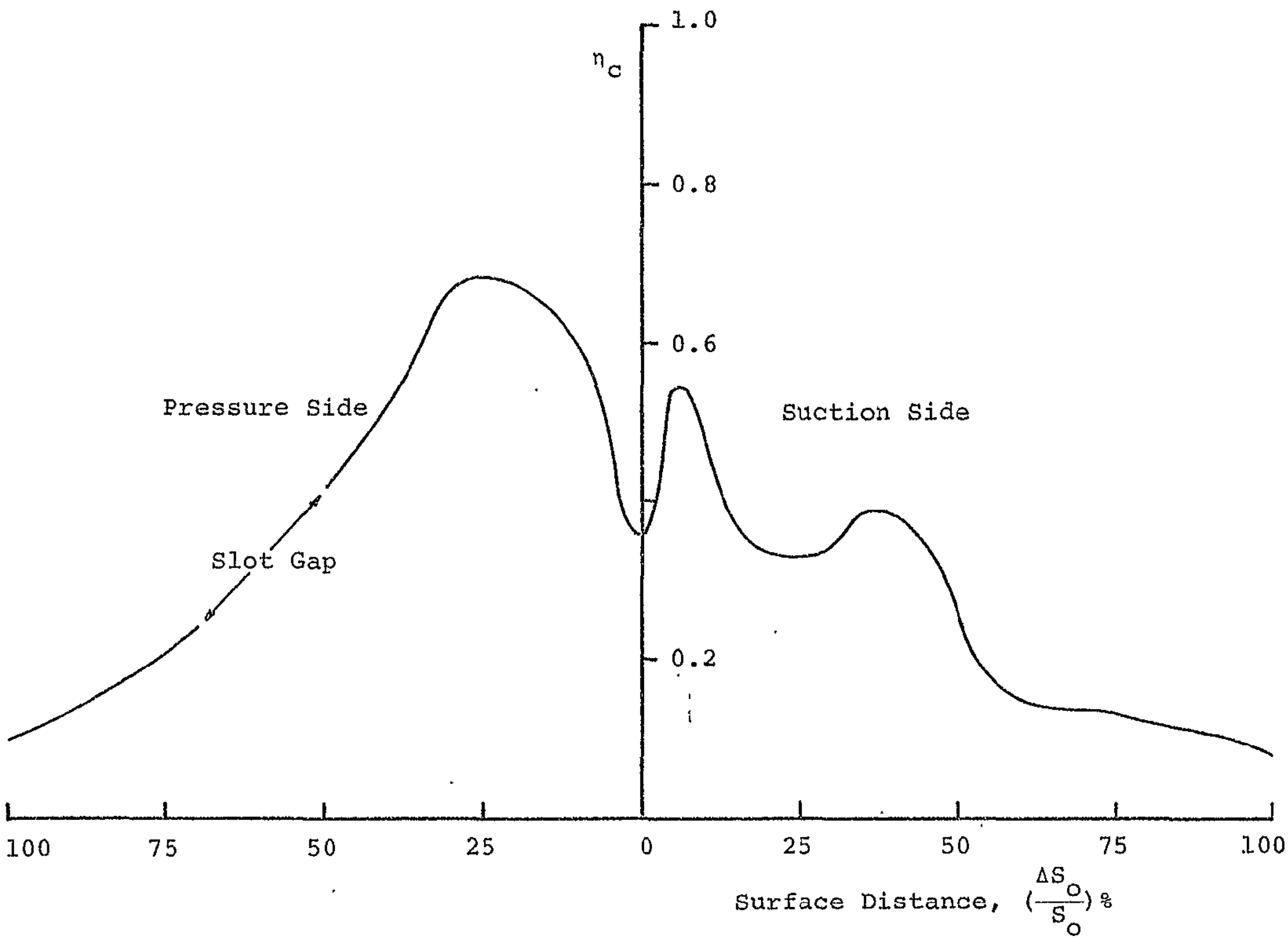


FIGURE 25. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION A3).

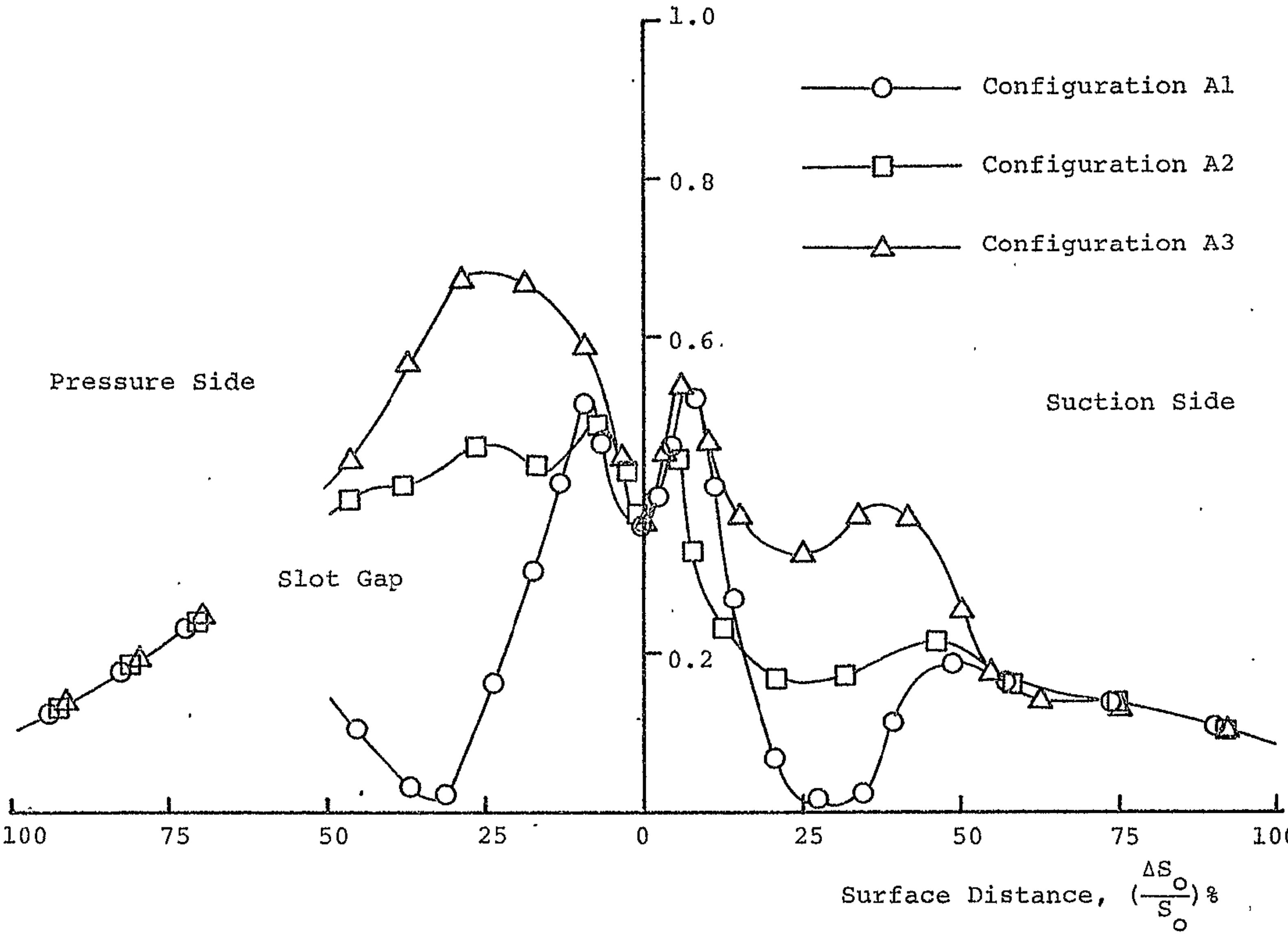


FIGURE 26. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT FOR DIFFERENT IMPINGING CONFIGURATIONS.

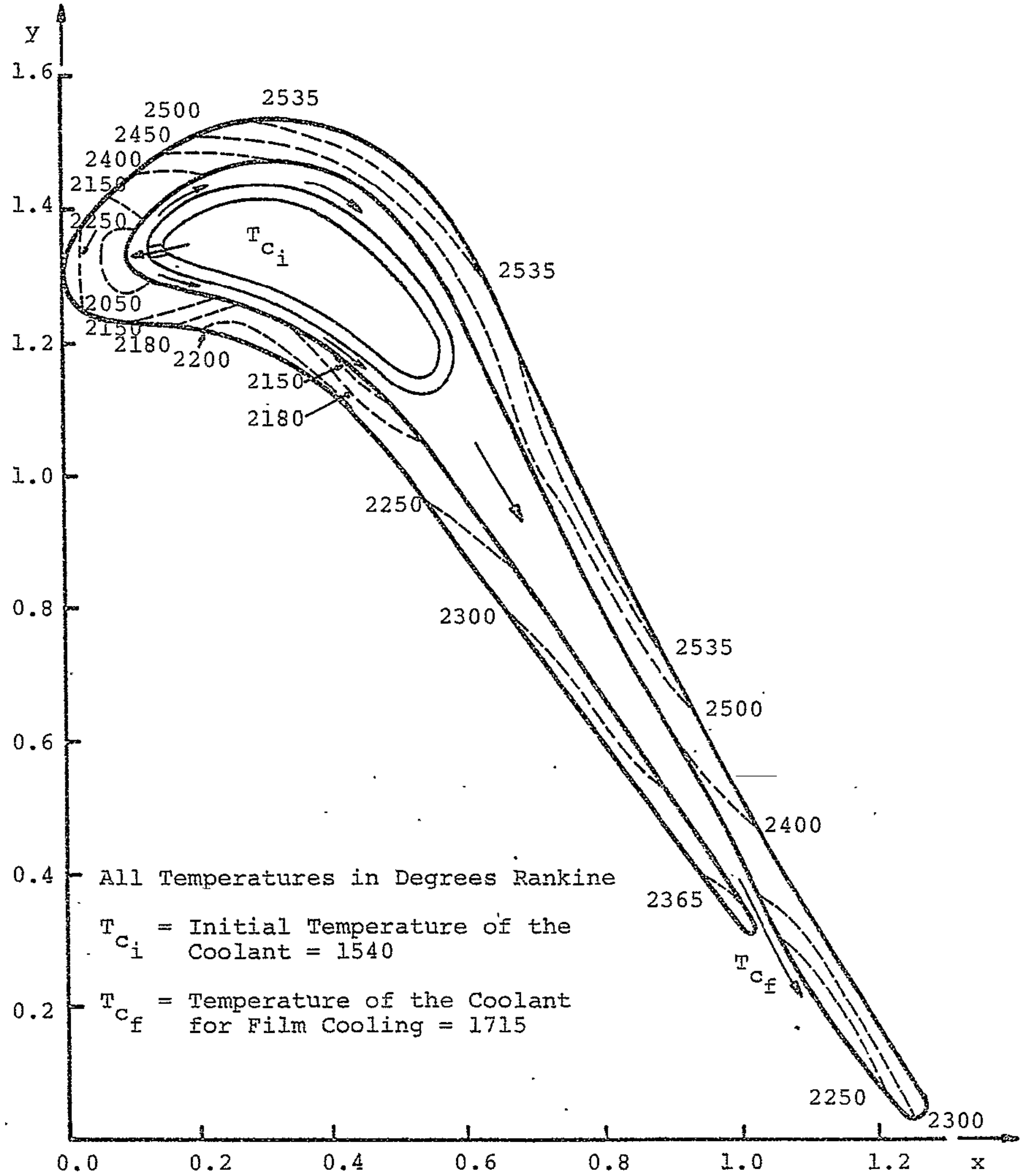


FIGURE 27. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION B2).

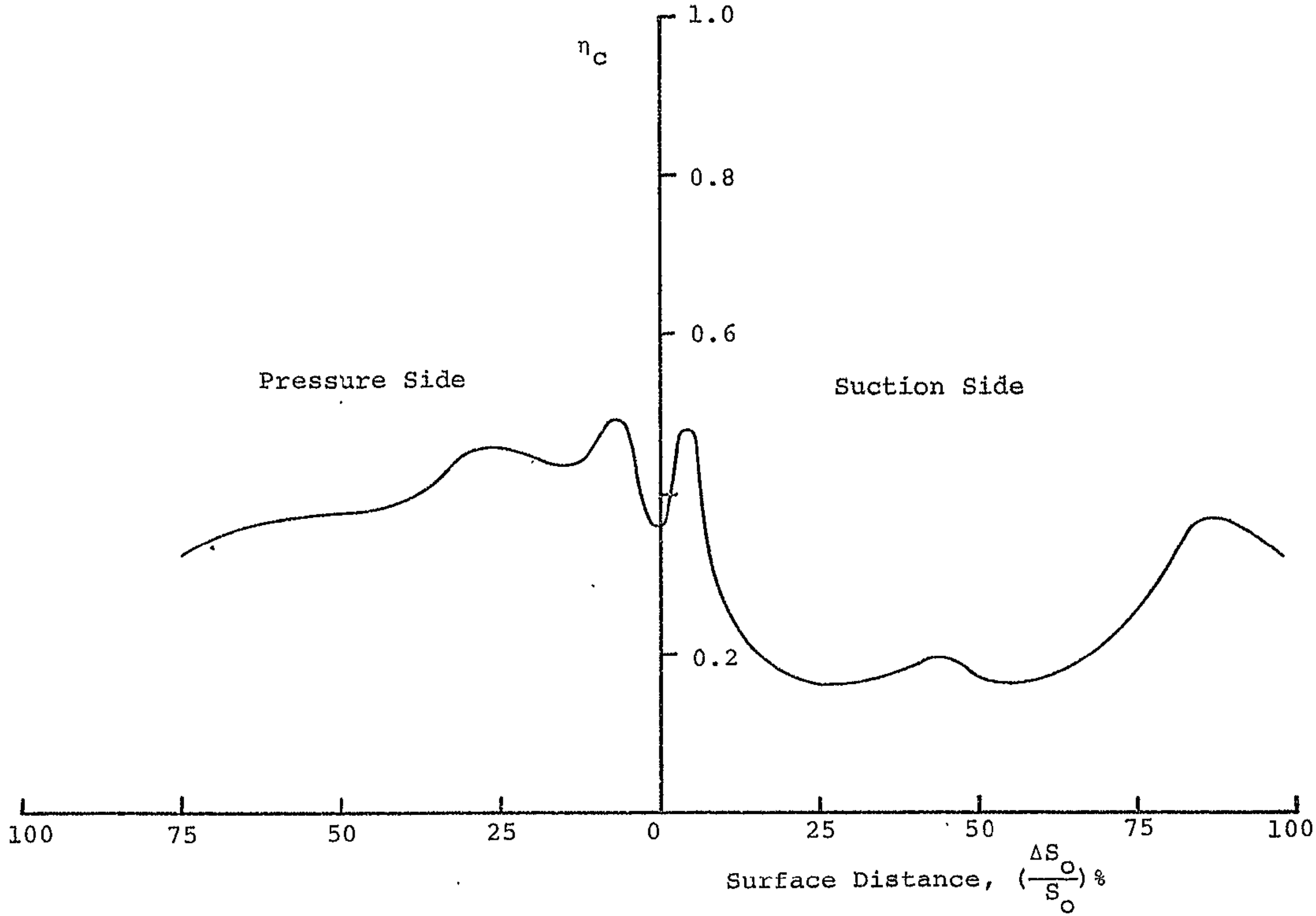


FIGURE 28. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION B2).

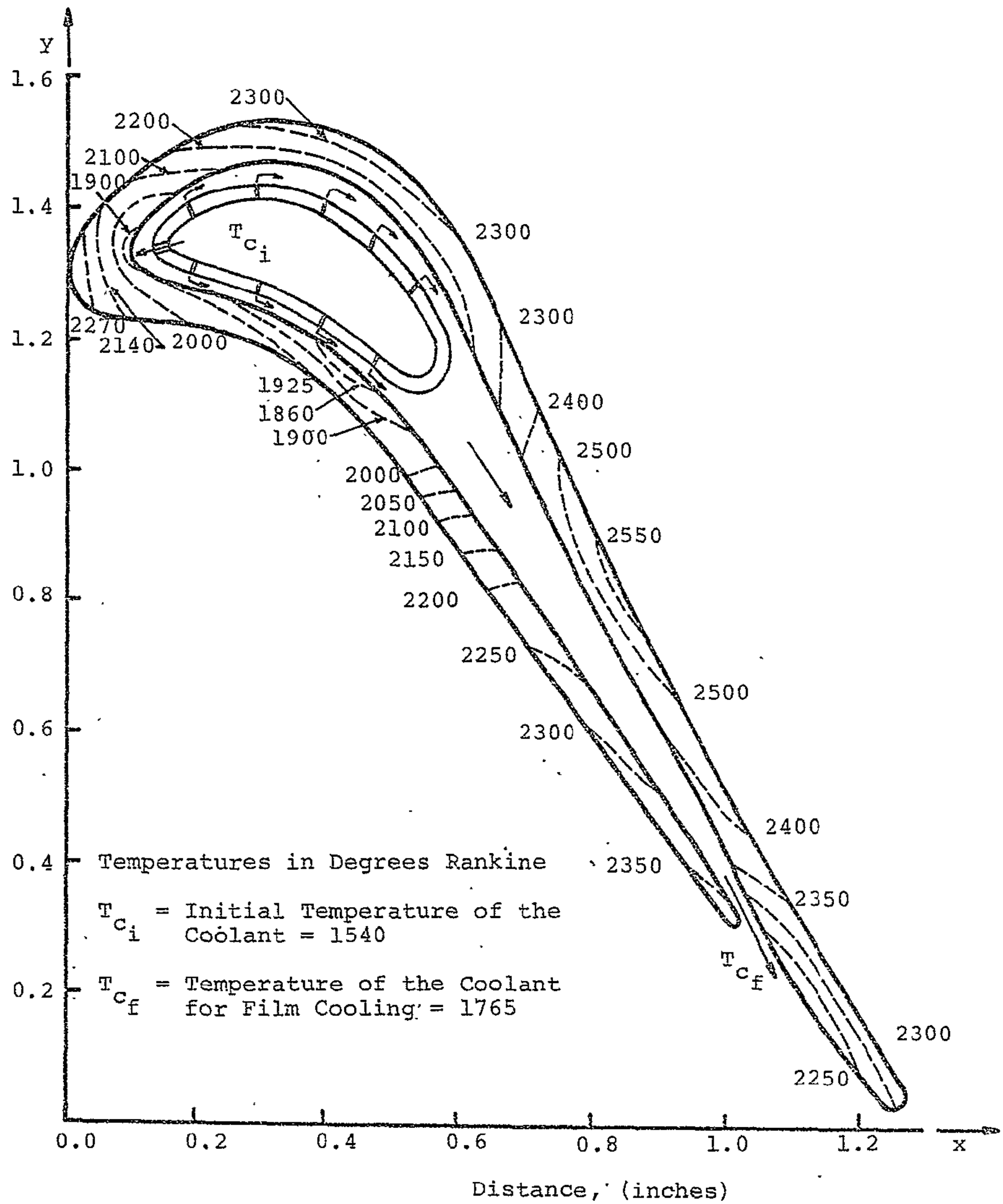


FIGURE 29. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION B3).

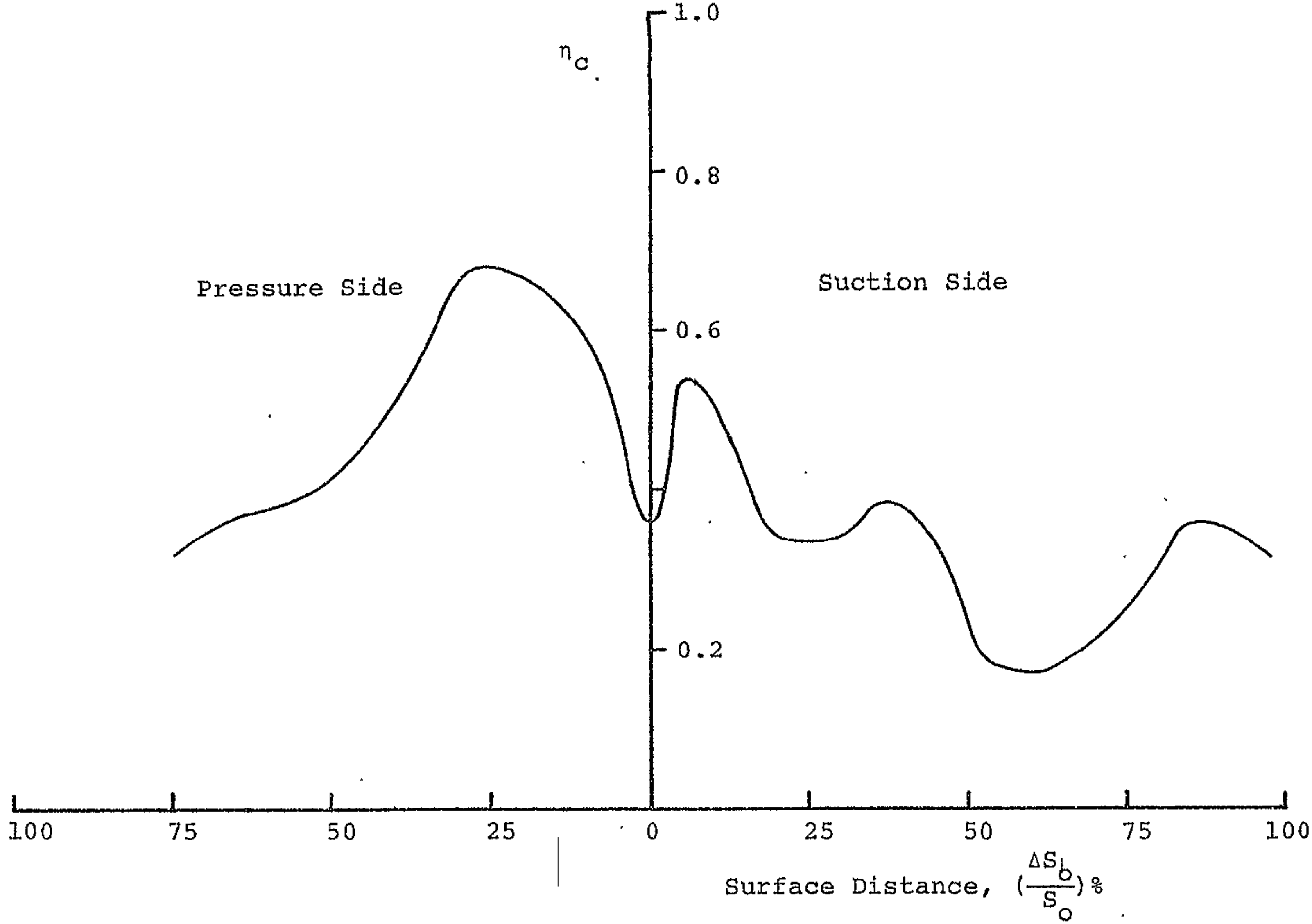


FIGURE 30. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION B3)

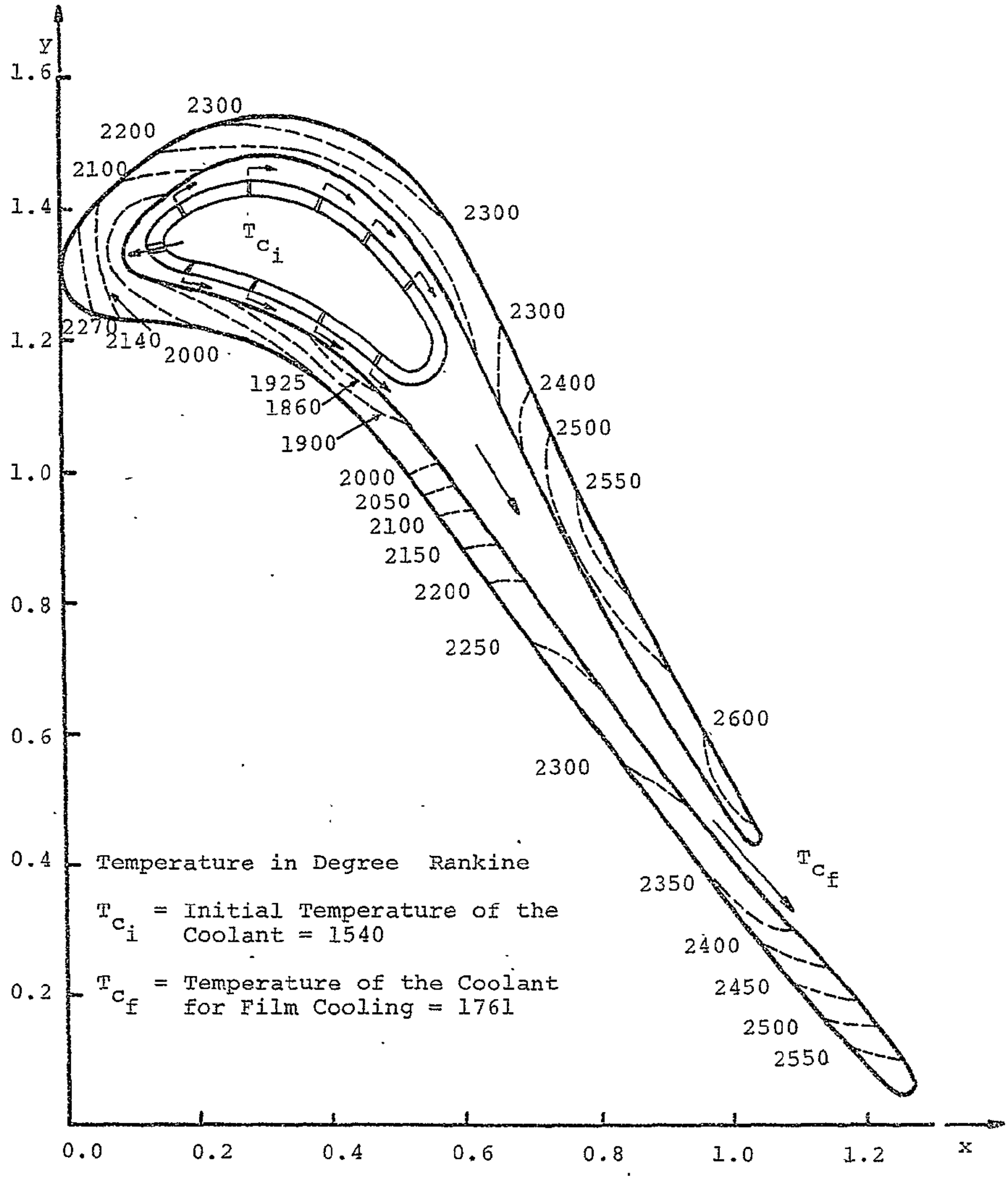


FIGURE 31. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION C3).

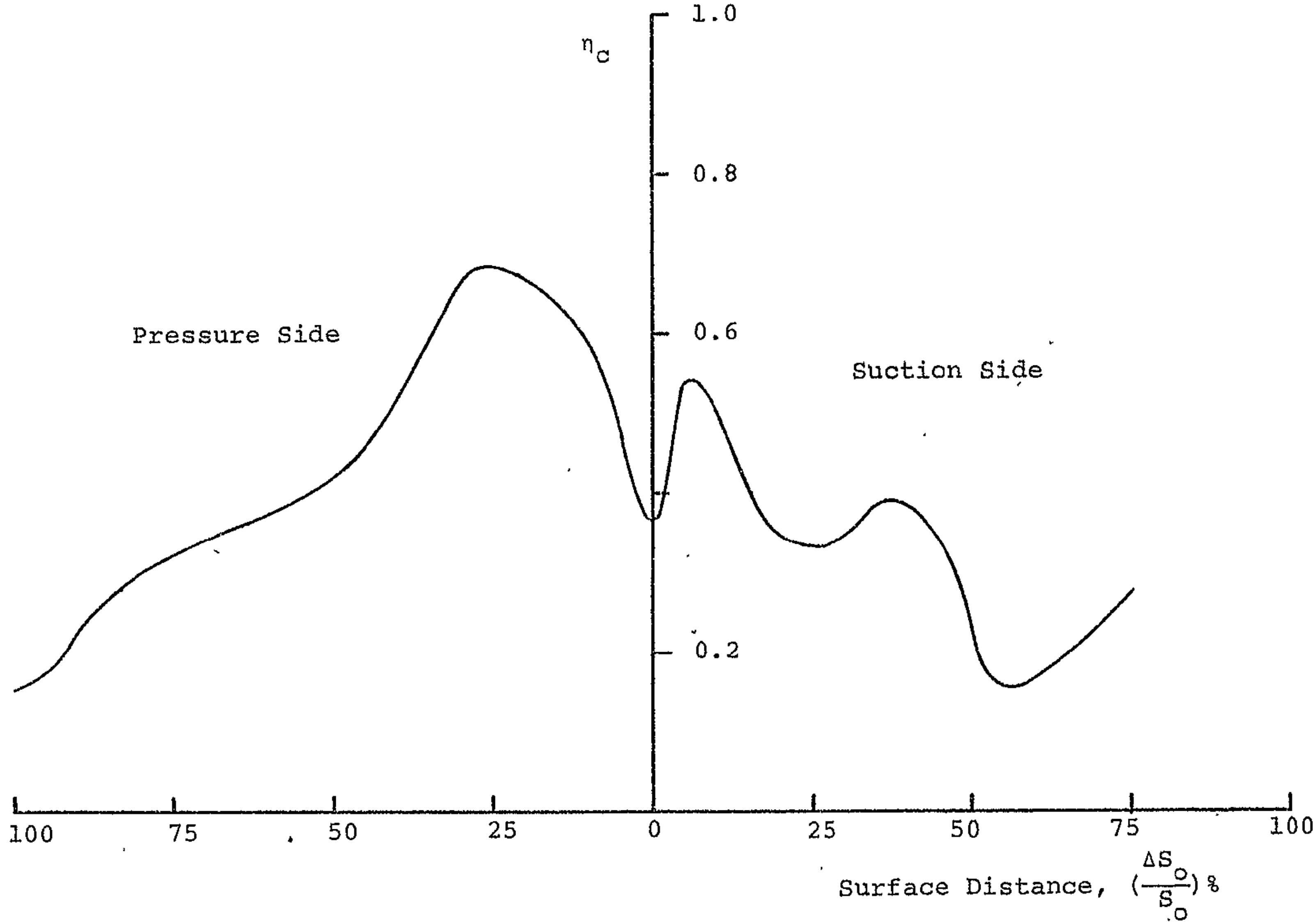


FIGURE 32. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION C3).

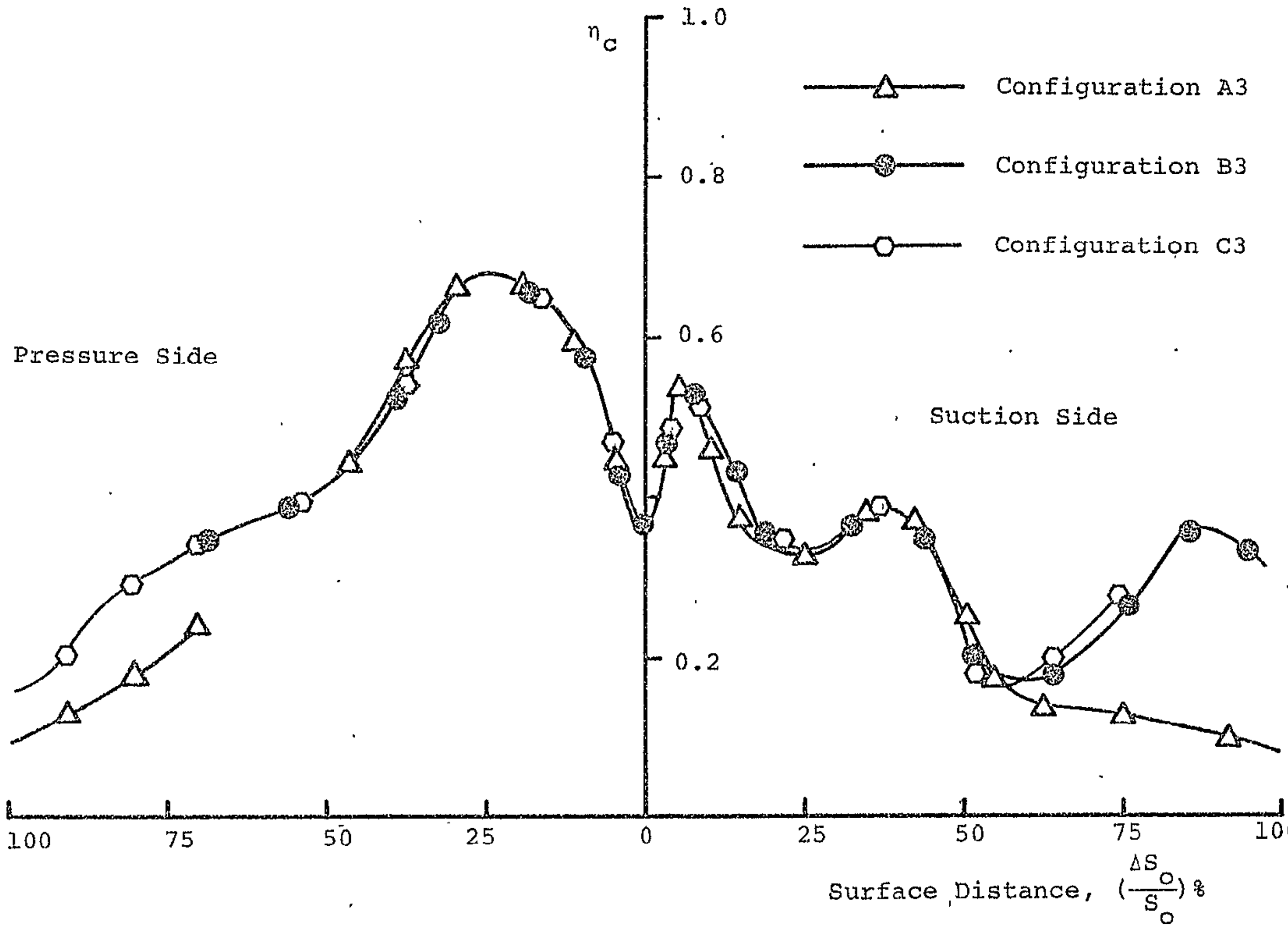


FIGURE 33. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT FOR DIFFERENT SLOT POSITIONS.

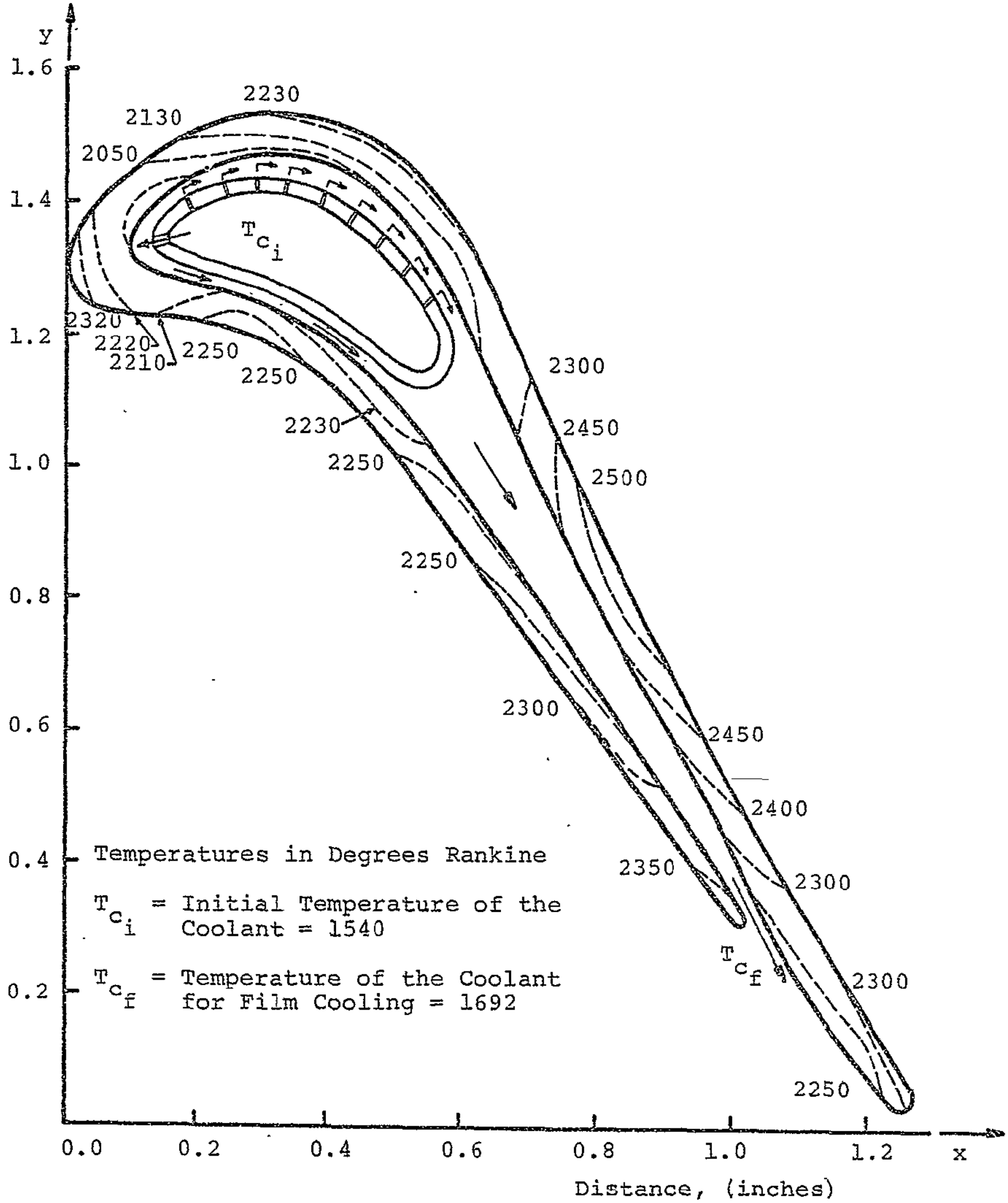


FIGURE 34. NOZZLE VANE TEMPERATURE DISTRIBUTION WITH 3% COOLANT (CONFIGURATION B4).

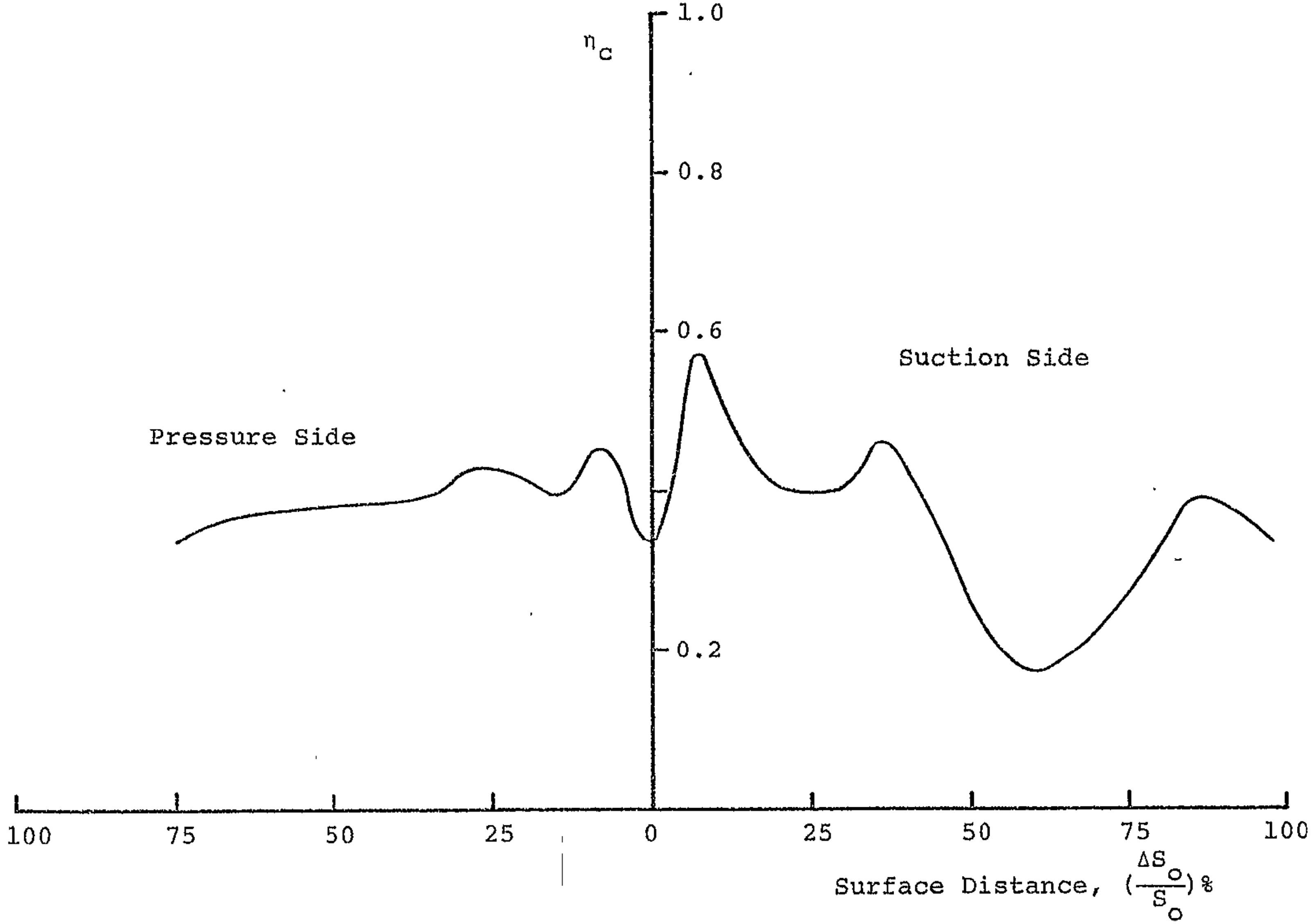


FIGURE 35. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATION B4).

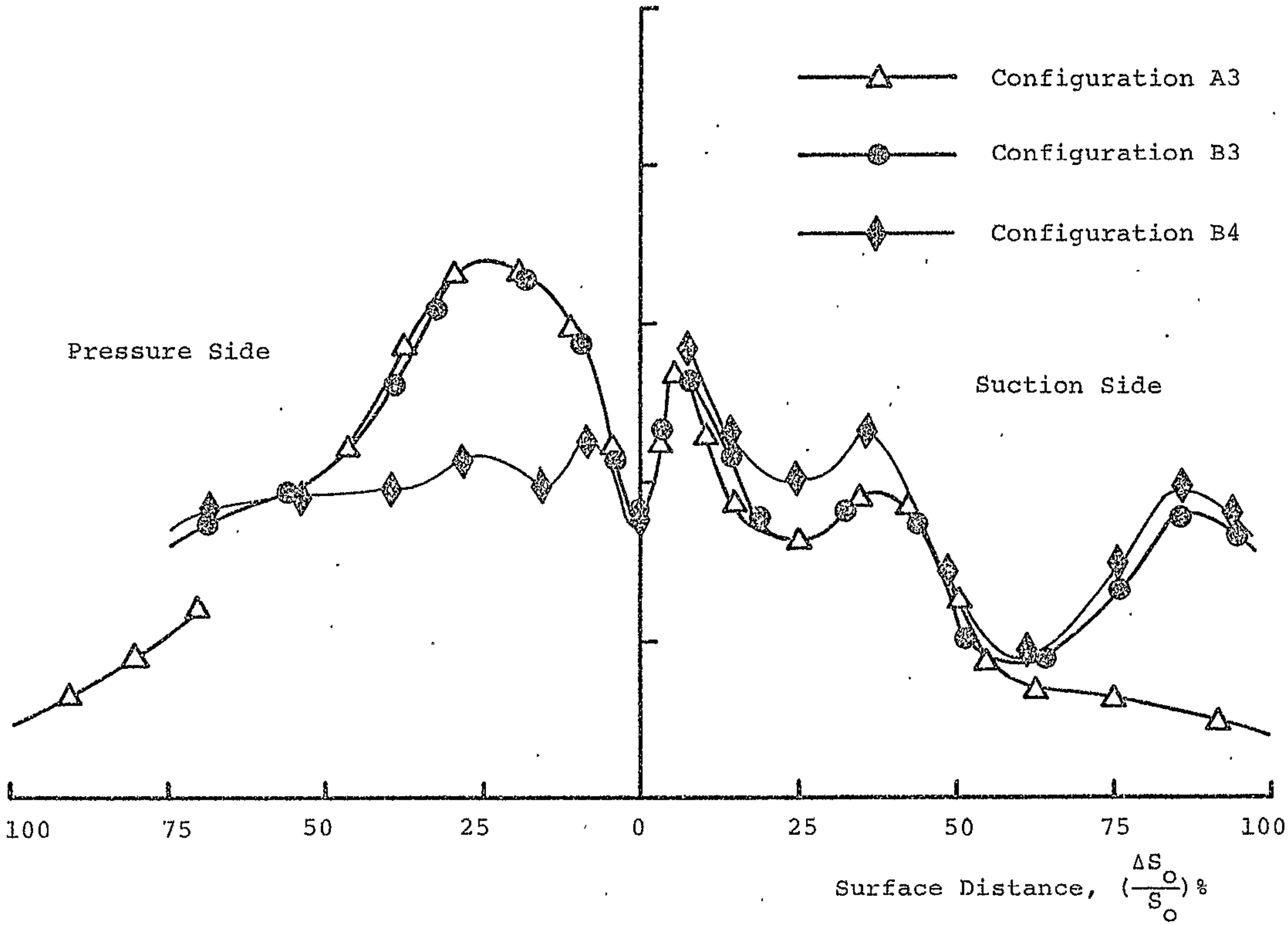


FIGURE 36. EFFECTIVENESS OF COOLING ON BLADE SURFACES WITH 3% COOLANT (CONFIGURATIONS A3, B3, B4).

APPENDIX A

PROGRAM LISTING AND SAMPLE OUTPUT

```
C     FORTRAN COMPUTER PROGRAM TO OBTAIN TEMPERATURE DISTRIBUTION
C     IN A BLADE USING THE FINITE DIFFERENCE METHOD
C
C
C                  ** MAIN PROGRAM NO.1 **
C
C     FOR COOLANT DISCHARGE ON LOWER OUTER SURFACE USE
C     MAIN PROGRAM NO.1 ALONG WITH ALL THE SUBROUTINES
C
C
      DIMENSION TGTEM(30),TGXTEM(65)
      DIMENSION XISO(30,30),YISO(30,30),TSOLT(30),NNY(30)
      DIMENSION DELS(30,4),MASC(30,4),Q(30,4),DELM(30,4)
      DIMENSION T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      DIMENSION VELG(30,65)
      DIMENSION EFF(30,65)
      REAL  LB,MASC
      INTEGER*4 TSOLT
      COMMON T,TB,TBX,TG,TGX,H,HX,Y1,Y2,Y3,Y4,X1,X2,X3,X4,BX,BY,BXI,BYI
     1,IP
      COMMON DX,DY,NX,NY,NXO,NXI,ITER,N1,N2,N3,N4,NP,N5,N6,N7
      COMMON/DAT/   NX1,NY1,N11,N21,N31,N41,N51,N52,N61,N62,N71,NSR,
     1DELM,MASC,DELS,XK,CME,SUMM,IMAX,NTE,TCOLN,LB,CP
      COMMON/TEM/  VELG,PRNO,TEMPC,CPC,VISCC,ALPHA,FLOC,DENSC
      COMMON/ISO/ XISO,YISO,TSOLT,NSOLT,NNY
      CALL DATA
  245 CONTINUE
C
C     TEMPORARY STORAGE FOR GAS TEMPS DOWNSTREAM OF SLOT
C
      DO 250  I=N52,NX1
      TGTEM(I)=TG(I,1)
  250 CONTINUE
      DO 255  J=2,N6
      TGXTEM(J)=TGX(1,J)
  255 CONTINUE
C
C  CALCULATES ADIABATIC WALL TEMPERATURES DOWNSTREAM OF SLOT
C
      DO 260  I=N52,NX1
      CALL TEMAD(I,1,TAD)
  260 TG(I,1)=TAD
      DO 265  J=2,N6
      CALL TEMAD(1,J,TAD)
  265 TGX(1,J)=TAD
      DO 266  I=N52,N21
      TG(I,4)=TG(I,1)
  266 CONTINUE
```

```
      DC 267  J=N31,N6
      TGX(4,J)=TGX(1,J)
 267  CCNTINUE
 300  CCNTINUE
C
C   STARTING OF GAUSS-SEIDELL METHCD CF ITERATION ON I LINES
C
      SUM=0.
      DC 510 I=2,NX1
      IL=IFIX(Y1(I)/DY+0.0001)+2
      IH=IFIX(Y2(I)/DY+0.0001)+1
      IF(ABS(Y2(I)-FLCAT(IH-1)*CY).LT.0.00001) IH=IH-1
C
C   CUTER LCWER BCUNDARY
C
C
C   CARDS WITH NECA CCMMENT TO SPECIFY SLCPES AT SPECIAL PCINTS
C
       B2=FLCAT(I-1)*CX
       C2=Y1(I)
       CALL CUR(B2,C2,B,C,0)
       XMN=B+2.*C*B2
      IF(I.EC.NX1) XMN=0.01
       YT=FLOAT(IL-1)*CY
       X=(C2-YT)*XMN+B2
      IF(I.EC.6)  XMN=-1.0
      IF(I.GT.N51.ANC.I.LT.N2)  GC TC 352
      IF(I.GE.N2)  GC TC 305
      IF(X.GT.X3(IL)) GC TC 330
 305  IF(ABS(XMN).LE.0.015) GC TC 335
       IF(X.LE.B2)GC TC  320
       XR=B2+CX
      IF(I.GE.N2)  GC TC 310
      IF ((X3(IL)-XR).GT.C.0001) GC TC 310
      TTT=T(I,IL)+(TBX(3,IL)-T(I,IL))*(X-B2)/(X3(IL)-B2)
      GC TC 325
 310  IF((X2(IL)-XR).GT.C.0001) GC TC 315
       TTT=T(I,IL)+(TBX(2,IL)-T(I,IL))*(X-B2)/(X2(IL)-B2)
       GC TC 325
 315   CCNTINUE
      IF((X-XR).GT.0.0001)  GC TC 337
       TTT=T(I,IL)+(T(I+1,IL)-T(I,IL))*(X-B2)/CX
      GC TC 325
 320   TTT=T(I,IL)-(T(I,IL)-T(I-1,IL))*(B2-X)/CX
 325    XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GC TC 340
 330  TTT=TBX(3,IL)
      XNL=(X3(IL)-B2)*(X3(IL)-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
```

```
      GO TO 340
 335   TTT=T(I,IL)
       XNL=YT-C2
      GO TO 340
 337  TTT=T(I+1,IL)
      XNL=DX*DX+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
 340    AL=H(I,1)/XK
       TNEW=(XNL*AL*TG(I,1)+TTT)/(1.+XNL*AL)
 350  CONTINUE
      TNEW=(1.-OME)*TB(I,1)+OME*TNEW
       SUM=SUM+(TB(I,1)-TNEW)*(TB(I,1)-TNEW)
       TB(I,1)=TNEW
 352  CONTINUE
      IF(I.GT.N1.AND.I.LT.N2) GO TO 360
C
C
C   INTERIOR POINTS STARTING FROM LOWER END FOR I LINES WHICH DO NOT CUT
C   THE INNER BOUNDARY
C
      DO 355 J=IL,IH
      CALL MESH(I,J,TNEW)
      TNEW=(1.-OME)*T(I,J)+OME*TNEW
      SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 355  T(I,J)=TNEW
      GO TO 480
C
C
C     FOR I LINES INTERSECTING INNER BOUNDARY-ALL INTERIOR POINTS FROM
C   LOWER SURFACE OF OUTER BOUNDARY TO LOWER SURFACE OF INNER BOUNDARY
C
 360   CONTINUE
      ILI=IFIX(Y3(I)/DY+0.0001)+1
      IF(ABS(Y3(I)-FLOAT(ILI-1)*DY).LT.0.00001) ILI=ILI-1
      IHI=IFIX(Y4(I)/DY+0.0001)+2
      IF(I.GT.N51.AND.I.LT.N2)  GO TO 375
      IF(IL.GT.ILI)  GO TO 375
      DO 365 J=IL,ILI
      CALL MESH(I,J,TNEW)
      TNEW=(1.-OME)*T(I,J)+OME*TNEW
      SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 365  T(I,J)=TNEW
C
C   INNER BOUNDARY - LOWER SURFACE
 375  B2=FLOAT(I-1)*DX
      C2=Y3(I)
      CALL CUR(B2,C2,B,C,1)
      XMN=B+2.*C*B2
      IF(I.EQ.N11)  XMN=-1.0
      YT=FLOAT(ILI-1)*DY
```

```
      X=(C2-YT)*XMN+B2
      IF(I.EQ.7)  XMN=-1.0
      IF(I.GT.N51.AND.I.LT.N2) GO TO 412
      IF(X.LT.X1(ILI))  GO TO 380
      IF((C2-YT).GE.(C2-Y1(I))) GO TO 380
      GO TO 382
 380  TTT=TBX(1,ILI)
      XNL=(C2-YT)*(C2-YT)+(X1(ILI)-B2)*(X1(ILI)-B2)
      XNL=SQRT(XNL)
      GO TO 400
 382  IF(ABS(XMN).LE.0.015) GO TO 395
      IF(X.LE.B2)  GO TO 383
      TTT=T(I,ILI)+(T(I+1,ILI)-T(I,ILI))*(X-B2)/DX
      GO TO 390
 383  CONTINUE
      XL=B2-DX
      IF((XL-X1(ILI)).GT.0.0001)  GO TO 385
      TTT=T(I,ILI)-(T(I,ILI)-TBX(1,ILI))*(B2-X)/(B2-X1(ILI))
      GO TO 390
 385  TTT=T(I,ILI)-(T(I,ILI)-T(I-1,ILI))*(B2-X)/DX
 390  XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 400
 395  TTT=T(I,ILI)
      XNL=C2-YT
 400  AL=H(I,3)/XK
      TNEW=(XNL*AL*TG(I,3)+TTT)/(1.+XNL*AL)
 410  CONTINUE
      TNEW=(1.-OME)*TB(I,3)+OME*TNEW
      SUM=SUM+(TB(I,3)-TNEW)*(TB(I,3)-TNEW)
      TB(I,3)=TNEW
 412  CONTINUE
C
C  CALCULATES NEW COOLANT TEMP USING HEAT BALANCE EQUATIONS
C
      IF(I.GT.N5 .AND.I.LT.N2)  GO TO 430
       Q(I,3)=H(I,3)*(TB(I,3)-TG(I,3))*DELS(I,3)*LB
       Q(I,4)=H(I,4)*(TB(I,4)-TG(I,4))*DELS(I,4)*LB
      IF(I.EQ.N71)  GO TO 415
      IF(I.LT.N71)  GO TO 420
      IF(I.GT.N71)  GO TO 425
 415  TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+MASC(I,4)*CP*TG(I,4)+Q(I,3)+Q(I,4)
     1)/(MASC(I+1,3)*CP)
      GO TO 430
 420  TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+DELM(I,3)*CP*TCOLN+ Q(I,3))/(MASC(
     1I+1,3)*CP)
      GO TO 430
 425  TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+ Q(I,3)+ Q(I,4))/(MASC(I+1,3)*CP)
C
C   UPPER SURFACE OF INNER BOUNDARY
```

```
C
 430  C2=Y4(I)
      CALL CUR(B2,C2,B,C,1)
      XMN=B+2.*C*B2
      IF(I.EQ.N11)  XMN=1.2
      YT=FLOAT(IHI-1)*DY
      IF(ABS(XMN).LE.0.015) GO TO 445
      X=(C2-YT)*XMN+B2
      XR=B2+DX
      XL=B2-DX
      IF(X.LE.B2) GO TO 435
      IF(X.GT.X2(IHI))  GO TO 446
      IF((X-XR).GT.0.0001)  GO TO 447
      IF((X2(IHI)-XR).LE.0.0001)  GO TO 434
      TTT=T(I,IHI)+(T(I+1,IHI)-T(I,IHI))*(X-B2)/DX
       GO TO 440
 434  TTT=T(I,IHI)+(TBX(2,IHI)-T(I,IHI))*(X-B2)/DX
      GO TO 440
 435  CONTINUE
      IF((X-X1(IHI)).LT.0.0001)  GO TO 448
      IF((XL-X1(IHI)).LT.0.0001)   GO TO 437
       TTT=T(I,IHI)-(T(I,IHI)-T(I-1,IHI))*(B2-X)/DX
      GO TO 440
 437  TTT=T(I,IHI)-(T(I,IHI)-TBX(1,IHI))*(B2-X)/(B2-X1(IHI))
 440   XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 450
 445   TTT=T(I,IHI)
      XNL=YT-C2
      GO TO 450
 446  IF((Y2(I+1)-YT).GT.0.0001)  GO TO 447
      TTT=TBX(2,IHI)
      XNL=(B2-X2(IHI))*(B2-X2(IHI))+DY*DY
      XNL=SQRT(XNL)
      GO TO 450
 447  TTT=T(I+1,IHI)
      XNL=DX*DX+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 450
 448  TTT=TBX(1,IHI)
      XNL=(B2-X1(IHI))*(B2-X1(IHI))+DY*DY
      XNL=SQRT(XNL)
 450   AL=H(I,4)/XK
      TNEW=(XNL*AL*TG(I,4)+TTT)/(1.+XNL*AL)
      TNEW=(1.-CME)*TB(I,4)+CME*TNEW
      SUM=SUM+(TB(I,4)-TNEW)*(TB(I,4)-TNEW)
      TB(I,4)=TNEW
C
C  CALCULATES NEW COOLANT TEMP USING HEAT BALANCE EQUATIONS
C
```

```
      IF(I.GT.N5 .AND.I.LT.N2)  GC TC 47C
      IF(I.EC.N71)  GC TC 455
      IF(I.LT.N71)  GC TC 460
      IF(I.GT.N71)  GC TC 465
 455  TG(I+1,4)=(MASC(I,3)*CP*TG(I,3)+MASC(I,4)*CP*TG(I,4)+ Q(I,3)+ Q(I,
     14))/(MASC(I+1,4)*CP)
      GC TC 470
 460  TG(I+1,4)=(MASC(I,4)*CP*TG(I,4)+DELM(I,4)*CP*TCCLN+ Q(I,4))/(MASC(
     1I+1,4)*CP)
      GC TC 470
 465  TG(I+1,4)=(MASC(I,4)*CP*TG(I,4)+ Q(I,3)+ Q(I,4))/(MASC(I+1,4)*CP)
C
C
C   INTERIOR POINTS FROM UPPER SURFACE CF INNER BCUNDARY TO THE UPPER
C   SURFACE OF CUTER BCUNCARY
C
 470  DC 475 J=IHI,IH
      CALL MESH(I,J,TNEW)
      TNEW=(1.-CME)*T(I,J) +CME*TNEW
      SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 475  T(I,J)=TNEW
 480   CCNTINUE
C
C   UPPER BOUNCARY CF CUTER SURFACE
C
       C2=Y2(I)
       CALL CUR(B2,C2,B,C,0)
      XMN=B+2.*C*B2
      IF(I.EQ.NX1)  XMN=-1.8
      IF(I.EC.18)  XMN=-2.8
      IF(I.EC.19)  XMN=-2.8
      YT=FLOAT(IH-1)*CY
      IF(ABS(XMN).LE.0.015) GC TC 500
      X=(C2-YT)*XMN+B2
      XR=B2+CX
       XL=B2-CX
      IF(I.EC.3)  XMN=1.0
      IF(I.LE.N1.CR.I.GE.N2)  GC TC 481
      IF((C2-(Y4(I)+Y4(I+1))/2).GE.CY)  GC TC 481
      TTT=T8(I,4)
      XNL=C2-Y4(I)
      GC TO 505
 481  CCNTINUE
      IF(X.LE.B2)  GC TC 483
      IF(IH.GE.N4.OR.IH.LE.N3)  GC TC 4840
      IF((X-X3(IH)).GT.0.0001)  GC TC 492
      IF(X3(IH).LE.XR)  GC TC 482
 4840  TTT=T(I,IH)+(T(I+1,IH)-T(I,IH))*(X-B2)/CX
      GC TC 495
 482  TTT=T(I,IH)+(TBX(3,IH)-T(I,IH))*(X-B2)/(X3(IH)-B2)
```

```
      GO TO 495
 483  CONTINUE
      IF(IH.GE.N4.OR.IH.LE.N3)  GO TO 485
      IF(X.LE.X4(IH))  GO TO 498
      IF((XL-X4(IH)).GT.0.0001)  GO TO 490
      TTT=T(I,IH)-(T(I,IH)-TBX(4,IH))*(B2-X)/(B2-X4(IH))
      GO TO 495
 485  CONTINUE
      IF(X.LE.X1(IH))  GO TO 497
      IF((XL-X1(IH)).GT.0.0001)  GO TO 490
       TTT=T(I,IH)-(T(I,IH)-TBX(1,IH))*(B2-X)/(B2-X1(IH))
       GO TO 495
 490  TTT=T(I,IH)-(T(I,IH)-T(I-1,IH))*(B2-X)/DX
      GO TO 495
 492  TTT=TBX(3,IH)
      XNL=(B2-X3(IH))*(B2-X3(IH))+DY*DY
       XNL=SQRT(XNL)
      GO TO 505
 495  XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 505
 497  TTT=TBX(1,IH)
      GO TO 503
 498  TTT=TBX(4,IH)
      GO TO 502
 500  TTT=T(I,IH)
 502  XNL=C2-YT
      GO TO 505
 503  XNL=(B2-X1(IH))*(B2-X1(IH))+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
 505  AL=H(I,2)/XK
      TNEW=(XNL*AL*TG(I,2)+TTT)/(1.+XNL*AL)
      TNEW=(1.-OME)*TB(I,2)+OME*TNEW
      SUM=SUM+(TB(I,2)-TNEW)*(TB(I,2)-TNEW)
      TB(I,2)=TNEW
 510   CONTINUE
C
C     ITER. FOR J LINES ONLY BOUNDARY POINTS INTERSECTED BY J LINES
C
      DO 650 J=2,NY1
      IN =IFIX(X1(J)/DX+0.0001)+2
      IA =IFIX(X2(J)/DX+0.0001)+1
       IF(ABS(X2(J)-FLOAT(IA-1)*DX).LT.0.00001) IA=IA-1
C
C    NEAR SURFACE OF OUTER BOUNDARY
C
       B2=X1(J)
      C2=FLOAT(J-1)*DY
      CALL CUR(B2,C2,B,C,0)
      XMN=B+2.*C*B2
```

```
      XT=FLOAT(IN-1)*DX
      IF(J.EQ.25)  XMN=-1.0
      IF(J.GE.N31.AND.J.LE.N61)  GO TO 547
       IF(ABS(XMN).GE.5.)GO TO 530
       XL=XT-DX
       X=-DY*XMN+B2
      IF(X.LE.B2) GO TO 520
      IF(J.LT.N6.OR .J.GE.N41)  GO TO 512
      IF(X.GT.X3(J+1))  GO TO 542
      IF((X3(J+1)-XT).GT.0.0001)  GO TO 515
      TTT=T(IN-1,J+1)+(TBX(3,J+1)-T(IN-1,J+1))*(X-XL)/(X3(J+1)-XL)
      GO TO 525
512   CONTINUE
      IF(X.GT.X2(J+1))  GO TO 532
      IF((X2(J+1)-XT).GT.0.0001) GO TO 515
       TTT=T(IN-1,J+1)+(TBX(2,J+1)-T(IN-1,J+1))*(X-XL)/(X2(J+1)-XL)
        GO TO 525
515   CONTINUE
      IF((XL-X1(J+1)).GT.0.0001)  GO TO 518
      IF(X.GT.XT)  GO TO 533
      TTT=T(IN,J+1)+(TBX(1,J+1)-T(IN,J+1))*(XT-X)/(XT-X1(J+1))
      GO TO 525
518   CONTINUE
       TTT=T(IN,J+1)+(T(IN-1,J+1)-T(IN,J+1))*(XT-X)/DX
      GO TO 525
520   CONTINUE
      X=DY*XMN+B2
      IF(XT.GE.X3(J-1))  GO TO 524
      IF(X.GT.XT)  GO TO 523
      TTT=T(IN,J-1)+(T(IN-1,J-1)-T(IN,J-1))*(XT-X)/DX
      GO TO 525
523   TTT=T(IN,J-1)
      GO TO 534
524   TTT=TBX(3,J-1)
      XNL=(B2-X3(J-1))*(B2-X3(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 535
525    XNL=(B2-X)*(B2-X)+DY*DY
       XNL=SQRT(XNL)
      GO TO 535
530    TTT=T(IN,J)
      XNL=XT-B2
      GO TO 535
532   TTT=TB(IN,2)
      XNL=(B2-XT)*(B2-XT)+(C2-Y2(IN))*(C2-Y2(IN))
      XNL=SQRT(XNL)
      GO TO 535
533   TTT=T(IN,J+1)
534   XNL=(B2-XT)*(B2-XT)+DY*DY
      XNL=SQRT(XNL)
```

```
 535  AL=HX(1,J)/XK
      TNEW=(TTT+AL*XNL*TBX(1,J))/(1.+XNL*AL)
      GO TO 545
 542  TNEW=(TB(IN,1)+TB(IN-1,1))/2
 545  CONTINUE
      TNEW=(1.-OME)*TBX(1,J)+OME*TNEW
      SUM=SUM+(TNEW-TBX(1,J))*(TNEW-TBX(1,J))
      TBX(1,J)=TNEW
 547  CONTINUE
      IF(J.GT.N3.AND.J.LT.N4) GO TO 550
      GO TO 625
C
C
C   FOR J LINES INTERSECTING INNER BOUNDARY-BOUNDARY POINTS ON THE
C   NEARER SURFACE OF INNER BOUDARY
C
C
 550   CONTINUE
      INI=IFIX(X3(J)/DX+0.0001)+1
      IF(ABS(X3(J)-FLOAT(INI-1)*DX).LT.0.00001) INI=INI-1
      IAI=IFIX(X4(J)/DX+0.0001)+2
      B2=X3(J)
      CALL CUR(B2,C2,B,C,1)
      XMN=B+2.*C*B2
      IF(J.EQ.N31)  XMN=-1.7
      XT=FLOAT(INI-1)*DX
      XR=XT+DX
      XL=XT-DX
      IF(J.EQ.28)  XMN=1.20
       IF(ABS(XMN).GE.5.)GO TO 565
      IF (J.GE.N3.AND.J.LE.N61)  GO TO 587
      IF((C2-(Y1(INI)+Y1(INI+1))/2.).LE.DY) GOTO 570
        X=DY*XMN+B2
      IF(X.GT.B2) GO TO 555
      IF(X.LE.X1(J-1))  GO TO 570
      IF((X1(J-1)-XT).GT.0.0001)  GO TO 552
      IF((XT-X).GT.0.0001)  GO TO 562
       TTT=T(INI,J-1)+(T(INI+1,J-1)-T(INI,J-1))*(X-XT)/DX
       GO TO 560
 552  TTT=TBX(1,J-1)+(T(IN+1,J-1)-TBX(1,J-1))*(X-X1(J-1))/(XR-X1(J-1))
      GO TO 560
 555  X=-DY*XMN+B2
      IF(X.LE.X1(J+1))  GO TO 572
      TTT=T(INI,J+1)+(T(INI,J+1)-T(INI+1,J+1))*(X-XT)/DX
      GO TO 560
 562  IF((XL-X1(J-1)).GT.0.0001)  GO TO 563
      TTT=TBX(1,J-1)+(T(INI,J-1)-TBX(1,J-1))*(XT-X)/(XT-X1(J-1))
      GO TO 560
 563  TTT=T(INI-1,J-1)+(T(INI,J-1)-T(INI-1,J-1))*(XT-X)/DX
 560   XNL=(B2-X)*(B2-X)+DY*DY
```

```
      XNL=SQRT(XNL)
      GO TO 575
565    TTT=T(INI,J)
      XNL=B2-XT
      GO TO 575
570   CONTINUE
      IF(INI.EQ.N51)  GO TO 577
      IF((XL-X1(J)).GT.0.0001)  GO TO 574
      TTT=(TB(INI,1)+TBX(1,J))/2.
      XNL=B2-(XT+X1(J))/2.
      GO TO 575
572   TTT=TB(INI,2)
      XNL=(XT-B2)*(XT-B2)+(Y2(INI)-C2)*(Y2(INI)-C2)
      XNL=SQRT(XNL)
      GO TO 575
574   TTT=TB(INI,1)
      XNL=(B2-XT)*(B2-XT)+(C2-Y1(INI))*(C2-Y1(INI))
      XNL=SQRT(XNL)
575    AL=HX(3,J)/XK
      TNEW=(TTT+AL*XNL*TGX(3,J))/(1.+XNL*AL)
      GO TO 585
577   TNEW=TB(INI,1)
585   CONTINUE
      TNEW=(1.-CME)*TBX(3,J)+CME*TNEW
      SUM=SUM+(TNEW-TBX(3,J))*(TNEW-TBX(3,J))
      TBX(3,J)=TNEW
587   CONTINUE
      DO 590 I=N1,N2
      XR = FLOAT(I-1)*DX
      IF( (XR-X3(J)).GE.0.0001)  I33=I-1
      IF( (ABS(XR-X3(J)).GE.(0.9999*DX))) I33=I33-1
      IF( (XR-X3(J)).GE.0.0001)  GO TO 593
590   CONTINUE
593   IF(I33.LT.N11)  I33=N11
      TGX(3,J)=TG(I33,3)
C
C   FARTHER SURFACE OF INNER BOUNDARY
C
      B2=X4(J)
      CALL CUR(B2,C2,B,C,1)
      XMN=B+2.*C*B2
      IF(J.EQ.N31)  XMN=-1.7
      XT=FLOAT(IAI-1)*DX
      XL=XT-DX
       IF(ABS(XMN).GE.5.)GO TO 605
       X=-DY*XMN+B2
      IF((X-XT).GT.0.0001)  GO TO 595
      IF(X.LT.X2(J+1))  GO TO 597
595   CONTINUE
      IF((Y2(IAI)-(C2+DY)).GT.0.0001)  GO TO  596
```

```
      TTT=TBX(2,J+1)
      XNL=(B2-X2(J+1))*(B2-X2(J+1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 610
 596  TTT=T(IAI,J+1)
      XNL=(B2-XT)*(B2-XT)+DY*DY
      XNL=SQRT(XNL)
      GO TO 610
 597  CONTINUE
      IF(X2(J+1).LE.XT)  GO TO 598
       TTT=T(IAI,J+1)+(T(IAI-1,J+1)-T(IAI,J+1))*(XT-X)/DX
      GO TO 600
 598  TTT=TBX(2,J+1)+(T(IAI-1,J+1)-TBX(2,J+1))*(X2(J+1)-X)/(X2(J+1)-XL)
 600   XNL=(B2-X)*(B2-X)+DY*DY
       XNL=SQRT(XNL)
      GO TO 610
 605  TTT=T(IAI,J)
      XNL=XT-B2
 610   AL=HX(4,J)/XK
      TNEW=(TTT+AL*XNL*TGX(4,J))/(1.+XNL*AL)
 612  TNEW=(1.-OME)*TBX(4,J)+OME*TNEW
      SUM=SUM+(TNEW-TBX(4,J))*(TNEW-TBX(4,J))
      TBX(4,J)=TNEW
      DO 615 I=I33,N2
      XR=FLOAT(I-1)*DX
      IF ((XR-X4(J)).GE.0.0001)  I4=I
      IF(ABS(XR-X4(J)).LE.0.0001) I4=I4+1
      IF((XR-X4(J)).GE.0.0001) GO TO 620
 615  CONTINUE
 620  IF(I4.GT.N21)  I4=N21
      TGX(4,J)=TG(I4,4)
C
C   FARTHER BOUNDARY POINTS ON OUTER SURFACE
C
 625  B2=X2(J)
      CALL CUR(B2,C2,B,C,G)
      XMN=B+2.*C*B2
      IF(J.EQ.15)  XMN=-2.0
      IF(J.EQ.17)  XMN=-2.0
      IF(J.EQ.2  ) XMN=6.0
      XT=FLOAT(IA-1)*DX
      IF(ABS(XMN).GE.5.) GO TO 645
       XR=XT+DX
      XL=XT-DX
       X=DY*XMN+B2
      IF(J.LE.N31)  GO TO 635
      IF(IA.GT.N21)  GO TO 627
      IF((C2-Y4(IA)).GT.DY)  GO TO 627
      TTT=TB(IA,4)
      XNL=(C2-Y4(IA))*(C2-Y4(IA))+(B2-XT)*(B2-XT)
```

```
      XNL=SQRT(XNL)
      GO TO 646
627   CONTINUE
      IF((XT-X4(J-1)).GT.0.00001) GO TO 629
      IF((X4(J-1)-X).GT.0.0001)  GO TO 633
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 628
      TTT=TBX(4,J-1)-(TBX(4,J-1)-TBX(2,J-1))*(X-X4(J-1))/(X2(J-1)-X4
     1 (J-1))
      GO TO 642
628   TTT=TBX(4,J-1)-(TBX(4,J-1)-T(IA+1,J-1))*(X-X4(J-1))/(XR-X4(J-1))
      GO TO 642
629   CONTINUE
      IF(X.GT.XT)  GO TO 640
630   CONTINUE
      IF((X-X4(J-1)).GT.0.0001)  GO TO 632
631   TTT=T(IA,J-1)
      XNL=(B2-XT)*(B2-XT)+(DY*DY)
      XNL=SQRT(XNL)
      GO TO 646
632   CONTINUE
      IF((XL-X4(J-1)).GT.0.0001)  GO TO 634
633   TTT=TBX(4,J-1)
      XNL=(B2-X4(J-1))*(B2-X4(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
634   TTT=T(IA-1,J-1)+(T(IA-1,J-1)-T(IA,J-1))*(X-XL)/DX
      IF((XL-X).GT.0.0001)  GO TO 631
      GO TO 642
635   CONTINUE
      IF((C2-Y1(IA)).GT.DY)  GO TO 636
      TTT=TB(IA,1)
      XNL=(C2-Y1(IA))*(C2-Y1(IA))+(B2-XT)*(B2-XT)
      XNL=SQRT(XNL)
      GO TO 646
636   CONTINUE
      IF((XT-X1(J-1)).GT.0.00001) GO TO 639
      IF((X1(J-1)-X).GT.0.0001)  GO TO 644
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 638
      TTT=TBX(1,J-1)-(TBX(1,J-1)-TBX(2,J-1))*(X-X1(J-1))/(X2(J-1)-X1
     1 (J-1))
      GO TO 642
638   TTT=TBX(1,J-1)-(TBX(1,J-1)- T(IA+1,J-1))*(X-X1(J-1))/(XR-X1(J-1))
      GO TO 642
639   CONTINUE
      IF(X.GT.XT)  GO TO 640
      IF((X-X1(J-1)).GT.0.0001)  GO TO 643
      GO TO 631
640   CONTINUE
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 641
      TTT=T(IA,J-1)+(TBX(2,J-1)-T(IA,J-1))*(X-XT)/(X2(J-1)-XT)
```

```
      GO TO 642
  641  TTT=T(IA,J-1)+(T(IA+1,J-1)-T(IA,J-1))*(X-XT)/DX
  642  XNL=(B2-X)*(B2-X)+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
  643 CONTINUE
      IF((XL-X1(J-1)).GT.0.0001)  GO TO 634
  644 TTT=TBX(1,J-1)
      XNL=(B2-X1(J-1))*(B2-X1(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
  645 TTT=T(IA,J)
      XNL=B2-XT
  646  AL=HX(2,J)/XK
      TNEW=(TTT+AL*XNL*TGX(2,J))/(1.+XNL*AL)
  647 CONTINUE
      TNEW=(1.-OME)*TBX(2,J)+OME*TNEW
      SUM=SUM+(TBX(2,J)-TNEW)*(TBX(2,J)-TNEW)
      TBX(2,J)=TNEW
  650  CONTINUE
      DO 655 I=1,NX1
      DO 655 J=1,NY1
      X=FLOAT(I-1)*DX
      TB(1,1)=TBX(1,J)
      Y=FLOAT(J-1)*DY
      IF(ABS(X-X1(J)).LT.0.00001.AND.ABS(Y-Y1(I)).LT.0.00001) T(I,J)=TB(
     1I,1)
      IF(ABS(X-X1(J)).LT.0.00001.AND.ABS(Y-Y2(I)).LT.0.00001) T(I,J)=TB(
     1I,2)
      IF(ABS(X-X2(J)).LT.0.00001.AND.ABS(Y-Y2(I)).LT.0.00001) T(I,J)=TB(
     1I,2)
  655 CONTINUE
      DO 657 I=N11,N21
      DO 657 J=N31,N41
      X=FLOAT(I-1)*DX
      Y=FLOAT(J-1)*DY
      IF(ABS(X-X3(J)).LT.0.00001.AND.ABS(Y-Y3(I)).LT.0.00001) T(I,J)=TB(
     1I,3)
      IF(ABS(X-X4(J)).LT.0.00001.AND.ABS(Y-Y4(I)).LT.0.00001) T(I,J)=TB(
     1I,4)
      IF(ABS(X-X3(J)).LT.0.00001.AND.ABS(Y-Y4(I)).LT.0.00001) T(I,J)=TB(
     1I,4)
  657 CONTINUE
C
C
C
C   COMPLETION OF AN ITERATION-CHECKS FOR REQUIRED CONVERGENCE
C
C
      IF(ITER.EQ.1) WRITE(6,660)
```

```
660  FORMAT(1H1)
     WRITE(6,665) ITER,SUM
665  FORMAT(5X,'ITERATION=',I3,3X,'ERROR=',E12.5 )
     IF(ITER.GE.IMAX) GO TO 2000
     IF(SUM.LE.SUMM) GO TO 670
     IF((ITER/NTE*NTE).EQ.ITER)  GO TO 670
     ITER=ITER+1
     GO TO 300
670  CONTINUE
     WRITE(6,700)
700  FORMAT(//,10X,'T-MATRIX,GIVES ALL INTERIOR POINTS.INITIAL SETTING
    1 OF T IS 1600.0'/10X,'ONLY THOSE INSIDE BLADE ARE CHANGED AND WILL
    2 BE DIFFERENT FROM 1600.00')
     WRITE(6,705)
705  FORMAT(/,2X,'J =',4X,'1',7X,'2',7X,'3',7X,'4',7X,'5',7X,'6',7X, '7
    1',7X,'8',7X,'9',6X,'10',6X,'11',6X,'12',6X,'13',6X,'14',6X,'15'/
    22X,'I')
     IF(NY.GT.15) WRITE(6,710)
710  FORMAT(8X,'16',6X,'17',6X,'18',6X,'19',6X,'20' ,6X,'21',6X,'22',6X
    1,'23',6X,'24',6X,'25')
     DO 715 I=1,NX
715  WRITE(6,720) I, (T(I,J),J=1,NY)
720  FORMAT(/,2X,I2,1X,15F8.2,2(/5X,15F8.2))
     WRITE(6,725)
725  FORMAT('1',//15X,'GAS TEMPERATURE DISTRIBUTION ON THE BLADE INNER
    1AND OUTER SURFACE',//)
     WRITE(6,745)  NX1
      WRITE(6,720) (TG(I,1),I=1,NX1)
     WRITE(6,750) NX1
     WRITE(6,720) (TG(I,2),I=1,NX1)
     WRITE (6,755) N11,N21
      WRITE(6,800) (TG(I,3),I=N11,N21)
     WRITE(6,760) N11,N21
     WRITE(6,800) (TG(I,4),I=N11,N21)
     WRITE (6,770) NY1
     WRITE(6,800) (TGX(1,J),J=2,NY1)
     WRITE(6,780) NY1
     WRITE(6,800) (TGX(2,J),J=2,NY1)
     WRITE(6,785) N31,N41
     WRITE(6,800) (TGX(3,J),J=N31,N41)
     WRITE(6,790) N31,N41
     WRITE(6,800) (TGX(4,J),J=N31,N41)
     WRITE(6,795)
     WRITE(6,740)
740  FORMAT('1',//,5X,'TEMPERATURE DISTRIBUTION ON THE BOUNDARY',////)
     WRITE(6,745)  NX1
745  FORMAT(//,10X,'BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE  OF
    1 OUTER BOUNDRY'/,12X,'(START FROM I= 2  TO   I=',I3,' )')
     WRITE(6,720) (TB(I,1),I=1,NX1)
     WRITE(6,750) NX1
```

```
750  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE  OF
    1 OUTER BOUNDRY'/,12X,'(START FROM I= 2  TO I=',I3,' )')
     WRITE(6,720) (TB(I,2),I=1,NX1)
     WRITE (6,755) N11,N21
755  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF
    1 INNER BOUNDARY'/,12X,'(START FROM I=',I3,'  TO  I=',I3,' )')
     WRITE(6,800)(TB(I,3),I=N11,N21)
     WRITE(6,760) N11,N21
760  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF
    1 INNER BOUNDARY'/,12X,'(START FROM I=',I3,'  TO  I=',I3,' )')
     WRITE(6,800)(TB(I,4),I=N11,N21)
     WRITE (6,770) NY1
770  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
    1 OUTER BOUNDARY'/,12X,'(START FROM J= 2  TO  J=',I3,' )')
     WRITE(6,800)(TBX(1,J),J=2,NY1)
     WRITE(6,780) NY1
780  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
    1 OUTER BOUNDARY'/,12X,'(START FROM J= 2  TO  J=',I3,' )')
     WRITE(6,800)(TBX(2,J),J=2,NY1)
     WRITE(6,785) N31,N41
785  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
    1INNER BOUNDARY '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
     WRITE(6,800)(TBX(3,J),J=N31,N41)
     WRITE(6,790) N31,N41
790  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
    1INNER BOUNDARY. '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
     WRITE(6,800)(TBX(4,J),J=N31,N41)
     WRITE(6,795)
795  FORMAT(1H1)
800  FORMAT(3(/,5X,15F8.2))
     WRITE (6,810)
810  FORMAT(1H1///,15X,'TEMPERATURE DISTRIBUTION INSIDE THE BLADE AND O
    XN ITS SURFACE'///2X,'J=')
     DO 885 J1=2,NY1
     DO 815 I=1,NX
     J=NY+1-J1
     XR=FLOAT(I-1)*DX
     IF(    (XR-X1(J)).GE.0.0001) I1=I
     IF(ABS(XR-X1(J)).LE.0.0001) I1=I+1
     IF(    (XR-X1(J)).GE.0.0001) GO TO 820
815  CONTINUE
820  DO 825 I=I1,NX
     XR=FLOAT(I-1)*DX
     IF(    (XR-X2(J)).GE.0.0001) I2=I-1
     IF(ABS(XR-X2(J)).GE.(0.9999*DX)) I2=I2-1
     IF(    (XR-X2(J)).GE.0.0001) GO TO 830
825  CONTINUE
830  CONTINUE
     J=NY+1-J1
     IF(J.GE.N4.OR.J.LE.N3)  GO TO 860
```

```
 835   CONTINUE
       DO 840 I=N1,N2
        J=NY+1-J1
       XR = FLOAT(I-1)*DX
       IF( (XR-X3(J)).GE.0.0001)  I33=I-1
       IF( (ABS(XR-X3(J)).GE.(0.9999*DX))) I33=I33-1
       IF( (XR-X3(J)).GE.0.0001)  GO TO 845
 840   CONTINUE
 845   DO 850 I=I33,N2
       XR=FLOAT(I-1)*DX
       IF ((XR-X4(J)).GE.0.0001)  I4=I
       IF(ABS(XR-X4(J)).LE.0.0001) I4=I4+1
       IF((XR-X4(J)).GE.0.0001) GO TO 855
 850   CONTINUE
 855   CONTINUE
       J= NY+1-J1
       WRITE(6,875) J,TBX(1,J),(T(I,J),I=I1,I33),TBX(3,J)
       WRITE(6,880) TBX(4,J),(T(I,J),I=I4,I2),TBX(2,J)
       GO TO 865
 860   WRITE(6,870) J,TBX(1,J),(T(I,J),I=I1,I2),TBX(2,J)
 865   CONTINUE
 870   FORMAT (/2X,I2,4X,15F8.2)
 875   FORMAT(/2X,I2,4X,15F8.2)
 880   FORMAT(/16X,'#####',3X,12F8.2)
 885   CONTINUE
       IF(SUM.LE.SUMM) GO TO 890
       ITER=ITER+1
       GO TO 300
 890   CONTINUE
       IF(ABS(TEMPC-TG(N51,3)).LE.5)  GO TO 910
C
C
C  NEW VALUE OF FILM COOLANT TEMP IS OBTAINED
C  THEN GOES BACK TO GET FINAL SET OF TEMPS
C
       TEMPC=TG(N51,3)
       DENSC=198.57/TG(N51,3)
       CPC=(9.87-36.1/(TG(N51,3)**0.5)-2389./TG(N51,3)+9.06E05/(TG(I,J)**
      12.))/28.97
       VISCC=7.77E-07*(TG(N51,3))**1.5/(TG(N51,3)+198.)
C
C  REASSIGNING GAS TEMPS DOWNSTREAM OF SLOT TO ORIG. VALUES
C
C
       DO 900  I=N52,NX1
       TG(I,1)=TGTEM(I)
 900   CONTINUE
       DO 905  J=2,N6
       TGX(1,J)=TGXTEM(J)
 905   CONTINUE
```

```
      GC TC 245
 910  CCNTINUE
C
C
C  CBTAINS ISCTHERMAL LINE LCCATICNS
C
 940  WRITE(6,945)
 945  FCRMAT(1H1,38X,'ISCTHERMAL LINE LCCATICNS '/,9X,'I',4X,'J',3X,'T',
     14X,'T - 1',6X,'T - 2  ',8X,'FRAC',/)
      CALL TEMP
 990  CCNTINUE
      WRITE(6,992)
 992  FCRMAT(1X,1H1)
      DC 999 NN=1,NSCLT
      WRITE(6,995) TSCLT(NN)
 995  FCRMAT(1X,'THE ISCTHERM AT ',I5,/)
      WRITE(6,997)
 997  FCRMAT(5X,'X-COORD ',10X,'Y-CCCRD ',/)
      NNZ=NNY(NN)
      WRITE(6,998) ((XISC(NN,NNL),YISC(NN,NNL)),NNL=1,NNZ)
 998  FCRMAT(5X,F8.4,10X,F8.4)
 999  CCNTINUE
      DC 1000  I=N52,NX1
      TG(I,1)=TGTEM(I)
 1000  CCNTINUE
      DC 1010 J=2,N6
      TGX(1,J)=TGXTEM(J)
 1010  CCNTINUE
      DC 680  I=2,NX1
      EFF(I,1)=(TG(I,1)-TB(I,1))/(TG(I,1)-TCCLN)
      EFF(I,2)=(TG(I,2)-TB(I,2))/(TG(I,2)-TCCLN)
 680  CCNTINUE
      DC 685  J=2,NY1
      EFF(1,J)=(TGX(1,J)-TBX(1,J))/(TGX(1,J)-TCCLN)
      EFF(2,J)=(TGX(2,J)-TBX(2,J))/(TGX(2,J)-TCCLN)
 685  CCNTINUE
      WRITE(6,687)
 687  FCRMAT('1',//,10X,'EFFICIENCY CF CCCLING',////)
      WRITE(6,745)  NX1
      WRITE(6,690)(EFF(I,1),I=1,NX1)
 690  FCRMAT(/,2X,I2,1X,15F8.4,2(/5X,15F8.4))
      WRITE(6,750)NX1
      WRITE(6,690)(EFF(I,2),I=1,NX1)
      WRITE(6,770) NY1
      WRITE(6,692)(EFF(1,J),J=2,NY1)
 692  FCRMAT(3(/,5X,15F8.4))
      WRITE(6,780)  NY1
      WRITE(6,692)(EFF(2,J),J=2,NY1)
 2000 CCNTINUE
      STCP

      END
```

```
C       FORTRAN  COMPUTER  PROGRAM  TO  OBTAIN  TEMPERATURE  DISTRIBUTION
C       IN  A  BLADE  USING  THE  FINITE  DIFFERENCE  METHOD
C
C
C                         ** MAIN  PROGRAM  NO.2  **
C
C       FOR  COOLANT  DISCHARGE  ON  UPPER  OUTER  SURFACE  USE
C       MAIN  PROGRAM  NO.2  ALONG  WITH  ALL  THE  SUBROUTINES
C
C
        DIMENSION XISO(30,30),YISO(30,30),TSCLT(30),NNY(30)
        DIMENSION DELS(30,4),MASC(30,4),G(30,4),DELM(30,4)
        DIMENSION T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
       1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
       2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
        DIMENSION TGTEM(30),TGXTEM(65)
        DIMENSION VELG(30,65)
        DIMENSION EFF(30,65)
        REAL  LB,MASC
        INTEGER*4 TSCLT
        COMMON T,TB,TBX,TG,TGX,H,HX,Y1,Y2,Y3,Y4,X1,X2,X3,X4,BX,BY,BXI,BYI
       %,IP
        COMMON DX,DY,NX,NY,NXC,NX1,ITER,N1,N2,N3,N4,NP,N5,N6,N7
        COMMON/DAT/   NX1,NY1,N11,N21,N31,N41,N51,N52,N61,N62,N71,NSR,
       1DELM,MASC,DELS,XK,OME,SUMM,IMAX,NTE,TCOLN,LB,CP
        COMMON/TEM/  VELG,PRNC,TEMPC,CPC,VISCC,ALPHA,FLOC,DENSC
        COMMON/ISO/ XISO,YISO,TSCLT,NSCLT,NNY
        CALL DATA
 245    CONTINUE
C
C       TEMPORARY STORAGE FOR GAS TEMPS DOWNSTREAM OF SLOT
C
        DO 256 I=N51,NX1
        TGTEM(I)=TG(I,2)
 256    CONTINUE
        DO 258  J=2,N62
        TGXTEM(J)=TGX(2,J)
 258    CONTINUE
C
C  CALCULATES ADIABATIC WALL TEMPERATURES DOWNSTREAM OF SLOT
C
 266    DO 267  I=N51,NX1
        CALL TEMAD(I,2,TAD)
 267     TG(I,2)=TAD
        DO 268  J=2,N62
        CALL TEMAD(2,J,TAD)
 268    TGX(2,J)=TAD
        DO 269  I=N51,N21
        TG(I,3)=TG(I,2)
 269    CONTINUE
```

```
 3CC   CCNTINUE
C
C    STAFTING OF GAUSS-SEIDELL NETHCD CF ITERATION ON I LINES
C
       SUM=0.
       DC 510 I=2,NX1
       IL=IFIX(Y1(I)/DY+0.0001)+2
       IH=IFIX(Y2(I)/DY+0.0001)+1
       IF(ABS(Y2(I)-FLOAT(IH-1)*DY).LT.0.00001) IH=IH-1
C
C    CUTEF LOWEF BOUNDARY
C
C      CAFCS WITH NECA COMMENT TO SPECIFY SLOPES AT SPECIAL PCINTS
C
C
        B2=FLOAT(I-1)*DX
        C2=Y1(I)
        XR=B2+DX
        CALL CUF(B2,C2,B,C,0)
        XMN=B+2.*C*B2
       IF(I.EQ.NX1) XMN=0.01
        YT=FLOAT(IL-1)*DY
        X=(C2-YT)*XMN+B2
       IF(I.EQ.6)  XMN=-1.0
       IF(IL.LE.N3)  GC TO 305
       IF(I.GE.N2)  GO TO 305
       IF(X.GT.X3(IL)) GO TO 330
 305   IF(ABS(XMN).LE.0.015) GC TC 335
        IF(X.LE.B2)GO TO  320
        XR=B2+DX
       IF(IL.LE.N3)  GC TO 310
       IF(I.GE.N2)  GC TC 310
       IF ((X3(IL)-XR).GT.0.0001) GC TC 310
       TTT=T(I,IL)+(TBX(3,IL)-T(I,IL))*(X-B2)/(X3(IL)-B2)
       GC TC 325
 310   IF((X2(IL)-XR).GT.0.0001) GC TC 315
        TTT=T(I,IL)+(TBX(2,IL)-T(I,IL))*(X-B2)/(X2(IL)-B2)
        GC TC 325
 315    CCNTINUE
       IF((X-XR).GT.0.0001)  GO TO 337
        TTT=T(I,IL)+(T(I+1,IL)-T(I,IL))*(X-B2)/DX
       GC TO 325
 320    TTT=T(I,IL)-(T(I,IL)-T(I-1,IL))*(B2-X)/DX
 325     XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
       XNL=SQRT(XNL)
       GC TC 340
 330   TTT=TBX(3,IL)
       XNL=(X3(IL)-B2)*(X3(IL)-B2)+(C2-YT)*(C2-YT)
       XNL=SQRT(XNL)
       GC TO 340
```

```
 335    TTT=T(I,IL)
        XNL=YT-C2
        GO TO 340
 337    TTT=T(I+1,IL)
        XNL=DX*DX+(C2-YT)*(C2-YT)
        XNL=SQRT(XNL)
 340     AL=H(I,1)/XK
        TNEW=(XNL*AL*TG(I,1)+TTT)/(1.+XNL*AL)
 350    TNEW=(1.-OME)*TB(I,1)+OME*TNEW
        SUM=SUM+(TE(I,1)-TNEW)*(TB(I,1)-TNEW)
        TE(I,1)=TNEW
        IF(I.GT.N1.AND.I.LT.N2) GO TO 360
C
C
C    INTERIOR POINTS STARTING FROM LOWER END FOR I LINES WHICH DO NOT CUT
C    THE INNER BOUNDARY
C
        DO 355 J=IL,IH
        CALL MESH(I,J,TNEW)
        TNEW=(1.-OME)*T(I,J)+ONE*TNEW
        SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 355    T(I,J)=TNEW
        GO TO 480
C
C
C       FOR I LINES INTERSECTING INNER BOUNDARY-ALL INTERIOR POINTS FROM
C    LOWER SURFACE OF OUTER BOUNDARY TO LOWER SURFACE OF INNER BOUNDARY
C
 360     CONTINUE
        ILI=IFIX(Y3(I)/DY+0.0001)+1
        IF(ABS(Y3(I)-FLOAT(ILI-1)*DY).LT.0.00001) ILI=ILI-1
        IHI=IFIX(Y4(I)/DY+0.0001)+2
        IF(IL.GT.ILI)  GO TO 375
        DO 365 J=IL,ILI
        CALL MESH(I,J,TNEW)
        TNEW=(1.-OME)*T(I,J)+OME*TNEW
        SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 365    T(I,J)=TNEW
C
C    INNER BOUNDARY - LOWER SURFACE
C
 375    E2=FLOAT(I-1)*DX
        C2=Y3(I)
        CALL CUR(E2,C2,B,C,1)
        XMN=B+2.*C*E2
        IF(I.EQ.N11)  XMN=-1.0
        YT=FLOAT(ILI-1)*DY
        X=(C2-YT)*XMN+E2
        IF(I.EQ.7)  XMN=-1.0
        IF(X.LT.X1(ILI)) GO TO 380
```

```
      IF((C2-YT).GE.(C2-Y1(I))) GO TO 380
      GO TO 382
  380 TTT=TEX(1,ILI)
      XNL=(C2-YT)*(C2-YT)+(X1(ILI)-B2)*(X1(ILI)-B2)
      XNL=SQRT(XNL)
      GO TO 400
  382 IF(ABS(XMN).LE.0.015) GO TO 395
      IF(X.LE.B2)  GO TO 383
      TTT=T(I,ILI)+(T(I+1,ILI)-T(I,ILI))*(X-B2)/DX
      GO TO 390
  383 CONTINUE
      XL=B2-DX
      IF((XL-X1(ILI)).GT.0.0001)  GO TO 385
      TTT=T(I,ILI)-(T(I,ILI)-TEX(1,ILI))*(B2-X)/(B2-X1(ILI))
      GO TO 390
  385 TTT=T(I,ILI)-(T(I,ILI)-T(I-1,ILI))*(B2-X)/DX
  390 XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 400
  395 TTT=T(I,ILI)
      XNL=C2-YT
  400 AL=H(I,3)/XK
      TNEW=(XNL*AL*TG(I,3)+TTT)/(1.+XNL*AL)
  410 TNEW=(1.-CME)*TB(I,3)+CME*TNEW
      SUM=SUM+(TB(I,3)-TNEW)*(TB(I,3)-TNEW)
      TB(I,3)=TNEW
C
C  CALCULATES NEW COOLANT TEMP USING HEAT BALANCE EQUATIONS
C
      IF(I.GT.NSR)  GO TO 430
       Q(I,3)=H(I,3)*(TB(I,3)-TG(I,3))*DELS(I,3)*LB
       Q(I,4)=H(I,4)*(TB(I,4)-TG(I,4))*DELS(I,4)*LB
      IF(I.EQ.N71)  GO TO 415
      IF(I.LT.N71)  GO TO 420
      IF(I.GT.N71)  GO TO 425
  415 TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+MASC(I,4)*CP*TG(I,4)+Q(I,3)+Q(I,4)
     1)/(MASC(I+1,3)*CP)
      GO TO 430
  420 TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+DELM(I,3)*CP*TCOLN+ Q(I,3))/(MASC(
     1I+1,3)*CP)
      GO TO 430
  425 TG(I+1,3)=(MASC(I,3)*CP*TG(I,3)+ Q(I,3)+ Q(I,4))/(MASC(I+1,3)*CP)
C
C   UPPER SURFACE OF INNER BOUNDARY
C
  430 C2=Y4(I)
      CALL CUR(B2,C2,B,C,1)
      XMN=B+2.*C*B2
      IF(I.EQ.N11)  XMN=1.2
      YT=FLOAT(IHI-1)*DY
```

```
      IF(I.GT.NS.AND.I.LT.N2)  GO TO 452
      IF(ABS(XMN).LE.0.015) GO TO 445
      X=(C2-YT)*XMN+B2
      XR=B2+DX
      XL=B2-DX
      IF(X.LE.B2) GO TO 435
      IF(X.GT.X2(IHI))  GO TO 446
      IF((X-XR).GT.0.0001)  GO TO 447
      IF((X2(IHI)-XR).LE.0.0001)  GO TO 434
      TTT=T(I,IHI)+(T(I+1,IHI)-T(I,IHI))*(X-B2)/DX
       GO TO 440
 434  TTT=T(I,IHI)+(TBX(2,IHI)-T(I,IHI))*(X-B2)/DX
      GO TO 440
 435  CONTINUE
      IF((X-X1(IHI)).LT.0.0001)  GO TO 448
      IF((XL-X1(IHI)).LT.0.0001)   GO TO 437
       TTT=T(I,IHI)-(T(I,IHI)-T(I-1,IHI))*(B2-X)/DX
      GO TO 440
 437  TTT=T(I,IHI)-(T(I,IHI)-TEX(1,IHI))*(B2-X)/(B2-X1(IHI))
 440   XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 450
 445   TTT=T(I,IHI)
      XNL=YT-C2  .
      GO TO 450
 446  IF((Y2(I+1)-YT).GT.0.0001)  GO TO 447
      TTT=TEX(2,IHI)
      XNL=(B2-X2(IHI))*(B2-X2(IHI))+DY*DY
      XNL=SQRT(XNL)
      GO TO 450
 447  TTT=T(I+1,IHI)
      XNL=DX*DX+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 450
 448  TTT=TEX(1,IHI)
      XNL=(B2-X1(IHI))*(B2-X1(IHI))+DY*DY
      XNL=SQRT(XNL)
 450   AL=H(I,4)/XK
      TNEW=(XNL*AL*TG(I,4)+TTT)/(1.+XNL*AL)
 451  CONTINUE                      .
      TNEW=(1.-OME)*TB(I,4)+OME*TNEW
      SUM=SUM+(TB(I,4)-TNEW)*(TB(I,4)-TNEW)
      TB(I,4)=TNEW
 452  CONTINUE
C
C  CALCULATES NEW COOLANT TEMP USING HEAT BALANCE EQUATIONS
C
       IF(I.GT.NSR)  GO TO 470
       IF(I.EQ.N71)  GO TO 455
       IF(I.LT.N71)  GO TO 460
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF(I.GT.N71)  GO TO 465
 455  TG(I+1,4)=(MASC(I,3)*CP*TG(I,3)+MASC(I,4)*CP*TG(I,4)+ Q(I,3)+ Q(I,
     14))/(MASC(I+1,4)*CP)
      GO TO 470
 460  TG(I+1,4)=(MASC(I,4)*CP*TG(I,4)+DELM(I,4)*CP*TCOLN+ Q(I,4))/(MASC(
     1I+1,4)*CP)
      GO TO 470
 465  TG(I+1,4)=(MASC(I,4)*CP*TG(I,4)+ Q(I,3)+ Q(I,4))/(MASC(I+1,4)*CP)
C
C
C    INTERIOR POINTS FROM UPPER SURFACE OF INNER BOUNDARY TO THE UPPER
C
 470  CONTINUE
      IF(I.GT.N5.AND.I.LT.N2)  GO TO 480
      IF(IHI.GT.IH)  GO TO 480
      IF (YT.GE.Y2(I))  GO TO 480
 471  DO 475 J=IHI,IH
      CALL MESH(I,J,TNEW)
      TNEW=(1.-OME)*T(I,J) +OME*TNEW
      SUM=SUM+(TNEW-T(I,J))*(TNEW-T(I,J))
 475  T(I,J)=TNEW
      GO TO 480
 478  T(I,IH)=TG(N52,3)
 480   CONTINUE
C
C    UPPER BOUNDARY OF OUTER SURFACE
C
       C2=Y2(I)
       CALL CUR(B2,C2,B,C,0)
      XMN=B+2.*C*B2
      IF(I.EQ.18)  XMN=-2.8
      IF(I.EQ.19)  XMN=-2.8
      YT=FLOAT(IH-1)*DY
      IF(ABS(XMN).LE.0.015) GO TO 500
      X=(C2-YT)*XMN+B2
      XR=B2+DX
       XL=B2-DX
      IF(I.EQ.3)  XMN=1.0
      IF(I.GT.N5.AND.I.LT.N2)  GO TO 509
      IF(I.LE.N1.OR.I.GE.N2)  GO TO 481
      IF((C2-(Y4(I)+Y4(I+1))/2).GE.DY)  GO TO 481
      TTT=TB(I,4)
      XNL=C2-Y4(I)
      GO TO 505
 481  CONTINUE
      IF(X.LE.B2)  GO TO 483
      IF(IH.GE.N4.OR.IH.LE.N3)  GO TO 4840
      IF((X-X3(IH)).GT.0.0001)  GO TO 492
      IF(X3(IH).LE.XR)  GO TO 482
 4840  TTT=T(I,IH)+(T(I+1,IH)-T(I,IH))*(X-B2)/DX
```

```
      GO TO 495
 482  TTT=T(I,IH)+(TEX(3,IH)-T(I,IH))*(X-B2)/(X3(IH)-B2)
      GO TO 495
 483  CONTINUE
      IF(IH.GE.N4.OR.IH.LE.N3)  GO TO 485
      IF(X.LE.X4(IH))  GO TO 498
      IF((XL-X4(IH)).GT.0.0001)  GO TO 490
      TTT=T(I,IH)-(T(I,IH)-TBX(4,IH))*(B2-X)/(B2-X4(IH))
      GO TO 495
 485  CONTINUE
      IF(X.LE.X1(IH))  GO TO 497
      IF((XL-X1(IH)).GT.0.0001)  GO TO 490
       TTT=T(I,IH)-(T(I,IH)-TBX(1,IH))*(B2-X)/(B2-X1(IH))
       GO TO 495
 490  TTT=T(I,IH)-(T(I,IH)-T(I-1,IH))*(B2-X)/DX
      GO TO 495
 492  TTT=TEX(3,IH)
      XNL=(B2-X3(IH))*(B2-X3(IH))+DY*DY
       XNL=SQRT(XNL)
      GO TO 505
 495  XNL=(X-B2)*(X-B2)+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
      GO TO 505
 497  TTT=TBX(1,IH)
      GO TO 503
 498  TTT=TBX(4,IH)
      GO TO 502
 500  TTT=T(I,IH)
 502  XNL=C2-YT
      GO TO 505
 503  XNL=(B2-X1(IH))*(B2-X1(IH))+(C2-YT)*(C2-YT)
      XNL=SQRT(XNL)
 505  AL=H(I,2)/XK
      TNEW=(XNL*AL*TG(I,2)+TTT)/(1.+XNL*AL)
 509  CONTINUE
      TNEW=(1.-OME)*TB(I,2)+OME*TNEW
      SUM=SUM+(TB(I,2)-TNEW)*(TB(I,2)-TNEW)
      TB(I,2)=TNEW
 510   CONTINUE
C
C     ITER. FOR J LINES ONLY BOUNDARY POINTS INTERSECTED BY J LINES
C
      DO 650 J=2,NY1
      IN =IFIX(X1(J)/DX+0.0001)+2
      IA =IFIX(X2(J)/DX+0.0001)+1
       IF(ABS(X2(J)-FLOAT(IA-1)*DX).LT.0.00001) IA=IA-1
C
C   NEAR SURFACE OF OUTER BOUNDARY
C
       B2=X1(J)
```

```
      C2=FLOAT(J-1)*DY
      CALL CUR(E2,C2,B,C,0)
      XMN=B+2.*C*E2
       XT=FLOAT(IN-1)*DX
      IF(J.EQ.25)  XMN=-1.0
       IF(ABS(XMN).GE.5.)GO TO 530
       XL=XT-DX
       X=-DY*XMN+E2
      IF(X.LE.B2) GO TO 520
      IF(J.LT.N3.OR .J.GE.N41)  GO TO 512
      IF(X.GT.X3(J+1))  GO TO 542
      IF((X3(J+1)-XT).GT.0.0001)  GO TO 515
      TTT=T(IN-1,J+1)+(TBX(3,J+1)-T(IN-1,J+1))*(X-XL)/(X3(J+1)-XL)
      GO TO 525
512   CONTINUE
      IF(X.GT.X2(J+1))  GO TO 532
      IF((X2(J+1)-XT).GT.0.0001) GO TO 515
       TTT=T(IN-1,J+1)+(TBX(2,J+1)-T(IN-1,J+1))*(X-XL)/(X2(J+1)-XL)
        GO TO 525
515   CONTINUE
      IF((XL-X1(J+1)).GT.0.0001)  GO TO 518
      IF(X.GT.XT)  GO TO 533
      TTT=T(IN,J+1)+(TEX(1,J+1)-T(IN,J+1))*(XT-X)/(XT-X1(J+1))
      GO TO 525
518   CONTINUE
       TTT=T(IN,J+1)+(T(IN-1,J+1)-T(IN,J+1))*(XT-X)/DX
      GO TO 525
520   CONTINUE
      X=DY*XMN+E2
      IF(XT.GE.X3(J-1))  GO TO 524
      IF(X.GT.XT)  GO TO 523
      TTT=T(IN,J-1)+(T(IN-1,J-1)-T(IN,J-1))*(XT-X)/DX
      GO TO 525
523   TTT=T(IN,J-1)
      GO TO 534
524   TTT=TEX(3,J-1)
      XNL=(E2-X3(J-1))*(B2-X3(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 535
525    XNL=(B2-X)*(B2-X)+DY*DY
       XNL=SQRT(XNL)
      GO TO 535
530    TTT=T(IN,J)
      XNL=XT-E2
      GO TO 535
532   TTT=TE(IN,2)
      XNL=(E2-XT)*(E2-XT)+(C2-Y2(IN))*(C2-Y2(IN))
      XNL=SQRT(XNL)
      GO TO 535
533   TTT=T(IN,J+1)
```

C-2

```
 534   XNL=(B2-XT)*(B2-XT)+DY*DY
       XNL=SQRT(XNL)
 535   AL=HX(1,J)/XK
       TNEW=(TTT+AL*XNL*TGX(1,J))/(1.+XNL*AL)
       GO TO 545
 542   TNEW=(TB(IN,1)+TB(IN-1,1))/2
 545   CONTINUE
       TNEW=(1.-OME)*TBX(1,J)+OME*TNEW
       SUM=SUM+(TNEW-TBX(1,J))*(TNEW-TBX(1,J))
       TEX(1,J)=TNEW
       IF(J.GT.N3.AND.J.LT.N4) GO TO 550
       GO TO 625
C
C
C    FOR J LINES INTERSECTING INNER BOUNDARY-BOUNDARY POINTS ON THE
C    NEARER SURFACE OF INNER BOUNDARY
C
C
 550    CONTINUE
       INI=IFIX(X3(J)/DX+0.0001)+1
       IF(ABS(X3(J)-FLOAT(INI-1)*DX).LT.0.00001) INI=INI-1
       IAI=IFIX(X4(J)/DX+0.0001)+2
       B2=X3(J)
       CALL CUR(B2,C2,B,C,1)
       XMN=B+2.*C*B2
       IF(J.EQ.N31)  XMN=-1.2
       IF(J.EQ.28)  XMN=1.2
       XT=FLOAT(INI-1)*DX
       XR=XT+DX
       XL=XT-DX
       IF(J.EQ.28)  XMN=1.20
        IF(ABS(XMN).GE.5.)GO TO 565
       IF((C2-(Y1(INI)+Y1(INI+1))/2.).LE.DY) GOTO 570
         X=DY*XMN+B2
       IF(X.GT.B2) GO TO 555
       IF(X.LE.X1(J-1))  GO TO 570
       IF((X1(J-1)-XT).GT.0.0001)  GO TO 552
       IF((XT-X).GT.0.0001)  GO TO 562
        TTT=T(INI,J-1)+(T(INI+1,J-1)-T(INI,J-1))*(X-XT)/DX
        GO TO 560
 552   TTT=TBX(1,J-1)+(T(IN+1,J-1)-TBX(1,J-1))*(X-X1(J-1))/(XR-X1(J-1))
       GO TO 560
 555   X=-DY*XMN+B2
       IF(X.LE.X1(J+1))  GO TO 572
       TTT=T(INI,J+1)+(T(INI,J+1)-T(INI+1,J+1))*(X-XT)/DX
       GO TO 560
 562   IF((XL-X1(J-1)).GT.0.0001)  GO TO 563
       TTT=TBX(1,J-1)+(T(INI,J-1)-TBX(1,J-1))*(XT-X)/(XT-X1(J-1))
       GO TO 560
 563   TTT=T(INI-1,J-1)+(T(INI,J-1)-T(INI-1,J-1))*(XT-X)/DX
```

```
 560    XNL=(B2-X)*(B2-X)+DY*DY
        XNL=SQRT(XNL)
        GO TO 575
 565    TTT=T(INI,J)
        XNL=B2-XT
        GO TO 575
 570    CONTINUE
        IF((XL-X1(J)).GT.0.0001)   GO TO 574
        TTT=(TB(INI,1)+TEX(1,J))/2.
        XNL=B2-(XT+X1(J))/2.
        GO TO 575
 572    TTT=TB(INI,2)
        XNL=(XT-B2)*(XT-B2)+(Y2(INI)-C2)*(Y2(INI)-C2)
        XNL=SQRT(XNL)
        GO TO 575
 574    TTT=TB(INI,1)
        XNL=(B2-XT)*(B2-XT)+(C2-Y1(INI))*(C2-Y1(INI))
        XNL=SQRT(XNL)
 575    AL=HX(3,J)/XK
        TNEW=(TTT+AL*XNL*TGX(3,J))/(1.+XNL*AL)
 585    TNEW=(1.-OME)*TBX(3,J)+OME*TNEW
        SUM=SUM+(TNEW-TBX(3,J))*(TNEW-TEX(3,J))
        TBX(3,J)=TNEW
        DO 590  I=N1,N2
        XR = FLOAT(I-1)*DX
        IF( (XR-X3(J)).GE.0.0001)   I33=I-1
        IF( (ABS(XR-X3(J)).GE.(0.9999*DX)))  I33=I33-1
        IF( (XR-X3(J)).GE.0.0001)   GO TO 593
 590    CONTINUE
 593    IF(I33.LT.N11)   I33=N11
        TGX(3,J)=TG(I33,3)
        IF(J.EQ.N62)   TGX(3,J) =TG(N51,3)
C
C  FARTHER SURFACE OF INNER BOUNDARY
        B2=X4(J)
        CALL CUR(B2,C2,B,C,1)
        XMN=B+2.*C*B2
        IF(J.EQ.N31)   XMN=-1.7
        XT=FLOAT(IAI-1)*DX
        XL=XT-DX
        IF(ABS(XMN).GE.5.)GO TO 605
        X=-DY*XMN+B2
        IF(J.GT.N3.AND.J.LT.N6)   GO TO 614
        IF((X-XT).GT.0.0001)   GO TO 595
        IF(X.LT.X2(J+1))   GO TO 597
 595    CONTINUE
        IF((Y2(IAI)-(C2+DY)).GT.0.0001)   GO TO   596
        TTT=TEX(2,J+1)
        XNL=(B2-X2(J+1))*(B2-X2(J+1))+DY*DY
        XNL=SQRT(XNL)
```

```
      GO TO 610
 596  TTT=T(IAI,J+1)
      XNL=(B2-XT)*(B2-XT)+DY*DY
      XNL=SQRT(XNL)
      GO TO 610
 597  CONTINUE
      IF(X2(J+1).LE.XT)  GO TO 598
       TTT=T(IAI,J+1)+(T(IAI-1,J+1)-T(IAI,J+1))*(XT-X)/DX
      GO TO 600
 598  TTT=TBX(2,J+1)+(T(IAI-1,J+1)-TBX(2,J+1))*(X2(J+1)-X)/(X2(J+1)-XL)
 600   XNL=(B2-X)*(B2-X)+DY*DY
       XNL=SQRT(XNL)
      GO TO 610
 605  TTT=T(IAI,J)
      XNL=XT-B2
 610   AL=HX(4,J)/XK
      TNEW=(TTT+AL*XNL*TGX(4,J))/(1.+XNL*AL)
 611  CONTINUE
 612  TNEW=(1.-CNE)*TBX(4,J)+CNE*TNEW
      SUM=SUM+(TNEW-TBX(4,J))*(TNEW-TBX(4,J))
      TBX(4,J)=TNEW
 614  CONTINUE
      DO 615 I=I33,N2
      XR=FLOAT(I-1)*DX
      IF ((XR-X4(J)).GE.0.0001)  I4=I
      IF(ABS(XR-X4(J)).LE.0.0001) I4=I4+1
      IF((XR-X4(J)).GE.0.0001) GO TO 620
 615  CONTINUE
 620  IF(I4.GT.N21)  I4=N21
      TGX(4,J)=TG(I4,4)
C
C   FARTHER BOUNDARY POINTS ON OUTER SURFACE
C
 625  B2=X2(J)
      CALL CUR(B2,C2,B,C,0)
      XMN=B+2.*C*B2
      IF(J.EQ.15)  XMN=-2.8
      IF(J.EQ.17)  XMN=-2.8
      IF(J.EQ.2  ) XMN=6.0
      XT=FLOAT(IA-1)*DX
      IF(ABS(XMN).GE.5.) GO TO 645
       XR=XT+DX
      XL=XT-DX
       X=DY*XMN+B2
      IF(J.GT.N3.AND.J.LT.N6)  GO TO 650
      IF(J.LE.N31)  GO TO 635
      IF(IA.GT.N21)  GO TO 627
      IF((C2-Y4(IA)).GT.DY)  GO TO 627
      TTT=TB(IA,4)
      XNL=(C2-Y4(IA))*(C2-Y4(IA))+(B2-XT)*(B2-XT)
```

```
      XNL=SQRT(XNL)
      GO TO 646
627   CONTINUE
      IF((XT-X4(J-1)).GT.0.00001) GO TO 629
      IF((X4(J-1)-X).GT.0.0001)  GO TO 633
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 628
      TTT=TBX(4,J-1)-(TBX(4,J-1)-TBX(2,J-1))*(X-X4(J-1))/(X2(J-1)-X4
     1 (J-1))
      GO TO 642
628   TTT=TBX(4,J-1)-(TBX(4,J-1)-T(IA+1,J-1))*(X-X4(J-1))/(XR-X4(J-1))
      GO TO 642
629   CONTINUE
      IF(X.GT.XT)  GO TO 640
630   CONTINUE
      IF((X-X4(J-1)).GT.0.0001)  GO TO 632
631   TTT=T(IA,J-1)
      XNL=(B2-XT)*(B2-XT)+(DY*DY)
      XNL=SQRT(XNL)
      GO TO 646
632   CONTINUE
      IF((XL-X4(J-1)).GT.0.0001)  GO TO 634
633   TTT=TBX(4,J-1)
      XNL=(B2-X4(J-1))*(B2-X4(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
634   TTT=T(IA-1,J-1)+(T(IA-1,J-1)-T(IA,J-1))*(X-XL)/DX
      IF((XL-X).GT.0.0001)  GO TO 631
      GO TO 642
635   CONTINUE
      IF((C2-Y1(IA)).GT.DY)  GO TO 636
      TTT=TB(IA,1)
      XNL=(C2-Y1(IA))*(C2-Y1(IA))+(B2-XT)*(B2-XT)
      XNL=SQRT(XNL)
      GO TO 646
636   CONTINUE
      IF((XT-X1(J-1)).GT.0.00001) GO TO 639
      IF((X1(J-1)-X).GT.0.0001)  GO TO 644
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 638
      TTT=TBX(1,J-1)-(TBX(1,J-1)-TBX(2,J-1))*(X-X1(J-1))/(X2(J-1)-X1
     1 (J-1))
      GO TO 642
638   TTT=TBX(1,J-1)-(TBX(1,J-1)- T(IA+1,J-1))*(X-X1(J-1))/(XR-X1(J-1))
      GO TO 642
639   CONTINUE
      IF(X.GT.XT)  GO TO 640
      IF((X-X1(J-1)).GT.0.0001)  GO TO 643
      GO TO 631
640   CONTINUE
      IF((XR-X2(J-1)).LT.0.0001)  GO TO 641
      TTT=T(IA,J-1)+(TBX(2,J-1)-T(IA,J-1))*(X-XT)/(X2(J-1)-XT)
```

```
      GO TO 642
  641  TTT=T(IA,J-1)+(T(IA+1,J-1)-T(IA,J-1))*(X-XT)/DX
  642  XNL=(B2-X)*(B2-X)+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
  643 CONTINUE
      IF((XL-X1(J-1)).GT.0.0001)  GO TO 634
  644 TTT=TEX(1,J-1)
      XNL=(B2-X1(J-1))*(B2-X1(J-1))+DY*DY
      XNL=SQRT(XNL)
      GO TO 646
  645 TTT=T(IA,J)
      XNL=B2-XT
  646  AL=HX(2,J)/XK
      TNEW=(TTT+AL*XNL*TGX(2,J))/(1.+XNL*AL)
  647 TNEW=(1.-OME)*TBX(2,J)+OME*TNEW
      SUM=SUM+(TEX(2,J)-TNEW)*(TEX(2,J)-TNEW)
      TEX(2,J)=TNEW
  650  CONTINUE
      DO 655 I=1,NX1
      DO 655 J=1,NY1
      X=FLOAT(I-1)*DX
      TE(1,1)=TBX(1,J)
      Y=FLOAT(J-1)*DY
      IF(ABS(X-X1(J)).LT.0.00001.AND.ABS(Y-Y1(I)).LT.0.00001) T(I,J)=TB(
     1I,1)
      IF(ABS(X-X1(J)).LT.0.00001.AND.ABS(Y-Y2(I)).LT.0.00001) T(I,J)=TB(
     1I,2)
      IF(ABS(X-X2(J)).LT.0.00001.AND.ABS(Y-Y2(I)).LT.0.00001) T(I,J)=TB(
     1I,2)
  655 CONTINUE
      DO 657 I=N11,N21
      DO 657 J=N31,N41
      X=FLOAT(I-1)*DX
      Y=FLOAT(J-1)*DY
      IF(ABS(X-X3(J)).LT.0.00001.AND.ABS(Y-Y3(I)).LT.0.00001) T(I,J)=TB(
     1I,3)
      IF(ABS(X-X4(J)).LT.0.00001.AND.ABS(Y-Y4(I)).LT.0.00001) T(I,J)=TB(
     1I,4)
      IF(ABS(X-X3(J)).LT.0.00001.AND.ABS(Y-Y4(I)).LT.0.00001) T(I,J)=TB(
     1I,4)
  657 CONTINUE
C
C    COMPLETION OF AN ITERATION-CHECKS FOR REQUIRED CONVERGENCE
C
C
      IF(ITER.EQ.1) WRITE(6,660)
  660 FORMAT(1H1)
      WRITE(6,665) ITER,SUM
  665 FORMAT(5X,'ITERATION=',I3,3X,'ERROR=',E12.5 )
```

```
      IF(ITER/NTE*NTE.EQ.ITER) GO TO 670
      IF(SUM.LE.SUMM) GO TO 670
      IF(ITER.GE.IMAX) GO TO 2000
      ITER=ITER+1
      GO TO 300
670   CONTINUE
      WRITE(6,700)
700   FORMAT(//,10X,'T-MATRIX,GIVES ALL INTERIOR POINTS.INITIAL SETTING
     1 OF T IS 1600.0'/10X,'ONLY THOSE INSIDE BLADE ARE CHANGED AND WILL
     2 BE DIFFERENT FROM 1600.00')
      WRITE(6,705)
705   FORMAT(/,2X,'J =',4X,'1',7X,'2',7X,'3',7X,'4',7X,'5',7X,'6',7X, '7
     1',7X,'8',7X,'9',6X,'10',6X,'11',6X,'12',6X,'13',6X,'14',6X,'15'/
     22X,'I')
      IF(NY.GT.15) WRITE(6,710)
710   FORMAT(8X,'16',6X,'17',6X,'18',6X,'19',6X,'20' ,6X,'21',6X,'22',6X
     1,'23',6X,'24',6X,'25')
      DO 715 I=1,NX
715   WRITE(6,720) I, (T(I,J),J=1,NY)
720   FORMAT(/,2X,I2,1X,15F8.2,2(/5X,15F8.2))
      WRITE(6,725)
725   FORMAT('1',//15X,'GAS TEMPERATURE DISTRIBUTION ON THE BLADE INNER
     1AND OUTER SURFACE',//)
      WRITE(6,745)  NX1
       WRITE(6,720) (TG(I,1),I=1,NX1)
      WRITE(6,750) NX1
      WRITE(6,720) (TG(I,2), I=1,NX1)
      IF(N1.EQ.N2.AND.N3.EQ.N4) GO TO 730
      WRITE (6,755) N11,N21
       WRITE(6,800) (TG(I,3),I=N11,N21)
      WRITE(6,760) N11,N21
      WRITE(6,800) (TG(I,4), I=N11,N21)
730   CONTINUE
      WRITE (6,770) NY1
      WRITE(6,800) (TGX(1,J),J=2,NY1)
      WRITE(6,780) NY1
      WRITE(6,800) (TGX(2,J),J=2,NY1)
      IF(N1.EQ.N2.AND.N3.EQ.N4) GO TO  735
      WRITE(6,785) N31,N41
      WRITE(6,800) (TGX(3,J),J=N31,N41)
      WRITE(6,790) N31,N41
      WRITE(6,800) (TGX(4,J),J=N31,N41)
      WRITE(6,795)
735   CONTINUE
      WRITE(6,740)
740   FORMAT('1',//,5X,'TEMPERATURE DISTRIBUTION ON THE BOUNDARY',////)
      WRITE(6,745)  NX1
745   FORMAT(//,10X,'BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE  OF
     1 OUTER BOUNDRY'/,12X,'(START FROM I= 2  TO  I=',I3,' )')
      WRITE(6,720) (TB(I,1), I=1,NX1)
```

```
      WRITE(6,750) NX1
750   FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE  OF
     1 OUTER BOUNDRY'/,12X,'(START FROM I= 2  TO I=',I3,' )')
      WRITE(6,720) (TB(I,2),I=1,NX1)
      IF(N1.EQ.N2.AND.N3.EQ.N4) GO TO 765
      WRITE (6,755) N11,N21
755   FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF
     1 INNER BOUNDARY'/,12X,'(START FROM I=',I3,'  TO  I=',I3,' )')
      WRITE(6,800)(TB(I,3),I=N11,N21)
      WRITE(6,760) N11,N21
760   FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF
     1 INNER BOUNDARY'/,12X,'( START FROM I=',I3,'  TO  I=',I3,' )')
      WRITE(6,800)(TB(I,4),I=N11,N21)
765   CONTINUE
      WRITE (6,770) NY1
770   FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
     1 OUTER BOUNDARY'/,12X,'( START FROM J= 2  TO  J=',I3,' )')
      WRITE(6,800)(TBX(1,J),J=2,NY1)
      WRITE(6,780) NY1
780   FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
     1 OUTER BOUNDARY'/,12X,'(START FROM J= 2  TO  J=',I3,' )')
      WRITE(6,800)(TBX(2,J),J=2,NY1)
      IF(N1.EQ.N2.AND.N3.EQ.N4) GO TO 805
      WRITE(6,785) N31,N41
785   FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
     1INNER BOUNDARY '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
      WRITE(6,800)(TBX(3,J),J=N31,N41)
      WRITE(6,790) N31,N41
790   FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
     1INNER BOUNDARY '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
      WRITE(6,800)(TBX(4,J),J=N31,N41)
      WRITE(6,795)
795   FORMAT(1H1)
800   FORMAT(3(/,5X,15F8.2))
805   CONTINUE
      WRITE (6,810)
810   FORMAT(1H1///,15X,'TEMPERATURE DISTRIBUTION INSIDE THE BLADE AND O
     XN ITS SURFACE'///2X,'J=')
      DO 885 J1=2,NY1
      DO 815 I=1,NX
      J=NY+1-J1
      XR=FLOAT(I-1)*DX
      IF(    (XR-X1(J)).GE.0.0001) I1=I
      IF(ABS(XR-X1(J)).LE.0.0001) I1=I+1
      IF(    (XR-X1(J)).GE.0.0001) GO TO 820
815   CONTINUE
820   DO 825 I=I1,NX
      XR=FLOAT(I-1)*DX
      IF(    (XR-X2(J)).GE.0.0001) I2=I-1
      IF(ABS(XR-X2(J)).GE.(0.9999*DX)) I2=I2-1
```

```
      IF(    (XR-X2(J)).GE.0.0001) GO TO 830
 825  CONTINUE
 830  CONTINUE
      J=NY+1-J1
      IF(J.GE.N4.OR.J.LE.N3)  GO TO 860
 835  CONTINUE
      DO 840 I=N1,N2
       J=NY+1-J1
      XR = FLOAT(I-1)*DX
      IF( (XR-X3(J)).GE.0.0001)  I33=I-1
      IF( (ABS(XR-X3(J)).GE.(0.9999*DX))) I33=I33-1
      IF( (XR-X3(J)).GE.0.0001)  GO TO 845
 840  CONTINUE
 845  DO 850 I=I33,N2
      XR=FLOAT(I-1)*DX
      IF ((XR-X4(J)).GE.0.0001)  I4=I
      IF(ABS(XR-X4(J)).LE.0.0001) I4=I4+1
      IF((XR-X4(J)).GE.0.0001) GO TO 855
 850  CONTINUE
 855  CONTINUE
      J= NY+1-J1
      WRITE(6,875) J,TBX(1,J),(T(I,J),I=I1,I33),TBX(3,J)
      WRITE(6,880) TBX(4,J),(T(I,J),I=I4,I2),TBX(2,J)
      GO TO 865
 860  WRITE(6,870) J,TBX(1,J),(T(I,J),I=I1,I2),TBX(2,J)
 865  CONTINUE
 870  FORMAT (/2X,I2,4X,15F8.2)
 875  FORMAT(/2X,I2,4X,15F8.2)
 880  FORMAT(/16X,'#####',3X,12F8.2)
 885  CONTINUE
      IF(SUM.LE.SUMM) GO TO 890
      ITER=ITER+1
      GO TO 300
 890  CONTINUE
      IF(ABS(TEMPC-TG(N5 ,3)).LE.5)  GO TO 910
C
C
C  NEW VALUE OF FILM COOLANT TEMP IS OBTAINED
C  THEN GOES BACK TO GET FINAL SET OF TEMPS
C
      TEMPC=TG(N5 ,3)
      DENSC=198.57/TG(N5 ,3)
      CPC=(9.87-36.1/(TG(N5 ,3)**0.5)-2389./TG(N5 ,3)+9.06E05/(TG(I,J)**
     12.))/28.97
      VISCC=7.77E-07*(TG(N5 ,3))**1.5/(TG(N5 ,3)+198.)
C
C  REASSIGNING GAS TEMPS DOWNSTREAM OF SLOT TO ORIG. VALUES
C
C
      DO 906  I=N51,NX1
```

```
      TG(I,2)=TGTEM(I)
 906  CCNTINUE
      DC 908  J=2,N62
      TGX(2,J)=TGXTEM(J)
 908  CCNTINUE
      GC TC 245
 910  CCNTINUE
C   CETAINS ISCTHERMAL LINE LCCATIONS
C
 940  WRITE(6,945)
 945  FCRMAT(1H1,38X,' ISOTHERMAL LINE LOCATIONS '/,9X,'I',4X,'J',3X,'T',
     14X,'T - 1',6X,'T - 2   ',8X,'FRAC',/)
      CALL TEMP
 990  CCNTINUE
      WRITE(6,992)
 992  FCRMAT(1X,1H1)
      DC 999 NN=1,NSCLT
      WRITE(6,995) TSCLT(NN)
 995  FCRMAT(1X,'THE ISCTHERM AT ',I5,/)
      WRITE(6,997)
 997  FCRMAT(5X,'X-COORD ',10X,'Y-COORD ',/)
      NNZ=NNY(NN)
      WRITE(6,998) ((XISO(NN,NNL),YISO(NN,NNL)),NNL=1,NNZ)
 998  FCRMAT(5X,F8.4,10X,F8.4)
 999  CCNTINUE
      DC 1000  I=N51,NX1
      TG(I,2)=TGTEM(I)
 1000  CONTINUE
      DC 1010  J=2,N62
      TGX(2,J)=TGXTEM(J)
 1010  CONTINUE
      DC 680  I=1,NX1
      EFF(I,1)=(TG(I,1)-TB(I,1))/(TG(I,1)-TCCLN)
      EFF(I,2)=(TG(I,2)-TB(I,2))/(TG(I,2)-TCCLN)
 680  CCNTINUE
      DC 685  J=2,NY1
      EFF(1,J)=(TGX(1,J)-TBX(1,J))/(TGX(1,J)-TCOLN)
      EFF(2,J)=(TGX(2,J)-TBX(2,J))/(TGX(2,J)-TCOLN)
 685  CONTINUE
      WRITE(6,687)
 687  FCRMAT('1',//,10X,'EFFICIENCY CF CCCLING',////)
      WRITE(6,745)  NX1
      WRITE(6,690)(EFF(I,1),I=1,NX1)
 690  FORMAT(/,2X,I2,1X,15F8.4,2(/5X,15F8.4))
      WRITE(6,750)NX1
      WRITE(6,690)(EFF(I,2),I=1,NX1)
      WRITE(6,770) NY1
      WRITE(6,692)(EFF(1,J),J=2,NY1)
 692  FORMAT(3(/,5X,15F8.4))
      WRITE(6,780)  NY1
      WRITE(6,692)(EFF(2,J),J=2,NY1)
 2000 CCNTINUE
      STCP
      END
```

```
      SUBROUTINE DATA
      DIMENSION DELS(30,4),MASC(30,4),C(30,4),DELM(30,4)
      DIMENSION T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      DIMENSION VELG(30,65)
      REAL  LB,MASC
      COMMON T,TB,TBX,TG,TGX,H,HX,Y1,Y2,Y3,Y4,X1,X2,X3,X4,BX,BY,BXI,BYI
     1,IP
      COMMON DX,DY,NX,NY,NXO,NXI,ITER,N1,N2,N3,N4,NP,N5,N6,N7
      COMMON/DAT/   NX1,NY1,N11,N21,N31,N41,N51,N52,N61,N62,N71,NSR,
     1DELM,MASC,DELS,XK,OME,SUMM,IMAX,NTE,TCOLN,LB,CP
      COMMON/TEM/  VELG,PRNO,TEMPC,CPC,VISCC,ALPHA,FLOC,DENSC
C
C  READS PHYSICAL PARAMETERS OF THE GRID SYSTEM
C
      READ(5,5) DX,DY,NXO,NXI
  5   FORMAT(2F5.3,2I5)
      READ(5,10) NX,NY,N1,N2,N3,N4,N5,N6,N7,NP
  10  FORMAT(10I5)
C
C  READS PROPERTIES OF THE BLADE AND THE COOLANT
C
      READ(5,15)  LB,XK,CP
  15  FORMAT(3F8.6)
      READ(5,20) PRNO,ALPHA,TCOLN,TEMPC
  20  FORMAT(4F8.2)
      READ(5,25) DENSC,CPC,VISCC,FLOC
  25  FORMAT(2F13.10,E16.10,F8.4)
C
C  READS ITERATION CHECKS
C
      READ(5,30) OME,SUMM,NTE,IMAX
  30  FORMAT(2F8.3,2I5)
      NX1=NX-1
      NY1=NY-1
      N11=N1+1
      N21=N2-1
      N31=N3+1
      N41=N4-1
      N51=N5+1
      N52=N5+2
      N61=N6-1
      N62=N6-2
      N71=N7-1
      NSR=N5-1
C
C  INIATIALLY TEMPS AT ALL POINTS ARE SET TO A CONVENIENT VALUE
C
      DO 31   I=1,NX
```

```
      DO 31   J=1,NY
   31 T(I,J) = 2400.
      DO 32   I=1,NX
      DO 32   J=1,4
   32 TB(I,J) = 2400.
      DO 33   J=1,NY
      DO 33   I=1,4
   33 TBX(I,J) = 2400.
C
C    READS IN BOUNDARY POINTS SET AND THE CORRESPONDING SEQUENCE NUMBERS
C     FOR THE BOUNDARY POINTS
C
C
      READ(5,35) (Y1(I),I=1,NX1)
   35 FORMAT((12F6.3))
      READ(5,40) (IP(I),I=1,NX1)
   40 FORMAT(20I4)
      DO 45 I=1,NX1
      X=FLOAT(I-1)*DX
      BX(IP(I))=X
   45 BY(IP(I))=Y1(I)
      READ(5,35) (Y2(I),I=1,NX1)
      READ(5,40) (IP(I),I=1,NX1)
      DO 50 I=1,NX1
      X=FLOAT(I-1)*DX
      BX(IP(I))=X
   50 BY(IP(I))=Y2(I)
      READ(5,35) (X1(J),J=1,NY1)
      READ(5,40)  (IP(J),J=1,NY1)
      DO 55 J=1,NY1
      Y=FLOAT(J-1)*DY
      BX(IP(J))=X1(J)
   55 BY(IP(J))=Y
      READ(5,35) (X2(J),J=1,NY1)
      READ(5,40) (IP(J),J=1,NY1)
      DO 60 J=1,NY1
      Y=FLOAT(J-1)*DY
      BX(IP(J))=X2(J)
   60 BY(IP(J))=Y
      READ(5,35) (Y3(I),I=N11,N21)
      READ(5,40) (IP(I),I=N11,N21)
      DO 65 I=N11,N21
      X=FLOAT(I-1)*DX
      BXI(IP(I))=X
   65 BYI(IP(I))=Y3(I)
      READ(5,35) (Y4(I),I=N11,N21)
      READ(5,40) (IP(I),I=N11,N21)
      DO 70 I=N11,N21
      X=FLOAT(I-1)*DX
      BXI(IP(I))=X
```

```
   70  BYI(IP(I))=Y4(I)
       READ(5,35) (X3(J),J=N31,N41)
       READ(5,40) (IP(J),J=N31,N41)
       DO 75 J = N31,N41
       Y=FLOAT(J-1)*DY
       BXI(IP(J))=X3(J)
   75  BYI(IP(J))= Y
       READ(5,35) (X4(J),J=N31,N41)
       READ(5,40) (IP(J),J=N31,N41)
       DO 77 J=N31,N41
       Y=FLOAT(J-1)*DY
       BXI(IP(J))=X4(J)
   77  BYI(IP(J))=Y
C
C    READS IN GAS TEMPS AND THEN HEAT TRANSFER COEFFICIENTS
C
       READ(5,90)(TG(I,1),I=1,NX1)
       READ(5,90)(TG(I,2),I=1,NX1)
       READ(5,90)(TG(I,3),I=N11,N21)
       READ(5,90)(TG(I,4),I=N11,N21)
   90  FORMAT(10F8.2)
       READ(5,90)(TGX(1,J),J=2,NY1)
       READ(5,90)(TGX(2,J),J=2,NY1)
       READ(5,90)(TGX(3,J),J=N31,N41)
       READ(5,90)(TGX(4,J),J=N31,N41)
       READ(5,99)(H(I,1),I=1,NX1)
       READ(5,99)(H(I,2),I=1,NX1)
       READ(5,99)(H(I,3),I=N11,N21)
       READ(5,99)(H(I,4),I=N11,N21)
   99  FORMAT(10F8.4)
       READ(5,99)(HX(1,J),J=2,NY1)
       READ(5,99)(HX(2,J),J=2,NY1)
       READ(5,99)(HX(3,J),J=N31,N41)
       READ(5,99)(HX(4,J),J=N31,N41)
C
C   READS VELOCITIES
C
       NR=1
       IF(NP.EQ.1)  NR=2
       READ(5,105)(VELG(I,NR),I=N51,NX1)
       READ(5,105)(VELG(NR,J),J=2,N6)
  105  FORMAT(10F8.2)
C
C   READS COOLANT MASS FLOW
C
       READ(5,110)(MASC(I,3),I=N11,N51)
       READ(5,110)(MASC(I,4),I=N11,N51)
       READ(5,110)(DELM(I,3),I=N11,N5)
       READ(5,110)(DELM(I,4),I=N11,N5)
C
```

```
C  READS SURFACE DISTANCES
C
      READ(5,110)(DELS(I,3),I=N11,N5)
      READ(5,110)(DELS(I,4),I=N11,N5)
 110  FORMAT(10F8.4)
C
C
C  ECHO CHECK FOR INPUT DATA
C
      ITER=1
      WRITE(6,115)
 115  FORMAT(//,25X,'INPUT DATA')
      WRITE(6,120)
 120  FORMAT(///,15X,'BLADE BOUNDARY POINTS COORDINATES'/////17X,'OUTER
     1 BOUNDARY',9X,'INNER BOUNDARY',//10X,'PT.NO.',5X,'X',9X,'Y',6X,
     2'PT.NO.',6X,'X',9X,'Y'/)
      NXI1=NXI+1
      WRITE(6,125)((I,BX(I),BY(I),I,BXI(I),BYI(I)),I=1,NXI)
      WRITE(6,130)((I,BX(I),BY(I)),I=NXI1,NXO)
 125  FORMAT(9X,I6,2F10.6,I6,2F10.6)
 130  FORMAT(9X,I6,2F10.6)
      WRITE(6,135)
 135  FORMAT('1',//15X,'INPUT DATA'//15X,'GAS TEMPERATURE DISTRIBUTION
     1ON THE BLADE INNER AND OUTER SURFACE',//)
      WRITE(6,745)  N1,NX1
       WRITE(6,720) (TG(I,1),I=1,NX1)
      WRITE(6,750)  N1,NX1
      WRITE(6,720) (TG(I,2),I=1,NX1)
      WRITE (6,755) N11,N21
       WRITE(6,800) (TG(I,3),I=N11,N21)
      WRITE(6,760) N11,N21
      WRITE(6,800) (TG(I,4),I=N11,N21)
      WRITE (6,770) NY1
      WRITE(6,800) (TGX(1,J),J=2,NY1)
      WRITE(6,780) NY1
      WRITE(6,800) (TGX(2,J),J=2,NY1)
      WRITE(6,785) N31,N41
      WRITE(6,800) (TGX(3,J),J=N31,N41)
      WRITE(6,790) N31,N41
      WRITE(6,800) (TGX(4,J),J=N31,N41)
      WRITE(6,795)
      WRITE(6,150)
 150  FORMAT('1',//15X,'INPUT DATA'//15X,'HEAT COEFFICIENT DISTRIBUTION
     1ON THE BLADE INNER AND OUTER SURFACE',//)
      WRITE(6,745)  N1,NX1
       WRITE(6,720) ( H(I,1),I=1,NX1)
      WRITE(6,750)  N1,NX1
      WRITE(6,720) ( H(I,2),I=1,NX1)
      WRITE (6,755) N11,N21
       WRITE(6,800) ( H(I,3),I=N11,N21)
```

```
      WRITE(6,760) N11,N21
      WRITE(6,800) ( H(I,4),I=N11,N21)
      WRITE (6,770) NY1
      WRITE(6,800) ( HX(1,J),J=2,NY1)
      WRITE(6,780) NY1
      WRITE(6,800) ( HX(2,J),J=2,NY1)
      WRITE(6,785) N31,N41
      WRITE(6,800) (HX(3,J),J=N31,N41)
      WRITE(6,790) N31,N41
      WRITE(6,800) ( HX(4,J),J=N31,N41)
      WRITE(6,795)
      WRITE(6,165)
165   FORMAT('1',//15X,'INPUT DATA'//15X,'COOLANT MASS FLOW DISTRIBUTION
     1',//)
      WRITE(6,755)  N11,N51
      WRITE(6,170)(MASC(I,3),I=N11,N51)
      WRITE(6,760)N11,N51
      WRITE(6,170)(MASC(I,4),I=N11,N51)
170   FORMAT(3(/,5X,15F8.4))
      WRITE(6,175)
175   FORMAT('1',//15X,'INPUT DATA',//15X,'VELOCITIES AT POINTS WHERE  A
     1DIABATIC WALL TEMP IS TO BE DETERMINED',//)
      IF(NP.EQ.1)  GO TO 178
      WRITE(6,745)  N51,NX1
      WRITE(6,180)  (VELG(I,NR),I=N51,NX1)
      WRITE(6,770) N6
      WRITE(6,180) (VELG(NR,J),J=2,N6)
      GO TO 182
178   CONTINUE
      WRITE(6,750) N51,NX1
      WRITE(6,180) (VELG(I,NR),I=N51,NX1)
      WRITE(6,780) N6
      WRITE(6,180) (VELG(NR,J),J=2,N6)
180   FORMAT(3(/ ,5X,15F8.2))
182   CONTINUE
      WRITE(6,185)
185   FORMAT('1',//15X,'INPUT DATA')
      WRITE(6,190)
190   FORMAT(//,15X,'PHYSICAL PARAMETERS OF THE GRID SYSTEM')
      WRITE(6,195) DX,DY,NXO,NXI
195   FORMAT(//,15X,'DX=',F5.3,5X,'DY=',F5.3,5X,'NXO=',I5,5X,'NXI=',I5)
      WRITE(6,200) NX,NY,N1,N2,N3,N4,N5,N6,N7,NP
200   FORMAT(//,15X,'NX=',I5,5X,'NY=',I5,5X,'N1=',I5,5X,'N2=',I5,//15X,
     1'N3=',I5,5X,5X,'N4=',I5,5X,'N5=',I5,5X,'N6=',I5,5X,'N7=',I5,
     25X,'NP=',I5)
      WRITE(6,205)
205   FORMAT(//,15X,'PROPERTIES OF THE BLADE AND THE COOLANT')
      WRITE(6,210) LB,XK,CP
210   FORMAT(//15X,'LB=',F8.6,5X,'XK=',F8.6,5X,'CP=',F8.6)
      WRITE(6,215) PRNC ,ALPHA ,TCOLN ,TEMPC
```

```
 215  FORMAT(//,15X,'PRNO=',F8.2,5X,'ALPHA=',F8.2,5X,'TCCLN=',F8.2,5X,
     1'TEMPC=',F8.2)
      WRITE(6,220) DENSC,CPC,VISCC,FLCC
 220  FORMAT(//,15X,'DENSC=',F13.10,5X,'CPC=',F13.10,5X,'VISCC=',E16.10
     1,5X,'FLCC=',F8.4)
      WRITE(6,225)
 225  FORMAT(//15X,'ITERATICN CHECKS')
      WRITE(6,230) OME,SUMM,NTE,IMAX
 230  FORMAT(//15X,'OME=',F8.3,5X,'SUMM=',F8.3,5X,'NTE=',I5 , 5X,
     1  'IMAX=',I5)
C
C   PRINT TITLE FOR SUBROUTINE CURVE OUTPUT
      WRITE(6,235)
 235  FORMAT(1H1,//,5X,'SUBROUTINE CURVE  -OUTPUT- '//5X,'I=0,OUTER BOUN
     1DARY. I=1,INNER BOUNDARY'/5X,'IP BOUNDARY POINT NUMBER')
      WRITE(6,240)
 240  FORMAT(     /,5X,'(B2,C2) IS THE POINT ON THE BOUNDARY. (B1,C1) AN
     1D (B3,C3) ARE THE SURROUNDING POINTS ON BLADE. A,B,C, ARE THE COEF
     2S'/5X ,'OF THE PARBOLIC CURVE Y=A+B*X+C*X*X',//,4X,'I',3X,'IP',5X,
     3'B1',8X,'C1',8X,'B2',8X,'C2',8X,'B3',8X,'C3',9X,'A',9X,'B',9X,'C',
     4//)
 244  CONTINUE
      IF(NP.EQ.1)  GO TO 245
      N5=N5-1
      N51=N5+1
      N52=N5+2
 245  CONTINUE
 720  FORMAT(/,2X,I2,1X,15F8.2,2(/5X,15F8.2))
 745  FORMAT(//,10X,'BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE  OF
     1 OUTER BOUNDARY'/,12X,'(START FROM I=',I3,' TO I=',I3,')')
 750  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE  OF
     1 OUTER BOUNDARY'/,12X,'(START FROM I=',I3,' TO I=',I3,')')
 755  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF
     1 INNER BOUNDARY'/,12X,'(START FROM I=',I3,'  TO  I=',I3,' )')
 760  FORMAT(//,10X,'BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF
     1 INNER BOUNDARY'/,12X,'(START FROM I=',I3,'  TO  I=',I3,' )')
 770  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
     1 OUTER BOUNDARY'/,12X,'(START FROM J= 2  TO  J=',I3,' )')
 780  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
     1 OUTER BOUNDARY'/,12X,'(START FROM J= 2  TO  J=',I3,' )')
 785  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF
     1INNER BOUNDARY '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
 790  FORMAT(//,10X,'BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF
     1INNER BOUNDARY '/12X,'(START FROM J=',I3,'  TO  J=',I3,' )')
 795  FORMAT(1H1)
 800  FORMAT(3(/,5X,15F8.2))
      RETURN
      END
```

```
      SUBROUTINE MESH(I,J,TNEW)
      COMMON    T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      COMMON DX,DY,NX,NY,NXC,NXI,ITER,N1,N2,N3,N4,NP,N5,N6,N7
C
C
C    SUBROUTINE FINDS SI-1,SI-2,DEL-U,DEL-2. FOR EACH INTERIOR POINT AND
C    GETS NEW TEMPS FOR EVERY INTERIOR POINT
C
C
      N51=N5+1
      PAR=FLOAT(I-1)*DX
      IF(NP.EQ.1)  GO TO 7
      IF(J.GT.N3.AND.J.LE.N6)  GO TO 6
      GO TO 7
    6 IF(I.GT.N51.AND.I.LE.N2)  GO TO 12
    7 CONTINUE
      IF((PAR-X1(J)).LE.DX)S1=(PAR-X1(J))/DX
      IF((PAR-X1(J)).LE.DX)T1=TBX(1,J)
      IF ((PAR-X1(J)).LE.DX) GO TO 20
      IF(J.GT.N3.AND.J.LT.N4)GO TO 10
       GO TO 15
   10 IF(PAR.GE.X4(J))GO TO 12
       GO TO 15
 12     IF((PAR-X4(J)).LE.DX)S1=(PAR-X4(J))/DX
      IF((PAR-X4(J)).LE.DX)T1=TBX(4,J)
      IF ((PAR-X4(J)).LE.DX) GO TO 20
   15  S1=1.0
      T1=T(I-1,J)
      IF(ABS(X1(J)-PAR+DX).LT.0.00001) T1=TBX(1,J)
      IF(ABS(X4(J)-PAR+DX).LT.0.00001) T1=TBX(4,J)
   20  CONTINUE
      IF((X2(J)-PAR).LE.DX)S2=(X2(J)-PAR)/DX
      IF((X2(J)-PAR).LE.DX) T3=TBX(2,J)
      IF((X2(J)-PAR).LE.DX) GO TO 35
      IF(J.GT.N3.AND.J.LT.N4)GO TO25
      GO TO 30
 25     IF(X3(J).GT.PAR)GO TO 27
      GO TO 30
 27     IF((X3(J)-PAR).LE.DX)S2=(X3(J)-PAR)/DX
      IF((X3(J)-PAR).LE.DX)T3=TBX(3,J)
      IF((X3(J)-PAR).LE.DX) GO TO 35
   30  S2=1.
      T3=T(I+1,J)
      IF(ABS(X2(J)-PAR-DX).LT.0.00001) T3=TBX(2,J)
      IF(ABS(X3(J)-PAR-DX).LT.0.00001) T3=TBX(3,J)
 35     PAR=FLOAT(J-1)*DY
      IF(NP.EQ.1)  GO TO 37
      IF(J.GT.N3.AND.J.LE.N6)  GO TO 36
```

```
      GO TO 37
36    IF(I.GT.N51.AND.I.LT.N2)   GO TO 42
37    CONTINUE
       IF((PAR-Y1(I)).LE.DY)D1=(PAR-Y1(I))/DY
       IF((PAR-Y1(I)).LE.DY)T2=TB(I,1)
      IF((PAR-Y1(I)).LE.DY) GO TO 50
       IF(I.GT.N1.AND.I.LT.N2)GO TO40
      GO TO 45
 40    IF(PAR.GE.Y4(I))GO TO42
      GO TO 45
 42    IF((PAR-Y4(I)).LE.DY)D1=(PAR-Y4(I))/DY
       IF((PAR-Y4(I)).LE.DY)T2=TB(I,4)
      IF((PAR-Y4(I)).LE.DY) GO TO 50
 45    D1=1.0
       T2=T(I,J-1)
      IF(ABS(Y1(I)-PAR+DY).LT.0.00001) T2=TB(I,1)
      IF(ABS(Y4(I)-PAR+DY).LT.0.00001) T2=TB(I,4)
 50    CONTINUE
       IF((Y2(I)-PAR).LE.DY)D2=(Y2(I)-PAR)/DY
       IF((Y2(I)-PAR).LE.DY)T4=TB(I,2)
      IF((Y2(I)-PAR).LE.DY) GO TO 65
       IF(I.GT.N1.AND.I.LT.N2)GO TO55
      GO TO 60
 55    IF(Y3(I).GT.PAR)GO TO57
      GO TO 60
 57    IF((Y3(I)-PAR).LE.DY)D2=(Y3(I)-PAR)/DY
       IF((Y3(I)-PAR).LE.DY)T4=TB(I,3)
      IF((Y3(I)-PAR).LE.DY) GO TO 65
 60    D2=1.0
       T4=T(I,J+1)
      IF(ABS(Y2(I)-PAR-DY).LT.0.00001) T4=TB(I,2)
      IF(ABS(Y3(I)-PAR-DY).LT.0.00001) T4=TB(I,3)
 65    CONTINUE
       A1=T1/S1/(S1+S2)
       A2=T2/D1/(D1+D2)
       A3=T3/S2/(S1+S2)
       A4=T4/D2/(D1+D2)
       E=1./S1/S2+(DX/DY)*(DX/DY)/D1/D2
       TNEW=(A1+A3+DX*DX/(DY*DY)*(A2+A4))/E
      RETURN
      END
```

```
      SUBROUTINE SLC(X1,Y1,X2,Y2,X3,Y3,A,B,C)
C
C
C    FINDS EQUATION THRU THREE POINTS AS Y=A+B*X+C*X*X
C
C
      CALL UNDFLW
      IF(ABS(X1-X2).LE.0.001.AND.ABS(X2-X3).LE.0.001) GO TO 10
      XS3=X3*X3
      XS2=X2*X2
      XS1=X1*X1
      D1=X2*XS3-X3*XS2
      D2=X1*XS3-X3*XS1
      D3=X1*XS2-X2*XS1
      D=D1-D2+D3
       IF(ABS(D).LE.0.000001) GO TO 10
      A1=Y1*(X2*XS3-X3*XS2)
      A2=Y2*(X1*XS3-X3*XS1)
      A3=Y3*(X1*XS2-X2*XS1)
      A=(A1-A2+A3)/D
      B1=Y2*XS3-Y3*XS2
      B2=Y1*XS3-Y3*XS1
      B3=Y1*XS2-Y2*XS1
      B=(B1-B2+B3)/D
      C1=X2*Y3-X3*Y2
      C2=X1*Y3-X3*Y1
      C3=X1*Y2-X2*Y1
      C=(C1-C2+C3)/D
      RETURN
 10   A=100.
      B=100.
      C=100.
       RETURN
      END
```

```
      SUBROUTINE CUR(B2,C2,B,C,I)
      COMMON    T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      COMMON DX,DY,NX,NY,NXO,NXI,ITER,N1,N2,N3,N4,NF,N5,N6,N7
C
C
C
C   SUBROUTINE TO FIND ADJACENT TWO POINTS FOR THE POINT(B2,C2) AND THEN
C   PASSES A CURVE Y=A+BX+CX2 AND DETERMINES A,B,C. B,C COEFFICIENTS
C   ARE RETURNED TO MAIN PROGRAM TO GET SLOPE OF CURVED BOUNDARY AT THE
C   POINT
C
C
C
      IF(I.EQ.1)GO TO 25
      DO 10 M=1,NXO
      IF(ABS(B2-BX(M)).LE.0.0001.AND.ABS(C2-BY(M)).LE.0.0001) GO TO 12
   10 CONTINUE
 12   IF(M.EQ.1) GO TO 15
      B1=BX(M-1)
      C1=BY(M-1)
      GO TO 13
 15   B1=BX(NXO)
      C1=BY(NXO)
 13   CONTINUE
      IF(M.EQ.NXO)  GO TO 100
      B3 = BX(M+1)
      C3 = BY(M+1)
 100  CONTINUE
      IF(M.EQ.NXO) B3=BX(1)
      IF(M.EQ.NXO) C3=BY(1)
      IF(B3.EQ.B2.AND.C3.EQ.C2)GO TO 14
      GO TO 37
   14 B3 = BX(M+2)
      C3 = BY(M+2)
      GO TO 37
   25 DO 30 M = 1,NXI
      IF(ABS(B2-BXI(M)).LE.0.0001.AND.ABS(C2-BYI(M)).LE.0.0001) GO TO 32
   30 CONTINUE
 32   IF(M.EQ.1) GO TO 34
      B1=BXI(M-1)
      C1=BYI(M-1)
      GO TO 33
 34    B1=BXI(NXI)
      C1=BYI(NXI)
 33   CONTINUE
      IF(M.EQ.NXI)  GO TO 200
      B3=BXI(M+1)
      C3=BYI(M+1)
```

```
200   CONTINUE
      IF(M.EQ.NXI) B3=BXI(1)
      IF(M.EQ.NXI) C3=BYI(1)
      IF(B3.EQ.B2.AND.C3.EQ.C2)GO TO 35
      GO TO 37
   35 B3=BXI(M+2)
      C3=BYI(M+2)
   37 CALL SLC(B1,C1,B2,C2,B3,C3,A,B,C)
      XMN=B+2.*C*B2
      IF(ITER.EQ.1) WRITE(6,40)I,M,B1,C1,B2,C2,B3,C3,A,B,C,XMN
  50  CONTINUE
   40 FORMAT(  3X,I2,I5,9F10.5,3X,'XMN=',F10.6)
      RETURN
      END
```

```
      SUBROUTINE TEMAD(I,J,TAD)
      DIMENSION VELG(30,65)
      COMMON    T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      COMMON DX,DY,NX,NY,NXC,NXI,ITER,N1,N2,N3,N4,NP,N5,N6,N7
      COMMON/TEM/  VELG,PRNO,TEMPC,CPC,VISCC,ALPHA,FLOC,DENSC
C
C
C
C  SUBROUTINE TO FIND ADIABATIC WALL TEMPERATURE
C
C  IN THE MAIN PROGRAM GAS TEMPERATURE AT POINTS DOWNSTREAM FROM THE SLO
C  BE REPLACED BY CORRESPONDING ADIABATIC WALL TEMPERATURE
C
C
      N51=N5+1
      REYNC=FLOC/VISCC
      IF(NP.EQ.1)  GO TO 6
      IF(I.EQ.1)  GO TO 10
      XRE=FLOAT(N51-1)*DX
      XP=FLOAT(I-1)*DX
      IF(I.GE.N2)  GO TO 5
      DIST=(((Y4(N51)-Y4(I))**2 +(XRE-XP)**2 )**0.5)/12.
      GO TO 9
    5 CONTINUE
      DIST=(((Y4(N51)-Y1(I))**2 +(XRE-XP)**2 )**0.5)/12.
      GO TO 9
    6 CONTINUE
      IF(I.EQ.1.OR.I.EQ.2)  GO TO 10
      XRE=FLOAT(N5 -1)*DX
      XP=FLOAT(I-1)*DX
      IF(I.GE.N2)  GO TO 8
      DIST=(((Y3(N5 )-Y3(I))**2+(XP-XRE)**2)**0.5)/12.
      GO TO 9
    8 DIST=(((Y3(N5 )-Y2(I))**2+(XP-XRE)**2)**0.5)/12.
    9 CONTINUE
      CPG=(9.87-36.1/(TG(I,J)**0.5)-2389./TG(I,J)+9.06E05/(TG(I,J)**2.))
     1/28.97
      VISCG=VISCC*((TG(I,J)/TEMPC)**1.5*(TEMPC+198.)/(TG(I,J)+198.))
      DENSG=DENSC*TEMPC/TG(I,J)
      GO TO 20
   10 CONTINUE
      IF(NP.EQ.1)  GO TO 16
      XRE=FLOAT(N51-1)*DX
      YP=FLOAT(J-1)*DY
      IF(J.LE.N3)  GO TO 15
      DIST=(((Y4(N51)-YP)**2+(XRE-X4(J))**2)**0.5)/12.
      GO TO 19
   15 CONTINUE
```

```
      DIST=(((Y4(N51)-YP)**2+(XRE-X1(J))**2)**0.5)/12.
      GO TO 19
 16   CONTINUE
      XRE=FLOAT(N5 -1)*DX
      YP=FLOAT(J-1)*DY
      IF(J.LE.N3)  GO TO 18
      DIST=(((Y3(N5 )-YP)**2+(X3(J)-XRE)**2)**0.5)/12.
      GO TO 19
 18   DIST=(((Y3(N5 )-YP)**2+(X2(J)-XRE)**2)**0.5)/12.
 19   CONTINUE
      CPG=(9.87-36.1/(TGX(I,J)**0.5)-2389./TGX(I,J)+9.06E05/(TGX(I,J)**2
     1.))/28.97
      VISCG=VISCC*((TGX(I,J)/TEMPC)**1.5*(TEMPC+198.)/(TGX(I,J)+198.))
      DENSG=DENSC*TEMPC/TGX(I,J)
 20   DIMP=DENSG*VELG(I,J)*DIST/FLCC
      BETA=1.+5E-04*REYNC*VISCC*SIN(ALPHA)/VISCG
      DNCM=1.+0.329*(REYNC)**(-0.2)*CPG/CPC*(VISCG/VISCC)**0.2*(DIMP)**(
     10.8)*BETA
      DNUME=1.9*PRNC**0.66666
      FCEFF=DNUME/DNCM
      IF(J.EQ.1.OR.J.EQ.2)  GO TO 30
      TAD=TGX(I,J)+FCEFF*(TEMPC-TGX(I,J))
      GO TO 40
30    TAD=TG(I,J)+FCEFF*(TEMPC-TG(I,J))
40    CONTINUE
      IF(TAD.LT.TEMPC)  TAD=TEMPC
50    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TEMP
      COMMON    T(30,65),TB(30,4),TBX(4,65),TG(30,4),TGX(4,65),
     1H(30,4),HX(4,65),Y1(30),Y2(30),Y3(30),Y4(30),X1(65),X2(65),
     2X3(65),X4(65),BX(200),BY(200),BXI(200),BYI(200),IP(200)
      COMMON DX,DY,NX,NY,NXC,NXI,ITER,N1,N2,N3,N4,NP,N5,N6,N7
C
C  ALONG I-LINE MIN TEMP STORED IN 'IL' AND MAX TEMP STORED IN 'IH'
C
      NX1=NX-1
      DO 987 I=2,NX1
      IF(I.GT.N1.AND.I.LT.N2) GO TO 967
      IA=IFIX(Y1(I)/DY+0.0001)+1
      ID=IFIX(Y2(I)/DY+0.0001)+2
      IT1=TB(I,1)
      IT2=TB(I,2)
      IL=IT1
      IH=IT2
      IF(IT1.GE.IT2) IL=IT2
      IF(IT1.GE.IT2) IH=IT1
      DO 955 J=IA,ID
      IF(T(I,J).LT.FLOAT(IL)) IL=IFIX(T(I,J))
      IF(T(I,J).GT.FLOAT(IH))IH=IFIX(T(I,J))
 950  CONTINUE
 955  CONTINUE
      DO 965 II=IL,IH
      DO 962 J=IA,ID
      IF((FLOAT(J )*DY).GT.Y2(I)) T(I,J+1)=TB(I,2)
      IF(Y1(I).GT.(FLOAT(J-1)*DY)) T(I,J)=TB(I,1)
      TV=FLOAT(II)
      IF(T(I,J).LT.TV.AND.T(I,J+1).GE.TV) GO TO 960
      IF(T(I,J).GE.TV.AND.T(I,J+1).LT.TV) GO TO 960
      GO TO 962
 960  CONTINUE
      RX=(T(I,J)-TV)/(T(I,J)-T(I,J+1))
      CALL ISOTH(II,I,J,DX,DY,RX)
      WRITE(6,972) I,J,II,T(I,J),T(I,J+1),RX
 962  CONTINUE
 965  CONTINUE
      GO TO 987
 967  CONTINUE
      IA=IFIX(Y1(I)/DY+0.0001)+1
      IB=IFIX(Y3(I)/DY+0.0001)+2
      IC=IFIX(Y4(I)/DY+0.0001)+1
      ID=IFIX(Y2(I)/DY+0.0001)+2
      IT1=TB(I,1)
      IT2=TB(I,3)
      IT3=TB(I,4)
      IT4=TB(I,2)
      IL=IT1
      IH=IT2
```

```
      IL1=IT3
      IH1=IT4
      IF(IT1.GE.IT2) IL=IT2
      IF(IT1.GE.IT2) IH=IT1
      IF(IT3.GE.IT4) IL1=IT4
      IF(IT3.GE.IT4) IH1=IT3
      DO 977 II=IL,IH
      DO 975 J=IA,IB
      TV=FLOAT(II)
      IF(Y1(I).GT.(FLOAT(J-1)*DY)) T(I,J)=TB(I,1)
      IF((FLOAT(J  )*DY).GT.Y3(I)) T(I,J+1)=TB(I,3)
      IF(T(I,J).LT.TV.AND.T(I,J+1).GE.TV) GO TO 970
      IF(T(I,J).GE.TV.AND.T(I,J+1).LT.TV) GO TO 970
      GO TO 975
970   RX=(T(I,J)-TV)/(T(I,J)-T(I,J+1))
      CALL ISOTH(II,I,J,DX,DY,RX)
      WRITE(6,972) I,J,II,T(I,J),T(I,J+1),RX
972   FORMAT(5X,3I5,3F11.4)
975   CONTINUE
977   CONTINUE
      DO 985 II=IL1,IH1
      DO 982 J=IC,ID
      TV=FLOAT(II)
      IF(Y4(I).GT.(FLOAT(J-1)*DY)) T(I,J)=TB(I,4)
      IF((FLOAT(J-1)*DY).GT.Y2(I)) T(I,J+1)=TB(I,2)
      IF(T(I,J).LT.TV.AND.T(I,J+1).GE.TV) GO TO 980
      IF(T(I,J).GE.TV.AND.T(I,J+1).LT.TV) GO TO 980
      GO TO 982
980   RX=(T(I,J)-TV)/(T(I,J)-T(I,J+1))
      CALL ISOTH(II,I,J,DX,DY,RX)
      WRITE(6,972) I,J,II,T(I,J),T(I,J+1),RX
982   CONTINUE
985   CONTINUE
987   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ISOTH(II,I,J,DX,DY,RX)
      DIMENSION XISO(30,30),YISO(30,30),TSOLT(30),NNY(30)
      INTEGER*4 TSOLT
      COMMON/ISO/ XISO,YISO,TSOLT,NSOLT,NNY
C
C
C   SUBROUTINE TO DETERMINE X-COORD AND Y-COORD OF POINTS ON A REQUIRED I
C
C
      DO 100  NN=1,NSOLT
      IF(TSOLT(NN)-II.EQ.0)  GO TO 200
      GO TO 100
  200 NNZ=NNY(NN) +1
      XISO(NN,NNZ)  =FLOAT(I-1)*DX
      YISO(NN,NNZ) =(FLOAT(J-1)+RX)*DY
      NNY(NN)=NNY(NN)+1
      GO TO 300
 100  CONTINUE
 300  RETURN
      END
      BLOCK DATA
      DIMENSION XISO(30,30),YISO(30,30),TSOLT(30),NNY(30)
      INTEGER*4 TSOLT
      COMMON/ISO/ XISO,YISO,TSOLT,NSOLT,NNY
      DATA  NNY,NSOLT,TSOLT/30*0,30,1900,1950,1960,2000,2020,2050,2080,2
     1100,2150,2200,2220,2235,2250,2260,2280,2300,2320,2350,2380,2400,24
     220,2440,2460,2480,2500,2520,2540,2550,2560,2570/
      END
```

SAMPLE OUTPUT CORRESPONDING TO

B3 COOLING CONFIGURATION

INPUT DATA

BLADE BOUNDARY POINTS COORDINATES

OUTER BOUNDARY			INNER BOUNDARY		
PT.NO.	X	Y	PT.NO.	X	Y
1	0.0	1.299999	1	0.100000	1.313999
2	0.024000	1.249999	2	0.114000	1.299999
3	0.050000	1.238000	3	0.150000	1.287999
4	0.100000	1.226999	4	0.200000	1.268999
5	0.150000	1.226000	5	0.250000	1.261000
6	0.200000	1.219000	6	0.285000	1.249999
7	0.250000	1.205000	7	0.300000	1.243999
8	0.264000	1.200000	8	0.350000	1.219000
9	0.300000	1.186999	9	0.381000	1.200000
10	0.350000	1.164000	10	0.400000	1.186999
11	0.371000	1.150000	11	0.450000	1.150000
12	0.400000	1.136000	12	0.498000	1.099999
13	0.429000	1.099999	13	0.500000	1.098000
14	0.450000	1.077999	14	0.538000	1.049999
15	0.475000	1.049999	15	0.550000	1.030999
16	0.500000	1.018999	16	0.572000	1.000000
17	0.514000	1.000000	17	0.600000	0.961000
18	0.550000	0.950000	18	0.607000	0.950000
19	0.585000	0.900000	19	0.643000	0.900000
20	0.600000	0.878000	20	0.650000	0.890000
21	0.621000	0.850000	21	0.678000	0.850000
22	0.650000	0.809000	22	0.700000	0.810000
23	0.657000	0.800000	23	0.713000	0.800000
24	0.693000	0.750000	24	0.745000	0.750000
25	0.700000	0.738000	25	0.750000	0.744000
26	0.726000	0.700000	26	0.779000	0.700000
27	0.750000	0.666000	27	0.800000	0.668000
28	0.763000	0.650000	28	0.811000	0.650000
29	0.798000	0.600000	29	0.843000	0.600000
30	0.800000	0.596000	30	0.850000	0.588000
31	0.833000	0.550000	31	0.875000	0.550000
32	0.850000	0.525000	32	0.900000	0.512000
33	0.869000	0.500000	33	0.908000	0.500000
34	0.900000	0.454000	34	0.938000	0.450000
35	0.903000	0.450000	35	0.950000	0.436000
36	0.940000	0.400000	36	0.972000	0.400000
37	0.950000	0.387000	37	1.000000	0.354000
38	0.976000	0.350000	38	1.002999	0.350000
39	1.000000	0.316000	39	1.033000	0.300000
40	1.013000	0.300000	40	1.049999	0.275000
41	1.049999	0.250000	41	1.063000	0.250000
42	1.091000	0.200000	42	1.099999	0.200000
43	1.099999	0.191000	43	1.138000	0.150000
44	1.138000	0.150000	44	1.150000	0.135000
45	1.150000	0.135000	45	1.181000	0.100000

46	1.181000	0.100000
47	1.200000	0.075000
48	1.224999	0.050000
49	1.249999	0.032000
50	1.270000	0.050000
51	1.249999	0.087000
52	1.240999	0.100000
53	1.210999	0.150000
54	1.200000	0.168000
55	1.181000	0.200000
56	1.150000	0.250000
57	1.122999	0.300000
58	1.099999	0.337000
59	1.092999	0.350000
60	1.063000	0.400000
61	1.049999	0.425000
62	1.037999	0.450000
63	1.009000	0.500000
64	1.000000	0.516000
65	0.982000	0.550000
66	0.956000	0.600000
67	0.950000	0.609000
68	0.929000	0.650000
69	0.901000	0.700000
70	0.900000	0.704000
71	0.877000	0.750000
72	0.851000	0.800000
73	0.850000	0.802000
74	0.827000	0.850000
75	0.803000	0.900000
76	0.800000	0.908000
77	0.781000	0.950000
78	0.758000	1.000000
79	0.750000	1.018000
80	0.736000	1.049999
81	0.713000	1.099999
82	0.700000	1.131000
83	0.691000	1.150000
84	0.670000	1.200000
85	0.650000	1.249999
86	0.625000	1.299999
87	0.600000	1.344999
88	0.598000	1.349999
89	0.564000	1.400000
90	0.550000	1.419000
91	0.523000	1.450000
92	0.500000	1.469000
93	0.453000	1.499999
94	0.450000	1.500999
95	0.400000	1.521000
96	0.350000	1.533000
97	0.300000	1.535999
98	0.250000	1.526999
99	0.200000	1.511000
100	0.180000	1.499999
101	0.150000	1.483000
102	0.104000	1.450000
103	0.100000	1.447000
104	0.050000	1.400000
105	0.013000	1.349999

46	1.183000	0.100000
47	1.150000	0.150000
48	1.113999	0.200000
49	1.099999	0.225000
50	1.085999	0.250000
51	1.058999	0.300000
52	1.049999	0.315000
53	1.033999	0.350000
54	1.009999	0.400000
55	1.000000	0.423000
56	0.986000	0.450000
57	0.961000	0.500000
58	0.950000	0.518000
59	0.933000	0.550000
60	0.905000	0.600000
61	0.900000	0.609000
62	0.876000	0.650000
63	0.850000	0.700000
64	0.822000	0.750000
65	0.800000	0.793000
66	0.798000	0.800000
67	0.773000	0.850000
68	0.750000	0.894000
69	0.748000	0.900000
70	0.724000	0.950000
71	0.700000	1.000000
72	0.677000	1.049999
73	0.654000	1.099999
74	0.650000	1.113000
75	0.633000	1.150000
76	0.610000	1.200000
77	0.600000	1.219999
78	0.585000	1.249999
79	0.561000	1.299999
80	0.550000	1.316000
81	0.528000	1.349999
82	0.500000	1.384999
83	0.485000	1.400000
84	0.450000	1.427999
85	0.413000	1.450000
86	0.400000	1.455000
87	0.350000	1.469000
88	0.300000	1.469999
89	0.250000	1.462999
90	0.213000	1.450000
91	0.200000	1.443999
92	0.150000	1.412999
93	0.138000	1.400000
94	0.100000	1.349999

```
INPUT DATA

GAS TEMPERATURE DISTRIBUTION  ON THE BLADE INNER AND OUTER SURFACE


    BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE  OF    OUTER BOUNDARY
      (START FROM I=  2 TO I= 26)

** 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00

    BOUNDARY POINTS ON I LINES FOR UPPER SURFACE  OF    OUTER BOUNDARY
      (START FROM I=  2 TO I= 26)

** 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00

    BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF     INNER BOUNDARY
      (START FROM I=  3  TO  I= 24 )

   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00
   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00

    BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF     INNER BOUNDARY
      (START FROM I=  3  TO  I= 24 )

   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00
   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00

    BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF    OUTER BOUNDARY
      (START FROM J= 2  TO  J= 31 )

   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00


    BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF   OUTER BOUNDARY
      (START FROM J= 2  TO  J= 31 )

   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00


    BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF   INNER BOUNDARY
      (START FROM J=  3  TO  J= 30 )

   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00
   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00

    BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF  INNER BOUNDARY
      (START FROM J=  3  TO  J= 30 )

   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00
   1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00 1540.00
```

INPUT DATA

HEAT COEFFICIENT DISTRIBUTION ON THE BLADE INNER AND OUTER SURFACE

BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE OF OUTER BOUNDARY
(START FROM I= 2 TO I= 26)

**	4.51	2.40	2.12	1.99	1.90	1.84	1.78	1.74	1.69	1.67	1.68	1.74	1.84	2.06	2.26
	2.38	2.53	2.81	3.27	3.85	4.31	4.69	4.94	5.17	5.28					

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF OUTER BOUNDARY
(START FROM I= 2 TO I= 26)

**	4.51	4.06	4.58	5.14	5.63	5.94	6.11	6.22	6.31	6.37	6.39	6.35	6.26	6.08	5.92
	5.76	5.67	5.57	5.47	5.40	5.33	5.28	5.24	5.21	5.21					

BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

9.03	8.19	5.56	5.42	5.49	5.56	5.42	5.56	5.21	3.47	2.08	1.39	1.39	1.53	1.72
1.91	2.08	2.36	2.57	2.78	2.78	2.78								

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

8.89	5.56	5.42	5.49	5.56	5.42	5.56	5.42	5.56	5.49	5.21	3.33	1.88	1.39	1.51
1.72	1.91	2.12	2.49	3.99	4.62	4.94								

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

5.24	5.10	4.86	4.61	4.31	3.99	3.63	3.06	2.81	2.57	2.47	2.37	2.26	2.13	2.04
1.88	1.78	1.72	1.68	1.67	1.67	1.71	1.77	1.88	6.28	6.67	6.30	4.51	4.06	5.00

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

5.21	5.21	5.21	5.22	5.24	5.26	5.28	5.31	5.35	5.38	5.44	5.47	5.51	5.56	5.60
5.66	5.72	5.78	5.82	5.88	5.94	6.01	6.11	6.18	6.26	6.32	6.35	6.39	6.39	6.31

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

2.78	2.78	2.78	2.78	2.78	2.64	2.49	2.29	2.10	1.98	1.88	1.75	1.61	1.51	1.39
1.39	1.39	1.88	2.78	3.85	5.21	5.56	5.49	5.56	8.68	8.89	6.94	5.42		

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

5.10	4.86	4.69	4.40	4.17	2.64	2.53	2.40	2.22	2.05	1.94	1.81	1.70	1.61	1.49
1.39	1.39	1.39	1.88	2.43	3.19	3.96	5.17	5.42	5.96	5.42	5.56	5.56		

INPUT DATA

COOLANT MASS FLOW DISTRIBUTION

BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 22)

0.0 4.4100 6.3700 6.3700 8.3300 8.3300 10.2900 10.2900 12.2500 12.2500 26.4600 26.4600 26.4600 26.4600 26.4600
26.4600 26.4600 26.4600 26.4600 26.4600

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 22)

0.0 4.4100 6.3700 6.3700 8.3300 8.3300 10.2900 10.2900 12.2500 14.2100 26.4600 26.4600 26.4600 26.4600 26.4600
26.4600 26.4600 26.4600 26.4600 26.4600

INPUT DATA

VELOCITIES AT POINTS WHERE ADIABATIC WALL TEMP IS TO BE DETERMINED

BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE OF OUTER BOUNDARY
(START FROM I= 22 TO I= 26)

1641.66 1835.00 2030.00 2266.00 2416.00

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 8)

2350.00 2150.00 1975.00 1796.66 1641.66 1408.00 1218.33

INPUT DATA

PHYSICAL PARAMETERS OF THE GRID SYSTEM

DX=0.050 DY=0.050 NXC= 105 NXI= 94

NX= 27 NY= 32 N1= 2 N2= 25

N3= 2 N4= 31 N5= 21 N6= 0 N7= 13 NP= 0

PROPERTIES OF THE BLADE AND THE COOLANT

LB=0.339000 XK=0.537500 CP=0.270200

PRAC= 0.72 ALPHA= 0.0 TCOLN= 1540.00 TEMPC= 1540.00

DENSC= 0.4513199925 CPC= 0.2685999870 VISCC=0.2690299080E-04 FLOC= 0.2602

ITERATION CHECKS

CHE= 1.000 SUMM= 0.020 NTE= 100 IMAX= 300

SUBROUTINE CURVE -OUTPUT-

I=0,OUTER BOUNDARY. I=1,INNER BOUNDARY
IP BOUNDARY POINT NUMBER

(B2,C2) IS THE POINT ON THE BOUNDARY. (B1,C1) AND (B3,C3) ARE THE SURROUNDING POINTS ON BLADE. A,B,C, ARE THE COEFS OF THE PARABOLIC CURVE Y=A+B*X+C*X*X

I	IP	B1	C1	B2	C2	B3	C3	A	B	C	
0	3	0.02400	1.25000	0.05000	1.23800	0.10000	1.22700	1.26489	-0.69665	3.17683	XMN= -0.378962
0	104	0.10000	1.44700	0.05000	1.40000	0.01300	1.35000	1.32936	1.64925	-4.72801	XMN= 1.176443
0	4	0.05000	1.23800	0.10000	1.22700	0.15000	1.22600	1.25900	-0.52005	1.99997	XMN= -0.120056
1	1	0.10000	1.35000	0.10000	1.31400	0.11400	1.30000	100.00000	100.00000	100.00000	XMN=119.999985
1	94	0.13800	1.40000	0.10000	1.35000	0.10000	1.31400	100.00000	100.00000	100.00000	XMN=119.999985
0	103	0.10400	1.45000	0.10000	1.44700	0.05000	1.40000	1.33539	1.46772	-3.51010	XMN= 0.765696
0	5	0.10000	1.22700	0.15000	1.22600	0.20000	1.21900	1.21100	0.28003	-1.19994	XMN= -0.079949
1	3	0.11400	1.30000	0.15000	1.28800	0.20000	1.26900	1.32872	-0.19011	-0.54251	XMN= -0.352866
1	92	0.20000	1.44400	0.15000	1.41300	0.13800	1.40000	1.09588	3.23474	-7.47174	XMN= 0.993222
0	101	0.18000	1.50000	0.15000	1.48300	0.10400	1.45000	1.34445	1.22121	-1.98394	XMN= 0.626027
0	6	0.15000	1.22600	0.20000	1.21900	0.25000	1.20500	1.20500	0.34999	-1.39997	XMN= -0.209995
1	4	0.15000	1.28800	0.20000	1.26900	0.25000	1.26100	1.41100	-1.15004	2.20009	XMN= -0.270001
1	91	0.21300	1.45000	0.20000	1.44400	0.15000	1.41300	1.24458	1.49955	-2.51628	XMN= 0.493033
0	99	0.25000	1.52700	0.20000	1.51100	0.18000	1.50000	1.28272	1.79873	-3.28668	XMN= 0.484061
0	7	0.20000	1.21900	0.25000	1.20500	0.26400	1.20000	1.21475	0.26279	-1.20549	XMN= -0.339958
1	5	0.20000	1.26900	0.25000	1.26100	0.28500	1.25000	1.21022	0.65697	-1.81558	XMN= -0.250813
1	69	0.30000	1.47000	0.25000	1.46300	0.21300	1.45000	1.24581	1.47579	-2.42940	XMN= 0.261087
0	98	0.30000	1.53600	0.25000	1.52700	0.20000	1.51100	1.37700	0.95017	-1.40001	XMN= 0.250165
0	9	0.26400	1.20000	0.30000	1.16700	0.35000	1.16400	1.20429	0.28724	-1.14973	XMN= -0.402599
1	7	0.28500	1.25000	0.30000	1.24400	0.35000	1.21900	1.23242	0.49882	-1.54046	XMN= -0.425460
1	68	0.35000	1.46900	0.30000	1.47000	0.25000	1.46300	1.30799	1.01994	-1.59976	XMN= 0.060080
0	97	0.35000	1.53300	0.30000	1.53600	0.25000	1.52700	1.30201	1.49987	-2.39964	XMN= 0.060081
0	10	0.30000	1.18700	0.35000	1.16400	0.37100	1.15000	1.01939	1.43274	-2.91105	XMN= -0.604996
1	8	0.30000	1.24400	0.35000	1.21900	0.38100	1.20000	1.24761	0.40591	-1.39434	XMN= -0.570126
1	67	0.40000	1.45500	0.35000	1.46900	0.30000	1.47000	1.20299	1.67014	-2.60023	XMN= -0.150015
0	96	0.40000	1.52100	0.35000	1.53300	0.30000	1.53600	1.36496	1.11009	-1.80008	XMN= -0.149967
0	12	0.37100	1.15000	0.40000	1.13600	0.42900	1.10000	-0.61078	9.59536	-13.07059	XMN= -0.861109
1	10	0.38100	1.20000	0.40000	1.18700	0.45000	1.15000	1.33760	-0.05368	-0.80883	XMN= -0.700739
1	66	0.41300	1.45000	0.40000	1.45500	0.35000	1.46900	1.33443	0.96731	-1.66220	XMN= -0.362450
0	95	0.45000	1.50100	0.40000	1.52100	0.35000	1.53300	1.39298	0.96017	-1.60012	XMN= -0.319927
0	14	0.42900	1.10000	0.45000	1.07800	0.47500	1.05000	1.24523	0.33498	-1.56896	XMN= -1.077091
1	11	0.40000	1.18700	0.45000	1.15000	0.49800	1.10000	0.92885	1.87681	-3.07883	XMN= -0.894144
1	64	0.48500	1.40000	0.45000	1.42600	0.41300	1.45000	1.16536	1.86765	-2.85358	XMN= -0.700575
0	94	0.45300	1.50000	0.45000	1.50100	0.40000	1.52100	1.90804	-1.47368	1.26316	XMN= -0.336842
0	16	0.47500	1.05000	0.50000	1.01900	0.51400	1.00000	0.92663	1.67391	-2.98696	XMN= -1.313042
1	13	0.49800	1.10000	0.50000	1.09800	0.53800	1.05000	-0.04059	5.56863	-6.58823	XMN= -1.019607
1	62	0.52600	1.35000	0.50000	1.38500	0.48500	1.40000	0.47525	4.72607	-5.81518	XMN= -1.089108
0	92	0.52300	1.45000	0.50000	1.46500	0.45300	1.50000	1.25980	1.60835	-2.37983	XMN= -0.771473
0	18	0.51400	1.00000	0.55000	0.95000	0.58500	0.90000	1.55580	-0.79414	-0.55896	XMN= -1.408993
1	15	0.53800	1.05000	0.55000	1.03100	0.57200	1.00000	3.41223	-7.14569	5.11920	XMN= -1.514571
1	60	0.56100	1.30000	0.55000	1.31600	0.52800	1.35000	2.97744	-4.54135	2.76692	XMN= -1.497745
0	90	0.56400	1.40000	0.55000	1.41900	0.52300	1.45000	0.58389	4.31923	-5.09615	XMN= -1.286537
0	20	0.58500	0.90000	0.60000	0.87800	0.62100	0.85000	3.04908	-5.83246	3.68586	XMN= -1.409425
1	17	0.57200	1.00000	0.60000	0.96100	0.60700	0.95000	0.04511	4.59130	-5.10435	XMN= -1.533912
1	77	0.61000	1.20000	0.60000	1.22000	0.58500	1.25000	2.40039	-1.98438	-0.01563	XMN= -2.003124
0	87	0.62500	1.30000	0.60000	1.34500	0.59800	1.35000	12.46875	-34.54544	26.72726	XMN= -2.472733
0	22	0.62100	0.85000	0.65000	0.80900	0.65700	0.80000	3.17572	-5.96721	3.57377	XMN= -1.321313
1	20	0.64300	0.90000	0.65000	0.89009	0.67800	0.85000	1.81919	-1.41739	-0.00870	XMN= -1.428695
1	74	0.65400	1.10000	0.65000	1.11300	0.63300	1.15000	-18.47298	63.29166	-51.04166	XMN= -3.062485
0	85	0.67000	1.20000	0.65000	1.25000	0.62500	1.30000	-1.96834	12.18037	-11.12732	XMN= -2.285142

129

```
ITERATION=  1    ERROR= 0.18564E 07
ITERATION=  2    ERROR= 0.65904E 06
ITERATION=  3    ERROR= 0.23859E 06
ITERATION=  4    ERROR= 0.12171E 06
ITERATION=  5    ERROR= 0.75727E 05
ITERATION=  6    ERROR= 0.54934E 05
ITERATION=  7    ERROR= 0.41626E 05
ITERATION=  8    ERROR= 0.32352E 05
ITERATION=  9    ERROR= 0.25437E 05
ITERATION= 10    ERROR= 0.20119E 05
ITERATION= 11    ERROR= 0.15992E 05
ITERATION= 12    ERROR= 0.12751E 05
ITERATION= 13    ERROR= 0.10198E 05
ITERATION= 14    ERROR= 0.81741E 04
ITERATION= 15    ERROR= 0.65669E 04
ITERATION= 16    ERROR= 0.52856E 04
ITERATION= 17    ERROR= 0.42614E 04
ITERATION= 18    ERROR= 0.34408E 04
ITERATION= 19    ERROR= 0.27819E 04
ITERATION= 20    ERROR= 0.22517E 04
ITERATION= 21    ERROR= 0.18242E 04
ITERATION= 22    ERROR= 0.14787E 04
ITERATION= 23    ERROR= 0.11991E 04
ITERATION= 24    ERROR= 0.97247E 03
ITERATION= 25    ERROR= 0.78858E 03
ITERATION= 26    ERROR= 0.63925E 03
ITERATION= 27    ERROR= 0.51008E 03
ITERATION= 28    ERROR= 0.41946E 03
ITERATION= 29    ERROR= 0.33936E 03
ITERATION= 30    ERROR= 0.27432E 03
ITERATION= 31    ERROR= 0.22151E 03
ITERATION= 32    ERROR= 0.17876E 03
ITERATION= 33    ERROR= 0.14406E 03
ITERATION= 34    ERROR= 0.11599E 03
ITERATION= 35    ERROR= 0.93221E 02
ITERATION= 36    ERROR= 0.74891E 02
ITERATION= 37    ERROR= 0.60119E 02
ITERATION= 38    ERROR= 0.48218E 02
ITERATION= 39    ERROR= 0.38641E 02
ITERATION= 40    ERROR= 0.30946E 02
ITERATION= 41    ERROR= 0.24762E 02
ITERATION= 42    ERROR= 0.19801E 02
ITERATION= 43    ERROR= 0.15827E 02
ITERATION= 44    ERROR= 0.12647E 02
ITERATION= 45    ERROR= 0.10107E 02
ITERATION= 46    ERROR= 0.80685E 01
ITERATION= 47    ERROR= 0.64427E 01
ITERATION= 48    ERROR= 0.51459E 01
ITERATION= 49    ERROR= 0.41113E 01
ITERATION= 50    ERROR= 0.32771E 01
ITERATION= 51    ERROR= 0.26174E 01
ITERATION= 52    ERROR= 0.20938E 01
ITERATION= 53    ERROR= 0.16717E 01
ITERATION= 54    ERROR= 0.13345E 01
ITERATION= 55    ERROR= 0.10656E 01
ITERATION= 56    ERROR= 0.85039E 00
ITERATION= 57    ERROR= 0.68103E 00
ITERATION= 58    ERROR= 0.54222E 00
ITERATION= 59    ERROR= 0.43458E 00
ITERATION= 60    ERROR= 0.34733E 00
```

T-MATRIX,GIVES ALL INTERIOR POINTS.INITIAL SETTING OF T IS 1600.0
ONLY THOSE INSIDE BLADE ARE CHANGED AND WILL BE DIFFERENT FROM 1600.00

J = / I	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	16	17	18	19	20	21	22	23	24	25					
1	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2333.60	2400.00	2400.00	2400.00
	2400.00	2400.00													
2	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2255.86	2181.92	2145.69	2099.66	2400.00
	2400.00	2400.00													
3	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2101.74	1992.52	1875.01	1996.63	2400.00
	2400.00	2400.00													
4	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2002.86	2400.00	2400.00	2400.00	2095.87
	2400.00	2400.00													
5	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1958.21	2400.00	2400.00	2400.00	2087.57
	2235.28	2400.00													
6	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1931.79	2400.00	2400.00	2400.00	2400.00
	2229.88	2400.00													
7	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1953.31	2400.00	2400.00	2400.00	2400.00	2400.00
	2231.77	2400.00													
8	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1912.25	2400.00	2400.00	2400.00	2400.00	2400.00
	2253.51	2400.00													
9	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1906.22	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2250.63	2400.00													
10	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1903.84	1853.27	2400.00	2400.00	2400.00	2400.00	2400.00	2226.25
	2342.81	2400.00													
11	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	1931.67	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2196.80	2310.24
	2400.00	2400.00													
12	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2400.00	2060.61	2066.41	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2199.46	2308.93	2400.00
	2400.00	2400.00													
13	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2400.00	2109.41	2090.76	2400.00	2400.00	2400.00	2400.00	2400.00	2158.02	2240.16	2400.00	2400.00	2400.00
	2400.00	2400.00													
14	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00	2180.76	2400.00	2400.00	2400.00	2400.00	2400.00	2256.12	2298.96	2296.84	2400.00	2400.00	2400.00	2400.00
	2400.00	2400.00													

GAS TEMPERATURE DISTRIBUTION ON THE BLADE INNER AND OUTER SURFACE

BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE OF OUTER BOUNDRY
(START FROM I= 2 TO I= 26)

```
** 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 1765.57 1765.57 1765.57 1865.15 1927.43
```

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF OUTER BOUNDRY
(START FROM I= 2 TO I= 26)

```
** 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
   2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
```

BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

```
1540.00 1601.52 1608.47 1626.05 1617.64 1629.92 1623.05 1633.28 1624.99 1635.81 1671.22 1688.05 1702.34 1712.81 1721.87
1731.61 1741.95 1753.34 1765.55 1540.00 1540.00 1540.00
```

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

```
1540.00 1605.92 1609.84 1636.13 1633.60 1654.86 1650.34 1669.84 1667.47 1669.75 1671.22 1688.05 1702.34 1712.61 1721.87
1731.61 1741.95 1753.34 1765.55 1765.57 1765.57 1765.57
```

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

```
1900.16 1827.17 1765.57 1765.57 1765.57 1765.57 1765.57 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
```

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

```
2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00 2760.00
```

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

```
1540.00 1540.00 1540.00 1540.00 1765.55 1765.55 1753.34 1741.95 1741.95 1731.61 1721.87 1721.87 1712.81 1702.34 1702.34
1688.05 1671.22 1671.22 1635.81 1624.99 1633.28 1623.05 1629.92 1626.05 1540.00 1540.00 1540.00 1608.47
```

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

```
1765.57 1765.57 1765.57 1765.57 1765.57 1765.57 1765.57 1765.55 1765.55 1753.34 1753.34 1741.95 1741.95 1731.61 1721.87
1721.87 1712.81 1712.81 1712.81 1702.34 1702.34 1688.05 1688.05 1671.22 1671.22 1669.75 1667.47 1669.84
```

TEMPERATURE DISTRIBUTION ON THE BOUNDARY

BOUNDARY POINTS ON I-LINES FOR LOWER SURFACE OF OUTER BOUNDRY
(START FROM I= 2 TO I= 26)

** 2271.97 2133.44 2040.33 2005.50 1999.09 1970.72 1958.56 1920.24 1927.80 1971.12 2068.61 2139.47 2225.48 2246.73 2295.20
2295.95 2313.64 2352.89 2365.85 2760.01 2400.00 2400.00 2400.00 2246.49 2287.32

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF OUTER BOUNDRY
(START FROM I= 2 TO I= 26)

** 2099.64 2099.64 2199.12 2267.69 2306.02 2330.80 2342.52 2346.23 2345.40 2352.12 2342.27 2278.67 2296.84 2354.52 2523.31
2548.49 2562.91 2534.89 2488.94 2446.68 2379.60 2318.13 2332.20 2341.65 2376.58

BOUNDARY POINTS ON I LINES FOR LOWER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

1964.58 1917.97 1925.76 1914.34 1892.72 1887.71 1871.10 1853.27 1874.44 1962.43 2077.11 2140.73 2210.82 2228.76 2256.72
2259.58 2320.23 2327.38 2662.64 2400.00 2400.00 2400.00

BOUNDARY POINTS ON I LINES FOR UPPER SURFACE OF INNER BOUNDARY
(START FROM I= 3 TO I= 24)

1875.01 1982.40 2070.26 2122.10 2141.22 2164.90 2162.32 2173.86 2167.65 2119.41 2124.46 2261.97 2436.80 2474.10 2485.85
2460.25 2413.16 2365.13 2319.93 2221.64 2222.48 2208.61

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

2281.42 2400.00 2400.00 2400.00 2400.00 2400.00 2564.93 2381.37 2381.37 2353.27 2304.79 2295.57 2295.57 2270.96 2236.10
2236.10 2182.47 2104.04 2104.04 2019.87 1949.46 1924.02 1939.40 1984.91 2340.34 2333.60 2352.32 2099.64 2099.58 2237.61

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF OUTER BOUNDARY
(START FROM J= 2 TO J= 31)

2367.41 2354.66 2369.99 2345.94 2350.45 2360.00 2349.90 2408.46 2412.40 2457.69 2447.26 2496.53 2492.20 2538.22 2536.21
2571.14 2545.72 2577.34 2535.35 2564.75 2527.55 2503.89 2393.78 2388.13 2302.56 2322.78 2301.92 2345.98 2350.44 2345.13

BOUNDARY POINTS ON J LINES FOR NEARER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

2400.00 2400.00 2400.00 2400.00 2400.00 2760.01 2333.97 2334.32 2341.52 2271.05 2249.23 2265.81 2245.57 2202.35 2215.18
2173.39 2090.37 2094.58 2000.93 1902.20 1854.12 1844.94 1887.03 1918.08 1936.84 1875.01 1926.65 2084.42

BOUNDARY POINTS ON J LINES FOR FARTHER SURFACE OF INNER BOUNDARY
(START FROM J= 3 TO J= 30)

2206.44 2210.31 2212.65 2195.57 2226.05 2299.52 2325.77 2358.36 2369.60 2408.16 2416.14 2458.68 2462.01 2491.67 2462.07
2498.15 2448.19 2474.57 2437.95 2341.32 2273.00 2187.54 2142.20 2108.27 2131.96 2155.52 2158.29 2166.33

TEMPERATURE DISTRIBUTION INSIDE THE BLADE AND ON ITS SURFACE

J=								
31	2237.61	2235.28	2229.88	2231.77	2253.91	2290.63	2342.81	2345.19
30	2099.58	2095.87	2087.57	2064.42				
		#####	2166.33	2226.25	2310.24	2350.44		
29	2099.64	1996.63	1926.65					
		#####	2158.29	2196.60	2308.93	2345.90		
28	2352.32	2145.69	1875.01					
		#####	2155.52	2199.46	2301.92			
27	2333.60	2181.92	1992.52	1936.84				
		#####	2131.96	2240.16	2322.78			
26	2340.34	2255.86	2101.74	2002.86	1958.21	1931.79	1918.08	
		#####	2108.27	2158.02	2302.56			
25	1984.91	1953.31	1912.25	1887.03				
		#####	2142.20	2258.96	2388.13			
24	1939.40	1906.22	1844.94					
		#####	2187.54	2256.12	2353.78			
23	1924.02	1903.84	1854.12					
		#####	2273.00	2432.48	2503.89			
22	1949.46	1931.67	1902.20					
		#####	2341.32	2419.07	2527.55			
21	2019.87	2006.41	2000.93					
		#####	2437.95	2538.22	2564.75			
20	2104.04	2090.76	2094.58					
		#####	2474.57	2502.74	2539.39			
19	2104.04	2109.41	2090.37					
		#####	2448.19	2456.76	2563.56	2577.34		
18	2182.47	2180.76	2173.39					
		#####	2498.15	2520.93	2543.72			

ISOTHERMAL LINE LOCATIONS

I	J	T	T - 1	T - 2	FRAC
2	28	2100	2145.6931	2099.6396	0.9922
2	28	2101	2145.6931	2099.6396	0.9705
2	28	2102	2145.6931	2099.6396	0.9487
2	28	2103	2145.6931	2099.6396	0.9270
2	28	2104	2145.6931	2099.6396	0.9053
2	28	2105	2145.6931	2099.6396	0.8836
2	28	2106	2145.6931	2099.6396	0.8619
2	28	2107	2145.6931	2099.6396	0.8402
2	28	2108	2145.6931	2099.6396	0.8185
2	28	2109	2145.6931	2099.6396	0.7968
2	28	2110	2145.6931	2099.6396	0.7750
2	28	2111	2145.6931	2099.6396	0.7533
2	28	2112	2145.6931	2099.6396	0.7316
2	28	2113	2145.6931	2099.6396	0.7099
2	28	2114	2145.6931	2099.6396	0.6882
2	28	2115	2145.6931	2099.6396	0.6665
2	28	2116	2145.6931	2099.6396	0.6448
2	28	2117	2145.6931	2099.6396	0.6230
2	28	2118	2145.6931	2099.6396	0.6013
2	28	2119	2145.6931	2099.6396	0.5796
2	28	2120	2145.6931	2099.6396	0.5579
2	28	2121	2145.6931	2099.6396	0.5362
2	28	2122	2145.6931	2099.6396	0.5145
2	28	2123	2145.6931	2099.6396	0.4928
2	28	2124	2145.6931	2099.6396	0.4710
2	28	2125	2145.6931	2099.6396	0.4493
2	28	2126	2145.6931	2099.6396	0.4276
2	28	2127	2145.6931	2099.6396	0.4059
2	28	2128	2145.6931	2099.6396	0.3842
2	28	2129	2145.6931	2099.6396	0.3625
2	28	2130	2145.6931	2099.6396	0.3408
2	28	2131	2145.6931	2099.6396	0.3190
2	28	2132	2145.6931	2099.6396	0.2973
2	28	2133	2145.6931	2099.6396	0.2756
2	28	2134	2145.6931	2099.6396	0.2539
2	28	2135	2145.6931	2099.6396	0.2322
2	28	2136	2145.6931	2099.6396	0.2105
2	28	2137	2145.6931	2099.6396	0.1888
2	28	2138	2145.6931	2099.6396	0.1670
2	28	2139	2145.6931	2099.6396	0.1453
2	28	2140	2145.6931	2099.6396	0.1236
2	28	2141	2145.6931	2099.6396	0.1019
2	28	2142	2145.6931	2099.6396	0.0802
2	28	2143	2145.6931	2099.6396	0.0585
2	28	2144	2145.6931	2099.6396	0.0368
2	28	2145	2145.6931	2099.6396	0.0151
2	27	2146	2181.9180	2145.6931	0.9915
2	27	2147	2181.9180	2145.6931	0.9639
2	27	2148	2181.9180	2145.6931	0.9363
2	27	2149	2181.9180	2145.6931	0.9087
2	27	2150	2181.9180	2145.6931	0.8811
2	27	2151	2181.9180	2145.6931	0.8535
2	27	2152	2181.9180	2145.6931	0.8259
2	27	2153	2181.9180	2145.6931	0.7983
2	27	2154	2181.9180	2145.6931	0.7707
2	27	2155	2181.9180	2145.6931	0.7431
2	27	2156	2181.9180	2145.6931	0.7155

THE ISOTHERM AT 1900

X-COORD	Y-COORD
0.3000	1.2440
0.3500	1.2250
0.4000	1.1589
0.4500	1.1038
0.5000	1.0777

THE ISOTHERM AT 1950

X-COORD	Y-COORD
0.1500	1.2811
0.2000	1.2626
0.2500	1.2365
0.3000	1.2027
0.3500	1.1552
0.5000	1.0268

THE ISOTHERM AT 1960

X-COORD	Y-COORD
0.1500	1.2752
0.2000	1.2481
0.2500	1.2290
0.3000	1.1808
0.5000	1.0141

THE ISOTHERM AT 2000

X-COORD	Y-COORD
0.1000	1.2966
0.1000	1.4041
0.1500	1.2517
0.1500	1.4078
0.2000	1.2058
0.5500	1.0073

THE ISOTHERM AT 2020

X-COORD	Y-COORD
0.1000	1.2874
0.1000	1.4064
0.1500	1.2271
0.1500	1.4166
0.5500	0.9891

THE ISOTHERM AT 2050

X-COORD	Y-COORD
0.1000	1.2737
0.1000	1.4098
0.1500	1.4290
0.5500	0.9650

EFFICIENCY OF COOLING

```
BCUNCARY POINTS CN I-LINES FCR LCWER SURFACE  OF    OUTER BCUNCRY
  (START FRCM I= 2  TO  I= 26 )

0   0.4000  0.5136  0.5899  0.6184  0.6237  0.6469  0.6569  0.6883  0.6821  0.6466  0.5667  0.5086  0.4381  0.4207  0.3810
    0.3804  0.3659  0.3009  0.3198 -0.0000  0.2951  0.2951  0.2951  0.4209  0.3874

BCUNCARY POINTS CN I LINES FOR UPPER SURFACE  OF    OUTER BCUNCRY
  (START FRCM I= 2  TC I= 26 )

**  0.3218  0.5413  0.4597  0.4035  0.3721  0.3518  0.3422  0.3392  0.3398  0.3343  0.3424  0.3945  0.3796  0.3324  0.1940
    0.1734  0.1615  0.1845  0.2222  0.2568  0.3116  0.3622  0.3507  0.3425  0.3143

BCUNCARY PCINTS CN J LINES FOR NEARER SURFACE OF    OUTER BOUNDARY
  (START FRCM J= 2  TC  J= 31 )

    0.3923  0.2951  0.2951  0.2951  0.2951  0.2951  0.1599  0.3103  0.3103  0.3334  0.3731  0.3807  0.3807  0.4009  0.4294
    0.4294  0.4734  0.5377  0.5377  0.6067  0.6644  0.6852  0.6726  0.6353  0.3440  0.3495  0.3342  0.5413  0.5413  0.4282

BCUNCARY POINTS ON J LINES FOR FARTHER SURFACE OF   OUTER BOUNDARY
  (START FRCM J= 2  TC  J= 31 )

    0.3218  0.3322  0.3197  0.3394  0.3357  0.3279  0.3361  0.2881  0.2849  0.2478  0.2563  0.2160  0.2195  0.1818  0.1834
    0.1548  0.1756  0.1497  0.1809  0.1600  0.1905  0.2099  0.3002  0.3048  0.3749  0.3584  0.3755  0.3394  0.3357  0.3401
```

APPENDIX B

HEAT BALANCE EQUATIONS TO OBTAIN COOLANT TEMPERATURES

In this appendix, the equations governing the coolant temperature distribution will be derived. A one-dimensional flow analysis is performed down each inter tube-vane passage, account being taken of the area changes, mass addition due to additional impingement from the flanks of the tube and heat addition from the vane walls. The temperature of the coolant inside the impingement tube is assumed to be constant.

Referring to Figure 16, I=N1 and I=N7 are the I mesh lines enclosing the impingement tube without intersecting it and N71 = N7-1. To derive equations governing the temperature distributions, consider the control volumes in the coolant flow path. A typical control volume is made up of two consecutive I mesh lines and two surfaces. The four types of control volumes (CV1, CV2, CV3 and CV4) which are considered to calculate the coolant temperature rise are shown in Figure 17. The sketches illustrating the mass balance and heat balance for control volumes are shown in Figures 18 and 19 respectively.

Consider first the control volume CV1, shown in Figure 19. The heat balance equation for this system can be expressed semantically as:

Influx of enthalpy at I mesh line	+	Influx of enthalpy from the impingement tube	+	Rate of heat inflow from the vane wall

= Efflux of enthalpy at (I+1) mesh line

or mathematically as

$$\dot{m}(i,3)C_p\ TG(i,3) + \Delta\dot{m}(i,3)C_p\ TCOLN$$

$$+ h(i,3)\ [TB(i,3) - TG(i,3)]\ \Delta s(i,3)\ LB$$

$$= \dot{m}(i+1,3)C_p\ TG(i+1,3) \quad \text{(b.1)}$$

where $\dot{m}(i,3)$ = total mass of coolant entering the control volume at I mesh line per unit time (see Figure 18).

C_p = specific heat of coolant at constant pressure.

$TG(i,3)$ = coolant temperature at I mesh line intersection point on the inner lower surface.

$\Delta\dot{m}(i,3)$ = coolant mass added to the control volume between I and I+1 mesh lines from the impingement tube, per unit time.

TCOLN = temperature of the coolant in the impingement tube.

$h(i,3)$ = heat transfer coefficient at I mesh line intersection point on the blade inner lower surface.

$TB(i,3)$ = blade temperature at I mesh line intersection point on the blade inner lower surface.

$\Delta s(i,3)$ = surface distance between consecutive points on inner lower surface intersected by I mesh lines, (see Figure 18).

LB = length of the blade.

$\dot{m}(i+1,3)$ = total mass of coolant leaving the control volume at (I+1) mesh line per unit time.

$TG(i+1,3)$ = coolant temperature at (I+1) mesh line intersection point on the inner lower surface.

The equation (b.1) can be written in a more convenient form as follows:

$$TG(i+1,3) = \frac{\dot{m}(i,3)C_p\ TG(i,3) + \Delta\dot{m}(i,3)C_p\ TCOLN + Q(i,3)\Delta s(i,3)LB}{\dot{m}(i+1,3)C_p} \qquad (b.2)$$

where $Q(i,3) = h(i,3)\ [TB(i,3) - TG(i,3)]$ (b.3)

Following the same procedure described above, a similar equation can be obtained for the control volume CV2 shown in Figure 19 as follows:

$$TG(i+1,4) = \frac{\dot{m}(i,4)C_p\ TG(i,4) + \Delta\dot{m}(i,4)C_p\ TCOLN + Q(i,4)\Delta s(i,4)LB}{\dot{m}(i+1,4)C_p} \qquad (b.4)$$

where the second subscript 'four' for the variables appearing in equation (b.4) refers to the conditions at a point on the upper inner surface.

Referring to Figure 19, the blade inner upper and lower surfaces and the mesh lines I=N7 and I=N71 define the boundaries for the control volume CV3. The two streams of coolant coming from the lower and upper surfaces of the impingement tube merge at I=N71 line. The heat balance equation for this system (CV3) can be written as follows:

$$\begin{aligned} \dot{m}(i,3)C_p\ TG(i,3) + \dot{m}(i,4)C_p\ TG(i,4) + Q(i,3)\ \Delta s(i,3)LB \\ + Q(i,4)\ \Delta s(i,4)LB \\ = \dot{m}(i+1,3)C_p\ TG(i+1,3) \\ = \dot{m}(i+1,4)C_p\ TG(i+1,4) \qquad (b.5) \end{aligned}$$

Since, for $I \geq N71$, $\dot{m}(i+1,3) = \dot{m}(i+1,4)$, equation (b.5) can be rewritten as follows:

$$TG(i+1,3 = TG(i+1,4)$$

$$= \frac{[\dot{m}(i,3)C_pTG(i,3)+\dot{m}(i,4)C_pTG(i,4)+Q(i,3)\Delta s(i,3)LB+Q(i,4)\Delta s(i,4)LB]}{\dot{m}(i+1,3)C_p} \qquad (b.6)$$

Finally, the heat balance equation for the system CV4 shown in Figure 19 can be written as follows:

$$\dot{m}(i,3)C_p\ TG(i,3) + Q(i,3)\ \Delta s(i,3)LB + Q(i,4)\ \Delta s(i,4)LB$$

$$= \dot{m}(i+1,3)C_p\ TG(i+1,3) \qquad (b.7)$$

Since, for $I > N71$, $\dot{m}(i+1,3) = \dot{m}(i+1,4)$, equation (b.7) can be rewritten as follows:

$$TG(i+1,3) = TG(i+1,4) =$$

$$= \frac{\dot{m}(i,3)C_p\ TG(i,3) + Q(i,3)\ \Delta s(i,3)LB + Q(i,4)\ \Delta s(i,4)LB}{\dot{m}(i+1,3)C_p} \qquad (b.8)$$

The equations (b.2), (b.4), (b.6) and (b.8) are incorporated in the main program to calculate the coolant temperature rise over an incremental length Δs.