MULTIPOLE METHOD TO COMPUTE THE CONDUCTIVE HEAT FLOWS TO AND BETWEEN PIPES IN A COMPOSITE CYLINDER

Johan Bennet
Johan Claesson
Göran Hellström

January 1987

Dep. of Building Technology and Mathematical Physics

CODEN: LUTVDG/(TVBH-7094)/1-42/(1987)
MULTIPOLE METHOD TO COMPUTE THE CONDUCTIVE HEAT FLOWS TO AND BETWEEN PIPES IN A COMPOSITE CYLINDER.

Johan Bennet, Johan Claesson, Göran Hellström

Lund Institute of Technology
January 1987
CONTENTS

1 INTRODUCTION 3
2 THERMAL PROBLEM 4
3 LINE SOURCE FOR THE COMPOSITE REGION 5
4 MULTipoLES FOR THE COMPOSITE REGION 8
5 GENERAL EXPRESSION FOR THE TEMPERATURE 10
6 EQUATIONS FROM THE BOUNDARY CONDITION AT THE OUTER CIRCLE 11
7 EQUATIONS FROM THE BOUNDARY CONDITION AT THE PIPES 13
8 FINAL EQUATIONS AND ITERATIVE SOLUTION 18
9 REQUIRED NUMBER OF MULTipoLES 21
  9.1 Error on the boundary circles 21
  9.2 A three-pipe problem 22
  9.3 Test of Maxwell's reciprocity theorem 23
  9.4 An example with 15 pipes 24
10 COMPUTER MODEL. EXECUTION TIMES 25
11 MANUAL FOR THE COMPUTER CODE 26

REFERENCES 31

Appendix 1. LISTING OF COMPUTER CODE 32
1. INTRODUCTION

A so-called multipole method to compute the conductive heat flows to and between pipes in a cylinder is presented in [1]. The steady-state heat conduction is two-dimensional in a circular region perpendicular to the pipes. The region is homogeneous. In this paper the method is extended to the case of a composite cylinder. The pipes lie in an inner circular region, which is surrounded by an annulus of different thermal conductivity.

The applications for which the method has been developed concern so-called heat extraction boreholes and certain types of heat stores in the ground. The boreholes are used for heat extraction or as heat exchangers for heat injection/extraction. The heat carrier fluid may for example flow in a U-shaped tube in the borehole. Outside the tubes the borehole may be filled with sand. The local thermal problem in and near the borehole is quite important for the heat transfer capacity of the heat exchanger. This problem is essentially a steady-state one in the region between the fluid in the pipes and a suitably chosen cylinder around the borehole. The pipes are imbedded in a composite cylinder with an internal boundary at the borehole wall. The thermal resistance between the pipes and the outer circle is of particular interest. We are also interested in the thermal resistances between the pipes. The presented method and the computer program give a very rapid and accurate way to obtain these resistances and the complete temperature field. The method is used extensively in [2].

![Figure 1.1. An example showing the temperature field for a case with 3 pipes. Data according to (8.12).](image)

Figure 1.1 shows the computed temperature field for an example with three pipes. The fluid temperatures in the pipes and on the outer circle are indicated in the figure. One of the pipes has as indicated a thermal resistance layer. The complete set of data for the example is given by (8.12). The computer
time for this case is only a few seconds on a main frame computer (Norsk Data ND-500) or about 40 seconds on an IBM-PC AT-3 (10 MHz) with a 80287 math coprocessor.

2. THERMAL PROBLEM

Figure 2.1 shows the considered thermal problem. There are $N$ pipes ($N \geq 1$), which lie within the inner circular region with the radius $r_b$ (b = borehole). The inner circle is surrounded by an annular region of another material. The outer circle has the radius $r_c$.

![Figure 2.1 Steady-state heat conduction in a composite circular region with heat flows between the N pipes and the outer circle.](image)

The outer radius of pipe $n$ is $r_{pn}$, and its center lies at $(x_n, y_n)$. The fluid in pipe $n$ has the constant temperature $T_{In}$, while the temperature outside the outer circle is $T_c$.

The annular region, $r_b < r < r_c$, is homogeneous with the thermal conductivity $\lambda$. The inner circular region outside the pipes has the thermal conductivity $\lambda_b$. The steady-state temperature $T(x, y)$ satisfies the heat conduction equation in the annular region and in the inner circle:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (2.1)$$

Cartesian, complex and polar coordinates will be used:

$$z = x + iy = re^{i\phi} \quad (2.2)$$

The center of pipe $n$ is in complex coordinates:

$$z_n = x_n + iy_n \quad (2.3)$$

We will use the local polar coordinates $\rho_n, \psi_n$ from the center of any pipe $n$. See Figure 2.2.

$$z - z_n = \rho_n e^{i\psi_n} \quad (2.4)$$
Figure 2.2. Local polar coordinates $\rho_n, \psi_n$ from the center of pipe $n$.

The temperature and the radial heat flux are continuous at the inner boundary $r = r_b$:

$$T|_{r_b-0} = T|_{r_b+0}$$

$$\lambda_b \frac{\partial T}{\partial r} |_{r_b-0} = \lambda \frac{\partial T}{\partial r} |_{r_b+0}$$ (2.5)

The boundary conditions at the pipes and the outer circle $r = r_c$ are the same as in [1]. The boundary condition at pipe $n$ is:

$$T - \beta_n r_p n \frac{\partial T}{\partial \rho_n} = T_f n \quad \rho_n = r_p n \quad 0 \leq \psi_n \leq 2\pi$$ (2.6)

The boundary condition at the outer circle is:

$$T + \beta_c r_c \frac{\partial T}{\partial \psi_c} = T_c \quad r = r_c \quad 0 \leq \phi \leq 2\pi$$ (2.7)

The dimensionless coefficient $\beta_n$ determines the thermal resistance between the fluid in pipe $n$ and the material just outside the pipe. This resistance is $\beta_n r_p n / \lambda_b$ (K/(W/m$^2$)). The corresponding thermal resistance $R_p n$ (K/(W/m)) per unit pipe length is obtained by division with the perimeter $2\pi r_p n$:

$$R_p n = \frac{\beta_n}{2\pi \lambda_b} \quad (K/(W/m))$$ (2.8)

The thermal resistance per unit length (in the axial direction) of the outer circle is:

$$R_p e = \frac{\beta_c}{2\pi \lambda} \quad (K/(W/m))$$ (2.9)

The thermal resistance per unit area at the outer circle is $\beta_c r_c / \lambda = 2\pi r_c R_p e$ (K/(W/m$^2$)). The thermal resistance coefficients $\beta_n$ and $\beta_c$ may take any non-negative value: $0 \leq \beta_n \leq \infty, 0 \leq \beta_c \leq \infty$. The value $\beta_n = +\infty$ means zero heat flux, $\frac{\partial T}{\partial \rho_n} = 0$. The value of $T_f n$ is then redundant.

3. LINE SOURCE FOR THE COMPOSITE REGION

The solution in the previous paper [1] for pipes in a homogeneous region is based on the line source solution in complex form. Suitable derivatives gave the multipoles, which were needed for the solution. We will here in the same way use the line source for a composite region.

The thermal problem for the basic line source solution is shown in Figure 3.1. The thermal conductivity is $\lambda_b$ in the circle $0 \leq r < r_b$, and $\lambda$ in the infinite region outside the circle, $r_b \leq r < \infty$. There is a line source with the strength $+q_n$ (W/m) at the point $(x_n, y_n)$, which lies within the circle $(r_n < r_b)$.

The solution to this problem is given in [2].
0 \leq r \leq r_b:

\begin{equation}
T(x, y) = \frac{q_n}{2\pi \lambda_b} \left\{ \ln \left( \frac{r_b}{\sqrt{(x-x_n)^2 + (y-y_n)^2}} \right) + \sigma \cdot \ln \left( \frac{r_b^2/r_n}{\sqrt{(x-x_n)^2 + (y-y_n)^2}} \right) \right\} \tag{3.1}
\end{equation}

\begin{equation}
r_b \leq r < \infty:
T(x, y) = \frac{q_n}{2\pi \lambda} \left\{ (1 - \sigma) \ln \left( \frac{r_b}{\sqrt{(x-x_n)^2 + (y-y_n)^2}} \right) + \sigma \cdot \ln \left( \frac{r_b}{\sqrt{x^2 + y^2}} \right) \right\} \tag{3.2}
\end{equation}

We have introduced the notation:

\begin{equation}
\sigma = \frac{\lambda_b - \lambda}{\lambda_b + \lambda} \quad (-1 < \sigma < 1) \tag{3.3}
\end{equation}

The case \( \sigma = 0 \), i.e. \( \lambda_b = \lambda \), is the one studied in [1].

The first term in (3.1) represents a line sink \( +q_n \) at \((x_n, y_n)\) in a material with the thermal conductivity \( \lambda_b \). The second term is due to the fact that the thermal conductivity is \( \lambda \) for \( r > r_b \). This term represents a line sink with the strength \( \sigma q_n \) situated at the mirror point \((x_n r_n^2/r_b^2, y_n r_n^2/r_b^2)\). The mirror point lies on the same radius as \((x_n, y_n)\). The product of the distances to the center is \( r_n r_b / r_n = r_b^2 \). The temperature field (3.2) in the outer region \( r \geq r_b \) consists of a line sink with the strength \( q_n (1 - \sigma) \) at \((x_n, y_n)\) and another one with the remaining heat \( q_n \cdot \sigma \) at the \((0,0)\). It shall be observed that the thermal conductivity is \( \lambda \) in (3.2) and \( \lambda_b \) in (3.1).

The solution (3.1-2) is rewritten in the complex form using the following expressions:

\begin{equation}
\ln \left( \frac{r_b}{\sqrt{(x-x_n)^2 + (y-y_n)^2}} \right) = \text{Re} \left[ \ln \left( \frac{r_b}{x-x_n} \right) \right] \tag{3.4}
\end{equation}

\begin{equation}
\ln \left( \frac{r_b}{\sqrt{x^2 + y^2}} \right) = \text{Re} \left[ \ln \left( \frac{r_b}{x} \right) \right] \tag{3.5}
\end{equation}
\[
\ln \left( \frac{r_b^2/r_n}{\sqrt{(x-x_nr_b^2/r_n)^2 + (y-y_nr_b^2/r_n)^2}} \right)
\]

\[
= \ln \left( \frac{r_b^2/|z_n|}{|z-x_nr_b^2/r_n^2|} \right) = \ln \left( \frac{r_b^2}{|z_n| \cdot |z-x_nr_b^2/(z_nz_n)|} \right)
\]

\[
= \ln \left( \frac{r_b^2}{|z_n-z_nz_n|} \right) = \ln \left( \frac{r_b^2}{|r_b^2-z_nz_n|} \right) = \text{Re} \left[ \ln \left( \frac{r_b^2}{r_b^2-z_nz_n} \right) \right]
\]

In the last two lines we used that \( r_b^2 = |z_n|^2 = z_nz_n \) and \( |w| = |\bar{w}| \) for any complex number \( w \) and its conjugate \( \bar{w} \).

The temperature (3.1-2) may now be written in the following form:

\[
T(x, y) = \frac{q_n}{2\pi\lambda_b} \cdot \text{Re}[W_{no}]
\]

The complex-valued function \( W_{no} \) is from (3.1-7) defined by:

\[
W_{no} = \begin{cases}
\ln \left( \frac{r_b}{z-x_n} \right) + \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2-z_nz_n} \right) & r \leq r_b \\
(1+\sigma) \ln \left( \frac{r_b}{z-x_n} \right) + \frac{\lambda \sigma}{\lambda} \ln \left( \frac{r_b}{z} \right) & r \geq r_b
\end{cases}
\]

The factor \( 1+\sigma \) in (3.9) is in accordance with (3.3) equal to \( \lambda_b(1-\sigma)/\lambda \). The simplicity of the expressions (3.7-9) is noteworthy.

Let us verify that (3.7-9) is indeed the solution to the fundamental line source problem for the composite region shown in Figure 3.1. The expression (3.9) is a regular (analytic) function in the region \( r > r_b \), since the sources lie within the circle. Its real part \( \text{Re}[W_{no}] \) satisfies the heat conduction equation \( \nabla^2 T = 0 \).

The expression (3.8) is also a regular function in \( r < r_b \) except at the point \( z = z_n \). It remains to verify that the boundary conditions (2.5) at \( r = r_b \) are satisfied. We have at the circle \( z = r_b \cdot e^{i\phi} \):

\[
W_{no}|_{r=r_b} = \ln \left( \frac{r_b}{r_b \cdot e^{i\phi} - z_n} \right) + \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2 - r_b e^{-i\phi} \cdot z_n} \right)
\]

\[
= \ln \left( \frac{r_b}{r_b \cdot e^{i\phi} - z_n} \right) + \sigma \cdot \ln \left( \frac{r_b \cdot e^{i\phi}}{r_b e^{i\phi} - z_n} \right)
\]

\[
W_{no}|_{r=r_b} = (1+\sigma) \ln \left( \frac{r_b}{r_b \cdot e^{i\phi} - z_n} \right) + \frac{\lambda \sigma}{\lambda} \ln \left( e^{-i\phi} \right)
\]

The difference is

\[
W_{no}|_{r=r_b} - W_{no}|_{r=r_b} = \sigma \cdot i\phi - \frac{\lambda}{\lambda} \cdot (-i\phi)
\]

or

\[
W_{no}|_{r=r_b} - W_{no}|_{r=r_b} = \frac{\lambda_b - \lambda}{\lambda} \cdot i\phi
\]

This difference contains only an imaginary part. The temperature, which is given by the real part, is therefore continuous at \( r = r_b \).

For the radial heat flux at \( r = r_b \), we have:

\[
\lambda_b \frac{\partial W_{no}}{\partial r} \bigg|_{r=r_b} = \lambda_b \left\{ \frac{-e^{i\phi}}{r_b e^{i\phi} - z_n} + \sigma \cdot \frac{(-z_n)e^{-i\phi}}{r_b^2 - r_b e^{-i\phi} \cdot z_n} \right\}
\]

\[
= \frac{\lambda_b}{r_b} \left( \frac{r_b e^{i\phi} - \sigma z_n}{r_b e^{i\phi} - z_n} \right)
\]

7
\[
\lambda \frac{\partial W_{n_0}}{\partial r} \bigg|_{r_b+0} = \lambda \left\{ (1+\sigma) \cdot \frac{-e^{i\phi}}{r_b e^{i\phi} - z_n} + \frac{\lambda_b}{\lambda} \cdot \frac{-e^{i\phi}}{r_b e^{i\phi} - z_n} \right\} \\
= \ldots = -\frac{\lambda_b}{r_b} \cdot \frac{r_b e^{i\phi} - \sigma z_n}{r_b e^{i\phi} - z_n}
\]

(3.14)

The heat flux, i.e. the real parts of (3.13-14), is therefore continuous at \( r = r_b \). This is in fact true also for the imaginary part of \( W_{n_0} \):

\[
\lambda_b \frac{\partial W_{n_0}}{\partial r} \bigg|_{r_b-0} = \lambda \frac{\partial W_{n_0}}{\partial r} \bigg|_{r_b+0}
\]

(3.15)

### 4. Multipoles for the Composite Region

The multipoles of order \( j \) were in complex form given by \((z - z_n)^{-j}\) in the case of a single region, [1]. They were obtained by derivatives of the complex logarithm \( \ln(z - z_n) \). Let \( W_{n_1} \) be the derivative of \( W_{n_0} \), (3.8-9), with respect to the complex variable \( z_n \):

\[
W_{n_1} = \frac{\partial}{\partial z_n} (W_{n_0})
\]

(4.1)

We get from (3.8) and (3.9):

\[
W_{n_1} = \frac{1}{z - z_n} + \sigma \cdot \frac{z}{r_b^2 - \bar{z}z_n} \quad r \leq r_b
\]

(4.2)

\[
W_{n_1} = (1 + \sigma) \cdot \frac{1}{z - z_n} \quad r \geq r_b
\]

(4.3)

The function \( W_{n_1} \) is regular. Thus, the real and imaginary parts, \( Re(W_{n_1}) \) and \( Im(W_{n_1}) \), both satisfy \( \nabla^2 T = 0 \), except at the point \( z = z_n \). At the boundary \( r = r_b \) the function \( W_{n_0} \) satisfies (3.12) and (3.15). The identities are valid for any \( z_n \), and hence for the derivative with respect to \( z_n \). But the derivative of \( \phi (=\arctan(y/x)) \) in (3.12) with respect to \( z_n \) is zero. Therefore we have:

\[
W_{n_1}|_{r_b-0} = W_{n_1}|_{r_b+0}
\]

(4.4)

\[
\lambda_b \frac{\partial W_{n_1}}{\partial r} \bigg|_{r_b-0} = \lambda \frac{\partial W_{n_1}}{\partial r} \bigg|_{r_b+0}
\]

(4.5)

This means that \( W_{n_1} \) and the radial heat flux are continuous at \( r = r_b \). This is true both for \( Re(W_{n_1}) \) and \( Im(W_{n_1}) \).

The multipole of any order \( j \) is given by the \( j \):th derivative of \( W_{n_0} \) with respect to \( z_n \). We define \( W_{n_j} \) by:

\[
W_{n_j} = \frac{1}{(j-1)!} \frac{\partial^j}{\partial z_n^j} (W_{n_0})
\]

(4.6)

From (3.8) and (3.9) we get the neat expressions:

\[
\begin{align*}
W_{n_j} &= \frac{1}{(z - z_n)^j} + \sigma \cdot \left( \frac{z}{r_b^2 - \bar{z}z_n} \right)^j \quad r \leq r_b \\
W_{n_j} &= (1 + \sigma) \cdot \frac{1}{(z - z_n)^j} \quad r \geq r_b
\end{align*}
\]

(4.7-8)

The expressions (4.7-8) are regular functions, which implies that the real and imaginary parts satisfy \( \nabla^2 T = 0 \), except at \( z = z_n \). The boundary conditions (4.4) and (4.5) may be derived with respect to \( z_n \) any number of times. This means that \( W_{n_j} \) and the radial heat flux are continuous at \( r = r_b \) for any \( j \):
The multipole (4.7) expressed in local polar coordinates around \( Z = Z_n \) becomes:

\[
W_{nj} = \frac{1}{(\rho_n e^{i\psi_n})^{\frac{1}{2}}} + \sigma \cdot \left( \frac{x_n + \rho_n e^{-i\psi_n}}{r_b^2 - r_n^2 - \rho_n e^{-i\psi_n} \cdot z_n} \right)^j
\]  
(4.11)

or

\[
W_{nj} = (\cos(j\psi_n) - i \cdot \sin(j\psi_n)) \cdot \rho_n^{-j} + \sigma \cdot \left( \frac{x_n + \rho_n e^{-i\psi_n}}{r_b^2 - r_n^2 - \rho_n e^{-i\psi_n} \cdot z_n} \right)^j \quad (\rho_n < r_b - r_n)
\]  
(4.12)

The first part represents a pure multipole behaviour, while the second part is a correction to account for the effect of the different thermal conductivity \( \lambda \) for \( r > r_b \). The real part of \( W_{nj} \), \( Re(W_{nj}) \), gives a variation \( \cos(j\psi_n) \) around the point \( z = z_n \), and the imaginary part, \( Im(W_{nj}) \), gives a variation \( \sin(j\psi_n) \). The temperature field from the general multipole of order \( j \) at \( Z = Z_n \) is:

\[
T = Re\left( P_{nj} \cdot W_{nj} \right) = Re\left( (c_{nj} + i \cdot s_{nj}) W_{nj} \right) = c_{nj} \cdot Re(W_{nj}) - s_{nj} \cdot Im(W_{nj})
\]  
(4.13)

Here \( P_{nj} \) is an arbitrary complex number.

The line sink and multipoles at \( z_n \) can be used to represent an arbitrary temperature solution outside pipe \( n \). We need a corresponding representation for the outer boundary circle \( r = r_c \). Here we need a solution in the composite region inside \( r = r_c \). The solution in complex form shall vary as \( e^{i \cdot j \phi} \) on \( r = r_c \). We have the corresponding solutions to \( \nabla^2 T = 0 \) in polar coordinates in \( r_b < r < r_c \):

\[
r^j \cdot e^{i \cdot j \phi} = z^j \quad r^{-j} e^{i \cdot j \phi} = \frac{1}{z^j}
\]  
(4.14)

In the inner region \( 0 \leq r \leq r_b \) we can use only the first solution \( z^j \), since the second one is infinite at \( r = 0 \) or \( z = 0 \).

Therefore we start with the following expressions:

\[
W_{cj} = A \cdot z^j \quad 0 \leq r \leq r_b
\]

\[
W_{cj} = z^j + B/z^j \quad r_b \leq r \leq r_c
\]  
(4.15)

Continuity at \( z = r_b e^{i \phi} \) requires:

\[
A r_b^j = r_b^j + B/r_b^j
\]  
(4.16)

The radial heat flux is also continuous at \( z = r_b e^{i \phi} \):

\[
\lambda_b \cdot \frac{\partial W_{cj}}{\partial r} \bigg|_{r_b} = \lambda_b A \cdot j r_b^{j-1} \cdot e^{i \cdot j \phi} = \lambda \cdot \frac{\partial W_{cj}}{\partial r} \bigg|_{r_b}
\]

\[
= \lambda \cdot j \cdot e^{i \cdot j \phi} \left( r_b^{j-1} - B/r_b^{j+1} \right)
\]  
(4.17)

The constants \( A \) and \( B \) are determined by (4.16-17). This gives the following basic expressions:

\[
\begin{cases}
W_{cj} = (1 - \sigma) z^j & r \leq r_b \\
W_{cj} = z^j - \sigma \cdot (r_b^2/z^j)^j & r \geq r_b
\end{cases}
\]  
(4.18)
We will call these solutions $W_{cj}$ multipoles at infinity. The index $c$ refers to the fact that they are needed at the outer boundary $r = r_c$. The multipole of order $j$ can represent the variation $e^{ij\phi}$ or in real form any combination of $\cos(j\phi)$ and $\sin(j\phi)$ at $r = r_c$.

5. GENERAL EXPRESSION FOR THE TEMPERATURE

The general expression for the temperature field uses all the line sinks and the multipoles of all orders at the pipes and at infinity. We have the following general expression:

$$T = T_0 + \text{Re} \left[ \sum_{n=1}^{N} P_n \cdot W_{no} + \sum_{n=1}^{N} \sum_{j} P_{nj} \cdot r_j^m \cdot W_{nj} + \sum_{j} P_{cj} r_c^{-j} W_{cj} \right]$$

(5.1)

Here $W_{no}$ is given by (3.8-9), $W_{nj}$ by (4.7-8) and $W_{cj}$ by (4.18). The factors $r_j^m$ and $r_c^{-j}$ are introduced for dimensional reasons. The dimension of $P_n$, $P_{nj}$, and $P_{cj}$ is that of a temperature.

The strength of the line source $q_n$ is in accordance with (3.7) related to $P_n$ by:

$$P_n = \frac{q_n}{2\pi A}$$

(5.2)

The multipole factors $P_{nj}$ and $P_{cj}$ are complex numbers, while $P_n$ of course is real. The temperature $T_0$ is an arbitrary constant. The summation in $j$ runs from 1 to infinity for an exact solution, and from 1 to $J$ in the truncated approximate solution. The multipoles are not used in the case $J = 0$.

The general temperature (5.1) becomes with the explicit formulas (3.8-9), (4.7-8) and (4.18):

$r \leq r_b$:

$$T = T_0 + \text{Re} \left[ \sum_{n=1}^{N} P_n \left\{ \ln \left( \frac{r_b}{z - z_n} \right) + \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2 - 2z_n} \right) \right\} + \sum_{n=1}^{N} \sum_{j} P_{nj} \left\{ \left( \frac{r_{pn}}{z - z_n} \right)^j + \sigma \left( \frac{r_{pn}^2}{r_b^2 - 2z_n} \right)^j \right\} + \sum_{j=1}^{N} P_{cj} (1 - \sigma) \left( \frac{z}{r_c} \right)^j \right]$$

(5.3)

$r_b \leq r \leq r_c$:

$$T = T_0 + \text{Re} \left[ \sum_{n=1}^{N} P_n \left\{ 1 + \sigma \cdot \ln \left( \frac{r_b}{z - z_n} \right) + \frac{\lambda_b}{\lambda} \sigma \cdot \ln \left( \frac{r_b}{z} \right) \right\} + \sum_{n=1}^{N} \sum_{j=1}^{N} P_{nj} \left( \frac{r_{pn}}{z - z_n} \right)^j \right]$$

(5.4)

The general expressions (5.1) or (5.3-4) satisfy the heat conduction equation (2.1) everywhere except at the points $z = z_n$ for any choice of the real constants $T_0$, $P_n$ and of the complex ones, $P_{nj}$ and $P_{cj}$. Each term satisfies the boundary conditions (2.5) at $r = r_b$. The constants $T_0$ and $P_{cj}$ are chosen so that the boundary condition (2.7) at the outer circle is fulfilled. This is done in the next chapter, while Chapter 7 deals with the boundary conditions at the pipes. We will get a set of equations for $T_0$, $P_n$, $P_{nj}$ and $P_{cj}$.

The method of solution presented in this paper and in [1] is essentially to use Fourier series expansions; one for each pipe and one for the outer circle. The problem is to transform the coordinates of the different
components into polar coordinates for each pipe and for the outer circle. A basic idea in the present
method is to use the complex form, which greatly facilitates the derivation of the equation system for
the boundary conditions.

6. EQUATIONS FROM THE BOUNDARY CONDITION
AT THE OUTER CIRCLE

The expressions (5.3) and (5.4) shall satisfy the boundary conditions (2.6-7). In the previous study
[1], each type of term was analysed separately (in Chapter 5 of [1]). We will here use a slightly different
approach. The temperature (5.3) is first expressed directly in the local polar coordinates of the considered
pipe. Then the expression is inserted in the boundary condition (2.6), and we get an equation system
that determines the line sinks $P_n$ and the multipoles $P_n$. This is done in the next chapter.

In this chapter we consider the boundary condition (2.7) at the outer circle $r = r_e$. Our first goal is to
express the temperature (5.4) for the outer region $r_e < r < r_c$ in polar coordinates: $T = T(r, \phi)$. This is
achieved by putting $z = re^{i\phi}$ in (5.4). We will need to distinguish between different powers $(e^{i\phi})^k$ or $z^k$
in the subsequent analysis. The functions of $z$ in (5.4) are therefore expanded in Taylor series.

For the logarithm of (5.4) we have:

$$\ln \left( \frac{r_b}{z - z_n} \right) = \ln \left( \frac{r_b}{z} \right) + \ln \left( \frac{1}{1 - z_n/z} \right)$$  \hspace{1cm} (6.1)

We need the Taylor series:

$$\ln \left( \frac{1}{1 - z} \right) = \sum_{k=1}^{\infty} \frac{1}{k} z^k \hspace{1cm} |z| < 1$$  \hspace{1cm} (6.2)

Then we have:

$$\ln \left( \frac{r_b}{z - z_n} \right) = \ln \left( \frac{r_b}{z} \right) + \sum_{k=1}^{\infty} \frac{1}{k} (z_n/z)^k \hspace{1cm} |z_n| < |z|$$  \hspace{1cm} (6.3)

We also need a Taylor expansion of the multipole term $(r_{pm}/(z - z_n))^j$ in (5.4). We have:

$$\left( \frac{r_{pm}}{z - z_n} \right)^j = \left( \frac{r_{pm}}{z} \right)^j \sum_{k'=0}^{\infty} \frac{j + k'-1}{j-1} \left( \frac{z_n}{z} \right)^{k'} \hspace{1cm} |z_n| < |z|$$  \hspace{1cm} (6.4)

Here we have used a Taylor series given in [1]:

$$\frac{1}{(1 - z)^j} = \sum_{k=0}^{\infty} \left( \frac{j + k - 1}{j - 1} \right) s^k \hspace{1cm} |s| < 1$$  \hspace{1cm} (6.5)

The multipoles (6.4) are summed over $j$ in (5.4). We need to rearrange the double sum in the following
way:

$$\sum_{j=1}^{\infty} P_{nj} \left( \frac{r_{pm}}{z - z_n} \right)^j = \sum_{j=1}^{\infty} \sum_{k'=0}^{\infty} P_{nj} \left( \frac{r_{pm}}{z} \right)^j \left( \frac{z_n}{z} \right)^{k'}$$

$$= \sum_{k=1}^{\infty} P_{nj} \sum_{j=1}^{\infty} \frac{(k-1)}{j-1} \frac{r_{pm}^j z_{k-j}}{z^k}$$  \hspace{1cm} \hspace{1cm} (6.6)

The change of summation from $j, k'$ to $k, j$ means that $j$ varies between 1 and $k$ for $k = 1, 2, \ldots$.

We insert (6.3) and (6.6) in (5.4) and put $z = re^{i\phi}$. Consider first the terms containing $ln(z)$. For
these we have from (5.4) and (6.3):

11
In the last line we used the definition (3.3) for $\sigma$. The temperature as a function of $r$ and $\phi$ is now using (5.4), (6.3), (6.6) and (6.7) with $z$ equal to $re^{i\phi}$:

\[
T(r, \phi) = T_o + \sum_{n=1}^{N} P_n \frac{\lambda_b}{\lambda} \ln \left( \frac{r_b}{r} \right) + \Re \left[ \sum_{n=1}^{N} \sum_{k=1}^{\infty} P_n (1 + \sigma) \frac{1}{k} \left( \frac{2 r}{r_n} \right)^k e^{-ik\phi} \right] + \sum_{n=1}^{N} \sum_{k=1}^{\infty} \sum_{j=1}^{k} P_n (1 + \sigma) \left( \frac{k-1}{j-1} \right) \frac{r_j^2 n^2 - j}{r^k} \cdot e^{-ik\phi} + \sum_{k=1}^{\infty} P_{ck} \left[ \left( \frac{r}{r_c} \right)^k - \sigma \left( \frac{r^2}{r_c^2} \right)^k \right] e^{ik\phi} \right] \quad (r_b \leq r \leq r_c)
\]

The summation index is in the last line changed from $j$ to $k$.

The temperature (6.8) shall satisfy the boundary condition (2.7). For the radial derivative of any power of $r$ we have:

\[
\beta_c r_c \frac{\partial}{\partial r} \left( r^k \right) \bigg|_{r=r_c} = \beta_c r_c \cdot k \cdot r_c^{k-1} = k \beta_c \cdot r_c^k
\]

\[
\beta_c r_c \frac{\partial}{\partial r} \left( r^{-k} \right) \bigg|_{r=r_c} = -k \beta_c \cdot r_c^{-k}
\]

We will separate different $k$, i.e. different orders in $e^{-ik\phi}$. The last line of (6.8) has a positive exponent, which is changed by complex conjugation:

\[
\Re \left[ P_{ck} \cdot e^{ik\phi} \right] = \Re \left[ P_{ck} \cdot e^{-ik\phi} \right]
\]

The boundary condition (2.7) may now be written:

\[
T_c = \left. \left( T + \beta_c \frac{\partial T}{\partial r} \right) \right|_{r=r_c} = T_o + \sum_{n=1}^{N} P_n \frac{\lambda_b}{\lambda} \left\{ \ln \left( \frac{r_b}{r_c} \right) + \beta_c r_c \cdot \left( \frac{1}{r_c} \right) \right\} + \sum_{n=1}^{N} \sum_{k=1}^{\infty} P_n (1 + \sigma) \frac{1}{k} \left( \frac{2 r}{r_n} \right)^k e^{-ik\phi} \left[ \sum_{n=1}^{N} \sum_{k=1}^{\infty} P_n (1 + \sigma) \left( \frac{k-1}{j-1} \right) \frac{r_j^2 n^2 - j}{r^k} \cdot e^{-ik\phi} \right]
\]

Equation (6.11) is valid for $0 \leq \phi \leq 2\pi$. The constant part (independent of $\phi$) must vanish. This determines $T_o$:

\[
T_o = T_c + \sum_{n=1}^{N} P_n \frac{\lambda_b}{\lambda} \left( \ln \left( \frac{r_c}{r_b} \right) + \beta_c \right)
\]

This equation relates the temperature level $T_o$ to the temperature $T_c$ at the outer circle and the line sources $P_n$.

The remaining part of (6.11) is an equation of the following type:
\[
0 = \sum_{k=1}^{\infty} \text{Re} \left[ e^{i k \phi} \cdot Z_k \right] \\
= \sum_{k=1}^{\infty} \{ \cos(k\phi) \cdot \text{Re}(Z_k) + \sin(k\phi) \cdot \text{Im}(Z_k) \} \quad 0 \leq \phi \leq 2\pi
\]  

All coefficients before \( \cos(k\phi) \) and \( \sin(k\phi) \) must vanish. This means that the complex factor \( Z_k \) is zero for all \( k \). We have the following equations:

\[
k = 1, 2, \ldots : \\
P_{ck} \cdot \left\{ 1 - \sigma \frac{1 - k\beta_c}{1 + k\beta_c} \left( \frac{r_b}{r_c} \right)^{2k} \right\} + \left( 1 + \sigma \right) \frac{1 - k\beta_c}{1 + k\beta_c} \left( \sum_{n=1}^{N} P_n \frac{1}{k} \left( \frac{z_n}{r_c} \right)^k + \sum_{n=1}^{N} \sum_{j=1}^{k} \frac{P_{nj}}{j-1} \left( \frac{r_{pm} z_n}{r_c^{k-j}} \right) \right) = 0
\]

This equation relates the value of a multipole \( P_{ck} \) at infinity to the values of line sources \( P_n \) and the multipoles \( P_{nj} \) at the pipes up to order \( j = k \).

7. EQUATIONS FROM THE BOUNDARY CONDITION AT THE PIPES

The boundary condition at pipe \( m \) is from (2.6):

\[
T_{fm} = \left( T - \beta_m \rho_m \frac{\partial T}{\partial \rho_m} \right) \bigg|_{\rho_m = r_{pm}} \quad 0 \leq \psi_m \leq 2\pi
\]

The temperature is given by (5.3). We will first express this temperature in the local polar coordinates \( \rho_m, \psi_m \) of pipe \( m \), which are given by (2.4):

\[
z - z_m = \rho_m e^{i\psi_m}
\]

We want to separate different powers \( (z - z_m)^k \) or \( (e^{i\psi_m})^k \). We therefore expand the various terms of (5.3) in Taylor series in \( z - z_m \).

We will use the following expressions.

A. \( n \neq m \)

\[
\ln \left( \frac{r_b}{z - z_m} \right) = \ln \left( \frac{r_b}{z - z_m + z_m - z_n} \right) \\
= \ln \left( \frac{r_b}{z_m - z_n} \right) + \ln \left( \frac{1}{1 - \frac{z - z_m}{z_n - z_m}} \right)
\]

The second logarithm is expanded in the Taylor series (6.2).

B.

\[
\ln \left( \frac{r_b^2}{r_b^2 - z_n \bar{z}} \right) = \ln \left( \frac{r_b^2}{r_b^2 - z_n z_m - z_n (z - \bar{z}_m)} \right) \\
= \ln \left( \frac{r_b^2}{r_b^2 - z_n \bar{z}_m} \right) + \ln \left( \frac{1}{1 - \frac{z_n (z - \bar{z}_m)}{r_b^2 - z_n \bar{z}_m}} \right)
\]
The second logarithm is expanded in the Taylor series (6.2).

C. \( n \neq m \)

\[
\frac{(r_{pn})^j}{(z - z_n)} = \left( \frac{r_{pn}}{z_m - z_n} \right)^j \cdot \frac{1}{(1 - \frac{z - z_m}{z_n - z_m})^j}
\]  (7.5)

The second factor is expanded with the Taylor series (6.5).

D.

\[
\left( \frac{r_{pn}(\bar{z})}{r_b^2 - z_n \bar{z}} \right)^j = \left( \frac{r_{pn}(\bar{z} + \bar{z} - \bar{z}_m)}{r_b^2 - z_n \bar{z}_m - z_n(\bar{z} - \bar{z}_m)} \right)^j
\]

\[
= \left( \frac{r_{pn}}{r_b^2 - z_n \bar{z}_m} \right)^j \cdot (\bar{z}_m + \bar{z} - \bar{z}_m)^j \cdot \frac{1}{(1 - \frac{z_n(\bar{z} - \bar{z}_m)}{r_b^2 - z_n \bar{z}_m})^j}
\]  (7.6)

The second factor is expanded as a binomial. The third factor is expanded in the series (6.5):

\[
(\bar{z}_m + \bar{z} - \bar{z}_m)^j = \sum_{j'=0}^{j} \binom{j}{j'} (\bar{z} - \bar{z}_m)^j' \cdot z_m^{j-j'}
\]  (7.7)

\[
\frac{1}{(1 - \frac{z_n(\bar{z} - \bar{z}_m)}{r_b^2 - z_n \bar{z}_m})^j} = \sum_{k'=0}^{\infty} \binom{j + k' - 1}{j - 1} \left( \frac{z_n(\bar{z} - \bar{z}_m)}{r_b^2 - z_n \bar{z}_m} \right)^{k'}
\]  (7.8)

The product of (7.7) and (7.8) is a double sum containing the powers \((\bar{z} - \bar{z}_m)^{j'} + j'\). We change the summation from \(k'\) and \(j'\) to \(k = k' + j'\) and \(j'\). The index \(k\) will vary from 0 to infinity. The values of \(j'\) run from 0 to \(\min(j, k)\). We have the expansion:

\[
\left( \frac{r_{pn}(\bar{z})}{r_b^2 - z_n \bar{z}} \right)^j = \sum_{k=0}^{\infty} \sum_{j'=0}^{\min(j, k)} \binom{j}{j'} (j + k - j' - 1) \cdot \binom{j'}{j - 1} \cdot \frac{r_{pn}^{j-j'} z_m^{j-k'} (\bar{z} - \bar{z}_m)^k}{(r_b^2 - z_n \bar{z}_m)^{j+k-j'}}
\]  (7.9)

E.

\[
\sum_{j=1}^{\infty} P_{ej} (1 - \sigma) \left( \frac{z}{r_c} \right)^j = \sum_{j=1}^{\infty} P_{ej} (1 - \sigma) \left( \frac{z_m + z - \bar{z}_m}{r_c} \right)^j
\]

\[
= \sum_{j=1}^{\infty} \sum_{k=0}^{j} P_{ej} (1 - \sigma) \binom{j}{k} z_m^{j-k} \frac{(z - z_m)^k}{r_c^k}
\]  (7.10)

The summation order is changed. Then we have:

\[
\sum_{j=1}^{\infty} \sum_{k=0}^{j} \ldots = \sum_{k=0}^{\infty} \sum_{j=\max(1, k)}^{\infty} \ldots
\]  (7.11)

The temperature (5.3) may now with the use of (7.2-11) be written in the following way:
\[ T = T_o + P_m \ln \left( \frac{r_b}{\rho_m} \right) \]
\[ + \text{Re} \sum_{n=1}^{N} \sum_{n \neq m} P_n \left\{ \ln \left( \frac{r_b}{z_m - z_n} \right) + \sum_{k=1}^{\infty} \frac{1}{k} \left( \frac{\rho_m}{z_n - z_m} \right)^k e^{i k \psi_m} \right\} \]
\[ + \sum_{n=1}^{N} P_n \sigma \left\{ \ln \left( \frac{r_b}{r_b - z_m z_n} \right) + \sum_{k=1}^{\infty} \frac{1}{k} \left( \frac{z_n \rho_m}{r_b - z_m z_n} \right)^k e^{-i k \psi_m} \right\} \]
\[ + \sum_{j=1}^{\infty} P_{mj} \left( \frac{r_{pm}}{\rho_m} \right)^j e^{-i j \psi_m} \]
\[ + \sum_{n=1}^{N} \sum_{j=1}^{\infty} \sum_{k=0}^{\infty} P_{nj} \left( \frac{r_{pm}}{z_m - z_n} \right)^j \left( \frac{\rho_m}{z_n - z_m} \right)^k e^{i k \psi_m} \]
\[ + \sum_{n=1}^{N} \sum_{j=1}^{\infty} \sum_{k=0}^{\infty} \sum_{j'=0}^{\min(j,k)} P_{nj} \sigma \left( \frac{z_n^{j+k-j'} \rho_m^{k-j'} e^{-i j' \psi_m}}{(r_b^2 - z_n z_m)^{j+k-j'}} \right) \]
\[ + \sum_{k=0}^{\infty} \sum_{j=\max(1,k)}^{\infty} P_{cj} (1 - c) \left( \frac{z_n^{j-k} \rho_m^k e^{i k \psi_m}}{r_b^k} \right) \]

(7.12)

The summation index on the fourth line (concerning \( P_{mj} \)) is changed to \( k \). The dependence on \( \psi_m \) lies in the exponents \( e^{i k \psi_m} \) and \( e^{-i k \psi_m} \). The latter terms may be changed to the positive exponent \( e^{i k \psi_m} \) by taking the complex conjugate as in (6.10). Expression (7.12) is with these modifications inserted in the boundary condition (7.1) for pipe \( m \). The derivatives of powers of \( \rho_m \) are simple. We have as in (6.9):

\[ -\beta_m r_{pm} \frac{\partial}{\partial \rho_m} \left( \rho_m^{k-1} \right) \big|_{\rho_m=r_{pm}} = -k \beta_m r_{pm}^k \]
\[ -\beta_m r_{pm} \frac{\partial}{\partial \rho_m} \left( \rho_m^{-k} \right) \big|_{\rho_m=r_{pm}} = k \beta_m r_{pm}^{-k} \]

(7.13)

We finally get the following expression for the boundary condition at pipe \( m \):
\[
T_{fm} = \left( T - \beta_m r_{pm} \frac{\partial T}{\partial \beta_m} \right)_{\beta_m=r_{pm}} = T_o + P_m \left\{ \ln \left( \frac{r_b}{r_{pm}} \right) - \beta_m r_{pm} \cdot \frac{-1}{r_{pm}} \right\} + \sum_{n=1}^{N} P_n \ln \left( \frac{r_b}{r_{mn}} \right) + \sum_{n=1}^{N} P_n \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2 - \bar{x}_m \bar{x}_n} \right) \\
+ \Re \left[ \sum_{n=1}^{N} \sum_{j=1}^{\infty} P_{nj} \left( \frac{r_{pm}}{z_m - z_n} \right)^i + \sum_{n=1}^{N} \sum_{j=1}^{\infty} P_{nj} \sigma \left( \frac{r_{pm} z_m}{r_b^2 - \bar{x}_m \bar{x}_n} \right)^j \right] \\
+ \sum_{j=1}^{\infty} P_{cj} (1 - \sigma) \left( \frac{z_m}{r_c} \right)^j
\]

(7.14)

Equation (7.14) is valid for all \(t \beta_m\) around the pipe. The last six lines of (7.14) contain the part that depends on \(t \beta_m\). It must be equal to zero. The expression is of the same type as (6.13). The complex factor for each component \(e^{i \cdot k \cdot \psi_m}\) must vanish. We get the following equations, which determine the multipoles \(P_{nj}\):

The first four lines on the right-hand side give the constant part; i.e. the part independent of \(\psi_m\). The length \(r_{mn}\) denotes the distance between the centers of pipes \(m\) and \(n\):

\[
r_{mn} = |z_m - z_n| = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2}
\]

(7.15)

Equation (7.14) is valid for all \(\psi_m\) around the pipe. The last six lines of (7.14) contain the part that depends on \(\psi_m\). It must be equal to zero. The expression is of the same type as (6.13). The complex factor for each component \(e^{i \cdot k \cdot \psi_m}\) must vanish. We get the following equations, which determine the multipoles \(P_{nj}\):
\[ m = 1, \ldots N \quad ; \quad k = 1, 2, \ldots : \]

\[
P_{mk} + \frac{1 - \kappa \beta_m}{1 + \kappa \beta_m} \left\{ \sum_{n=1, n \neq m}^N P_{nj} \left( \frac{1}{k} \left( \frac{r_{pm}}{z_m - z_n} \right)^k + \sum_{n=1}^N P_n \sigma \left( \frac{1}{k} \left( \frac{r_{pm}}{r_b^2 - z_m - z_n} \right)^k \right) \right) \right. \\
+ \sum_{n=1}^N \sum_{j=1, j \neq m}^\infty P_{nj} \left( j + k - 1 \left( \frac{r_{pm}}{z_m - z_n} \right)^j \left( \frac{r_{pm}}{z_n} \right)^k \right) \\
+ \sum_{n=1}^N \sum_{j=1}^\infty \sum_{j' = 0}^\infty P_{nj}(1 - \sigma) \left( j \left( \frac{z_m - z_n}{r_b^2} \right)^j \right) \right\} = 0 \]

(7.16)

The constant part of (7.14) must also vanish. This means that the first four lines on the right-hand side are equal to \( T_{jm} \). We can eliminate \( T_0 \) from (6.12). Then we get the following equations:

\[ m = 1, \ldots N: \]

\[
T_{jm} - T_0 = \sum_{n=1}^N q_n \cdot R_{mn}^o \\
+ \text{Re} \left[ \sum_{n=1}^N \sum_{j=1, j \neq m}^\infty P_{nj} \left( \frac{r_{pm}}{z_m - z_n} \right)^j + \sum_{n=1}^N \sum_{j=1}^\infty P_{nj}(1 - \sigma) \left( \frac{z_m - z_n}{r_b^2} \right)^j \right] \]

(7.17)

The first line involves on the right-hand side the line sources \( P_n \). We have returned to \( q_n \) via (5.2). The coefficients \( R_{mn}^o \) are given by:

\[
R_{mm}^o = \frac{1}{2 \pi \lambda_b} \left\{ \beta_m + \ln \left( \frac{r_b}{r_{pm}} \right) + \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2 - r_m^2} \right) \right\} + \frac{1}{2 \pi \lambda} \left\{ \ln \left( \frac{r_c}{r_b} \right) + \beta_c \right\} \quad m = 1, \ldots N \\
R_{mn}^o = \frac{1}{2 \pi \lambda_b} \left\{ \ln \left( \frac{r_{mn}}{r_{pm}} \right) + \sigma \cdot \ln \left( \frac{r_b^2}{r_b^2 - z_m - z_n} \right) \right\} + \frac{1}{2 \pi \lambda} \left\{ \ln \left( \frac{r_c}{r_b} \right) + \beta_c \right\} \quad m \neq n \]

(7.18)
8. FINAL EQUATIONS AND ITERATIVE SOLUTION

The final equations for $q_n$ (or $P_n$), $P_{nj}$ and $P_{ej}$ are (7.17-18), (7.16) and (6.14). Multipoles of all orders are needed in an exact solution. In the numerical one we truncate the equation system and consider multipoles up to order $J$ at each pipe and at infinity. The sine- and cosinevariation around the pipes and around the outer circle can then be satisfied up to order $J$ only. The truncation error is discussed in the next chapter.

We have from (7.17), (7.16), (5.2) and (6.14) the following final equations: $m = 1, \ldots, N$:

$$
T_{jm} - T_c = \sum_{n=1}^{N} q_n \cdot R_{mn}^0
+ \text{Re} \left[ \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj} \left( \frac{r_{pn}}{z_n - z_m} \right)^j + \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj} \sigma \left( \frac{r_{pn} z_m}{r^2_n - z_n z_m} \right)^j \right]
+ \sum_{j=1}^{J} P_{ej} (1 - \sigma) \left( \frac{z_m}{r_c} \right)^j
$$

$$
m = 1, \ldots, N; \quad k = 1, \ldots, J:
$$

$$
P_{mk} + \frac{1 - k \beta_c}{1 + k \beta_c} \left\{ \sum_{n=1}^{N} \frac{q_n}{2 \pi \lambda k} \left( \frac{r_{pn}}{z_n - z_m} \right)^k + \sum_{n=1}^{N} \frac{q_n}{2 \pi \lambda k} \sigma \left( \frac{r_{pn} z_m}{r^2_n - z_n z_m} \right)^k \right\}
+ \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj} \left( j + k - 1 \right) \left( \frac{r_{pn}}{z_n - z_m} \right)^j \left( \frac{r_{pn}}{z_n - z_m} \right)^k
+ \sum_{n=1}^{N} \sum_{j=1}^{J} \sigma \left( j' \right) \left( j + k - 1 \right) \left( \frac{r_{pn} z_n z_m}{r^2_n - z_n z_m} \right)^{j + k - j'}
+ \sum_{j=k}^{J} P_{ej} (1 - \sigma) \left( \frac{z_m}{r_c} \right)^k
= 0
$$

$$
k = 1, \ldots, J:
$$

$$
P_{ck} \cdot \left\{ 1 - \sigma \frac{1 - k \beta_c}{1 + k \beta_c} \left( \frac{r_b}{r_c} \right)^{2k} \right\}
+ (1 + \sigma) \frac{1 - k \beta_c}{1 + k \beta_c} \left\{ \sum_{n=1}^{N} \frac{q_n}{2 \pi \lambda k} \left( \frac{z_n}{r_c} \right)^k + \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj} \left( j + k - 1 \right) \left( \frac{r_{pn}}{r^2_n - z_n z_m} \right)^k \right\} = 0
$$

The thermal resistances $R_{mn}^0$ are given by (7.18):

$$
R_{mn}^0 = \frac{1}{2 \pi \lambda} \left\{ \beta_m + \ln \left( \frac{r_b}{r_{mn}} \right) + \sigma \cdot \ln \left( \frac{r^2_b - z_m z_n}{r^2_{mn} - z_n z_m} \right) \right\} + \frac{1}{2 \pi \lambda} \left\{ \ln \left( \frac{r_c}{r_b} \right) + \beta_c \right\} \quad m = 1, \ldots, N
$$

$$
R_{mn}^0 = \frac{1}{2 \pi \lambda} \left\{ \ln \left( \frac{r_b}{r_{mn}} \right) + \sigma \cdot \ln \left( \frac{r^2_b - z_m z_n}{r^2_{mn} - z_n z_m} \right) \right\} + \frac{1}{2 \pi \lambda} \left\{ \ln \left( \frac{r_c}{r_b} \right) + \beta_c \right\} \quad m \neq n \quad (8.4)
$$

There are $N$ real-valued equations (8.1) and $N \cdot J + J$ complex-valued ones (8.2-3). This corresponds to the $N$ line sources $q_n$ and the $N \cdot J + J$ complex-valued multipoles $P_{nj}$ and $P_{ej}$. 

18
The equation system (8.1-4) is solved iteratively as in [1]. Let \( K_{mn}^{o} \) be the elements of the inverse matrix of the resistance matrix with the elements \( R_{mn}^{o} \):

\[
(K_{mn}^{o}) = (R_{mn}^{o})^{-1}
\]

(8.5)

Inversion of (8.1) gives the heat fluxes:

\[
q_{m'} = \sum_{m=1}^{N} K_{m'm}^{o} \left\{ T_{m} - T_{c} \right\}
- \text{Re} \left\{ \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj} \left( \frac{r_{pn}}{z_{n} - z_{m}} \right)^{j} + \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj'} \sigma \left( \frac{r_{pn}z_{m}}{r_{b}^{2} - z_{m}z_{n}} \right)^{j} \right\}
+ \sum_{j=1}^{J} P_{ej}(1 - \sigma) \left( \frac{z_{m}}{r_{c}} \right)^{j} \left\} \right.
\]

(8.6)

Let \( q_{n}^{o} \), \( P_{nj}^{o} \) and \( P_{ej}^{o} \) denote the values of our variables for iteration step \( \nu \). We start for \( \nu = 0 \) with the values:

\[
P_{nj}^{o} = 0 \quad P_{ej}^{o} = 0 \quad q_{n}^{o} = \sum_{m=1}^{N} K_{m'm}^{o} (T_{m} - T_{c})
\]

(8.7)

This means that we start without multipoles and compute the heat fluxes with (8.6).

For \( \nu = 1, 2... \) we use the following recursive formulas in accordance with (8.2), (8.3) and (8.6):

\[
p_{mk}^{\nu+1} = -\frac{1 - k\beta_{m}}{1 + k\beta_{m}} \left\{ \sum_{n=1}^{N} \frac{q_{n}^{\nu}}{2\pi\lambda_{b}} \frac{1}{k} \left( \frac{r_{pn}}{z_{m} - z_{n}} \right)^{k} + \sum_{n=1}^{N} \frac{q_{n}^{\nu}}{2\pi\lambda_{b}} \sigma \frac{1}{k} \left( \frac{r_{pn}z_{m}}{r_{b}^{2} - z_{m}z_{n}} \right)^{k} \right\}
+ \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj}^{\nu} \left( \frac{j+k-1}{j-1} \right) \left( \frac{r_{pn}}{z_{m} - z_{n}} \right)^{j} \left( \frac{r_{pn}}{z_{m} - z_{n}} \right)^{k}
+ \sum_{j=0}^{J} P_{nj}^{\nu} \sigma \left( \frac{j+k-1}{j-1} \right) \left( \frac{r_{pn}^{2}}{r_{b}^{2} - z_{m}z_{n}} \right)^{j+1}
+ \sum_{j=1}^{J} P_{ej}^{\nu}(1 - \sigma) \left( \frac{z_{m}}{r_{c}} \right)^{j} \left\} \right.
\]

(8.8)

\[
p_{ek}^{\nu+1} \left\{ 1 - \sigma \frac{1 - k\beta_{e}}{1 + k\beta_{e}} \left( \frac{r_{b}}{r_{c}} \right)^{2k} \right\} =
- (1 + \sigma) \frac{1 - k\beta_{e}}{1 + k\beta_{e}} \left\{ \sum_{n=1}^{N} \frac{q_{n}^{\nu}}{2\pi\lambda_{b}} \frac{1}{k} \left( \frac{z_{n}}{r_{c}} \right)^{k} + \sum_{n=1}^{N} \sum_{j=1}^{J} P_{nj}^{\nu} \left( \frac{k-1}{j-1} \right) \frac{r_{pn}z_{n}}{r_{b}^{2} - z_{m}z_{n}} \right\}
\]

(8.9)
\[ m' = 1, \ldots N: \]

\[ q_{m'}^{n+1} = \sum_{m=1}^{N} K_{m'm} \left\{ T_{Im} - T_c \right\} \]

\[- \text{Re} \left\{ \sum_{n \neq m}^{N} \sum_{j=1}^{J} p_{nj}^{n+1} \left( \frac{r_{pn}}{z_n - z_m} \right)^j + \sum_{n=1}^{N} \sum_{j=1}^{J} p_{nj}^{n+1} \sigma \left( \frac{r_{pn}z_m}{r_c^2 - z_n z_m} \right)^j \right\} \]

\[ + \sum_{j=1}^{J} p_{nj}^{n+1} (1 - \sigma) \left( \frac{z_m}{r_c} \right)^j \}\]

\[(8.10)\]

The iterative solution (8.7-10) is physically reasonable. Consider an iteration step \( \nu \). There is an approximate solution with line sinks and multipoles. The equations (8.1-3) are not satisfied exactly. The equations (8.8) and (8.9) mean that we change the multipoles so that the variation vanishes exactly (up to order \( J \)) at the considered boundary circle. But the difference \( P_{mk}^{n+1} - P_{mk}^n \) will induce a multipole behaviour at the other pipes. These secondary disturbances are however damped with factors of the type:

\[ \left( \frac{z_n}{r_c} \right)^k \; ; \; \left( \frac{r_{pn}}{r_c} \right)^{k'} \; ; \; \left( \frac{r_{pn}}{z_n - z_m} \right)^{k''} \]

The convergence of the iterative procedure is therefore rapid.

The iteration procedure is quite robust. It has worked without any problems for all cases that we have tested. An example is given below.

The iterations are performed until the following criterion is satisfied.

\[ \frac{|P_{nj}^{n+1} - P_{nj}^n|}{\max_{1 \leq k \leq J} |P_{nk}|} < \varepsilon \quad \text{for all } n \text{ and all } j (j \leq J) \text{ with } |P_{nj}| \neq 0 \]

\[ \frac{|P_{cj}^{n+1} - P_{cj}^n|}{\max_{1 \leq k \leq J} |P_{cj}|} < \varepsilon \quad \text{for all } j (j \leq J) \text{ with } |P_{cj}| \neq 0 \]

\[(8.11)\]

Here \( \varepsilon \) is a measure of the iteration accuracy. The multipole differences are divided by the largest multipole up to the same order \( j \) according to the expression in the denominators. Normally, this is the multipole of first order. This will mean that \( \varepsilon \) gives the accuracy relative to the magnitude of the strongest multipole.

Example 8.1. Three pipes with different temperatures. We take the following data:

\[ N = 3 \quad \lambda_b = 0.6 \quad \lambda = 3.6 \]
\[ r_b = 2 \quad r_c = 4 \quad \beta_1 = 0 \quad T_c = 0 \]
\[ x_1 = 1 \quad y_1 = 0 \quad \beta_1 = 0 \quad T_{f1} = 1 \quad r_{p1} = 0.5 \]
\[ x_2 = 0 \quad y_2 = 1.5 \quad \beta_2 = 0 \quad T_{f2} = -3 \quad r_{p2} = 0.25 \]
\[ x_3 = -1 \quad y_3 = -0.5 \quad \beta_3 = 0.5 \quad T_{f3} = 2 \quad r_{p3} = 0.5 \]

\[(8.12)\]

The temperature field of this case is shown in Figure 1.1 (and on the cover).

The number of iterations in order to obtain the required accuracy \( \varepsilon \) is given in Table 8.1 for different \( J \) and \( \varepsilon \).
Table 8.1. Required number of iterations in example 8.1.

We have found as in [1] that the value of $\varepsilon$ is not critical. One may use $\varepsilon = 10^{-4}$ or $10^{-5}$.

9. REQUIRED NUMBER OF MULTipoles

The boundary conditions at the pipes and the outer circle are satisfied up to the order $J$ in Fourier terms i.e. up to the order $\cos(J\psi_m)$ and $\sin(J\psi_m)$. The approximation becomes better, when $J$ is increased, but the computational effort and the execution time also increase. It is an important question what value of $J$ to choose in any particular case. One can always increase $J$, until the solution does not change.

The accuracy as a function of $J$ will be studied and illustrated in this chapter with a few examples.

9.1 ERROR ON THE BOUNDARY CIRCLES

The example shown in Figure 9.1.1. is used to illustrate the error on the boundary circles.

The following data are used:

\[ \begin{align*}
N &= 2 \\
\lambda_b &= 1 \\
\lambda &= 5 \\
\varepsilon &= 10^{-5} \\
r_b &= 2 \\
r_c &= 10 \\
\beta_1 &= 0 \\
T_{r1} &= 0 \\
x_1 &= 1 \\
y_1 &= 0 \\
\beta_2 &= 0 \\
T_{r2} &= 0 \\
x_2 &= -1 \\
y_2 &= 0 \\
\beta_3 &= 0 \\
T_{r3} &= 0 \\
\end{align*} \]

Figure 9.1.1. Example with two pipes.

The heat fluxes $q_1$ and $q_2$ are equal. The computed values for different $J$ are given in Table 9.1.1. The computer model described in the following chapters is used. We see that $J = 1$ gives an error for $q_1$ of 2
%, and it is 0.3% for J = 3. The first order, J = 1, is sufficient in this case in order to calculate the heat fluxes.

\[
\begin{array}{ccccccccc}
J & 0 & 1 & 2 & 3 & 4 & 5 & 10 \\
q_1(J)/q_1(10) & 0.73 & 0.981 & 0.990 & 0.997 & 0.9989 & 0.9994 & 1
\end{array}
\]

Table 9.1.1. Calculated heat fluxes for example (9.1.1) for different J.

The polar coordinates of pipe 1 are \( \rho_1, \psi_1 \). The temperature on the uninsulated pipe is \( T(1, \psi_1) \). In an exact solution \( T(1, \psi_1) \) is equal to \( T_{f1} = 0 \), but for a finite \( J \) there will be a certain error or variation around the pipe. This boundary temperature is shown in Figure 9.1.2 for \( J = 10 \). The calculations are made with the computer program described in Chapters 10-11. The temperature \( T(1, \psi_1) \) is obtained from the general formula (5.1) with summation in \( j \) up to \( J = 10 \).

![Figure 9.1.2. Temperature variation on pipe 1. Data according to (9.1.1) with \( J = 10 \).](image)

The largest deviation from \( T_{f1} = 0 \) is 0.002. The largest error of boundary temperature is therefore 0.2% (of the temperature difference \( T_c - T_{f1} \)).

9.2 A THREE-PIPE PROBLEM

As a second example we take the three-pipe problem of example 8.1. The data are given by (8.12). The temperature field of this case is shown in Figure 1.1. Table 9.2 gives the calculated heat fluxes and the temperatures in a few points for different \( J \).
\begin{table}
\begin{tabular}{cccccccc}
J & 0 & 1 & 2 & 3 & 5 & 10 & 15 \\
\hline
$q_3$ & 4.570 & 4.766 & 4.792 & 4.792 & 4.792 & 4.792 & 4.792 \\
T(0,0) & 0.7851 & 0.7343 & 0.7267 & 0.7245 & 0.7243 & 0.7243 & 0.7243 \\
T(0,-2) & 1.2410 & 0.1452 & 0.1724 & 0.1704 & 0.1706 & 0.1706 & 0.1706 \\
\end{tabular}
\end{table}

Table 9.2.1. Heat flows and temperatures in a few points for different $J$. Data according to (8.12).

The table shows that the error in heat fluxes is at most 7\% for $J = 0$ and 0.8\% for $J = 1$. The temperatures differ with up to a factor 2 for $J = 0$, while the largest error is 0.03 temperature units for $J = 1$. This difference decreases to 0.0024 for $J = 2$. The value $J = 1$ should be sufficient in this example.

9.3 Test of Maxwell's reciprocity theorem

We do not have analytical solutions to compare with in more complicated cases. We will therefore use a general theorem due to J.C. Maxwell (and others). Consider a steady-state heat conduction problem. The region is bounded by a number of boundary surfaces $S_1, S_2, S_3, \ldots$. The temperature is zero on all surfaces except one. Two cases are considered:

\begin{align}
A: & \quad T = 1 \quad \text{on} \quad S_1 \quad T = 0 \quad \text{on} \quad S_2 \quad (\text{and} \quad S_3, \ldots) \\
B: & \quad T = 1 \quad \text{on} \quad S_2 \quad T = 0 \quad \text{on} \quad S_1 \quad (\text{and} \quad S_3, \ldots) \\
\end{align}

(9.3.1)

Let $q_2^A$ be the heat flux from surface 2 in case A, and $q_1^B$ the heat flux from surface 1 in case B. The reciprocity theorem states that these two fluxes are equal:

\begin{equation}
q_2^A = q_1^B
\end{equation}

(9.3.2)

We have found that the reciprocity theorem is valid for our truncated problem (8.1-3) as well. (We have not taken the trouble to try to prove this).

In a test of the reciprocity theorem we use example (8.1). The data (8.12) are valid except for the boundary temperatures. We use:

\begin{equation}
T_c = 0 \quad T_{f1} = 0 \quad J = 5
\end{equation}

(9.3.3)

and

\begin{align}
\text{case } A: & \quad T_{f2} = 1 \quad T_{f3} = 0 \\
\text{case } B: & \quad T_{f2} = 0 \quad T_{f3} = 1 \\
\end{align}

(9.3.3A) (9.3.3B)

The calculated heat fluxes became equal with seven digits:

\begin{align}
q_2^A &= -0.1752401 \\
q_2^B &= -0.1752401
\end{align}

(9.3.4)
Another example concerns a case with a single region \( \lambda_b = \lambda \). The first five pipes of the example in section 9.3 in [1] are used. The following data are valid:

\[
\begin{align*}
N = 5 & \quad \lambda = \lambda_b = 1 & \quad J = 5 \\
T_c = 0 & \quad r_c = 10 & \quad \beta_c = 0 \\
x_1 = 1 & \quad y_1 = 0 & \quad r_{p1} = 0.5 & \quad \beta_1 = 0 \\
x_2 = 2 & \quad y_2 = 1 & \quad r_{p2} = 0.5 & \quad \beta_2 = 0 & \quad T_{f2} = 0 \\
x_3 = 1 & \quad y_3 = 2 & \quad r_{p3} = 0.5 & \quad \beta_3 = 0 & \quad T_{f3} = 0 \\
x_4 = 0 & \quad y_4 = 2 & \quad r_{p4} = 0.5 & \quad \beta_4 = 0 & \quad T_{f4} = 0 \\
x_5 = 2 & \quad y_5 = -1 & \quad r_{p5} = 0.5 & \quad \beta_5 = 0.3 \\
\end{align*}
\]

case A: \quad T_{f1} = 1 & \quad T_{f5} = 0 \\
case B: \quad T_{f1} = 0 & \quad T_{f5} = 1
\]

(9.3.5)

The computed reciprocal heat fluxes became again equal with 7 digits:

\[
\begin{align*}
q_n^A &= -1.890565 \\
q_n^B &= -1.890565
\end{align*}
\]

(9.3.6)

### 9.4 AN EXAMPLE WITH 15 PIPES

We use the example with 15 pipes in [1], section 9.3. The data according to (9.3.1) in [1] are valid. The corresponding temperature field is shown on the cover of [1]. The outer radius \( r_c \) is equal to 10. We take \( r_b \) equal to this value and add an outer region out to \( r_c = 20 \). The thermal conductivity is taken to be ten times higher than in the inner region. The data of (9.3.1) in [1] are supplemented with:

\[
\begin{align*}
r_b = 10 & \quad \lambda_b = 1 & \quad r_c = 20 & \quad \lambda = 10
\end{align*}
\]

(9.4.1)

Table 9.4.1 gives the computed heat fluxes \( q_n \), \( 1 \leq n \leq 15 \), for different \( J \). The values do not differ much from those in [1], (Table 9.3.1). We see that \( J = 1 \) is not acceptable, while \( J = 5 \) is quite sufficient.

<table>
<thead>
<tr>
<th>( J )</th>
<th>( q_1 )</th>
<th>( q_2 )</th>
<th>( q_3 )</th>
<th>( q_4 )</th>
<th>( q_5 )</th>
<th>( q_6 )</th>
<th>( q_7 )</th>
<th>( q_8 )</th>
<th>( q_9 )</th>
<th>( q_{10} )</th>
<th>( q_{11} )</th>
<th>( q_{12} )</th>
<th>( q_{13} )</th>
<th>( q_{14} )</th>
<th>( q_{15} )</th>
</tr>
</thead>
</table>

Table 9.4.1. Computed heat fluxes \( q_n \) for different \( J \) for a case with 15 pipes.
10. COMPUTER MODEL. EXECUTION TIMES

The computer model calculates the line sources and the multipoles up to the given order $J$, and it gives the corresponding temperature field. The next chapter is a manual for the computer program. The source code is given in appendix 1.

The input data of the model are:

$$
\lambda, \lambda_b, N, J, r_b, r_e, \beta_c, T_e, \ x_n, y_n, r_{pn}, \beta_n, T_{f_n} \text{ for } n = 1, \ldots N \tag{10.1}
$$

There are the following restrictions on the input variables:

$$
\lambda > 0 \quad \lambda_b > 0 \quad N = 1, 2, \ldots \quad J = 0, 1, \ldots \\
0 < r_b < r_e \quad \beta_c \geq 0 \quad \text{for } n = 1, \ldots N \\
r_{pn} > 0 \quad \beta_n \geq 0 \quad \text{for } n = 1, \ldots N \\
r_n = \sqrt{x_n^2 + y_n^2} \leq r_b - r_{pn} \quad \text{for } n = 1, \ldots N \\
r_{mn} \geq r_{pn} + r_{pm} \quad \text{for } m \neq n \\
r_{mn} = r_{pm} + r_{pn} \quad \text{only if } \beta_m + \beta_n > 0 \quad \text{or } T_{f_m} = T_{f_n} \tag{10.2}
$$

The condition $r_{mn} \geq r_{pm} + r_{pn}$ ensures that the pipes do not cover each other. They may touch each other, if there is a thermal insulation ($\beta_m + \beta_n > 0$) or if the temperatures $T_{f_m}$ and $T_{f_n}$ are equal, so that the heat flux between the pipes remains finite.

The first step after input and test of the restrictions (10.2) is to calculate auxiliary variables and the resistance matrix ($R_{mn}$) with the elements (8.4). The inverse matrix ($R_{mn}^\circ$) is then calculated. The initial values of line sources and multipoles for $\nu = 0$ are given by (8.7). Formulas (8.8-10) are used in successive iterations, until the $\epsilon$-criterion (8.11) is met.

The output is the values of the line sources and multipoles and, if requested, the temperature field $T(x, y)$, which is obtained from (5.3-4) with summation in $j$ up to $J$.

The steady-state heat conduction problem may be solved numerically with finite difference or finite element methods. The present method is however more rapid, and it is simpler to get a high and controlled accuracy. We will give the execution time for a number of cases in order to show this.

Consider the case with 15 pipes in section 9.4. In the first examples we use the first 5 pipes ($N = 5$), then the first 10 pipes ($N = 10$) and finally all 15 pipes. The accuracy $\epsilon$ is $10^{-4}$ and $10^{-5}$, while $J$ is 5 and 10. The execution times in CPU-seconds on a ND-500 computer from Norsk Data are given in Table 10.1.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$J$</th>
<th>$\epsilon = 10^{-4}$</th>
<th>$\epsilon = 10^{-5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>5</td>
<td>13 s</td>
<td>18 s</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>140 s</td>
<td>200 s</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>44 s</td>
<td>60 s</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>460 s</td>
<td>620 s</td>
</tr>
<tr>
<td>15</td>
<td>5</td>
<td>110 s</td>
<td>150 s</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1200 s</td>
<td>1700 s</td>
</tr>
</tbody>
</table>

Table 10.1. Execution time in CPU-seconds on ND-500 for 5 to 15 pipes and multipole order $J = 5, 10$.

Table 10.1 shows that the CPU-time is below a few minutes even for 15 pipes, if the order of multipoles is kept below $J \leq 5$. The CPU-time for a more moderate case with $N \leq 3$, $J \leq 2$ is only a few seconds. The execution times for an IBM-PC AT-3 (10 MHz) with a 80287 math co-processor are about 60 times longer.

The program has been adapted to run on IBM-PC and other compatible computers. The source code is written in FORTRAN77. It has been compiled with the MS-FORTRAN77 compiler V3.3.

The executable code supplied on the disk may be used for cases with a maximum of 15 pipes. The maximum order of multipoles is 10. The size of the program is then 133K. These restrictions can be removed by changing the dimensions in the PARAMETER statement at the beginning of the main program. The maximum number of pipes is IW and the maximum order of multipoles is IJ.

The basic version of the source code is listed in Appendix 1. The current version of the program differs from the basic version according to the changes listed on the file README.DOC. This file may also contain important information about changes in this manual. Be sure to read this file before you run the program.

Files on the Disk.

A brief description of the files on the enclosed disk is given below.

- _MPC.EXE_ Executable code
- _MPC.FOR_ Source code for the current version of the program
- _SAMPLE.DAT_ Input data file for the example. See below
- _SAMPLE.OUT_ Output file for the example. See below
- _MPC.BAT_ Batch file
- _SIGNAL.COM_ Program that generates a signal when the job is completed
- _README.DOC_ Contains updates of this manual and the basic version of the source code.

Using the Program.

To run the program MPC (MultiPoleComposite), insert the disk into a drive, make that drive the default drive, and type the program name:

```
MPC
```

The program is intended for interactive use. The user is prompted for the names of an input data file (optional) and the output file. The input data may be entered interactively on the screen or read directly from a disk file. If the input data is entered interactively it may be saved on a disk file.

Input Data.

The input data must satisfy the restrictions (10.2). The program indicates any violation of these restrictions. The input data list is read in free format, i.e. the values must be separated by one or more contiguous blanks or a comma. The input data records are specified below.

1. \( \lambda_b \) Thermal conductivity in the inner region, (W/mK)
2. \( \lambda \) Thermal conductivity in the outer region, (W/mK)
3. \( N \) Number of pipes
4. \( J \) Maximal order of multipoles
5. \( r_b \) Radius of the inner region, (m)
6. \( r_c \) Radius of the outer region, (m)
7. \( \beta_c \) Thermal resistance coefficient at the outer circle, (-)
8. \( T_c \) Temperature at outer circle, (°C)

The following record must be repeated for each pipe, i.e. N times.

1. \( x_i \) x-coordinate of the center of pipe i, (m)
2. \( y_i \) y-coordinate of the center of pipe i, (m)
3. \( r_{pi} \) Radius of pipe i, (m)
4. \( \beta_i \) Thermal resistance coefficient of pipe i, (-)
5. \( T_i \) Fluid temperature in pipe i, (°C)
6. \( \epsilon \) Iteration accuracy. See section 8. A suitable value is \( 10^{-4} \)
7. \( ITMAX \) Maximum number of iterations. A suitable value is 500

The temperatures are calculated in a rectangular grid within a rectangular region with the corners determined by XMIN, XMAX, YMIN, and YMAX. The number of grid points in the X- and Y-direction are NX and NY respectively. If a grid point lies within the circle of one of the pipes or outside the
outer boundary, then there will be no temperature calculation in that point. There is no calculation of temperatures if \(NX \leq 0\) or \(NY \leq 0\).

5. \(XMIN\) Minimum \(x\)-coordinate of rectangular region for temperature calculation, (m)

\(XMAX\) Maximum \(x\)-coordinate of rectangular region for temperature calculation, (m)

\(YMIN\) Minimum \(y\)-coordinate of rectangular region for temperature calculation, (m)

\(YMAX\) Maximum \(y\)-coordinate of rectangular region for temperature calculation, (m)

\(NX\) Number of grid points in the \(x\)-direction

\(NY\) Number of grid points in the \(y\)-direction

The program is the same as in [1] except for the necessary modifications. The source code is given in appendix 1.

AN EXAMPLE

The example 8.1 with 3 pipes is used. The complete set of data is given by (8.12). The accuracy \(\epsilon\) is set to \(10^{-4}\) and \(ITMAX\) to 500. The corner points of the region of temperature calculation are \(XMIN = -2\), \(YMIN = -2\), \(XMAX = 2\), and \(YMAX = 2\). The number of grid points in the \(x\)-direction, \(NX\), and the number of grid points in the \(y\)-direction, \(NY\), are both 5. This gives the following indata list:

1. 0.6 3.6 3 10
2. 2.0 4.0 0.0 0.0
3. 1.0 0.0 0.5 0.0 1.0
4. 0.0 1.5 0.25 0.0 -3.0
5. -1.0 -0.5 0.5 0.5 2.0
6. 1.0E-4 500
7. -2.0 2.0 -2.0 2.0 5 5

The output is given on the next two pages. The line sources and multipoles are denoted in the following way: \(q_n = q(n)\), \(P_{nk} = P(n,k)\), \(P_{ck} = PC(k)\).

Correction on p.28, line 35-36:

Number of grid points along the \(x\)-axis 5
Number of grid points along the \(y\)-axis 5
MULTIPOLE METHOD - Pipes in a composite cylinder

INPUT DATA

Input file a: sample.dat
Output file a: sample.out

Thermal conductivity in inner region: .600 W/(m*K)
Thermal conductivity in outer region: 3.600 W/(m*K)
Number of pipes: 3
Order of multipoles: 10

Radius of inner region: 2.000 m
Radius of outer region: 4.000 m

Thermal resistance coefficient: .000E+00
Temperature: .000 C

Pipe x(n) y(n) rp(n) beta(n) Temp C
1 1.000 .000 .500 .000E+00 1.000
2 .000 1.500 .250 .000E+00 -3.000
3 -1.000 -.500 .500 .500E+00 2.000

Iteration accuracy: .10E-03
Maximum number of iterations: 500

TEMPERATURE OUTPUT: Definition of rectangular area
Minimum x-value: -2.000
Maximum x-value: 2.000
Minimum y-value: -2.000
Maximum y-value: 2.000
Number of grid points along x-axis: 4
Number of grid points along y-axis: 4

Pipe Initial values for q(n) (Order of multipole = 0)
1 .3701710E+01
2 -.8120926E+01
3 .4570385E+01

Number of iterations: 6

Pipe q(n)
1 .3775983E+01
2 -.8688749E+01
3 .4791657E+01

Pipe Order P(n,k)
1 1 .85094E-01 .26170E+00
1 2 -.30722E-01 -.72742E-01
1 3 .18398E-01 .80464E-02
1 4 -.27537E-02 .11994E-02
TEMPERATURES (Deg C)

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>Temp</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2.00000</td>
<td>-2.00000</td>
<td>14027</td>
</tr>
<tr>
<td>-2.00000</td>
<td>-1.00000</td>
<td>28004</td>
</tr>
<tr>
<td>-2.00000</td>
<td>.00000</td>
<td>20526</td>
</tr>
<tr>
<td>-2.00000</td>
<td>1.00000</td>
<td>-.00343</td>
</tr>
<tr>
<td>-2.00000</td>
<td>2.00000</td>
<td>-.09997</td>
</tr>
<tr>
<td>-1.00000</td>
<td>-2.00000</td>
<td>.20032</td>
</tr>
<tr>
<td>-1.00000</td>
<td>-1.00000</td>
<td>1.37087</td>
</tr>
<tr>
<td>-1.00000</td>
<td>.00000</td>
<td>1.31407</td>
</tr>
<tr>
<td>-1.00000</td>
<td>1.00000</td>
<td>-.31126</td>
</tr>
<tr>
<td>-1.00000</td>
<td>2.00000</td>
<td>-.35746</td>
</tr>
<tr>
<td>.00000</td>
<td>-2.00000</td>
<td>.17064</td>
</tr>
<tr>
<td>.00000</td>
<td>-1.00000</td>
<td>.82908</td>
</tr>
<tr>
<td>.00000</td>
<td>.00000</td>
<td>.72430</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>115</td>
<td>.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>116</td>
<td>.00000</td>
<td>2.00000</td>
</tr>
<tr>
<td>117</td>
<td>1.00000</td>
<td>-2.00000</td>
</tr>
<tr>
<td>118</td>
<td>1.00000</td>
<td>-1.00000</td>
</tr>
<tr>
<td>119</td>
<td>1.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>120</td>
<td>1.00000</td>
<td>2.00000</td>
</tr>
<tr>
<td>121</td>
<td>2.00000</td>
<td>-2.00000</td>
</tr>
<tr>
<td>122</td>
<td>2.00000</td>
<td>-1.00000</td>
</tr>
<tr>
<td>123</td>
<td>2.00000</td>
<td>.00000</td>
</tr>
<tr>
<td>124</td>
<td>2.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>125</td>
<td>2.00000</td>
<td>2.00000</td>
</tr>
</tbody>
</table>
REFERENCES


APPENDIX 1. LISTING OF COMPUTER CODE

1 C----------------------------------------------
2 C
3 C PROGRAM MPC
4 C
5 C
6 C
7 C
8 C
9 C Multipole method to compute the heat flows
10 C and temperatures to and between insulated
11 C circular pipes in a composite cylinder with
12 C two concentric regions with different
13 C thermal conductivities
14 C
15 C Authors: Johan Bennet, Johan Claesson, Goran Hellstrom
16 C
17 C Departments of Building Technology and Mathematical Physics
18 C
19 C Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden
20 C
21 C Date: 1987-03-25
22 C
23 C Reference: Notes on Heat Transfer 3 - 1987
24 C----------------------------------------------
25 PARAMETER (IW=15, IJ=10, IW2=IW*IW)
26 C
27 COMPLEX*8 Z(IW), P(IW,IJ), P2(IW,IJ), PC(IJ), ZRC(IW,IJ), TERM
28 + ,RPZMN(IW,IW,IJ), RPMZN(IW,IW,IJ), PMK, PRB, ZPR, RPZ(IW,IJ)
29 DIMENSION RP(IW), BETA(IW), TF(IW), QBEG(IW)
30 + ,Q(IW), RRR(IW), QM(IW)
31 DOUBLE PRECISION RKO(IW,IW), RKOVEC(IW2), MC(IW), MD(IW)
32 CHARACTER*16 READFI, OUTFI
33 CHARACTER ANS
34 LOGICAL INFIL, OK
35 DATA INFIL/.FALSE.I
36 C********** OPEN INPUT AND OUTPUT FILES **********
37 WRITE(*,441)
38 WRITE(*,*)
39 2 WRITE(*,*)' Input data from file? (Y/N)'
40 READ(*,5) ANS
41 IF((ANS.EQ.'Y').OR.(ANS.EQ.'y')) THEN
42 WRITE(*,*) ' Name of input file ?'
43 READ(*,10) READFI
44 INQUIRE(FILE=READFI, EXIST=OK)
45 IF (OK) THEN
46 OPEN(UNIT=5, FILE=READFI)
47 INFIL=.TRUE.
48 ELSE
49 WRITE(*,*)' *** ERROR ***','READFI', 'File not found'
50 GO TO 2
51 ENDIF
52 WRITE(*,*) ' Name of output file ?'
53 READ(*,10) OUTFI
54 INQUIRE(FILE=OUTFI, EXIST=OK)
55 IF (OK) THEN
56 OPEN(UNIT=6, FILE=OUTFI)
57 IFIL=.TRUE.
58 ELSE
59 OPEN(UNIT=6, FILE=OUTFI, STATUS='NEW')
60 ENDIF
61 5 FORMAT(A1)
62 10 FORMAT(A16)
63 C********** READ INPUT DATA **********
64 IF(INFIL) THEN
65 READ(5,*) RLAMB, RLM, NW, J
READ(5,*) RB,RC,BETAC,TC
DO 15 M=1,NW
   READ(5,*) ZRE,ZIM,RP(M),BETA(M),TF(M)
   Z(M)=CMPLX(ZRE,ZIM)
15 CONTINUE
READ(5,*) EPS,ITMAX
READ(5,*) XMIN,XMAX,YMIN,YMAX,NX, NY
ELSE
WRITE(*,*)'Thermal conductivity in the inner region ?'
READ(*,*) RLAMB
WRITE(*,*)'Thermal conductivity in the outer region ?'
READ(*,*) RLAM
WRITE(*,*)'Number of pipes ?'
READ(*,*) NW
WRITE(*,*)'Order of multipoles ?'
READ(*,*) J
WRITE(*,*)'Radius of inner region ?'
READ(*,*) RB
WRITE(*,*)'Radius of outer region ?'
READ(*,*) RC
WRITE(*,*)'OUTER BOUNDARY'
WRITE(*,*)'Thermal resistance coefficient ?'
READ(*,*) BETAC
WRITE(*,*)' Temperature ?'
READ(*,*) TC
DO 16 M=1,NW
   WRITE(*,443) M
   WRITE(*,*)' x-value ?'
   READ(*,*) ZRE
   WRITE(*,*)' y-value ?'
   READ(*,*) ZIM
   Z(M)=CMPLX(ZRE,ZIM)
   WRITE(*,*)' Pipe radius ?'
   READ(*,*) RP(M)
   WRITE(*,*)' Thermal resistance coefficient ?'
   READ(*,*) BETA(M)
   WRITE(*,*)' Temperature ?'
   READ(*,*) TF(M)
16 CONTINUE
WRITE(*,*)
WRITE(*,*)'Iteration accuracy ?'
READ(*,*) EPS
WRITE(*,*)'Maximum number of iterations ?'
READ(*,*) ITMAX
WRITE(*,*)
WRITE(*,*)'TEMPERATURE OUTPUT: Define rectangular area'
WRITE(*,*)' Minimum x-value ?'
READ(*,*) XMIN
WRITE(*,*)' Maximum x-value ?'
READ(*,*) XMAX
WRITE(*,*)' Minimum y-value ?'
READ(*,*) YMIN
WRITE(*,*)' Maximum y-value ?'
READ(*,*) YMAX
WRITE(*,*)' Number of grid points along the x-axis ?'
READ(*,*) NX
WRITE(*,*) 'Number of grid points along the y-axis?' READ(*,*) NY WRITE(*,*) IF(.NOT.INFIL) THEN WRITE(*,*)'Save input data on file? (Y/N)'
READ(*,5) ANS IF((ANS.EQ.'Y').OR.(ANS.EQ.'y')) THEN WRITE(*,*) 'Name of input file'
READ(*,10) READFI INQUIRE(FILE=READFI,EXIST=OK)
IF (OK) THEN OPEN(UNIT=5,FILE=READFI)
ELSE OPEN(UNIT=5,FILE=READFI,STATUS='NEW') ENDIF WRITE(5,701) RLAMB,RLAM,NW,J WRITE(5,702) RB,RC,BETAC,TC DO 17 M=1,NW
ZRE=REAL(Z(M))
ZIM=AIMAG(Z(M))
WRITE(5,702)
ZRE,ZIM,RP(M),BETA(M),TF(M) CONTINUE WRITE(5,703) EPS,ITMAX WRITE(5,704) XMIN,XMAX,YMIN,YMAX,NX,NY CLOSE(UNIT=5) ENDIF C********** CONSISTENCY TESTS OF INPUT DATA ********** IF(RLAMB.LE.0.) STOP ' RLAMB < 0.' IF(RLAM.LE.0.) STOP ' RLAM < 0.' IF(NW.LT.1) STOP ' NW < 1' IF(J.LT.0) STOP ' J < 0' IF(RB.LE.0.) STOP ' RB < 0.' IF(RC.LE.0.) STOP ' RC < 0.' IF(BETAC.LT.0.) STOP ' BETAC < 0.' DO 20 N=1,NW
IF(RB.LT.CABS(Z(N))+RP(N)-1.E-6) STOP ' RB < R(N)+RP(N)' IF(BETA(N).LT.0.) STOP ' BETA(N) < 0.' IF(RP(N).LE.0.) STOP ' RP(N) < 0.' DO 20 M=1,NW
IF(M.NE.N) THEN
IF(CABS(Z(N)-Z(M))-RP(N)-RP(M)-1.E-6) STOP ' RMN < RP(N)+RP(M)' IF(CABS(Z(N)-Z(M))-RP(N)-RP(M).LE.1.E-6) THEN
IF(BETA(N)+BETA(M).EQ.0. AND TF(N).NE.TF(M)) +STOP ' RMN=RP(N+RP(N) AND BETA(N)+BETA(M)=O. AND TF(N) NE TF(N)' END IF END IF 20 CONTINUE IF(EPS.LE.0.) STOP ' EPS LE 0.' C********** LISTING OF INPUT DATA ********** WRITE(*,+) WRITE(*,441) WRITE(*,450) RLAMB,RLAM,NW,J WRITE(*,450) RB,RC,BETAC,TC WRITE(*,470) (M,Z(M),RP(M),BETA(M),TF(M),M=1,NW)
WRITE(*,480) EPS,ITMAX
WRITE(*,490) XMIN,XMAX,YMIN,YMAX,NX,NY
WRITE(6,441)
WRITE(6,442) READFI,OUTFI
WRITE(6,450) RLAMB,RLAM,NW,J
WRITE(6,460) RB,RC,BETAC,TC
WRITE(6,470) (M,Z(M),RP(M),BETA(M),TF(M),M=1,NW)
WRITE(6,480) EPS,ITMAX
WRITE(6,490)
WRITE(6,410)
IIT=0
PI=3.14159265
PILAMB=1./(2.*PI*RLAMB)
PILAM=1./(2.*PI*RLAM)
SIGMA=(RLAMB-RLAM)/(RLAMB+RLAM)
ALBETC=BETAC+ALOG(RC/RB)
C********** SET CONSTANTS **********
DO 60 M=1,NW
RBM=RB**2/(RB**2-CABS(Z(M)**2)
RKO(M,M)=PILAMB*(ALOG(RB/RP(M))+BETA(M)+SIGMA*ALOG(RBM)+
+ PILAM*ALBETC
IF(J.GE.1) THEN
DO 30 K=1,J
ZRC(M,K)=(O.,O.)
RPZ(M,K)=(O.,O.)
RPMZN(M,N,K)=(RP(M)*CONJG(Z(M))*RBM/RB**2)**K
IF(CABS(Z(M)) .NE.O. ) THEN
ZRC(M,K)=(Z(M)/RC)**K
RPZ(M,K)=(RP(M)/Z(M)**K
END IF
30 CONTINUE
END IF
DO 50 N=1,NW
IF(M.NE.N) THEN
PMK=Z(N)-Z(M)
RMN=CABS(PMK)
RBM=RB**2/CABS(RB**2-Z(N)*CONJG(Z(M))
PRB=RP(M)*CONJG(Z(N))/(RB**2-CONJG(Z(N))*Z(M))
RKO(M,N)=PILAMB*(ALOG(RB/RMN)+SIGMA*ALOG(RBM))+
+ PILAM*ALBETC
IF(J.GE.1) THEN
DO 40 K=1,J
RPZMN(M,N,K)=(RP(M)/PMK)**K
END IF
40 CONTINUE
END IF
DO 60 N=1,NW
60 CONTINUE
C********** CALCULATION OF MATRIE RKO **********
C********** AND AUXILIARY ARRAYS **********
DO 64 JJ=1,NW
DO 62 I=1,NW
K=K+1
RKOVEC(K)=RKO(I,JJ)
62 CONTINUE
64 CONTINUE
C********** CONVERT MATRIX RKO TO VECTOR RKOVEC **********
K=0
DO 66 JJ=1,NW
DO 66 I=1,NW
K=K+1
RKOVEC(K)=RKO(I,JJ)
66 CONTINUE
CONTINUE
C*********** INVERSION OF MATRIX RKO ***********
CALL INV1(RKOVEC,MC,MD,NW,DET)
C*********** CONVERT VECTOR RKOVEC TO MATRIX RKO
K=0
DO 68 JJ=1,NW
  DO 66 I=1,NW
    K=K+1
    RKO(I,JJ)=RKOVEC(K)
  CONTINUE
68 CONTINUE
66 CONTINUE
C******************** INITIAL VALUES OF ENERGY FLOWS **********
C******************** AND MULTIPOLES **********
DO 80 M=1,NW
  QBEG(M)=0.
  DO 10 N=1,NW
    QBEG(M)=QBEG(M)+RKO(M,N)*(TF(N)-TC)
  CONTINUE
Q(M)=QBEG(M)
80 CONTINUE
WRITE(*,430) (N,Q(N),N=1,NW)
WRITE(6,430) (N,Q(N),N=1,NW)
IF(J.EQ.0) GO TO 280
DO 90 M=1,NW
  DO 90 K=1,J
    P(M,K)=(0.,0.)
  CONTINUE
PC(K)=(0.,0.)
90 CONTINUE
C****************** START OF ITERATION LOOP ****************
WRITE(*,495) EPS
DO 270 IIT=1,ITMAX
  EPSMAX=0.
  C****************** MULTIPOLES AT THE PIPES ****************
  DO 160 M=1,NW
    PMMAX=0.
    DO 150 K=1,J
      PRB=1./(RB**2-CONJG(Z(N)*Z(M))*KFAK=1
      DO 120 JJ=1,J
        IF(N.NE.M) PMK=PMK+P(N, JJ)*RPZMN(N, M, JJ)
        *RPZMN(M, N, K)*KFAK
        JPEND=MINO(JJ,K)
        KFAK1=KFAK
        KFAK2=1
        DO 110 JPRIM=0,JPEND
          JJPRIM=JJ-JPRIM
          KJPRIM=K-JPRIM
          TERM=(1.,0.)
          IF(JPRIM.GE.1) TERM=CONJG(RPMZ(N, M, JJPRIM))
          IF(KPRIM.GE.1) TERM=TERM*RPZMN(M, N, KPRIM)
          PMK=PMK+CONJG(P(N, JJ))*SIGMA*TERM*(RP(N)*RP(N)
          +PRB)**JPRIM*KFAK1*KFAK2
36
IF(JPRIM.NE.JPEND) THEN
  KFAK1=KFAK1*KJPRIM/(K+JJ-1-JPRIM)
  KFAK2=KFAK2*JJPRIM/(JPRIM+1)
END IF

CONTINUE

KFAK=KFAK*(K+JJ)/JJ

CONTINUE

IF(N.NE.M) THEN
  PMK=PMK+Q(N)*PILAMB*RPZMN(M,N,K)/K
  PMK=PMK+Q(N)*PILAMB*SIGMA*RPMZN(M,N,K)/K
END IF

DO 140 JJ=K,J
  PMK=PMK+PC(JJ)*(1.-SIGMA)*RPZ(M,K)*ZRC(M,JJ)*KFAK
  KFAK=KFAK*(JJ+1)/(JJ+1-K)
CONTINUE

PMK=CONJG(PMK)*(BETA(M)*K-1.)/(BETA(M)*K+1.)

PMMAX=AMAX1(CABS(PMK),PMMAX)

IF(CABS(PMK).GT.1.E-1) THEN
  EPSMAX=AMAX1(EPSMAX,CABS(PMK-P(M,K)/PMAX))
  PC (K)=PMK
END IF

CONTINUE

DO 170 M=1,NW
  DO 170 K=1,J
    P(M,K)=P2(M,K)
  170 CONTINUE

************ CALCULATION OF MULTIPOLES **************

DO 200 K=1,J
  PMK=(0.,0.)
  DO 190 M=1, NW
    KFAK=1
    DO 180 JJ=1,K
      PMK=PMK+P(M,JJ)*ZRC(M,K)*RPZ(M,JJ)*KFAK
    180 CONTINUE
    PMK=PMK+ZRC(M,K)*Q(M)*PILAMB/K
  190 CONTINUE

PMK=CONJG(PMK)*((1.-SIGMA)*X/(SIGMA*X*(RB/RC)**2-K-1.))

PMAX=AMAX1(CABS(PMK),PMAX)

IF(CABS(PMK).GT.1.E-7) THEN
  EPSMAX=AMAX1(EPSMAX,CABS(PMK-PC(K)/PMAX))
  PC (K)=PMK
END IF

CONTINUE

DO 240 M=1,NW
  QQQ=0.
  DO 220 N=1,NW
    DO 210 JJ=1,J
      IF(M.NE.N) THEN
        QQQ=QQQ+REAL(RPZMN(N,M,JJ)*P(N,JJ))
      ELSE
        QQQ=QQQ+REAL(P(N,JJ)*RPZMN(N,M,JJ))
      END IF
    210 CONTINUE
  220 CONTINUE
  DO 230 JJ=1,J
    XXX=1./(1.+BETAC*K)
    PMK=CONJG(PMK)*XX*(SIGMA+1.)/(SIGMA*X*(RB/RC)**(2*K)-1.)
    PMMAX=AMAX1(CABS(PMK),PMAX)
    IF(CABS(PMK).GT.1.E-7) THEN
      EPSMAX=AMAX1(EPSMAX,CABS(PMK-PC(K)/PMAX))
      PC (K)=PMK
    END IF
  230 CONTINUE

************ AT THE OUTER CIRCLE **************

PMAX=0.

DO 200 K=1,J
  PMK=(0.,0.)
  DO 190 M=1, NW
    KFAK=1
    DO 180 JJ=1,K
      PMK=PMK+P(M,JJ)*ZRC(M,K)*RPZ(M,JJ)*KFAK
    180 CONTINUE
    PMK=PMK+ZRC(M,K)*Q(M)*PILAMB/K
  190 CONTINUE

XX=(1.-BETAC*K)/(1.+BETAC*K)

PMK=CONJG(PMK)*XX*(SIGMA+1.)/(SIGMA*X*(RB/RC)**(2*K)-1.)

PMMAX=AMAX1(CABS(PMK),PMAX)

IF(CABS(PMK).GT.1.E-7) THEN
  EPSMAX=AMAX1(EPSMAX,CABS(PMK-PC(K)/PMAX))
  PC (K)=PMK
END IF

CONTINUE

DO 240 M=1,NW
  QQQ=0.
  DO 220 N=1,NW
    DO 210 JJ=1,J
      IF(M.NE.N) THEN
        QQQ=QQQ+REAL(RPZMN(N,M,JJ)*P(N,JJ))
      ELSE
        QQQ=QQQ+REAL(P(N,JJ)*RPZMN(N,M,JJ))
      END IF
    210 CONTINUE
  220 CONTINUE
  DO 230 JJ=1,J
340   QQQ=QQQ+(1.-SIGMA)*REAL(PC(JJ)*ZRC(M,JJ))
341  230  CONTINUE
342   QM(N)=QQQ
343  240  CONTINUE
344   DO 260 M=1,NW
345     QQQ=0.
346   DO 250 N=1,NW
347     QQQ=QM(N)+RKO(M,N)+QQQQ
348  250  CONTINUE
349     Q(N)=QBEG(M)-QQQQ
350  260  CONTINUE
351 C******************************************************************************
352 C******************************************************************************
353 WRITE(*,600) IIT,EPSMAX
354 IF(EPSMAX.LT.EPS) GO TO 280
355 270 CONTINUE
356 C******************************************************************************
357 C******************************************************************************
358 280 WRITE(6,370) IIT
359     WRITE(*,370) IIT
360     WRITE(6,380)(N,Q(N),N=1,NW)
361     WRITE(*,380) (N,Q(N),N=1,NW)
362 IF(J.GE.1) THEN
363     WRITE(6,390)((N,K,Q(N,K),K=1,J),N=1,NW)
364     WRITE(6,420)(K,PC(K),K=1,J)
365 END IF
366     WRITE(6,440)
367     WRITE(6,395)
368 C******************************************************************************
369 C******************************************************************************
370   DO 360 MPR=0,NX
371   DO 360 NPR=0,NY
372     ZP=CMPLX((XMAX-XMIN)*MPR/NX+XMIN,(YMAX-YMIN)*NPR/NY+YMIN)
373   C
374   DO 360 MPR=0,100
375     C
376   DO 290 N=1,NW
377     IF(CABS(ZP-Z(N)).LT.RP(N)-1.E-6) GO TO 360
378  290 CONTINUE
379     IF(CABS(ZP).GT.RC) GO TO 360
380     PMK=(0.,0.)
381     IF(CABS(ZP).LE.RB) THEN
382     DO 310 N=1,NW
383     IF(J.GE.1) THEN
384     PRB=1./(RB**2-Z(N)*CONJG(ZP))
385     DO 300 K=1,J
386     PMK=PMK+P(N,K)*((RP(N)/(ZP-Z(N)))**K+SIGMA*(RP(N)*CONJG(ZP)*PRB)**K)
387     300 CONTINUE
388     END IF
389     RON=CABS(ZP-Z(N))
390     PMK=PMK+Q(N)*(PILAM+ALBETC+
391     PILAM*(ALOG(RB/RON)+SIGMA*ALOG(RB**2+CABS(PRB))))
392  310 CONTINUE
393     IF(J.GE.1) THEN
394     DO 320 K=1,J
395     END IF
396     DO 320 K=1,J
397     END IF
398     DO 320 K=1,J
PMK=PMK+(1.-SIGMA)*PC(K)*(ZPR/RC)**K
CONTINUE

ELSE

C****************************** CALCULATION OF TEMPERATURES ***************
C****************************** WHEN ZPR GREATER THEN RB ***************

DO 340 N=1,NW
   IF(J.GE.1) THEN
      DO 330 K=1,J
         PMK=PMK+P(N,K)*(1.+SIGMA)*(RP(N)/(ZPR-Z(N)```**K
      CONTINUE
      RON=CABS(ZPR-Z(N))
      PMK=PMK+Q(N)*PILAM*(ALBETC+SIGMA*ALOG(RB/CONJG(ZPR)))
      CONTINUE
      IF(J.GE.1) THEN
         DO 350 K=1,J
            PMK=PMK+PC(K)*ZPR/RC)**K-SIGMA*(RB**2/(RC*CONJG(ZPR))**K)
         CONTINUE
      END IF
   END IF
   TPR=REAL(PMK)+TC
   X=REAL(ZPR)
   Y=AIMAG(ZPR)
   WRITE(6,400) X,Y,TPR
   END IF

WRITE(*,510)
FORMAT(/' Number of iterations: ',14/)
370 FORMAT(/' Pipe q(n)'/(I5,3X,E14.7/)
380 FORMAT(/' Pipe',4X,'Order',11X,'P(n,k)'/(I5,I8,3X,2E12.5/)
390 FORMAT(/'X',13X,'Y',12X,'Temp',/)
400 FORMAT(2X,F10.5,4X,F10.5,4X,F10.5)
410 FORMAT(/,75('*')/)
420 FORMAT(/' Order',13X,'PC(k)'/(I5,4X,2E12.5/)
430 FORMAT(/' Pipe Initial values for q(n) (Order of multipole = 0)'/
   +/(I5,E17.7/)
440 FORMAT(/' TEMPERATURES (Deg C)'/)
450 FORMAT(/' MULTIPOLE METHOD - Pipes in a composite cylinder'/,
   +1X,48(IH=),//,' INPUT DATA')
460 FORMAT(/' Output file ',A16,/,' Output file ',A16,/)
470 FORMAT(/' Pipe n','I2)
490 FORMAT(/' Outer boundary'/,
   +' Thermal resistance coefficient ',2X,E10.3,/
   +' Temperature',2X,E10.3,/
   +' Iteration accuracy',13X,E10.2/
+ ' Maximum number of iterations',2X,I10)
459 490 FORMAT('/' TEMPERATURE OUTPUT: Definition of rectangular area',/,
460 + ' Minimum x-value',10X,F10.3/
461 + ' Maximum x-value',10X,F10.3/
462 + ' Minimum y-value',10X,F10.3/
463 + ' Maximum y-value',10X,F10.3/
464 + ' Number of grid points along x-axis',111/
465 + ' Number of grid points along y-axis',111)
466 495 FORMAT('/' Starting iterations Goal: Iteration accuracy ',E10.2,/)n
467 500 FORMAT(' Number of iterations',I4,' Iteration accuracy ',E10.2)
468 510 FORMAT(' Job ended. Output written to ',A16)
469 701 FORMAT(1X,2(E12.5,2X),216)
470 702 FORMAT(1X,6(E12.5,2X),216)
471 703 FORMAT(1X,E12.5,I7)
472 704 FORMAT(1X,4(E12.6,2X),215)
473 END
474 SUBROUTINE INV1(B,MC,MD,N,D)
475 C-----------------------------------------------------------------------
476 C Calculate inverse of a matrix
477 C-----------------------------------------------------------------------
478 IMPLICIT REAL*8 (A-H,O-Z)
479 DIMENSION B(l),MC(l),MD(l)
480 DATA Cl/1.0DO/
481 DATA C2/1.0D50/
482 C
483 D=C1
484 NK=-N
485 DO 80 K=1,N
486 NK=NK+N
487 MC(K)=K
488 MD(K)=K
489 KK=NK+K
490 BIGA=B(KK)
491 DO 20 J=K,N
492 IZ=N*(J-l)
493 DO 20 I=K,N
494 IJ=IZ+I
495 Rl=B(IJ)
496 IF(Rl.LT.0.0DO) Rl=-Rl
497 R2=BIGA
498 IF(R2.LT.0.0DO) R2=-R2
499 R2=R2-Rl
500 IF(R2.GT.0.0DO) GO TO 20
501 BIGA=B(IJ)
502 MC(K)=I
503 MD(K)=J
504 20 CONTINUE
505 J=MC(K)
506 IF(J-K.LE.0) GO TO 35
507 KI=K-N
508 DO 30 I=1,N
509 KI=KI+N
510 HOLD=-B(KI)
JI = KI - K + J
B(KI) = B(JI)
30 B(JI) = HOLD
35 I = MD(K)
IF(I - K .LE. 0) GO TO 45
JP = N * (I - 1)
DO 40 J = 1, N
JK = NK + J
JI = JP + J
HOLD = B(JK)
B(JK) = B(JI)
40 B(JI) = HOLD
45 IF(ABS(BIGA) .GT. 0.0) GO TO 48
46 D = 0.0
RETURN
48 DO 65 I = 1, N
IF(I - K .EQ. 0) GO TO 55
IK = NK + I
B(IK) = B(IK) / (-BIGA)
55 CONTINUE
DO 65 I = 1, N
IK = IK + I
B(IK) = B(IK) / BIGA
55 CONTINUE
DO 75 I = 1, N
IK = IK + I
HOLD = B(IK)
IJ = I - N
DO 65 J = 1, N
IJ = IJ + N
IF(I - K .EQ. 0) GO TO 55
IF(J - K .EQ. 0) GO TO 65
KJ = IJ - I + K
B(IJ) = HOLD * B(KJ) + B(IJ)
65 CONTINUE
KJ = K - N
DO 75 J = 1, N
KJ = KJ + N
IF(J - K .EQ. 0) GO TO 75
B(KJ) = B(KJ) / BIGA
75 CONTINUE

UNDEFINED VALUE OF D IF ABS(D) .GT. C2
IF(D .LT. C2 .AND. D .GT. -C2) D = D * BIGA
B(KK) = C1 / BIGA
80 CONTINUE
K = N
100 K = K - 1
IF(K .LE. 0) GO TO 150
I = MC(K)
108 JQ = N * (K - 1)
JR = R * (I - 1)
DO 110 J = 1, N
JK = JQ + J
HOLD = B(JK)
JI = JR + J
B(JK) = B(JI)
110 B(JI) = HOLD

41
568  120  J=MD(K)
569       IF(J-K.LE.0) GO TO 100
570  125  KI=K-N
571  130    DO 130 I=1,N
572  130    KI=KI+N
573  130    HOLD=B(KI)
574  130    JI=KI-K+J
575  130    B(KI)=-B(JI)
576  130    B(JI)=HOLD
577  150       GO TO 100
578  150       CONTINUE
579  150       RETURN
580  150       END