National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Canada

A STUDY OF THE THERMAL FIELD SURROUNDING BURIED DISTRICT HEATING PIPES

Helen E. Reeve

A thesis submitted to the School of Graduate Studies and Research

in partial fulfillment of the requirements for the

degree of

MASTER OF APPLIED SCIENCE

in the Department of Chemical Engineering

University of Ottawa

August 1995

ISBN 0-612-11591-7

Canada

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

ABSTRACT

Traditional deep buried water and sewer services in northern communities can be prohibitively expensive due to deep frost penetration. Alternative municipal systems can also be uneconomic owing to cost-intensive freeze protection schemes. The integration of district heating (DH) systems with domestic services allows shallow burial of all services in a common trench by using the waste heat from the DH pipes as freeze protection for the nearby municipal lines, thus providing an economically attractive solution. A steady-state 2-dimensional numerical model, based on finite volume principles, was developed as a tool for the design of integrated services systems. The model can predict the thermal field surrounding any number of pipes in any configuration and under a wide variety of ground conditions. Insulation boards can be modeled in any configuration in the soil surrounding the pipes. The main grid system is a rectangular coordinate system with local embedded sub-grids in cylindrical coordinates inserted around each pipe to reflect the radial flow of heat in the vicinity of the pipes. This numerical model was tested and proved to be self-consistent. Predicted temperatures were compared with temperatures taken around experimental pipe loops. Two systems of pipe loops with different diameters were tested, each system having an insulation board placed directly above the pipes. The predicted temperatures agreed well with the experimental temperatures. Anomalies found were explained by settling of the sand and deficiencies in backfilling in the region of the pipes due to the proximity of the insulation board to the pipes. These effects were modeled by inserting a thin air gap around the pipes, resulting in closer agreement between predicted and experimental temperatures. The model was determined to be a valid tool for the design of integrated services systems.

Dedicated to my parents, Sheila and David, for all their love and support, especially my father who first introduced me to heat transfer.

ACKNOWLEDGMENTS

I would like to sincerely thank my thesis supervisor, Dr. W. Haliett, for all of his help and advice throughout the course of this project. His thoroughness and his constant enthusiastic support have been greatly appreciated.

The design, planning, purchasing, construction, supervision, instrumentation and implementation of the experimental system as well as the design, development and programming of the numerical model was done by the writer. However, as this work was done at CANMET's Energy Research Lab I would like to acknowledge the help of some of the staff at CANMET:

Thanks to Chris Snoek of CANMET, the scientific authority for this project, for setting and overseeing the direction of the work. I appreciated his continuous involvement and support, especially with the experimental component of the project.

Grateful acknowledgment is given to Don Hampton of CANMET for his help in the design of the experimental loops and to Stephane Joanis for his help with numerous computer applications.

Many thanks to CANMET for funding the project, especially Michael Wiggin for his interest and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

The purpose of this work is to develop tools for the thermal design of integrated heating, water and sewer services for northern communities where separate installation of these services may be prohibitively expensive. To accomplish this, a study of the thermal field surrounding buried district heating pipes was initiated. The project consists of two components: the development of a numerical model to predict the thermal field surrounding buried pipes and the verification of this model with experimental soil temperature data, measured around experimental hot water pipe loops.

1.1 Background

There are many northern communities which experience frost penetration deep into the earth. Excavation for traditional municipal water and sewer lines becomes uneconomic in these locations. Alternatives to traditional systems do exist; however, the associated operation and maintenance costs of freeze protection for these systems, which typically involve recirculation and/or heat tracing, can become prohibitively expensive. Eastern regions of Canada also suffer from expensive excavation costs owing to the extremely rocky terrain. The integration of district heating (DH) systems with domestic services offers an economically attractive, technically feasible, and environmentally friendly solution to this problem.

District heating systems deliver heat produced at a central plant to buildings in a community as hot water via an underground double pipeline system. Heat is used for space heating, domestic hot water and/or for industrial processes. District heating thus displaces the generation of energy within each individual building. At each building, a small heat exchanger extracts the heat required for that building from the main DH supply line to which it is connected. The cooled water returns to the central plant via the main DH return line to be re-heated. District heating systems are particularly appropriate for serving more densely populated areas, within which consumers can be both large and small. Entire independent communities are, therefore, ideally suited for district heating systems. The distribution network pipelines of a district heating system are integral to the integrated technology concept.

Integration of DH systems with domestic water and sewer systems entails burial of all service lines (DH and municipal) in one common trench. Heat losses from the DH pipes are used as freeze protection for the municipal lines. Specifically, with an insulation board buried directly on top of the shallow buried DH pipes, the thermal field created by the heat loss from the DH lines can be sufficiently warm to prevent the municipal lines buried nearby from freezing. In this manner, integrated services technology facilitates shallow burial of municipal lines, providing more economically viable systems to northern communities. Not only are excavation costs reduced due to shallow burial and the use of one common trench for all service lines, but using waste heat as freeze protection also eliminates the substantial operating and maintenance costs associated with

the freeze protection of existing systems. Further, there are many benefits a district heating system offers to independent communities, some of which translate into dollar figures, but many others of which do not. One such benefit is the opportunity to achieve self-sufficiency by using local resources to fuel the district heating system rather than importing oil and gas and depending on provincial utilities. Details of the benefits of district heating systems will not be discussed here but play an important role in the marketing of this new integrated technology.

1.2 Objectives

The primary objective of this research project is to develop tools for facilitating the thermal design of integrated services systems with any number of pipes, in any configuration, under a wide variety of ground conditions. This was intended specifically for northern communities; however, meeting broader requirements increases the possible applications and facilitates future work.

To meet the primary objective, a numerical model, based on finite volume principles, was developed to predict the thermal field surrounding buried pipelines and the heat losses from those pipelines. The model can handle any number of pipes, in any configuration, with or without insulation on the pipes and insulation in the soil. To verify this model, temperature measurements were taken in the soil surrounding experimental hot water pipe loops. Each experimental system consists of two parallel pipes, running side by side, simulating supply and return district heating pipes. Soil temperatures were

monitored over the winter and spring of 1994/1995. Experimental temperature data from these loops were compared with predicted temperatures from the model, run with the experimental configuration. Further successful verification of the model was accomplished through comparison with the well-established analytical solution for a single buried pipe, and through checking that energy is conserved within the predicted calculation domain.

This report begins with a literature survey, then details the development of the numerical model, the description of experimental loops and the set-up and functioning of the experimental equipment. Experimental and model results are presented and comparisons are drawn between the two for verification of the model.

2.0 LITERATURE REVIEW

The purpose of this literature review was to gain insight into the research that has been done world-wide in the general field of heat transfer to or from buried pipelines. Most of the papers reviewed concentrate on heat conduction through the soil, but a few of them examine heat transfer in pipeline trenches and flow in the pipe as well. Some works are contradictory, in that what may be dismissed as unimportant by one researcher may be another researcher's entire focus. The following cites the research collected, gives the purpose of the paper, its conclusions and a general idea of the methods used to reach those conclusions.

2.1 Experiments and Analyses of Heat Transfer through Soils from Pipelines

The research most directly relevant to district energy is that of Phetteplace and co-workers at the US Army Cold Regions Engineering Laboratory in Hannover, New Hampshire. Phetteplace (1991a) compares actual heat losses from an operational system with calculated results. Four different configurations were studied: (a) pipes in a shallow concrete trench; (b) pipes within a common conduit; (c) pipes buried in individual conduits; and (d) two pre-insulated buried pipes. Heat loss from these systems was estimated via five different methods (the specific method depending on the system): (i) the insulation method, which involves calculating the heat flow through the insulation from temperature readings inside and outside the insulation; (ii) the soil method, in which the soil temperature is taken at the

depth of burial of the pipe to approximate the variations in heat transfer rate from a buried heat distribution system due to variations at the surface (used for the common conduit system only); (iii) a method for two buried pipes in individual conduits, which involves calculation of the steady-state conductance through the pipe, insulation and soil, while accounting for the thermal resistance of the air space and thermal interaction between pipes; (iv) a method for two pipes buried in a common conduit, which also employs a convective heat transfer coefficient; and (v) the soil method for two direct buried pipes, which uses a coupled two-pipe resistance formulation. The heat losses found experimentally proved reliable and repeatable. For example, the heat loss found for the low temperature hot water site (LTHW) at a supply temperature of 86°C and a soil temperature of 26°C was 18.7 W/m via the insulation method and 20.7 W/m via the soil method.

Phetteplace and Meyer (1990) provide an extensive report on many aspects of thermal distribution systems, including piping system layout, types of system construction, thermal design considerations, thermal properties, methods of heat transfer analysis, hydraulic considerations, pipe sizing and network calculations. The section on methods of heat transfer analysis provides detailed descriptions of analytical models for single uninsulated buried pipes, a single insulated buried pipe, a single buried pipe in a conduit with an air space, two pipes in a common conduit, two buried pipes or conduits, pipes in trenches or tunnels and pipes in shallow trenches. This paper's methods use steady-state analyses, as the authors believe that adjustments to changes in surface temperature occur sufficiently quickly that the difficult transient solution is not worthwhile. The variation in measured soil

properties is reported as ±25%, leading the authors to question the value of highly detailed numerical models.

Much of the work in the previous two papers is also presented in Phetteplace et al. (1991b), including quantification of heat losses from some of the same experimental distribution systems for purposes of identifying possible improvements in the existing methods and systems. Again, the heat losses measured by the different methods proved reliable and repeatable. Phetteplace et al. (1986) present a thermal analysis of a shallow utilidor intended to prevent freezing of utility lines in cold climates. A two dimensional steady-state finite element program was used to calculate heat loss. The system model was based on the resistance method: thermal resistances to heat transfer through the pipe, the insulation, the air space between the outer pipe casing and the utilidor wall, the utilidor wall, the soil and to the atmosphere are estimated. Another finite element model of the system was developed, reflecting the actual dimensions so as to facilitate comparison between measured soil temperatures and calculated temperatures. Results were good, considering the use of the steady state approximation, the uncertainty in the thermal properties of the soil, and possible errors in thermocouple readings.

Bohm (1992) derives the efficiency of a district heating system as a function of line heat demand and the heat loss. Six experimental methods for determining heat loss are presented. A new method divides the instantaneous heat loss into steady-state and transient (ie. due to climatic changes) components. This latter determines the temperature field in the

ground based on the assumption that the transient component is one-dimensional and the thermal conductivity of the ground is known. It is shown that heat flux meters are more accurate for measurement than directly measuring the surface temperature of the pipe casing; however, there were problems with calibration of the flux meters. "Sandbox" experiments and finite element analysis are used to show that when steady-state heat loss equations are used to calculate instantaneous (transient) heat losses the use of a soil temperature between the pipe centre and the ground surface gives better results than the undisturbed soil temperature at the level of the pipe.

Kusada (1973) examines a system of two pipes at different temperatures buried in a trench. A mathematical model is presented which uses the method of superposition and the mirror image technique to combine solutions for single pipes into a matrix of equations for temperatures and heat fluxes. The model is substantiated with results from an experimental system in Washington, D.C.

Huovilainen (1985) analyses two dual pipe systems, one with pipes buried separately in their own jackets and one double-pipe prefabricated pipe system. Equations are presented from which factors affecting heat losses are determined; these show that the factors tending to increase the heat loss the most in both systems are the temperature levels, the thermal conductivity of the insulation, and the anchor point elements (if placed too closely together). The factors which have only marginal effects are the installation depth (for depths of burial greater than 100 mm), the heat transfer coefficient from earth to air, the

eccentricity of the carrier pipe with respect to the jacket, and the thermal conductivity of the earth. The thermal conductivity of the earth has little effect on the heat loss only because the pipes are insulated and the insulation is the main thermal resistance to heat transfer.

Hadvig (1982) gives some guidelines for avoiding potential problems which may arise in the areas of design, utilization of thermal energy and heat loss in district heating networks. The theory for heat loss from single buried pipes is presented as well as for two pipes, taking into account the transfer of heat between the two pipes. Explicit examples are presented, showing clearly the steady-state method used. The thermal loss is given as a function of the difference between the temperatures of the earth and water in the pipe for the comparison of theoretical and measured heat loss. The results show good agreement between measured and theoretical losses.

Battara, Mariani and Viggiani (1987) present a numerical solution (finite difference) for a buried pipe with a sinusoidal ground surface temperature variation, leading to a correction factor which can be used to make a steady state solution (using the minimum seasonal temperature at burial depth) more accurate. This correction is of the order 10-20%, which suggests transient effects are not too important.

Coulter (1976) applies the classical buried pipe solution to a buried liquid natural gas pipeline. His analysis includes the heat transfer resistance from the flowing fluid to the pipe wall, which is not usually considered important.

Shapiro and Moran (1978) examine heat transfer together with the soil moisture diffusion caused by heating of the soil, with a particular view to the feasibility of heating soil for agriculture. The model uses finite differences to solve the governing equations for heat and mass transfer, using volume averaging to obtain general equations for porous media. With relationships describing the transport processes in the porous medium it determines accurate values for the associated transport coefficients for soil. A simulation of a soil warming system is presented and the equations are applied to it to predict steady-state temperature and moisture content profiles in the soil. This is unlikely to be very relevant to district heating unless moisture migration produces large changes in soil properties.

A number of Russian authors have produced analytical solutions of some complexity for heat transfer from buried pipes. Dyachuk, Kornilov and Furman (1981) present an analytical solution for steady state heat loss from a buried pipe with an insulating blanket on top of the soil. Krivoshein and Agapkin (1977) present analytical expressions for the unsteady temperature distribution of the ground and the heat losses of an underground pipeline for an arbitrary variation of the temperature of the medium being transferred and certain boundary conditions imposed at the pipe wall and ground surface. To find unsteady heat losses the conduction equation was solved as the sum of the natural temperature distribution of the ground and the thermal perturbation caused by the pipeline. The accuracy of the solution was checked using numerical modelling. However, it should be noted that some of the other papers reviewed questioned the need for a transient solution in

view of the fact that steady state is achieved quickly. Other transient analytical solutions from the Russian literature, all quite difficult to apply to practical calculations, include Novakovskii (1985), Agapkin et al. (1983), and Krivoshein and Novakovskii (1975).

MacDavid (1991) uses infrared imaging (thermography) to calculate the temperature of buried steam lines used for district heating. Formulas for calculating the heat loss and temperature of a buried steam line are presented. Under steady-state conditions a heat loss from the ground surface is found which is equivalent to the heat loss from the pipe to the surface. The temperature of the line is estimated knowing the ground temperature and total heat coming from the steam line, then applying the fundamental heat transfer equation.

2.2 Heat Transfer Between Pipelines and Enclosures

One possible configuration for a district heating pipeline is that of one or more pipes enclosed in a buried concrete duct or circular conduit, and several studies of heat transfer in such enclosures were reviewed. Babus'Haq et al. (1984a) examine natural convection heat transfer between two pipes in an air space in a concrete trench. The effect of configuration on the overall heat loss in such a system is small, according to Huovilainen (1985). Babus'Haq et al., however, found that the optimum pipe location within the same enclosure is hot (supply) directly above cool (return) and this optimum leads to a 17% decrease in the overall heat loss for a 20°C temperature difference between the wall of the trench and the outer surface of the supply pipe. These results are in comparison to the typical side-by-side configuration or cool on top of hot. Babus'Haq et al. (1984b) found similarly for district

cooling pipes in a circular conduit that the optimal configuration places the cool pipe above the return, reducing the heat gain by 35% over the side-by-side configuration. Babus'Haq et al.(1985) examine experimental data and theoretical predictions for steady-state heat transfer to/from horizontal single or double pipelines enclosed in a horizontal circular or rectangular enclosure. Again, the focus of the paper is a topic which in other papers was dismissed as unimportant as a contributor to the overall heat loss from piping systems: identifying the optimal configuration of the pipes for maximal thermal resistance of the air-filled cavity. A correlation is presented for the combined convective/conductive resistances of the air for Grashof numbers in the range of 3E3 to 10E8 and diameter ratios, D/d (enclosure diameter/pipe diameter), of 1.3 to 7.5. Similar work is reported by Naylor, Badr and Tarasuk (1989) for natural convection between eccentric tubes, with experiments performed using a Mach-Zehnder interferometer and solutions of the governing equations using a finite element method.

2.3 Soil Properties

A report by the U.S. Army Corps of Engineers (Kersten, 1949), prepared at the Laboratory for the Determination of the Thermal Properties of Soils, presents extensive research done to determine the thermal conductivities of soils of varying properties. Although this reference is quite old, it is considered to be the most comprehensive and accurate research done to date (Phetteplace, 1994). Nineteen different soils were tested and the effects of moisture content and density for each were recorded. (Density in this context refers to void fraction or degree of compaction.) Equations were presented for

calculating the conductivity of different classes of soils, based on the experimental findings. Exact properties (composition) of the soil, however, were found to be very important, as the values could vary by more than 100% for two different soils at the same moisture content and density. Within any one class of soil type, the following trends were found: on average, at a constant moisture content, a 16 kg/m^3 increase in density resulted in a 3% increase in conductivity. At constant density, an increase in moisture content resulted in an increase in conductivity: for sandy soils, the thermal conductivity was found to vary between 0.5768 W/m°C and 3.4608 W/m°C for a corresponding range of moisture content of 0 to 30 % (% of dry soil weight). These measurements, however, for both density and moisture effects, were reported with an accuracy of only ±25%. As well as moisture and density, frost heave contributes a further source of variability to the measurement of the thermal conductivity. The main conclusion drawn from this paper is that the extent of variation in thermal conductivity in the natural environment is such that one could not give accurate figures for it.

2.4 Miscellaneous Works

The remaining papers reviewed concentrate on the flow in the pipe; they do not necessarily pertain to district energy lines. Bau and Sadhal (1982) derive expressions for the temperature distribution and velocity profile shape factor for turbulent and laminar flow in a buried pipe. Thermal resistance between the fluid and medium surfaces as a function of the burial depth and properties of the fluid and solid medium are also found. The turbulent case

assumed a uniform heat transfer coefficient at the pipe surface and used a convective boundary condition. The energy equations inside and outside the pipe were solved simultaneously for laminar flow, assuming linear temperature variation along the pipe axis. The pipe surface temperature was found to be uniform for depths of burial exceeding 2 radii. In turbulent flow (ie for district energy systems) the heat transfer coefficient is high enough compared to the thermal resistance of the soil that the temperature of the pipe wall is essentially identical to that of the flowing liquid, so that an analysis of the flow in the pipe is not required.

Hooker and Brigham (1978) discuss the differences between frictional heating and the Joule-Thompson effect of expansion cooling in a liquid. A model for steady-state heat transfer to/from a buried pipeline is developed under the assumptions that the soil is homogeneous with constant thermal conductivity, the surface temperature is constant, the film coefficient of the fluid at the pipe wall is negligible and the thermal conductivity of the pipe is infinite. The effects described are only important for exceedingly long pipelines such as oil pipelines (the example given in the paper is 60 miles long).

2.5 Conclusions from the Literature

The papers reviewed lead to several useful conclusions about the heat transfer phenomena involved in buried pipelines:

1. For insulated pipes the pipe insulation is the main resistance to heat transfer; the soil resistance is generally less. However, for uninsulated pipes the soil is the controlling

resistance. For either case. the resistance of the pipe wall and the resistance for convection from the liquid to the wall are small.

2. The depth of burial has little effect on heat transfer except for very shallow burial. Huovilainen (1985) finds no effect of burial depth beyond 100 mm depth. for an insulated pipe.

3. The pipe surface temperature is nearly uniform in the circumferential direction as long as the burial depth is sufficient (Bau and Sadhal (1982) give depth/radius >2).

4. A number of writers (Phetteplace et al. 1991b, Bøhm 1992, Kusuda 1973) have used heat flux meters to measure heat losses from the pipe. These are sensitive to small errors in temperature measurement and are difficult to calibrate and match to the soil properties. They do not appear to be a very reliable means of measurement. It appears that they were used by some authors only because circumstances did not allow performance of an overall energy balance on the pipeline. Schwerdtfeger (1970) presents a useful summary of heat flux meters, including design parameters, calibration, and use.

5. The response of the soil to transients in surface temperatures is sufficiently fast that quasi-steady state heat conduction models are quite adequate for predicting heat losses. Hadvig (1982) shows that the temperature underground at the pipe burial depth is more or less in phase with the surface temp and does not differ from it by more than 5°C.

6. Soil properties can vary widely: Phetteplace and Meyer (1990) give a range of ±25%. Kersten (1994) reported a wide variation in measured thermal conductivity dependant on moisture content and density of the particular soil. These measurements were reported with an accuracy of ±25%.

From the work reviewed here, it seems that there has not been any research done on the specific applications of this research project: heat transfer from buried distribution systems from the point of view of system integration (utilizing the lost heat to prevent other utility lines, run beside the district heating lines, from freezing) and different insulation configurations such as the use of a flat insulation blanket above the pipes as opposed to individual pipes being insulated.

3.0 NUMERICAL MODEL DESCRIPTION

A numerical model based on the finite volume method was developed to facilitate the design of integrated services systems for any configuration, under a wide variety of ground conditions. To this end, the model was designed to simulate the geometry of the experimental set-up for comparison purposes, but also to simulate larger, more complex systems with any number of pipes, with or without insulation, in any configuration, with any number of different conductive media in the soil surrounding the pipes. The model solves the 2-dimensional steady-state heat conduction equation in discretized form for the region surrounding the buried pipes and insulation. The model does not include transients, as the literature indicates that the temperature field around a buried pipeline can be considered quasi-steady within the time scale of seasonal temperature variation (Battara et al., 1987).

The main calculation grid is a rectangular system beginning at the ground surface and extending downward. Since the flow of heat in the vicinity of the pipes is in the radial direction and thermal gradients are steepest in these regions, local embedded grids in cylindrical coordinates are inserted into the main rectangular grid system to more realistically reflect the flow of heat in these regions. There are two types of these embedded sub-grids: "full" sub-grids and "reduced" sub-grids. The choice of sub-grid type depends on the configuration of the particular system being simulated. Figure 1 shows an example of a grid simulating the experimental trench with two pipes buried directly below an insulation board.

3.1 Main Rectangular Grid

The main rectangular grid has the ground surface as its top boundary and extends downward into the earth with the distance to the remaining three boundaries in the earth defined by the user for each simulation. The top surface is assumed isothermal and a zero heat flux condition is imposed across the remaining three boundaries. If the distance to the grid boundaries from the heat source (pipe) is sufficiently large, this assumption is reasonable. This is further illustrated in Section 3.4.2. It is assumed that the heat flux entering the bottom boundary of the grid from the earth's center is negligible.

Node spacing is defined for each simulation and can be uniform or non-uniform in the x and y directions. Some typical main grid nodes are illustrated in the bottom left corner of the grid in Figure 1, while Figure 2 gives details of a single cell (central node). Non-uniform grid spacing allows for refinement in regions of steep thermal gradients and coarser spacing as the gradients flatten out. This reduces computing time while maintaining accuracy. The nodes are placed mid-way between control volume faces, which implies that when defining the grid for a particular simulation, the control volumes should be drawn first and the nodes second. Figure 2 illustrates a typical rectangular cell "P" drawn in this fashion. With the node at the geometric center of the control volume, the temperature of the node "P", T_P, can be regarded as a good representative value for the control volume in the calculation of quantities such as the thermal conductivity. Following this practice, the assumption that the heat flux between two nodes prevails

across the entire control volume interface between those nodes ($\Delta x*1$ or $\Delta y*1$) is more reasonable (Patankar, 1980). Furthermore, this practice facilitates handling of composite materials: control volume boundaries can be defined easily at the point of material change (thermal conductivity change), such as at the border of an insulation board in the soil.

3.1.1 Discretization Equations: Rectangular Grid

The finite volume numerical method involves solving discretization equations, in this case for temperature, which are derived for the discretized calculation domain from the governing differential equation.

The governing differential equation for the main grid is the two dimensional steady-state conduction equation in rectangular coordinates:

$$\frac{\partial}{\partial x}\left(\frac{k\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\frac{k\partial T}{\partial y}\right)=0 \qquad (1)$$

This equation is integrated over each control volume in the grid to obtain the discretization equations in temperature for each node. Again, the hatched area in Figure 2 illustrates a typical control volume around a node "P" of the rectangular grid. The "east", "west", "north" and "south" control volume interfaces are indicated as well as the dimensions of the control volume.

The following discretization equation results from integration of the 2-dimensional conduction equation over the control volume from the west to the east interface and from

the south to the north interface (Patankar, 1980). Capital subscripts refer to nodes and lower case subscripts refer to cell (control volume) boundaries. For example, k_e refers to the thermal conductivity at the control volume interface between nodes P and E (which is influenced by the conductivities of both nodes P and E), and T_E refers to the temperature at node E (which is assumed to prevail everywhere in the control volume surrounding the node E).

$$a_P T_P = a_E T_E + a_W T_W + a_N T_N + a_S T_S \qquad (2)$$

where $a_E = k_e \Delta y_P / (\delta x)_e$

$a_W = k_w \Delta y_P / (\delta x)_w$

$a_N = k_n \Delta x_P / (\delta y)_n$

$a_S = k_s \Delta x_P / (\delta y)_s$

$a_P = a_E + a_W + a_N + a_S$

The temperature at each node is then dependent on the neighboring node temperatures. It is assumed that the flow of heat is uniform across each control volume face. Since the grid is discretized into cells within which properties are constant, cell sizes for a simulation must be chosen such that the edges of insulation boards coincide with cell boundaries. The discretization equations derived for the boundary nodes of the rectangular main grid are different than the above interior node equation due to differing geometry and the imposed zero heat flux condition. Nodes lie along the four boundaries of the grid, thereby creating "half" control volumes on the sides and "quarter" control volumes on the corners (see Figure 1). The discretization equations for these boundary nodes are outlined in Appendix A.

3.1.2 Thermal conductivity

The model assumes a uniform thermal conductivity for the soil. In fact, conductivity varies substantially with compaction of the soil and with moisture content. As stated in the literature review, the variation in measured soil properties is reported as ±25% (Phetteplace and Meyer, 1990), so attempting to measure soil conductivity experimentally is not worthwhile. It is, therefore, reasonable to treat it as a fitted parameter which is constant in the model. Thermal conductivity is fitted using the results of experimental heat balances; this is further discussed in section 5.3.1.

Dealing with the change in conductivity from the soil to another material within the soil, such as the insulating board, however, is a different issue. The thermal conductivity within the confines of any given control volume is assumed uniform. If the neighboring control volume has a differing thermal conductivity, an interface conductivity is used in the calculation of the heat flux across that control volume boundary from one node to the next. The interface conductivity is influenced by the conductivity of both nodes and is dependent on the geometry of the two nodes. Patankar (1980) derives an interface conductivity, k_e, between points P and E as:

$$k_e = \{(1-f_e)/k_p + f_e/k_e\}^{-1} \qquad (3)$$

where f_e is an interpolation factor defined in terms of distance:

$$f_e = (\delta x)_{e+} / (\delta x)_e$$

The subscript "e" refers to the entire distance between nodes P and E; the subscript "e^+" refers to the distance from the interface (cell boundary) to the node E.

This equation applies to the steady-state one-dimensional flow of heat between two nodes and is derived to obtain a good representative heat flux, q_e, at the interface boundary.

3.2 Sub-grids

Two types of sub-grids in cylindrical coordinates were designed to provide a realistic representation of the radial flow of heat out from the pipes. The sub-grids are embedded into the main rectangular grid and surround each pipe. The "reduced" sub-grid was designed specifically to deal with a pipe placed immediately below an insulation board. The "full" sub-grid accommodates all other buried pipes.

3.2.1 Full Sub-Grid

Figure 3 illustrates the "full" sub-grid in cylindrical coordinates embedded in the rectangular main grid. The full sub-grid occupies four equally spaced square main grid nodes, creating a square with the pipe placed at the center where the four control volume interfaces meet. Eight radial lines extend to the square sub-grid boundaries at 45° angles from one another, beginning at the horizontal. These lines make up the "east" and "west" control volume faces for the sub-grid nodes, which lie on radial lines exactly between them, also at 45° from one another, starting at 22.5° from the horizontal. The circular divisions radiating outward from the pipe make up the "north" and "south" control

volume faces. The first node along all eight radial lines lies on the inside wall of the pipe: the first cell is therefore a "half" control volume. The north control volume interface of the first node is the outside wall of the pipe. The temperature of the first node on the inner pipe wall assumes the temperature of the water in the pipe and the thermal conductivity of the control volume is that of the pipe material . Each subsequent node along the radial line lies in the geometric center of its control volume. The number of nodes along any of the radial lines (and thus the number of circular divisions) is defined by the user for each simulation. Spacing between these nodes is also defined by the user and can be uniform or non-uniform. The pipe can be insulated or not insulated. If there is insulation on the pipe, the edge of the insulation coincides with one of the circular division lines ("north" or "south" control volume interface), so that the change in conductivity lies at the control volume interface. Interface thermal conductivity between radial nodes along the same radial line is calculated as described in Section 3.1.2.

3.2.2 Reduced Sub-Grids

Figure 4 illustrates the "reduced" sub-grid integrated into the main grid with the insulation board above it. The reduced sub-grid was designed specifically to handle the experimental configuration of the pipes with insulation board lying directly above. As in the full sub-grid configuration, the reduced sub-grid occupies four main grid nodes; however only the bottom two main grid nodes are square. The top two main grid nodes have a reduced height (Δy) to reflect the smaller gap between the pipe and the insulation board. This small space occupying the top two main grid nodes is represented by a

rectangular coordinate grid. Flow of heat in this region will not be radial; the isotherms will be almost perpendicular to the insulation board due to the much lower thermal conductivity of the insulation board (0.036 W/mC) as compared to the soil (approximately 0.8 W/mC), and this grid reflects this. The radial flow of heat out from the pipe below is represented by the cylindrical coordinate system which occupies the two lower, square main grid nodes. Heat flows radially out along the four radial lines indicated in Figure 4. This cylindrical section is set up exactly as in the full sub-grids. Again, the nodes must lie at the geometric center of the control volumes and the first node along any radial line lies on the inner pipe wall. Each node along a radial line has a corresponding node in the rectangular section of the sub-grid.

There can be no insulation around the pipe in this sub-grid. If the pipe were insulated, a full sub-grid would suffice, since most of the temperature drop occurs within the insulation and therefore the sub-grid borders could be very close to the outside casing of the insulation.

3.2.3 Discretization Equations for Sub-Grids

The governing differential equation for the cylindrical sub-grid geometry is still the steady-state conduction equation, but in cylindrical coordinates:

$$\frac{1}{r}\frac{\partial}{\partial\theta}\left(\frac{k}{r}\frac{\partial T}{\partial\theta}\right)+\frac{1}{r}\frac{\partial}{\partial r}\left(rk\frac{\partial T}{\partial r}\right)=0 \qquad (4)$$

As in the rectangular grid system, the governing differential equation is integrated over each control volume to obtain discretization equations from which nodal temperatures are calculated. Figure 5 depicts a typical interior node cylindrical coordinate control volume with "east", "west", "north" and "south" interfaces and dimensions shown. The discretization equation for node P is the same as the equation for rectangular coordinates, but with coefficients reflecting the cylindrical geometry (Patankar, 1980):

$$a_P T_P = a_E T_E + a_W T_W + a_N T_N + a_S T_S \qquad (5)$$

where $a_E = k_e \Delta r / (\delta\theta)$

$a_W = k_w \Delta r / (\delta\theta)$

$a_N = k_n r_n \Delta\theta / (\delta r)_n$

$a_S = k_s r_s \Delta\theta / (\delta r)_s$

$a_P = a_E + a_W + a_N + a_S$

$\delta\theta$ and $\Delta\theta$ are always $\pi/4$ due to the equal set spacing of the cylindrical sub-grid nodes.

The discretization equations for the rectangular nodes of the reduced sub-grids, for the cylindrical nodes bordering on rectangular nodes in the reduced sub-grids and for the rectangular nodes of the reduced sub-grid bordering on main grid nodes and the pipe, are derived in Appendix B. The equation for the cylindrical sub-grid nodes which border onto main grid nodes is also derived in Appendix B by integration of the steady-state conduction equation across the control volume. These cells technically violate the requirement for an orthogonal grid in Patankar's control volume method (Patankar, 1980): the gradient from the sub-grid boundary node to the main grid node is not exactly

orthogonal. The later validation studies of this model (see Section 3.4), however, provide verification that this violation has a negligible effect on the accuracy of the solution.

3.3 Overall Solution Algorithm

The overall solution algorithm comprises an algorithm for the main grid alone and separate algorithms for each embedded sub-grid. All employ the line by line method (Patankar. 1980) with equations manipulated to suit each geometry. Although each sub-grid and the main grid are set up to be solved separately, all are converged simultaneously and are linked to one another through the sub-grid boundary node temperatures and the adjacent main grid node temperatures. That is, sub-grid boundary node temperatures influence their adjacent neighbor main grid node temperatures and vice versa. The scheme involves consecutive sweeps of the main grid and each sub-grid, always using the most recently updated temperatures as the sweeps progress.

Simultaneous convergence and use of the most recently updated temperatures results in a reduced number of iterations to meet the convergence criterion and a faster, more efficient program. Convergence for the entire solution is achieved when the largest difference between successive calculated temperatures of the same node (from one iteration to the next) is less than a pre-set value, an absolute error of 0.0001°C being a reasonable standard.

3.3.1 Main Grid Algorithm Scheme

The line by line algorithm for the main grid entails solving a 1-dimensional tridiagonal matrix algorithm (Patankar, 1980) for each column of the 2-dimensional grid. Each column is solved, sweeping from left to right across the grid. Temperatures of all nodes in the grid are continuously updated: that is, each column uses the last updated temperatures from the neighboring columns in its solution. When the main grid algorithm meets the main grid cells in which the sub-grids are located, it simply ignores the contribution of those main grid nodes. Only the sub-grid boundary node temperatures influence the main grid nodes on which they border.

3.3.2 Sub-Grid Algorithm Scheme

Each sub-grid also employs the line by line solution algorithm, with equations manipulated to suit its geometry. TDMA equations are solved along each radial line in succession, with the first node of each line being the given water temperature in the pipe. The final node along a radial line is influenced by the temperature of its neighboring main grid node, updated by the last sweep of the solution algorithm of the main grid. This creates the link between the sub-grid and the main grid. After one sweep through each sub-grid consecutively, another sweep through the main grid is completed using the latest updated sub-grid boundary node temperatures.

3.4 Testing of the Numerical Model

A series of tests were performed on the numerical model to verify its accuracy. These included tests to confirm energy conservation within the grid and tests against the classical analytical solution for a single buried pipe. Tests were also conducted to determine the effect of nodal spacing and the distance of the calculation domain boundary from the pipes on the accuracy of the solution (as compared to the analytical solution).

3.4.1 Testing for the Conservation of Energy

Numerous configurations were simulated with the numerical model and the heat flux was calculated at various points to ensure uniformity in its value throughout the grid.

First, a single uninsulated pipe was tested. The total heat flow out from the surface of the pipe was calculated and compared with heat flow out from a square of main grid nodes surrounding the pipe. It was also compared with the heat flow leaving the grid from the top boundary. The heat flow leaving the top boundary represents the total heat flow in the grid due to the imposed zero heat flux at the other three boundaries. The test was run with the full sub-grid geometry and then with the half sub-grid geometry.

A second test compared the sum of the heat flow out from two buried, uninsulated pipes with the total heat flow leaving the top boundary of the grid. This test was performed using full sub-grids around both pipes, reduced sub-grids around both pipes and, finally,

with one full sub-grid and one reduced sub-grid. These two test configurations were run again with insulation on the pipes.

All tests were repeated, alternating temperature extremes for the pipe(s) and ground surface. In each of the above tests, the heat flows were equal up to the fifth significant figure, a very acceptable level of confidence.

Further tests included combinations of geometries with insulation boards in the soil. The first geometry consisted of a single pipe in a reduced sub-grid geometry with one insulation board above it. Three more tests added an insulation board per test on the other three sides of the pipe. These tests were all re-run with a full sub-grid geometry. A series of tests were then done with two pipes, with and without insulation, full and reduced sub-grid (and combination of the two), and an increasing number of insulation boards placed around the two pipes together. The total heat flow was again calculated at various different points within the grid to confirm energy conservation. Again, results showed agreement to within five significant figures.

3.4.2 Comparison of Heat Flow with Analytical Solution and Spacing Effects

The total heat flow from a model grid simulating a single, uninsulated buried pipe, was compared with the classical analytical solution for this configuration, which has long been accepted as accurate (Holman, 1990). The model was run many times with this configuration, varying the nodal spacing and boundary widths each time, to determine the

effects of changing these variables on the accuracy of the solution. With the analytical solution as a reference, some general guidelines as to reasonable, practically sized boundary widths and nodal spacing were determined.

The heat flow from a single, uninsulated, 1/2″ copper pipe, buried at 80 cm depth, with a temperature difference (between the water in the pipe and the ground surface) of 85°C in soil with a thermal conductivity of 0.6 W/m°C, was calculated analytically. The resultant heat flux value of 63.46 W/m was used as the benchmark for comparing model runs with the same configuration, temperatures and conductivities. A series of trial-and-error simulations with different boundary widths and spacing were run until a grid was found which yielded a heat flow less than 2% under the analytical solution. Table 1 gives the details (dimensions, spacing) of this grid. Figure 6 illustrates the grid system corresponding to Table 1. This grid can be used as a guideline for other grid designs. The grid dimensions are 74 nodes (x-axis) x 50 nodes (y-axis) and its side boundaries are approximately 18 m apart. The nodes directly around the pipe are spaced 4 cm apart because this allows ample room for representation of the radial heat flow out from the 1/2" pipe in the sub-grid. The pipe was placed mid-way between "east" and "west" grid boundaries and the space between each node from either side of the pipe out to each grid boundary increased successively by 10% per node. The refinement of the sub-grid node spacing also contributed to the accuracy of the solution. Spacing in the order of 19 nodes per radial line for a 4 cm x 4 cm sub-grid around the 1/2″ pipe was found to be optimal:

that is, no significant increase in accuracy was found for sub-grids with greater than 19 nodes.

Slightly better results were obtained for a grid with boundaries 18 m apart, but no further advantage was found past 25 m; a boundary width to depth of burial ratio of 12.5 is sufficient for this configuration. It was found that reducing the spacing between nodes from a 10% to a 5% successive increase in spacing per node improved results by only 0.2%, not enough to warrant the significant increase in the number of nodes. When using the reduced sub-grid, it was found that for the top two nodes a height 1/2 that of the two bottom nodes yielded the most accurate results.

The results found for the single buried pipe configuration will change slightly for different applications. For example, deeper burial would require an increase in the boundary widths, as the region of significant heat flow would extend farther in the horizontal direction. Similarly, configurations with more than one pipe would require larger boundary widths to accommodate the increase in space needed between the pipes. The specifications for the grid given in Table 1, however, can be used as a general guide for designing grids; they will give reasonably accurate results for most applications.

3.4.3 Comparison of Temperature Fields: Analytical with Predicted

The temperature field surrounding a single uninsulated pipe buried 74 cm deep was calculated using an analytical solution (Eckert et al., 1972). The numerical model was run with the same configuration and the resulting predicted temperature field in the soil was compared with the temperature field calculated with the analytical solution. The grid used for the model prediction followed the guideline for dimensions outlined in Section 3.4.2. Figure 7 illustrates a comparison of analytical and predicted temperatures along a vertical line from the ground surface down to the pipe wall. Both main grid and sub-grid temperatures are included in the comparison of Figure 7. The excellent agreement between temperatures validates the accuracy of the model. Since one of the non-orthogonal boundary sub-grid nodes is represented in Figure 7, the temperature agreement in Figure 7 also verifies the negligible effect these non-orthogonal boundary sub-grid nodes have on the solution.

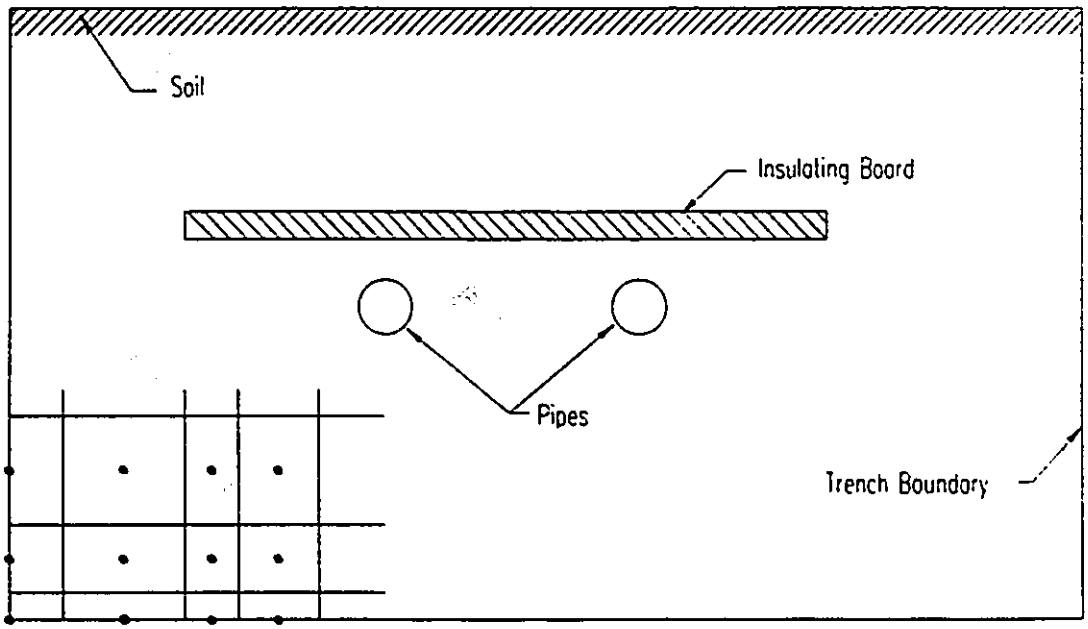

Figure 1: Sample Grid Simulating Experimental Trench

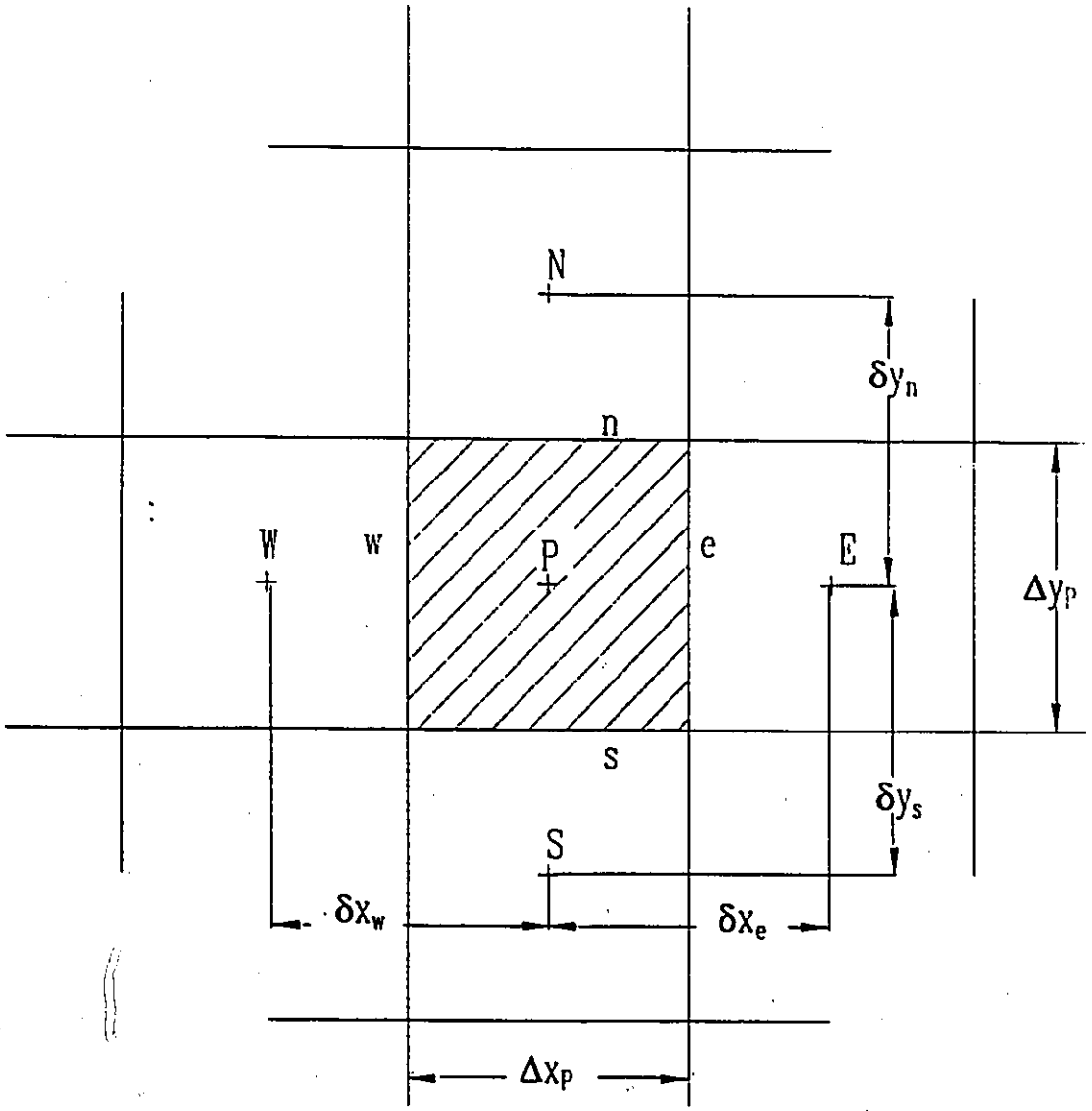

Figure 2: Typical Control Volume: Interior Rectangular Grid Node
(Uppercase letters refer to nodes; lowercase letters refer to cell boundaries)

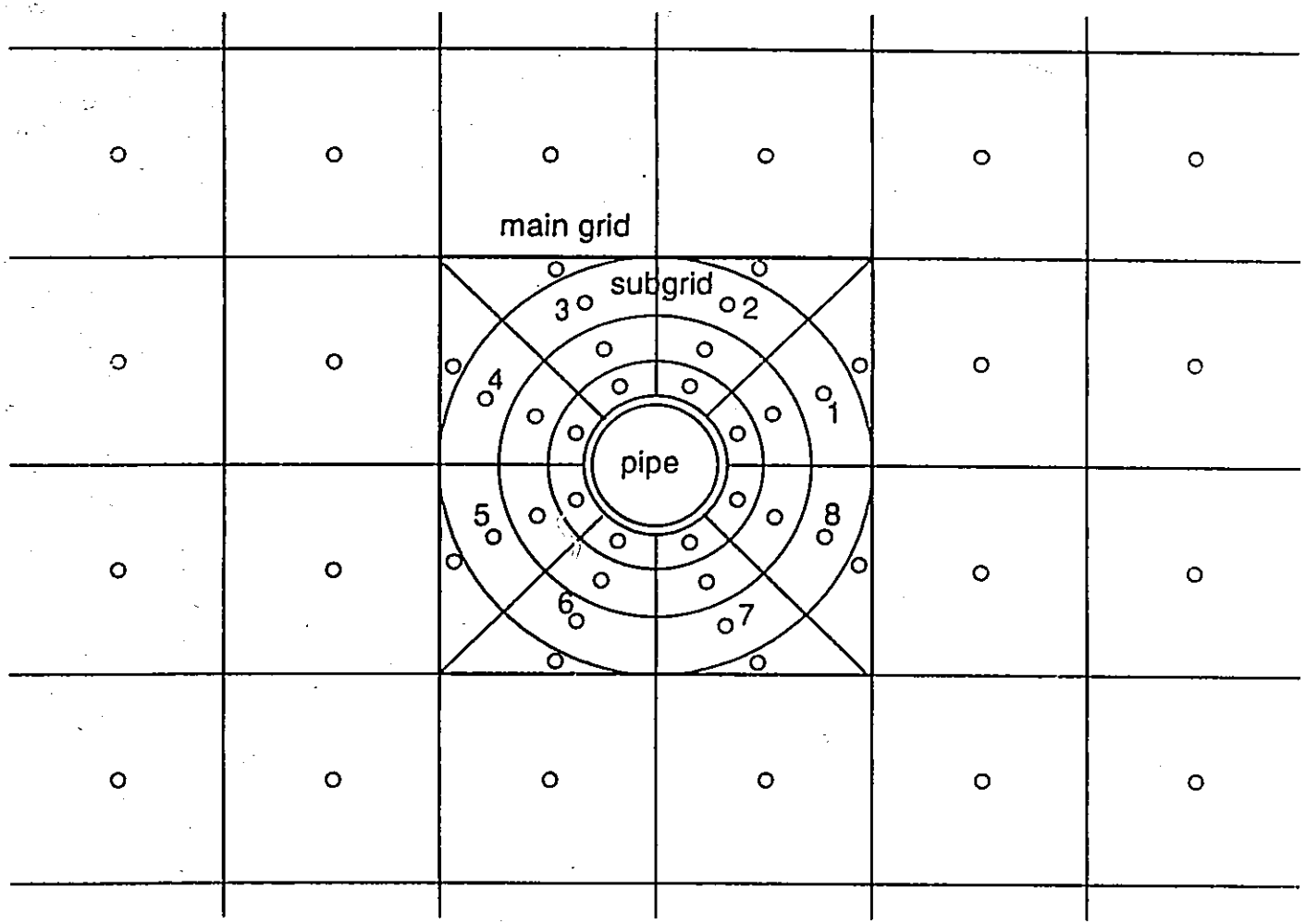

Figure 3: Full Sub-Grid Embedded into Main Grid

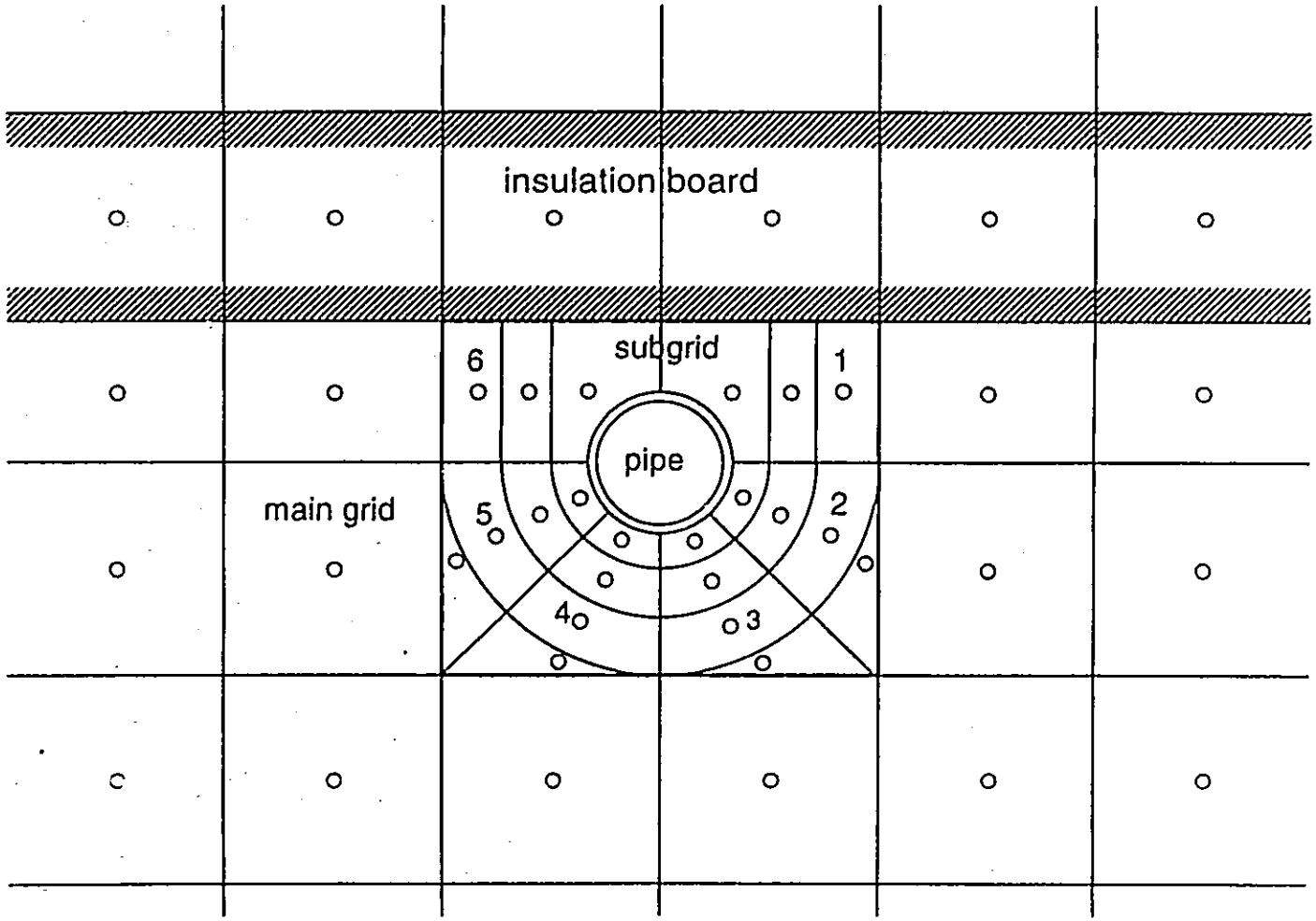

Figure 4 Reduced Sub-Grid Embedded into Main Grid

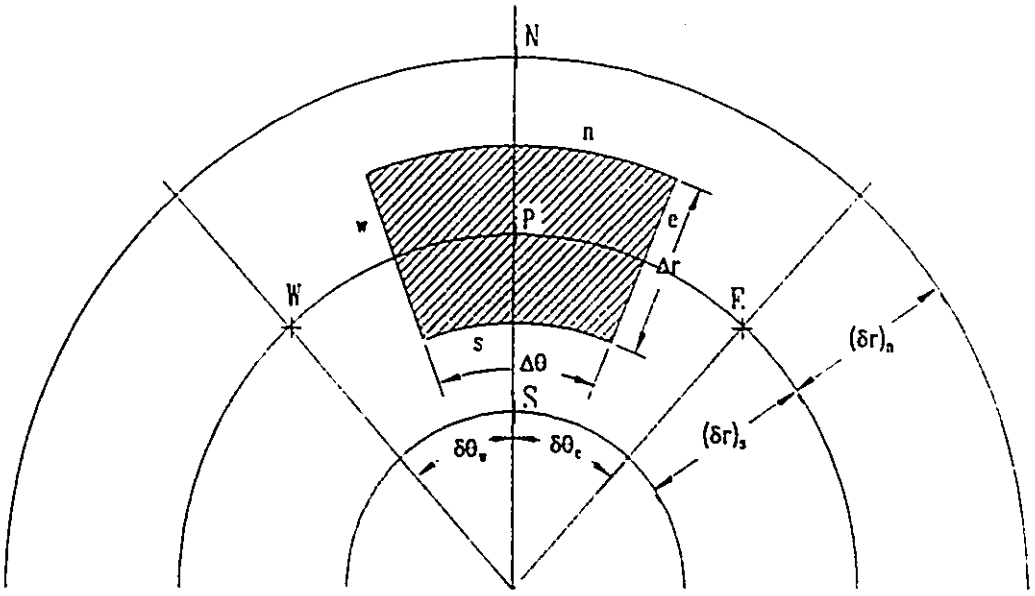

Figure 5: Typical Control Volume: Interior Cylindrical Grid Node

Table 1: Guideline Grid Specifications (Single Buried 1/2" Pipe)

MAIN GRID	
NUMBER OF NODES	
x-axis	75
y-axis	50
NODE SPACING	
x-axis (beginning from node 38)*	(4, 4, 4, 4, 3, 2.5, 3.5, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18, 20, 22, 24, 26, 29, 32, 35, 38, 41, 45, 50,55,60,66,73,80,82)
y-axis (beginning from node 1)	(4, 5, 5, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2.5, 1.5, 3, 3, 3, 4, 4, 4, 4, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 8, 10, 10, 10, 10, 10, 10, 10, 10, 10)
DEPTH OF PIPE BURIAL	83 cm
SUB-GRID	
SUB-GRID COORDINATES **(TOP LEFT MAIN GRID NODE)**	(40, 22)
NUMBER OF NODES PER RADIAL LINE	19
DIMENSIONS OF EACH MAIN GRID NODE OF SUB-GRID	4 cm x 4 cm
SUB-GRID NODE SPACING	(0.2, 0.1, 0.1, 0.15, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2)

* node 38 is mid-way along the x-axis - spacing given symmetric about node 38 - i.e. it is the same from nodes 38 to 1

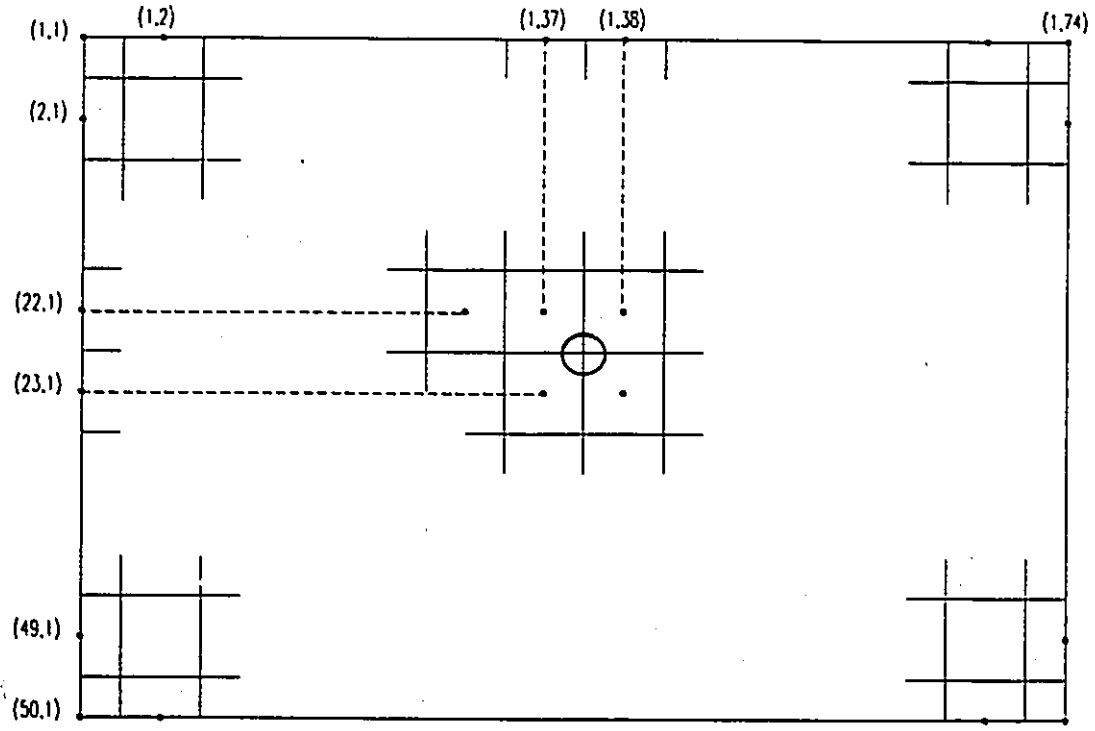

Figure 6: Model Grid Corresponding to Table 1: Single 1/2" Pipe

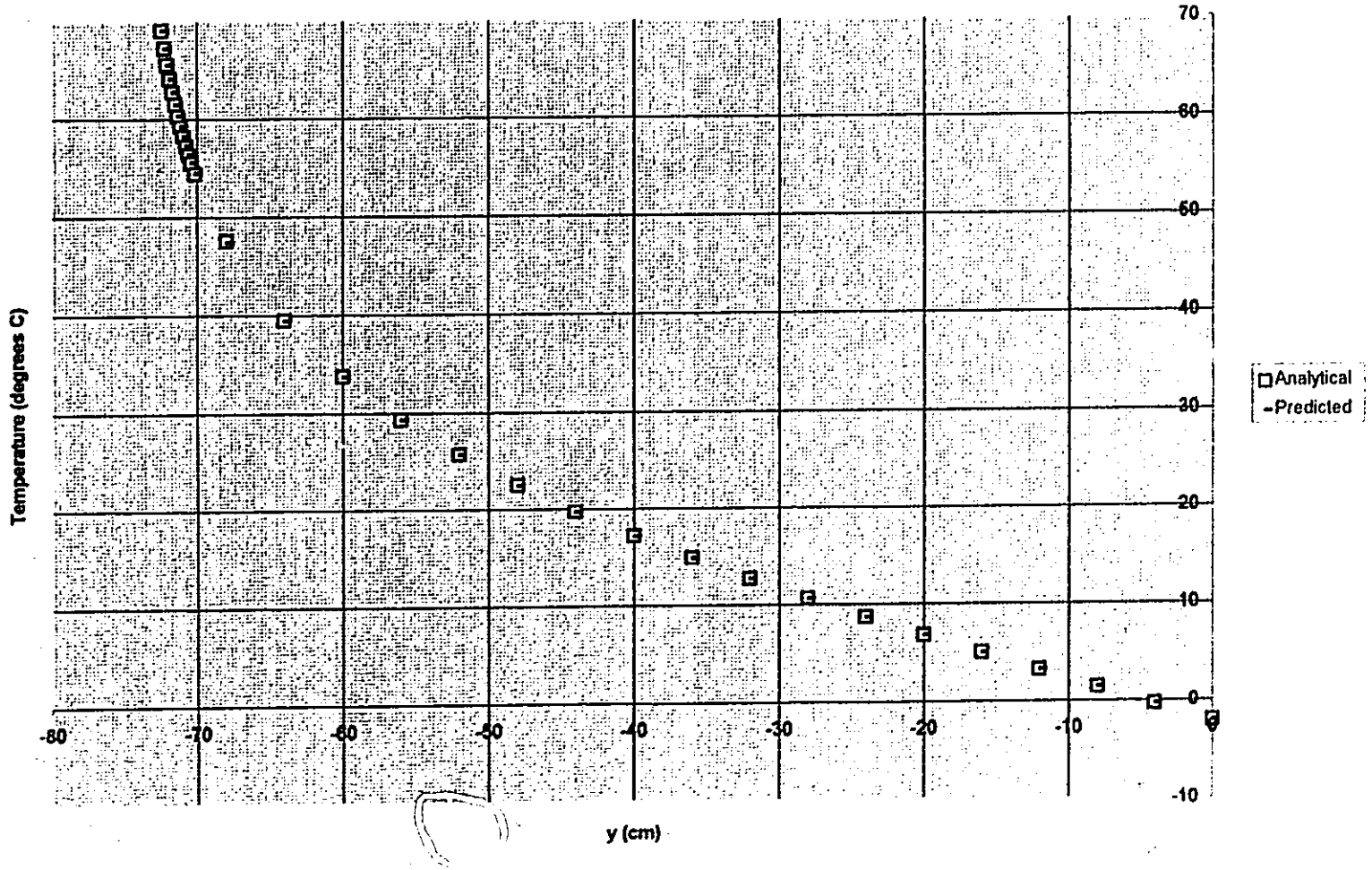

Figure 7: Comparison of Analytical and Predicted Temperatures from Pipe to Ground Surface (along a vertical line through MG and SG nodes)

4.0 EXPERIMENTAL

The purpose of the experiments is the verification of the numerical model. Results from the model, run at experimental conditions and with the experimental configuration, are compared with data collected from experimentation. Specifically, soil temperature data are taken below experimental hot water pipe loops at discrete locations, using thermistors. These temperatures are compared with discrete nodal temperatures calculated by the numerical model run at the same conditions. Water temperatures are required at the locations of the soil temperature measurements for completing an energy balance on the experimental loops. An experimental energy balance is required for comparison with the numerical output for fitting the soil thermal conductivity used in the numerical model.

4.1 Description of Experimental Setup

A total of five experimental systems were studied. Two of the five systems were designed specifically to provide data to verify the numerical model while the remaining three systems were intended to provide additional information for a more comprehensive analysis in future work. This report focuses on the verification of the numerical model; therefore, only those two systems which serve this purpose will be discussed in detail here. Each of the two systems consists of two separate hot water pipe loops running side by side. The two loops operate at different temperatures, simulating supply and return district heating lines. Pipes in both systems were left uninsulated to allow freedom in varying the surface temperature and thus flexibility in simulating different conditions.

Pipe surface temperatures were set to reflect the approximate jacket temperatures of typical insulated district heating lines.

4.1.1 Pipeline Systems

Figure 8 shows a cross sectional view of the trench containing the primary experimental system. Figure 9 shows the piping layout in detail for both experimental systems. The primary system consists of 2x100 m of 1/2" bare copper pipe buried 0.75 m deep in a 1 m wide trench. The trench was dug to just over 1 meter in depth and backfilled with masonry sand to a depth of 0.75 m, on top of which the pipes were then laid. Sand was used for backfill rather than the original earth because its properties are more uniform. A 5 cm thick x 60 cm wide extruded Styrofoam insulation board was placed directly above the two pipes. This insulation board serves to restrict the heat losses from the pipes to the ground surface, thereby allowing more heat transfer between the pipes and producing a warm thermal field around the pipes. The trench was again backfilled with masonry sand to ground surface level. Sand was compacted with a mechanical compactor before the pipes were laid and again after backfill to ground surface level. The pipe diameter, length and flowrate of this system were chosen to maximize the temperature drop around the loop so that an accurate heat balance could be performed. This is essential for comparison of the experimental results with those of the numerical model. The pipes are 20 cm apart, leaving space for hypothetical water and sewer lines between them. There is an insulated flexible pipe section at the 90 degree bends in the loop to handle

expansion and there is a 3/4" hose connection at the middle of the loop for drainage/blowout.

The second system consists of two loops of 3" bare copper pipe, placed side by side, 30 cm apart, each measuring a length of 20 m. The two loops are buried in the same trench configuration as the primary system, again with the Styrofoam insulation board placed directly above the two pipes. Backfill and compaction procedures were carried out as for the primary system. This system's pipe diameter is approximately that of the insulation jacket of actual district heating run-off lines (service connection lines). Heat loss from this loop is being studied to verify the model at larger pipe diameters and to confirm scaling.

The three remaining systems studied are of different configurations to provide additional information for future work. The first system consists of pre-insulated underground pipe loops equivalent to those of actual district heating systems. The second system consists of bare copper pipe loops enclosed in a Styrofoam insulation box and covered with sand on the earth's surface. The third system consists of a trench with sand backfill only. An insulation board was placed at a depth of 50 cm across half the length of this trench to study the disturbance of the earth's natural thermal field by the insulation board.

4.1.2 Controls and Equipment

Controls for both experimental systems are shown in the piping layout diagram of Figure 9. Although the four individual loops share two hot water tanks, the temperature in each loop could be maintained at a different set-point by automated proportional three-way valves. A proportional three-way valve on the supply line for each of the four separate loops regulates the mixing of return water from each loop with the supply water for that loop from the hot water tank, thus regulating the temperature to meet the set-point. The flowrate in the pipes is controlled by manual adjustment of a bypass loop on the supply line for each of the four loops.

Mass flowmeters monitor the flowrate for the 1/2" pipe loops. Velocity in the pipes was maintained at 0.4 - 0.5 m/s and the Reynolds number is approximately 9 000, ensuring turbulent flow. The flowrate was not measured for the 3" loops as no heat balance was to be calculated for the 3" system.

The two hot water tanks are Cascade 60 standard domestic hot water tanks. One of these tanks provides water for the "return" loops of both systems while the other tank provides water for the "supply" loops of both systems (note: "supply" and "return" here refer to the separate pipe loops which simulate the supply and return lines in an actual district heating system). The electrical heating capacity on the "supply" water tank is 5300 W and that on the "return" water tank is 3800 W. A higher capacity was required on the "supply" water tank to maintain a high temperature because the temperature drop around the 1/2"

supply loop and the rate of circulation around the loop were high. Higher electric capacity was obtained by forcing the top and bottom heaters on the supply water tank to operate simultaneously rather than one at a time as on the return water tank.

Each individual loop has its own pump, which is located on the supply line leaving the tank. The 1/2" system loops have Armstrong model SSC-30 pumps, and the 3" system loops have Armstrong model SSC-50 pumps. Both are two speed pumps.

4.2 System Operation

After the pipes were tested for leaks with high pressure air and the trenches were backfilled, the two tanks, and thus the closed loops branching from the tanks, were filled with water. When the pressure gauge on the top of the tank began to rise, indicating the systems were full, air venting procedures began. Air was released through automatic vents in the lines at the top of the tanks; however, due to the many elbows and the vertical height covered by the piping, most air had to be released by manual venting. This entailed trapping the air by periodically shutting valves in the main lines and allowing the air bubbles to rise through the bypass pipes, where they were vented through tubes inserted into Peterson plugs. (Peterson plugs are fittings mounted axially onto the pipe with a neoprene seal into which a sensor or tube can be inserted.) These plugs are located at measuring stations, where Resistance Temperature Detectors (RTD's) can be inserted to measure water temperatures, and also at the high points of the piping near the controls and pumps for the insertion of tubes for manual venting of air. Opening and shutting

main line valves in all areas where air was most easily diverted to venting places was continued until most of the air in the pipes was vented. As the air was being released, water was being added to the system to maintain pressure. System pressure was monitored during this manual venting process by the relief valve on the top of the tanks. If the tanks reached a pressure of 30 psi, the relief valve would open automatically, releasing water from the tank and lowering the pressure. Most air was released through manual venting and the small amounts remaining were removed over time by the automatic air vent on the line near the top of the tank, where the air eventually rose.

The temperatures in the loops were maintained by the proportional three-way valves. The temperature for the 1/2" system "supply" loop was set at 50°C and the 1/2" system "return" loop was set at 25 °C; the 3" system "supply" loop was set at 40°C and the 3" system "return" loop was set at 25°C. Unfortunately, many problems were encountered with the functioning of these three-way valves and, although constant for long periods of time, some of the set-points did change from their original settings. The most prominent change was in the return loop of the 1/2" system, which was actually maintained at 15°C during most of the testing period.

4.3 Instrumentation and Data Collection

4.3.1 Thermistor Grids

In each of the two systems, the two-dimensional thermal field is measured by a buried grid of thermistors placed transverse to the pipes. Figure 10 shows a cross-section of the trench for the first measuring station ("Station A") of the 1/2" system, illustrating the grid of thermistors extending down into the sand below the level of the pipes. There are two grids per system, one 5 m from the beginning of the loop ("Station A") and one 5 m from the end of the loop ("Station B"), as shown in Figure 8. The remaining three measuring stations have the same cross sectional configuration. The exact spacing between thermistors is illustrated in Appendix C for each measuring station grid. The thermistors are Fenwell Electronics part number 192-303 KET-A01. They have a listed accuracy of ±0.2°C and a resistance of 30kΩ at 25°C. Each thermistor is glued into a hole drilled through a Lexan board. The Lexan board thus positions all thermistors in a set configuration; it also holds the pipes in position near the top of the board. Figure 11 illustrates a typical thermistor protruding through the Lexan board near one of the pipes. Since the thermal conductivity of Lexan is 0.19 W/mC (ASHRAE, 1982) and that of the surrounding soil is about 0.8 W/mC, the presence of the Lexan board should not disturb the ground temperature measurements. Furthermore, the thermistor tips extends 5 cm out from the board. Glue-lined heat-shrink tubing is melted around the bulb of each thermistor to protect it from ground moisture. Gel-filled burial cable protects the thermistor wire leads.

Sand backfill surrounding the thermistor grids was hand packed and drenched with water to achieve maximum sand density. A high degree of compaction is desirable below the insulation board to minimize settling. Settling would cause progressively increasing void fractions extending from the board downward and thus a non-uniform thermal conductivity in the measurement region. It is desirable to have a uniform actual soil thermal conductivity, as it is assumed uniform in the numerical model. Unfortunately, in reality, soil thermal conductivity varies substantially with degree of compaction of the sand as well as with moisture content, both of which are difficult to measure. As stated in the literature review, the exact nature of the sand, in terms of its moisture content and density everywhere, would have to be known accurately in order to predict the value of the conductivity. Even if this were possible, the value could only be predicted to ±25% accuracy (Kersten, 1949). In the natural environment around the experimental pipe loops, it is impossible to predict what the moisture content will be under changing seasonal conditions. Further, if any settling occurs under the insulation board, the density will not only be unpredictable, but will be variable throughout the region surrounding the thermistors. Since the model assumes a uniform conductivity, every effort was made to ensure uniform soil conditions and packing density.

4.3.2 Data Acquisition System

Each thermistor is wired to a computer-based data acquisition (DAQ) system which monitors temperatures continuously. The computer uses the Steinhart equation to convert the thermistor resistance measurement into temperature (Macuga, W., 1994):

$$1/T = A + B(\ln R) + C(\ln R)^3 \qquad (6)$$

where $A = 9.163517787 \times 10^{-4}$

$B = 2.242583609 \times 10^{-4}$

$C = 1.148213378 \times 10^{-7}$

R = resistance

The manufacturer guarantees the thermistors to be precise to within ±0.2°C. A 50°C circulating water bath was used to test each thermistor against the data acquisition system temperature reading and their degree of accuracy was confirmed.

Experimental points were monitored continuously over the course of the fall, winter and spring of 1994/1995. During this period, data were recorded every four hours and were downloaded onto spreadsheet software. "Surfer", a three-dimensional graphics package, was used to create isotherm plots for comparison with numerical model results. These are discussed in detail in section 5.2.

4.3.3 Water Temperature Measurement with RTD's

The data acquisition system also continuously monitors the water temperatures in the pipes with RTD's which are inserted directly into the water flow. RTD's are located just downstream of the proportional three-way valves as the controlling sensors for the valves. RTD's are also located in each pipe just before the pipes enter the ground (supply to each loop), just after they exit from the ground (return from each loop), and in each pipe at

each ground temperature measuring station. Water temperatures at the measuring stations are required as input to the numerical model to calculate the local heat losses for comparison with measured thermistor temperatures. After installation and backfill, however, it was discovered that the sensors in these locations were not long enough to extend into the water flow and actually sat above the stream in the stem of the Peterson plug, with the result that the temperatures measured here were clearly in error. An alternative method then had to be used to obtain the water temperatures at the measuring stations. The temperature drops around each loop were measured at the point of pipe entry and exit from the ground. Knowing the total measured length of each pipe loop, the temperature drop per meter ($\Delta T/m$) of pipe length was calculated for each loop. Assuming a constant $\Delta T/m$ and knowing the distance to each measuring station, the water temperature for each pipe at each station was calculated.

4.3.4 Sources of Error

As with any experimental system, there are many sources of error. This system in particular is conducive to error due to its size, scope and the fact that it is subject to the natural environment. Firstly, there is much vegetation in the region and growth of roots through the sand trenches is inevitable. In fact, in one of the loops, a root actually grew through the Styrofoam insulation board. Secondly, although the thermistors are glued to the Lexan boards, the 5 cm tip protruding beyond the board is not stiff. Sand was packed by hand in these regions; however, with the weight of the sand and the flexibility of the sensor tips it is possible that a sensor could be displaced from its intended x-y coordinate

position on the Lexan board. This would cause inaccuracy in results, since the temperatures recorded are assumed to be at the discrete locations of the holes in the Lexan board through which the thermistors are placed.

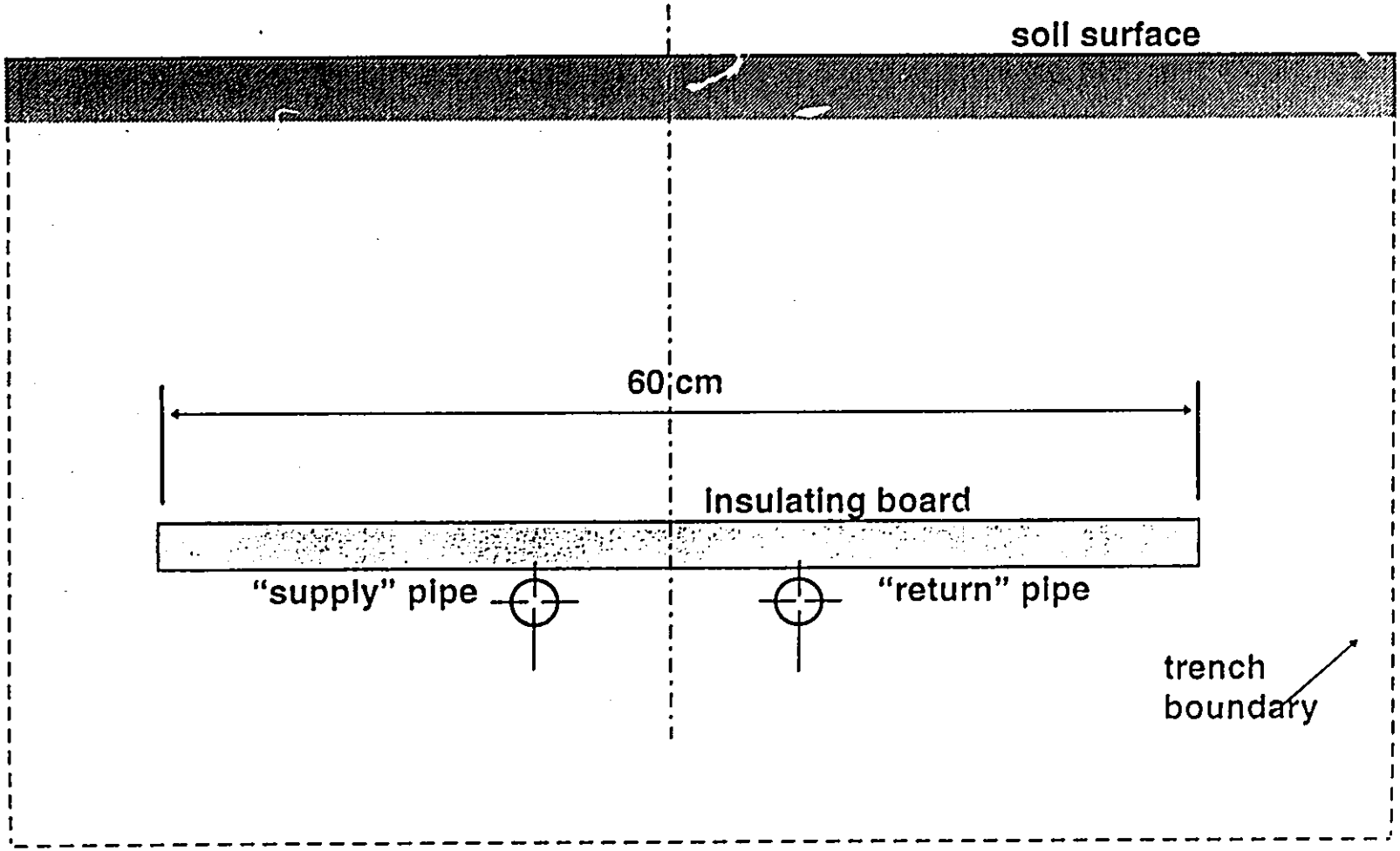

Figure 8 : Cross-Section of Trench

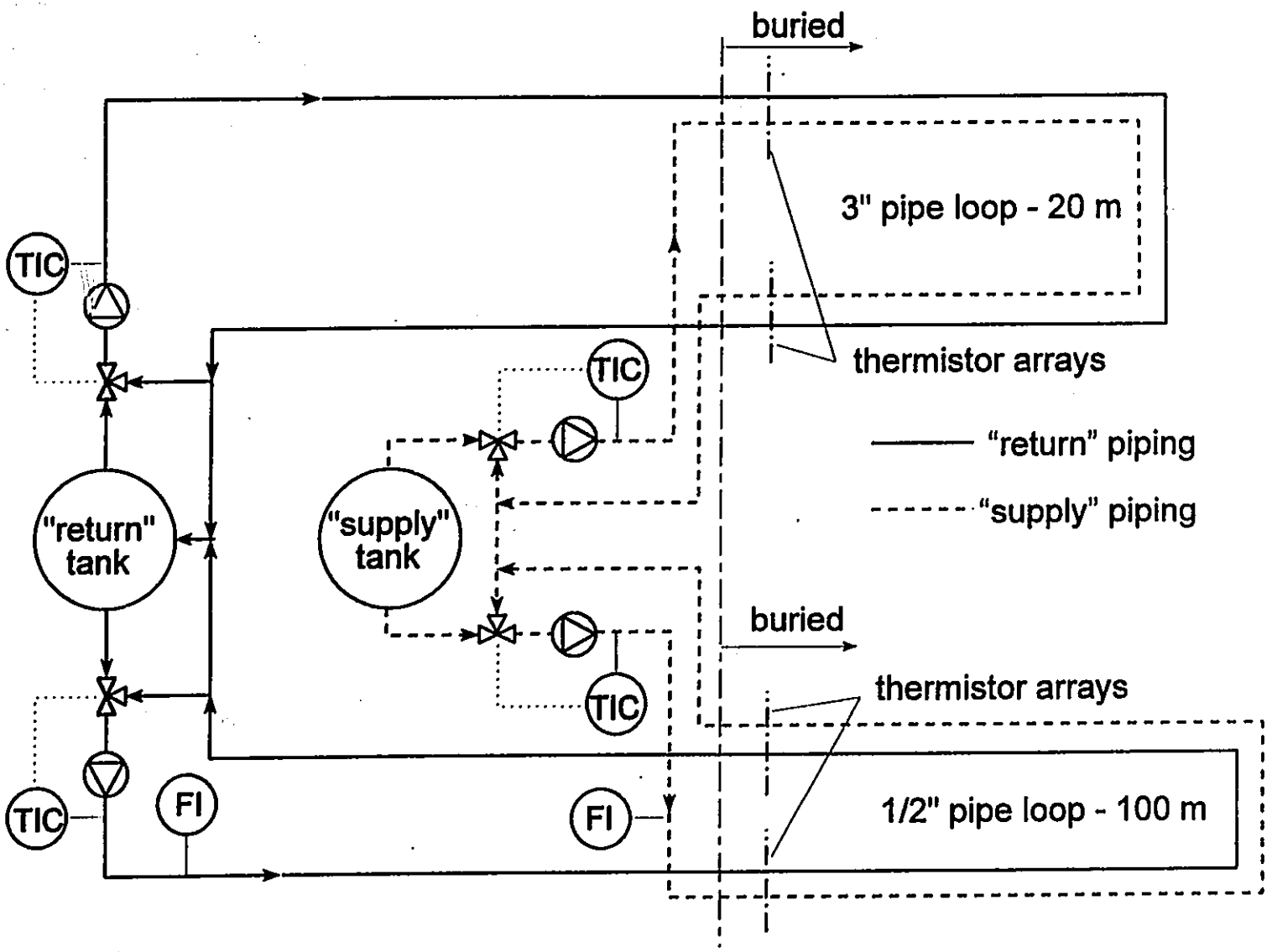

Figure 9: Piping and Controls Layout for 1/2" and 3" Systems

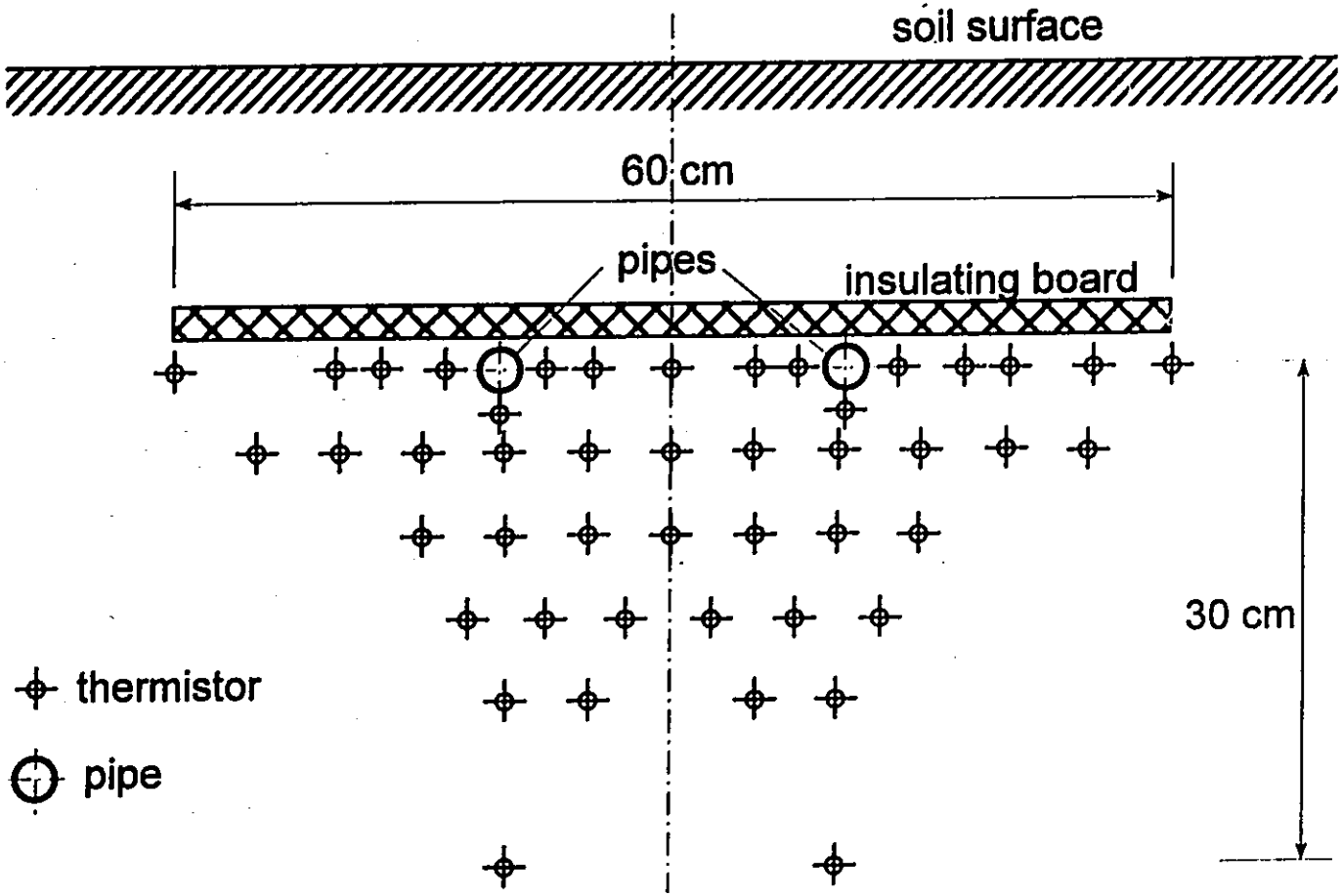

Figure 10: Cross-Section of Trench at Measuring Station

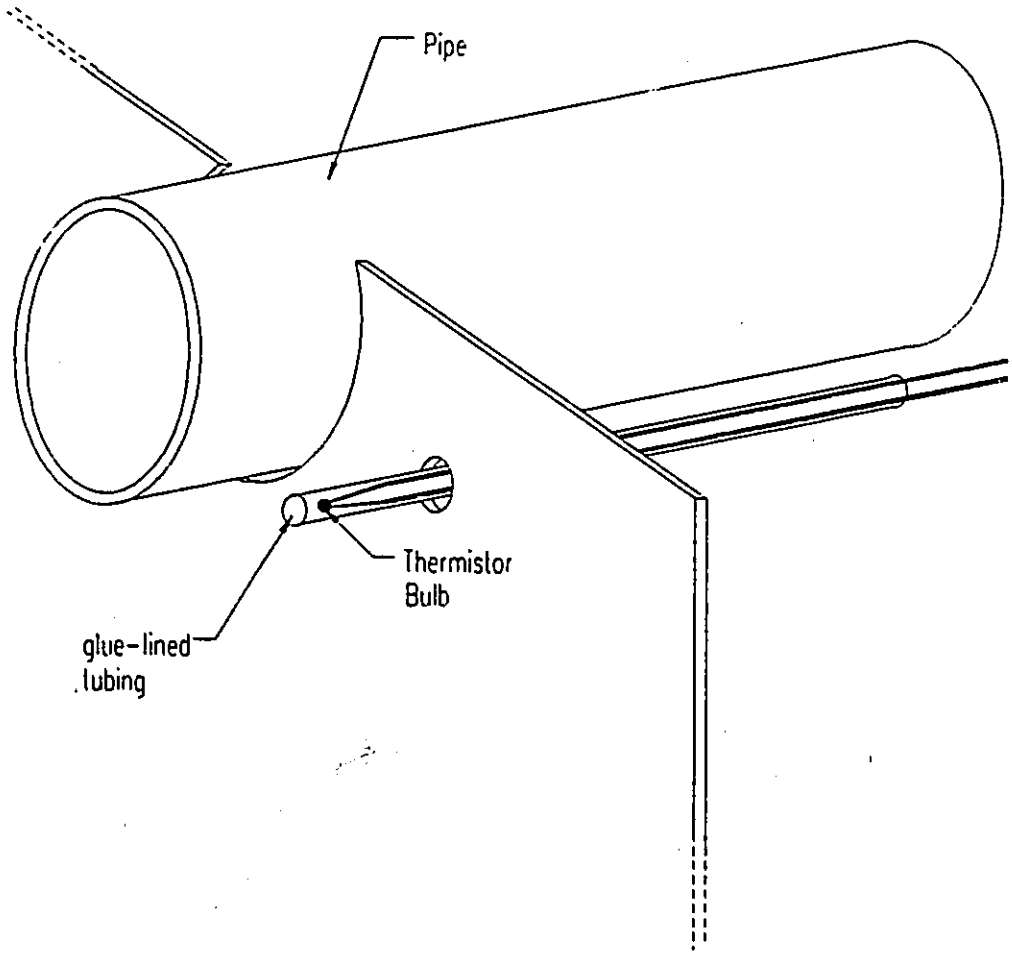

Figure 11: Thermistor Protruding Through Lexan Grid

5.0 RESULTS AND DISCUSSION

This chapter presents the experimental data collected over the five month testing period, focusing on five days representing the different conditions which occurred over that period. The experimental data are then compared with predicted temperatures resulting from model simulations run at the experimental conditions of each representative day.

5.1 Representative Experimental Data for Analysis

Unfortunately, the winter of 1994/1995 was not very cold and ground temperatures above the experimental pipe loops barely fell below freezing even in the coldest months. Figure 12 depicts ground temperatures from late November through to late February taken far from the experimental loops. These measurements were taken by RTD's placed at 0, 5, 20, 50, 80, 120, 150 and 180 cm below the ground surface. Ground surface temperatures above the experimental systems, required as input to the numerical model, were measured by extrapolation of readings from a line of five thermistors placed in the ground above measuring Station A of the 1/2" system. The thermistors were located between the top of the insulation board and 7 cm below surface level. They read higher than the temperatures shown in Figure 12 as there is a heat source below them.

Experimental data from five individual days were chosen from the five month testing period as representative data for that period. Data were chosen to reflect the different conditions occurring over the course of the testing period: pre-winter conditions, the

initial stages of ground frost penetration, steady mid-winter conditions, and pre-summer conditions, after the ground had warmed to steady spring temperatures. Since the model is a steady-state model, most data were taken after prolonged periods of steady ground and water temperature conditions. The pre-winter period, however, does reflect gradually changing ground conditions. This period, as well as a day during rapidly changing ground conditions, was studied to determine the effect of transients on the accuracy of the model results as compared to the steady-state conditions for which the model was designed.

December 26, 1994 reflects pre-winter ground conditions: the ground surface temperature above the 1/2" loops was 1°C. February 16, 1995 reflects the coldest mid-winter period. February was the coldest month with air temperatures measuring from -10°C down to -25°C; the ground surface temperature above the 1/2" loop on February 16 was -1°C. Again, this temperature is warmer than that measured in undisturbed soil, as shown in Figure 12, due to the presence of the hot pipes. The -1°C measurement is also higher than expected because of extensive snow cover. Data from March 6, 1995 reflects end of winter ground conditions, just before the ground warmed again. The ground temperature above the 1/2" loop was 0.3°C. April 3, 1995 represents steady spring ground conditions, well after the frost had left the ground; the ground temperature above the 1/2" loop on April 3 was 4.5°C. Finally, data from March 22, 1995 were chosen to represent a period of rapidly changing ground conditions. Temperatures had suddenly jumped up and a very heavy rainfall drenched the earth, causing flooding at the experimental field site. It is

expected that the thermal conductivity would have been changing as the water penetrated into the sand.

5.1.1 Heat Balance Results for Representative Experimental Data

A heat balance was performed on the 1/2" experimental loops for all five representative days to compare the experimental heat flow from each day with the corresponding day's predicted heat flux, resulting from a simulation of the experimental set-up. This comparison is required to determine the actual ground thermal conductivity on each of the five days. Thermal conductivity is a fitted model parameter; its extraction from the experimental heat balance results is discussed further in section 5.3.1. Table 2 gives the results of the experimental heat balances on the 1/2" system for all days under analysis:

Table 2: Heat Flows from Experimental Heat Balances

Date	q (W /m C)	
	1/2" System	
	supply	return
Dec. 26/94	27.5	0.01
Feb. 16/95	26.5	0.07
Mar. 6/95	29.12	0.61
Mar. 22/95	32.18	0.49
Apr. 3/95	29.74	0.01

The "supply" in Table 2 is the hotter of the two loops, simulating a supply district heating line. The "return" in Table 2 runs parallel to the supply loop, simulating a return district heating line. As is evident from the low values of the heat flux for the return loop, the

temperature in this loop is similar to the surrounding ground temperature and therefore the temperature drop around the loop is very small.

A heat balance could not be performed on the 3" loop since the temperature drop is so small (less than 1°C in most cases) and because the flowrate was not monitored on the 3" loops. Thermal conductivity for all days, found using results from the 1/2" loops, are assumed to be the same for the soil surrounding the 3" loops since the two systems are buried in close proximity to one another in the same field.

5.2 Trends in Physical Behavior: Isotherm Plots of Experimental Data

"Surfer", a three-dimensional graphics package, interpolates between the discrete experimental temperature points, recorded at each measuring station, to create isotherm plots surrounding the experimental pipes. Since Surfer recognizes only the temperatures at the thermistor locations and interpolates through the rest of the grid, an isotherm at the pipe temperature is imposed around the pipe surface to prevent unrealistic interpolations through the location of the pipes. Figures 13 through 16 illustrate plots for Station A (the first measuring station of the loop, 5 m from the beginning of the loop) for the 1/2" system on December 26, February 16, and March 6, respectively. Figures 17 through 19 illustrate some plots for Station A (the first measuring station of the loop, 5 m from the beginning of the loop) for the 3" system for December 26, February 16 and April 3, respectively.

The coordinates of all isotherm plots reflect the actual dimensions of the Lexan thermistor grid boards with the pipes sitting near the top of the board at the y = 0 line and the grid extending 30 cm downward below the pipes. Actual locations of thermistors in these boards are shown in Appendix C for each measuring station of the two systems.

5.2.1 General Trends

All isotherm plots of Figures 13 through 19 demonstrate expected trends: thermal gradients are steepest close to the hot pipe (supply pipe), illustrating a large heat loss, and decrease further away from the pipe. The isotherms are not crowded around the return pipe in this manner. The isotherms around the return pipe in the 1/2" systems in particular behave as though the pipe were not even there; this is because the thermal field set up in the ground by both pipes is very close to the return pipe temperature. Since none of the isotherms around the return pipe are hotter than the return pipe water temperature, there appears to be no net heating of the return pipe and the return pipe must be losing only a very small amount of heat.

The effectiveness of the insulation board is also seen in Figure 13 through 19. The insulation board is not shown on the plots but sit directly above the pipes shown. The isotherms approaching the insulation board (top of the plot at y=0) become nearly perpendicular, reflecting the minimal flow of heat through the board compared to the flow in the horizontal directions. This is expected as the conductivity of the insulation

board is 0.036 W/mC (AHRAE. 1982). whereas that of the soil is approximately 0.8 W/mC, almost thirty times greater than the insulation board.

Figures 13 through 16 exhibit small circular isotherms at approximately the same location in the grid (approximately x = -3 and y = -14). These are most likely spurious and caused by Surfer's interpretation of small errors either in measured temperature or in thermistor location.

5.2.2 Specific Comparisons Between Experimental Isotherm Plots

The isotherms of December 26 and February 16 in Figure 13 and 14, respectively, illustrate the differences in ground temperatures from pre-winter to mid-winter conditions. The lowest point of the 15°C isotherm is at approximately y = -20 cm in Figure 13, but at approximately y = -12 cm in Figure 14, demonstrating the effect of the colder ground temperature in February. The isotherm plot for April 3, Figure 16, shows the low point of this 15°C isotherm back down at y = -22 cm, illustrating the return to warmer ground conditions.

The isotherm plots of three of these same days for the 3" system, shown in Figures 17 through 19, exhibit similar trends. The 20°C isotherm in Figure 17 (December) has its lowest point at approximately y = -28 cm. Figure 18 shows that in February, the colder ground conditions push the lowest point of this 20°C isotherm up to y = -18 cm. The

warm spring conditions are again reflected in Figure 19 for April, with the lowest point of the 20°C isotherm back down to y = -25 cm.

Figure 20 shows an isotherm plot for the 1/2" system, Station B on March 6, 1995. This is the measuring station 5 m from the end of the 1/2" system. This plot is included to illustrate consistency in trends at this station. Comparison of Figure 15 with Figure 20 shows the 11.5°C supply pipe water temperature drop from Station A to Station B. The lower supply temperature at Station B is reflected by the less steep thermal gradients near the pipe.

Figure 21 shows an isotherm plot for the 3" system, Station B on March 6, 1995. This plot again shows consistency in trends. Since there is virtually no temperature drop between Station A and station B for the 3" loops, the isotherms plots for both Stations are very similar; gradients are equally as steep around the pipes.

5.3 Comparisons of Model Predictions and Experimental Results

5.3.1 Thermal Conductivity as a Fitted Parameter

The model assumes a uniform thermal conductivity and requires it as an input parameter. Since measurements of the thermal conductivity of soil have such a high degree of uncertainty (±25%; Phetteplace and Meyer, 1990), the most effective means of

determining its value for experimental simulation model runs is to treat the soil conductivity as a fitted parameter. This method provides the most accurate simulation of experimental conditions and thus provides the most realistic comparison between experimental and model results. Experimental heat balances, necessary for fitting the thermal conductivity, could only be performed on the 1/2" system, results of which were presented in Table 2. The soil conductivity calculated for each day using results from the 1/2" system are assumed the same for the 3" system as the 3" system is buried close to the 1/2" system.

To fit the thermal conductivity, the experimental heat flux is compared with the predicted heat flux resulting from a simulation of experimental conditions, using mean loop temperatures and a guessed value of the thermal conductivity. Since the model requires specific pipe temperatures as input, mean loop temperatures are used to best represent the heat flux from the entire loop for comparison with the energy balance, which is based on the temperature drop around the entire loop. The model is re-run at the same conditions, changing only the thermal conductivity, until the heat flux from the model matches the heat flux from the energy balance. It is important to note that the thermal conductivity essentially controls only the heat flow; the conductivity has very little effect on the thermal field set up in the soil.

The grid used for model runs to determine the conductivity was an 82 (x-axis) x 50 (y-axis) node grid with the pipes buried under insulation in the same configuration as the

experimental set-up. A reduced sub-grid geometry was used around both 1/2" pipes; the main grid nodes in which the sub-grid was placed were spaced 4 cm apart. Each sub-grid radial line held 23 nodes spaced no more than 0.2 cm apart. The main grid node spacing at the center of the x-axis, near the pipes, was 4 cm. The nodes extending to the boundaries on either side of the pipes increased successively in distance from one another by 10% per node. Main grid nodes along the y-axis were spaced at 4 cm from the top of the grid to the pipe level; below the pipes, distance between successive nodes increased by 10%.

Table 3 shows the fitted thermal conductivity found for each of the days under analysis:

Table 3: Fitted Thermal Conductivity for Representative Data

Date	k (W/mC)
Dec. 26/94	0.90
Feb. 16/95	0.85
Mar. 6/95	0.95
Mar. 22/95	1.30
Apr. 3/95	1.10

The soil thermal conductivity shows a marked increase on March 22, 1995, remaining high through to April 3, 1995. As mentioned in section 4.1, there was a heavy rainfall on March 21 and March 22, causing floods in the area of the experimental loops. Evidently, the excess rain did penetrate deep into the soil to the level of the thermistor grids, increasing the thermal conductivity as expected. The high thermal conductivity on April 3 indicates either that the soil is still wet or that the rainfall altered the soil density to

make the thermal conductivity of the dried soil higher. Since the grade of the earth is sloping, it is possible that a horizontal flow of water past the pipes caused packing of the sand.

The value of the conductivities in Table 3 fall within the range of values found in the literature for sandy soils: Kersten (1940) reported a range of thermal conductivity for sandy soils of 0.6 to 3.2 W/m°C. This range reflects a gradual change in soil properties from 0% moisture content and low density to 30% moisture content, 100% saturation and very high density. The values given by Kersten resulted from experimental measurements of soils and have an accuracy of ±25%.

5.3.2 Comparison of Predicted and Experimental Temperatures: 1/2" System

Figure 22 illustrates the major points on the grid used for the simulations of the 1/2" system for all days. The grid has 74 nodes along the x-axis and 50 nodes along the y-axis. The two pipes are placed at the center of the x-axis and are surrounded by the reduced sub-grid geometry. The node spacing from the pipes out to the main grid boundaries increases by 10% per node.

Figures 23, 24 and 25 illustrate comparisons between predicted and experimental temperatures for the 1/2 " system on December 26, 1994, February 16, 1995 and March 6, 1995, respectively. All Figures are of Station A, 5 m from the beginning of the pipe loops. Temperature is plotted on the y-axis with distance across the width of the trench

on the x-axis. The experimental points plotted are the thermistor measurements across the width of the trench at the level of the centerline of the pipes. Again, the exact locations of the thermistors in the Lexan grids are given in Appendix C. The predicted temperatures are at discrete nodal positions across the width of the trench; points outside the sub-grids are at a depth of 2 cm below the level of the pipe centerlines and points within the sub-grid are plotted along the two radial lines extending from the inner pipe wall out to the edges of the sub-grids at angles of 22.5° below the horizontal on either side of the pipe centerline. It is not possible to obtain main grid predicted temperatures at exactly the pipe centerline level due to the manner in which nodes and sub-grids are set up in the model's calculation domain; the pipe center is at the meeting point of four control volume faces. It was found, however, that there was very little difference between predicted temperatures 2 cm below the pipe centerline and those 1 cm above the pipe centerline; comparing predicted temperatures 2 cm below the pipe centerlines with experimental points at the pipe centerline is then quite reasonable. The pipe centerline locations for the 1/2" system, Station A, are at +10 cm and -10 cm for "supply" and "return", respectively.

Figures 23 through 25 all exhibit similar trends: on all three days, the predicted temperatures are consistently higher than the experimental temperatures. The predicted temperatures peak at the actual "supply" water temperature because the nodes extend as far as the inner pipe wall. The thermistors come only to within 2 cm of the pipe wall and therefore the experimental points do not show temperatures as high. Since not all

predicted nodes are vertically aligned with thermistors (at the same x-coordinate), four points which are aligned (at -20.5 cm, -9 cm, +8 cm, +20.5 cm) will be used in the following discussion of direct comparisons between predicted and experimental temperatures.

The predicted and experimental temperatures at -20.5 cm, on the x-axis, disagree by 1°C on December 26 and February 16, and by 1.5-2°C on March 6. Such a small discrepancy can be assumed negligible. At -9 cm, directly to the left of the return pipe, the predicted and experimental temperatures agree quite well on all three days. In the region surrounding either side of the supply pipe, the discrepancy between predicted and experimental temperatures becomes much more marked. Just to the left side of the "supply" pipe, at +8 cm, the experimental temperature is lower than the predicted temperature by 8°C on all three days. Further away from the supply pipe, at +20.5 cm, the experimental temperature is 4°C below the predicted temperature on December 26 and 6°C below the predicted temperature on February 16 and March 6.

The similarity in trends between the predicted and the experimental temperatures in each of the Figures 23 through 25 instills confidence in the accuracy of the model. Although not all values agree, the trends are consistent over all three representative days which raises this level of confidence. In fact, the similarities in trends and between temperatures are reasonable, considering the many unknowns and uncontrollable factors of the experimental system in natural environmental conditions.

Figure 26 also shows experimental and predicted temperatures for December 26 plotted as a function of distance across the width of the trench, but both are at a depth of 5 cm below the centerline level of the pipes. Figure 26 shows less detail than Figure 23, simply because the thermal gradients are not as steep farther away from the pipe centerline. The figure does, however, show the same discrepancies between predicted and experimental temperatures and exhibits the same trends. At -20 cm, the model temperature is again 1° higher than the experimental temperature. The predicted and experimental temperatures exhibit very good agreement in the region of the "return" pipe and become much more discrepant around the "supply" pipe. At +11 cm, 5 cm below the "supply" pipe, the predicted temperature is approximately 7° higher than the experimental temperature, which is consistent with Figure 23.

5.3.2.1 Air Gaps

The fact that the discrepancies between predicted and experimental temperatures are consistent over the three days, which are representative of three months, leads to the conclusion that there is a plausible explanation for the discrepancy. As was discussed in Section 4.3, the sand directly around the pipes, below the insulation board, was compacted by hand. The insulation board was placed directly on top of the two pipes, with no sand backfill between the top of the pipes and the board. Unfortunately, in reality this critical area directly around the pipes cannot be not perfectly compacted; there are bound to be some void areas underneath the board or, more likely, areas of more loosely packed sand. This problem would only worsen as the sand inevitably settled over time

underneath the board. If such voids or pockets of looser sand did form, there would be a change in thermal conductivity of the sand near the pipes, causing a change in the temperature distribution throughout the trench.

To simulate the effects of this settling, the model was run with a small air gap between the pipe and the soil. Small, pure air gaps are imposed to represent slightly larger areas of looser (more void fractions) sand. To simulate an air gap, the thermal conductivity of the second circle of nodes (first circle of nodes outside the pipe wall) was forced to be that of air at the approximate temperature of the pipe (0.027 W/m°C at 40°C). Since reduced sub-grids were used in the 1/2" loop simulations, the two rectangular nodes directly above the pipes were also forced to be air, simulating the possible gap between the top of the pipe and the board. Figure 27 illustrates the air gap (hatched area) imposed around the pipe in the reduced sub-grid geometry. Figures 28, 29 and 30 illustrate comparisons between predicted and experimental temperatures again for December 26, February 16 and March 6, with the predicted temperatures resulting from simulations with imposed air gaps around each pipe. A 0.035 cm air gap resulted in the closest fit of predicted temperatures to experimental temperatures for all three days. This would imply that there was no progressive settling underneath the insulation board over time after December.

Figures 28 through 30 are again of temperature vs. distance across the width of the trench. Experimental temperatures are the same as those in Figures 23 through 25 and

predicted temperatures are calculated at the same locations. The Figures show very good agreement between experimental and predicted temperatures. Each graph shows two pairs of isolated points forming a peak at the "supply" pipe. The first pair are the two sub-grid node temperatures on the inside of the pipe wall. The second pair are the next two sub-grid nodes which fall inside the imposed air gap around the pipe. Similar pairs of predicted temperatures can be seen forming a peak at the "return" pipe; however, the peak is much less marked because the "return" pipe temperature is close enough to the surrounding ground temperature that the air gap has a much smaller effect on the results. The remaining predicted temperatures throughout the grid are lower than the temperatures at the corresponding positions in Figures 23-25, reflecting the lower temperature distribution resulting from the pipes with the insulation of an air pocket around them.

Comparison of the same four points across the trench as were compared in Figures 23 through 25, illustrates how well these new predicted temperatures agree with the experimental temperatures. At +8 cm, just to the left of the "supply" pipe, there is very good agreement between the predicted and experimental temperatures on each of the three days: all three days dropped from a discrepancy of 8°C between model and experiment to a 0°C discrepancy. At +20.5 cm, on the right side of the "supply" pipe, the predicted temperature is approximately 1°C lower than the experimental temperature on December 26 and February 16. Again, a 1°C discrepancy is small enough to be considered negligible. The two temperatures at this location are in very close agreement on March 6. At -20.5 cm, on the left side of the "return" pipe, similar discrepancies are

seen: the model temperature is 1°C below the experimental temperature on December 26 and February 16. Temperatures at this location are in very close agreement on March 6. Again, the temperature discrepancies to the left of the "return" pipe on December 26 and February 16 are small and can be considered negligible.

The success of the air gap model would seem to indicate that there is indeed a non-uniform thermal conductivity around the pipes caused by more loosely packed sand, resulting from settling of the sand below the insulation boards, right on top of the pipes and/or poor packing of the sand in this region to begin with. The comparison plots of Figures 28 through 30, discussed above, support this theory very well.

Introducing an air gap around the 1/2" pipes increases the soil thermal conductivities above those fitted for the systems without air gaps, given in Table 3. Table 4 gives fitted thermal conductivities for the five representative days for the experimental set-up with a 0.035 cm air gap imposed around the pipes. It is again important to note that the thermal conductivity controls the heat flow and has very little effect on the thermal field. Re-calculation of the thermal field with the new values of k did not show any noticeable difference in temperatures; hence, one can conclude that k does not affect the thermal field, even though it directly controls the heat transfer rate. This is entirely in agreement with the classical theory for a single buried pipe, in which the equation for the location of the isotherms is entirely independent of k.

Table 4: Fitted Thermal Conductivities: Air Gaps Imposed Around Pipes

Date	k (W/m C)
Dec. 26/94	3.0
Feb. 16/95	2.6
Mar. 6/95	3.1
Mar. 22/95	5.0
Apr. 3/95	3.8

The conductivities in Table 4 lie at the high end and beyond the range of thermal conductivity found in the literature for sandy soils, given in Section 5.3.1. The values in Table 4 were fitted with the assumption that the heat flux along the pipe is uniform. In reality, there is not likely a uniform air gap along the length of the pipe; it is more likely to be localized at the measurement stations where the sand had to be hand-packed around the sensor grid. This would result in a lower heat flux at the measuring station than assumed so that the thermal conductivity would be lower than in Table 4.

5.3.2.2 Anomalies

There are two experimental data points (temperatures) in particular which seem to be somewhat anomalous in all of the comparison plots discussed in this section. In both the comparison plots without the imposed air gaps (Figures 23 through 25), and in the comparison plots with the imposed air gaps (Figures 28 through 30), the experimental temperatures at 0 cm and +12 cm along the trench width do not fit the trend of the temperatures surrounding them.

In Figures 23 through 25 the temperature at +12 cm agrees well with the predicted temperature; this is inconsistent with all surrounding experimental temperatures, which are 4-6° lower than the corresponding predicted temperatures. In Figures 28 through 30, the experimental temperature at this location is again inconsistent with its surrounding temperatures. In both sets of figures, it appears to be consistently too far to the right: that is, it appears that the temperature recorded is correct but the physical location in the experimental grid is not. If this is the case and the locations were to be moved to the left by 1-2 cm on the x-axis, the measurement would conform nicely to the trend of the surrounding temperatures. The error at this +12 cm point could be caused by a combination of misalignment due to the flexibility of the thermistor tip protruding through the board, and local anomalies in thermal conductivity caused by settling of the sand. This sensor is right next to the pipe wall and is therefore in the most difficult location to backfill properly.

The other anomalous point, at 0 cm, does not have a similar suitable explanation for its non-conformity because the problem lies in the temperature reading rather than the horizontal distance: the temperature is too low. The consistency of the behavior of this point through all figures presented in this section leads to the conclusion that the thermistor itself is defective.

5.3.2.3 Data Comparison During Transient Conditions

Figure 31 illustrates a comparison between predicted and experimental temperatures on March 22, 1995, during a period of rapidly changing ground conditions. A heavy rainfall had drenched the earth, causing flooding and penetration of water to the level of the thermistor grids. The increase in fitted soil conductivity, seen in Table 3 for this day, indicates that indeed it is a period of changing ground conditions. Data from this day is represented to examine the efficiency with which the steady-state model predicts temperatures during very transient conditions.

Figure 31 exhibits the same trends as seen in Figures 23 through 25, however the discrepancies are larger in Figure 31. At +8 cm, directly beside the supply pipe, the predicted temperature is 10°C higher than the experimental temperature, whereas in Figures 23 through 25, the discrepancy at this location was only 6°C. Similarly, the predicted temperatures near the "return" pipe are approximately 4°C higher than the experimental, and to the right of the "supply" pipe, they are approximately 5-6°C higher than the experimental temperatures. These are all higher than the discrepancies seen at the same locations in Figures 23 through 25. Although the discrepancies are higher in Figure 31, the amount by which they are higher is not substantial and the trends in Figure 31 are the same as those seen in Figures 23 through 25. Imposing a 0.035 cm air gap around the pipes on March 22, as seen in Figure 32, shows the predicted temperatures to fit quite reasonably to the experimental temperatures, in fact almost as well as those in Figures 28 through 30 for the other days with imposed air gaps. It would seem, therefore,

that the steady-state model handles transient conditions well: although the results are not quite as accurate, there is no abnormality in the predicted temperature trends. It should be pointed out that this is an extreme case: not only has there been a sharp transient in temperatures, but the ground has been heavily soaked as well, and it is possible that some flow of water may have occurred through the test area causing a convective flow of heat.

5.3.2.4 Conclusions

This section has demonstrated that the predicted temperatures match well with the experimental temperatures. The selected conditions on the days studied exhibited consistencies in predicted temperature trends and degrees of discrepancy. Discrepancies between predicted and experimental data were reasonably accounted for, and rectified, by imposing a small air gap around the pipes. The air gap represents the effect of actual air pockets in the soil around the pipe as well as larger void fractions in poorly compacted sand. Data from December 26 showed trends which were very consistent with those seen in February 16 and March 6. As the ground is undergoing gradual cooling in December, the model's ability to handle gradually changing soil conditions is demonstrated. Analysis of data from March 22 demonstrated that the model is also able to handle fairly rapid transient conditions. Although the fitted thermal conductivity changed on March 22, the model predictions were still reliable, and the change can be attributed entirely to the increased moisture content of the soil.

5.3.3 Comparison of Predicted and Experimental Temperatures: 3" System

Figure 33 illustrates the major points on the grid used for the model simulations of the 3" system for all days. The grid has 75 nodes along the x-axis and 50 nodes along the y-axis. The two pipes are placed at the center of the x-axis and are surrounded by the reduced sub-grid geometry. The node spacing from the pipes out to the main grid boundaries increases by 10% per node.

Figures 34, 35 and 36 illustrate comparisons between predicted and experimental temperatures for the 3" system, Station A, on December 26, 1994, February 16, 1995 and March 6, 1995, respectively. The graphs are in the same format as Figures 23-25 and 28-32. Again, the experimental points plotted are the thermistor measurements across the width of the trench at the level of the centerline of the pipes. The predicted temperatures outside the sub-grids are at a depth of 4 cm below the level of the pipe centerlines, rather than the 2 cm in the 1/2" system plots, because the sub-grid nodes take up more physical space around the 3" pipes than around the 1/2" pipes. The first node below the pipe centerline corresponds to the level of the lower sub-grid nodes, which for the 3" pipes are at 4 cm below the pipe centerline. Again, it was found that there was little difference between predicted temperatures 4 cm below the pipe centerline and those 2 cm above the pipe centerline, so Figures 34 through 36 do not include predictions at 2 cm above the pipes. Points within the sub-grid are plotted as in the 1/2" graphs: along the two radial lines extending from the inner pipe wall out to the edges of the sub-grids at angles of

22.5° below the horizontal on either side of the pipe centerline for each pipe. The pipe centerline locations for the 3″ system, Station A, are at +14 cm and -14 cm for "supply" and "return", respectively.

Figures 34 through 36 again all exhibit similar trends. On all three days, the model points show consistently higher temperatures than the experimental temperatures around the return pipe and on the right hand side of the "supply" pipe. The discrepancy between the two, however, is greater on the right hand side of the "supply" pipe than around the "return" pipe and there is virtually no discrepancy between the two on the left hand side of the "supply" pipe. Predicted temperatures in Figures 35 and 36 (February 16 and March 6, respectively), are 6°C higher than experimental temperatures on the left hand side of the "return" pipe at -24 cm and approximately 1°C higher directly to the right of the "return" pipe at -8 cm. Agreement in this region is much better for December 26: Figure 32 shows predicted temperatures to be only 1°C higher than experimental temperatures at -24 cm. The predicted and experimental temperatures on the left hand side of the "supply" pipe, however, agree quite well with one another. It would seem that the explanation for this behavior is slightly more complicated than an evenly distributed air gap around each pipe.

Although the trends are different than for the 1/2″ loops, the fact that they are consistent for the 3″ loop through Figures 34, 35 and 36, leads again to the conclusion that the model is consistent in its predictions over the three month period and there must be a

reasonable explanation for the discrepancies that exist. The predicted and experimental temperatures agreed very well for the 1/2" loops with evenly distributed air gaps imposed around the pipes. The reasoning behind the imposition of the air gaps was to represent the settling, or perhaps poor packing, of the sand directly beneath the insulation board, which created areas of looser packed sand in the critical areas around the pipes. The same insulation board configuration and method of packing was used for the 3" loops; therefore, one could presume there is a similar possibility for error and the likelihood that loosely packed sand exists just below the insulation board.

5.3.3.1 Air gaps

Figures 37, 38 and 39 illustrate comparisons between predicted and experimental temperatures again for December 26, February 16 and March 6, with the predicted temperatures resulting from simulations with air gaps inserted around each pipe. Air gaps were inserted around the pipes in the same manner as for the 1/2" loops, as explained in Section 5.3.2.1 Figures 37 through 39 show that the imposition of an air gap results in good agreement between predicted and experimental temperatures around the "return" pipe and on the right hand side of the "supply" pipe. An air gap of 0.20 cm was imposed around both pipes in Figures 38 and 39, but a smaller air gap, only 0.05 cm, was required for December 26 (Figure 37). Figures 38 and 39 for February 16 and March 6, respectively, show virtually no discrepancy between predicted and experimental temperatures on the return pipe and the right hand side of the supply pipe. Predicted temperatures are lower than the experimental temperatures on the right hand side of the

"return" pipe but only by approximately 2°C. Figure 37 shows virtually no discrepancies on either side of the "return" pipe and only about 1°C on the right hand side of the "supply" pipe, with the model prediction slightly higher than the experimental temperature. It appears, therefore, that a small air gap is a reasonable explanation for predicted and temperature discrepancies for the regions around the "return" pipe and to the right of the "supply" pipe. However, introduction of the air gap has produced a strong disagreement between predictions and measurements on the *left* hand side of the "supply" pipe, and it seems that the sand must actually be compacted on this side. An explanation for this is required.

A major construction difference between the 1/2" pipe loops and the 3" pipe loops, which may provide some insight into the observed trends near the supply pipe, is that the 1/2" loops have flexible couplings at the 90° elbows to allow for expansion, whereas the much larger 3" pipes do not have any means of accommodating expansion. The pipes were laid in the summertime when the ground was warm (approximately 20°C). When the system rose to the set-point temperatures, with supply at approximately 40°C and return at approximately 25°C, expansion of the "supply" pipe would have caused it to shift in the sand. Looking at Figures 34 through 39, it would appear that the "supply" pipe did indeed shift horizontally. Since the imposition of only a 2 mm air gap can change the predicted temperatures near the pipe greatly, as seen, for example, in Figures 38 and 39, it is quite reasonable to assume that the supply pipe shifted slightly over to the left, causing the formation of a larger void fraction on the right hand side of the pipe and compaction

of the sand on the left hand side, between the two pipes. As discussed previously, it also appears reasonable to assume that there was an existing area of looser sand between the top of the pipes and the insulation board, unrelated to the shift and caused by poor packing or settling of the sand in this region. The pipe shifting would account for the fact that in Figures 34 through 39 the predicted and experimental temperature discrepancies are less on the right hand side of the "return" pipe than on the left hand side. A shift to the left of the "supply" pipe would cause the sand in between the two pipes to become more compact, the effect being more pronounced directly beside the "supply" pipe but also extending somewhat towards the "return" pipe. The shift also explains why the temperature discrepancies are much higher to the right of the "supply" pipe than to the left of the "return pipe": the already existing small area of looser sand, due to poor compaction or sand setting, would be loosened further by the shift to the left of the "supply" pipe.

Figure 40 illustrates a comparison plot between predicted and experimental temperatures with a 0.30 cm air gap imposed around the entire return pipe, and a 0.50 cm air gap imposed around only the right hand side of the "supply" pipe. The air gap around the "supply" pipe is imposed by forcing the second ring of sub-grid radial nodes on the right hand side only to have a thermal conductivity of air (0.027 W/mC for a temperature of 40°C). Figure 40 thus substantiates the explanation proposed above: there is no discrepancy between predicted and experimental temperatures anywhere across the width of the trench. The higher air gap required to fit predicted to experimental temperatures on

the right hand side of the "supply" pipe indicates that there were indeed greater void fractions in the sand in this region due to a shift of the "supply" pipe.

5.3.4 Predicted Isotherm Plots

Figures 41 and 42 show isotherm plots resulting from model simulations for the 1/2" system, Station A and the 3" system, Station A, respectively, on December 26, 1994. Since Figures 28 and 37 showed that the actual situation in the ground is most accurately represented by imposing small air gaps around the pipes to reflect areas of loosely packed sand near the pipes, the isotherm plots in Figures 41 and 42 were created from a model simulation run with imposed air gaps: a 0.035 cm air gap for the 1/2" loops and 0.30 cm around the "return" pipe and 0.50 cm around the right hand side of the "supply" pipe for the 3" loops.

Figures 41 and 42 exhibit the same trends as discussed in Section 5.2 for the experimental isotherm plots. The thermal gradients are steepest close to the "supply" pipe, as seen by the crowding of isotherms in this region on both plots, and they decrease further away from the pipe. Comparison of Figure 41 with Figure 13, presented in section 5.2, reinforces the theory that there is loosely packed sand around the experimental pipes. Figure 41 shows a temperature distribution very similar to Figure 13, although the model is obviously smoother than the experimental plot, as it has no experimental error.

5.4 Technology Application: Design for the Community of Davis Inlet

The numerical model detailed and proven in this report was used for the design of an integrated services system for Little Sango Pond, the new site of the people of Davis Inlet. The concept of integrated services was seen to be ideal for Little Sango Pond, as the frost penetration in the northern site is too deep for traditional deep buried municipal services systems. Also, the integrated services concept is attractive to native communities because the use of natural resources from within the community or nearby regions as fuel for the District Heating system promotes self-reliance.

A preliminary design for the building connection trenches (from main DH lines to buildings) is presented here as this section of the system is the most critical in terms of potential freezing. It is also often the most cost-intensive section of traditional services systems in cold climates due to the effect of heat tracing and/or recirculation system requirements. Integrated services systems eliminate these costs.

Since Newfoundland law states that low pressure sewer systems must be buried in a separate trench from watermains with a minimum 3 m separation distance between the two trenches, the design for Davis Inlet consists of two trenches running parallel to one another. Figure 43 shows one trench containing the pressurized sewer line buried beside the supply district heating pipe and the other containing the watermain buried beside the return district heating pipe. The watermain is placed in the trench beside the return DH

line rather than the supply DH line to minimize excessive heating of the water during summer months. Both supply and return DH lines are 25 mm nominal diameter polyethylene pipes, pre-insulated with polyurethane foam.

The numerical model was used to determine the best insulation configuration to obtain a warm thermal field in the region of the municipal lines. (The municipal lines were not modeled; it is sufficient that the thermal field in the region where they will be placed is kept above freezing). Since there is only one DH line per trench, it was found that a box of insulation surrounding both pipes (in each trench) was the most effective configuration (see Figure 43). The model was used again to determine the thickness of the box walls and the thickness of the insulation on the pipes which would produce a warm thermal field within the box. An insulation box of 1 cm thickness was used with a pipe insulation thickness of 5 cm on the return pipe and 3 cm on the supply pipe. The ground temperature was set at -15°C, representing the ground temperature under snow cover, rather than the colder air temperature. No air gaps were inserted because there is more space between the pipes and the insulation board than there was in the experimental loops described in section 4, allowing easier compaction of the sand around the pipes.

Figure 44 illustrates the thermal field inside the insulation box for the return/water trench resulting from the model simulation. The figure shows a thermal field greater than 0°C, in the range of 5°C. This is sufficient to prevent the municipal lines from freezing. The

supply/sewer trench is not included because the hotter 90°C supply pipe ensures this trench too will have a warm enough thermal field.

When compared with designs for separately installed water and sewer services (Davis, 1994), the integrated services system showed substantial savings through elimination of the traditional freeze protection schemes such as heat tracing and recirculation. Further cost savings were also identified such as reduced excavation costs and reduced piping costs due to reduced insulation requirements. The design for this site showed the practicality and economic benefits of integrated services.

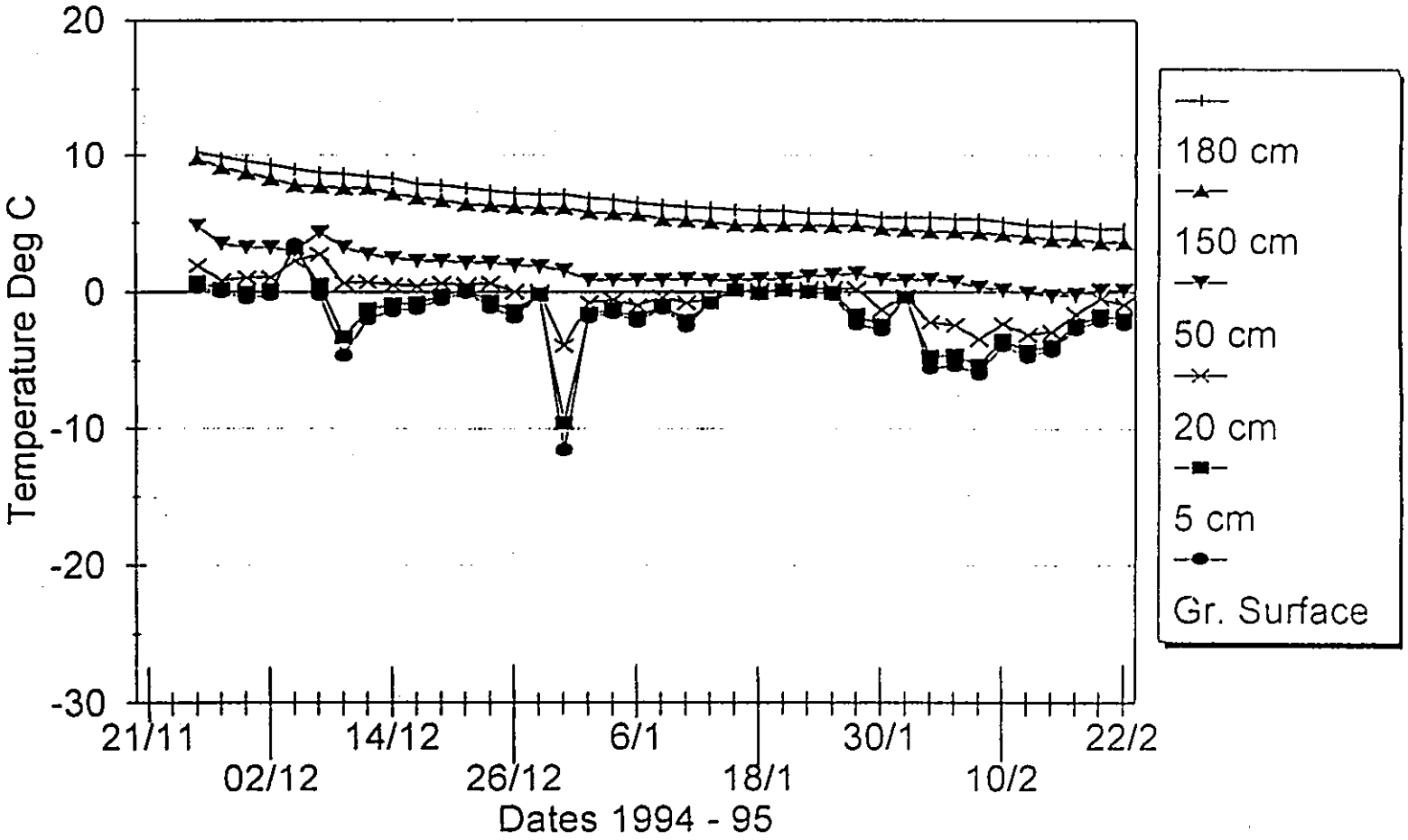

Figure 12: Ground Temperatures Measured Far From Experimental Pipes

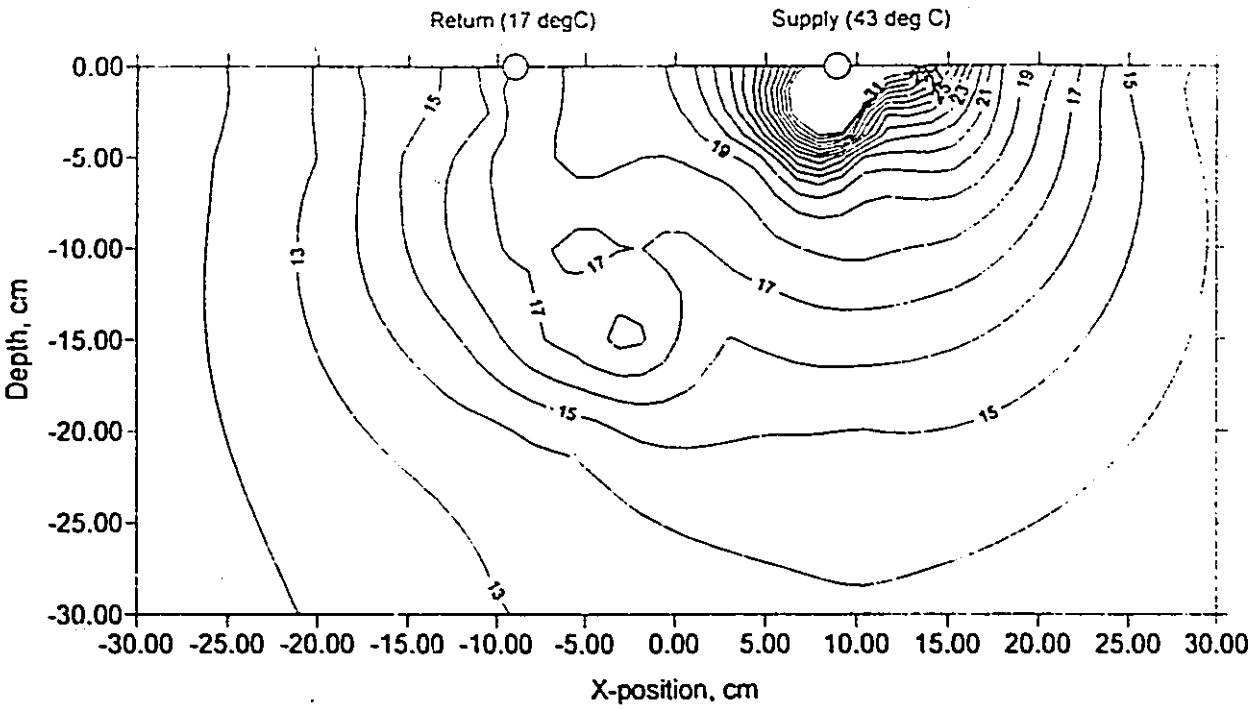

Figure 13: Isotherms from Experimental Data
1/2" System, Station A: December 26, 1994

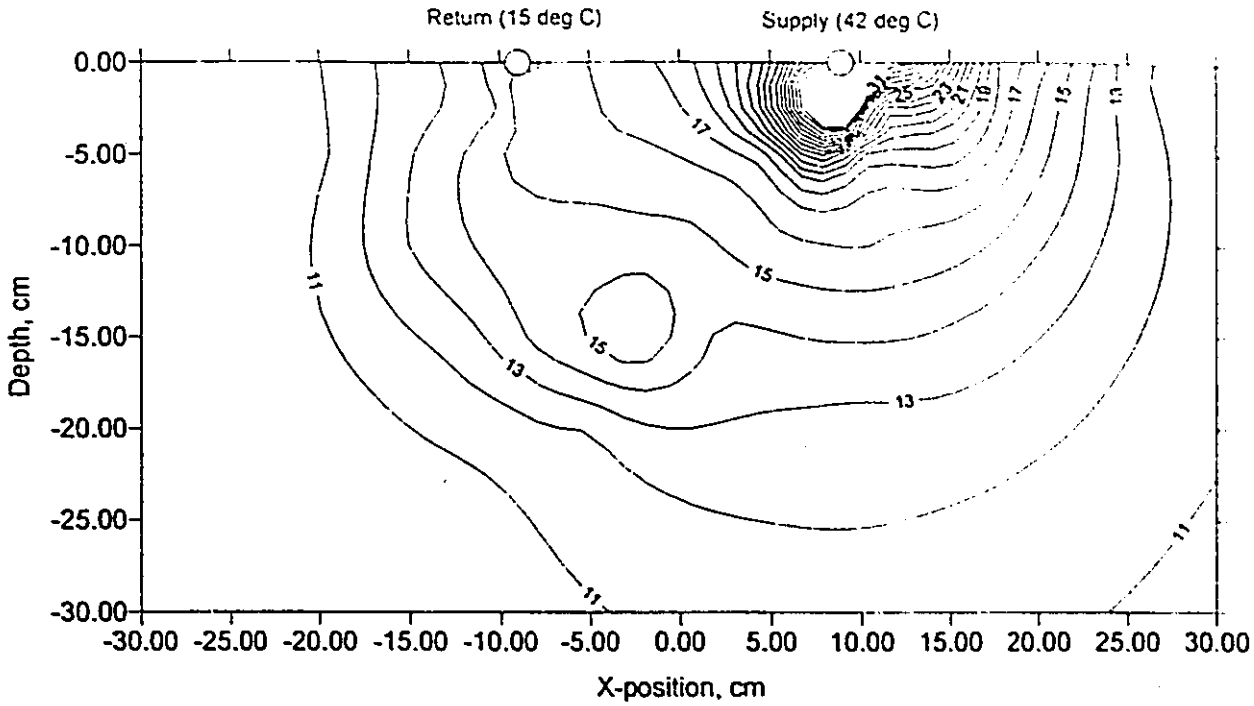

Figure 14: Isotherms from Experimental Data
1/2" System, Station A: February 16, 1995

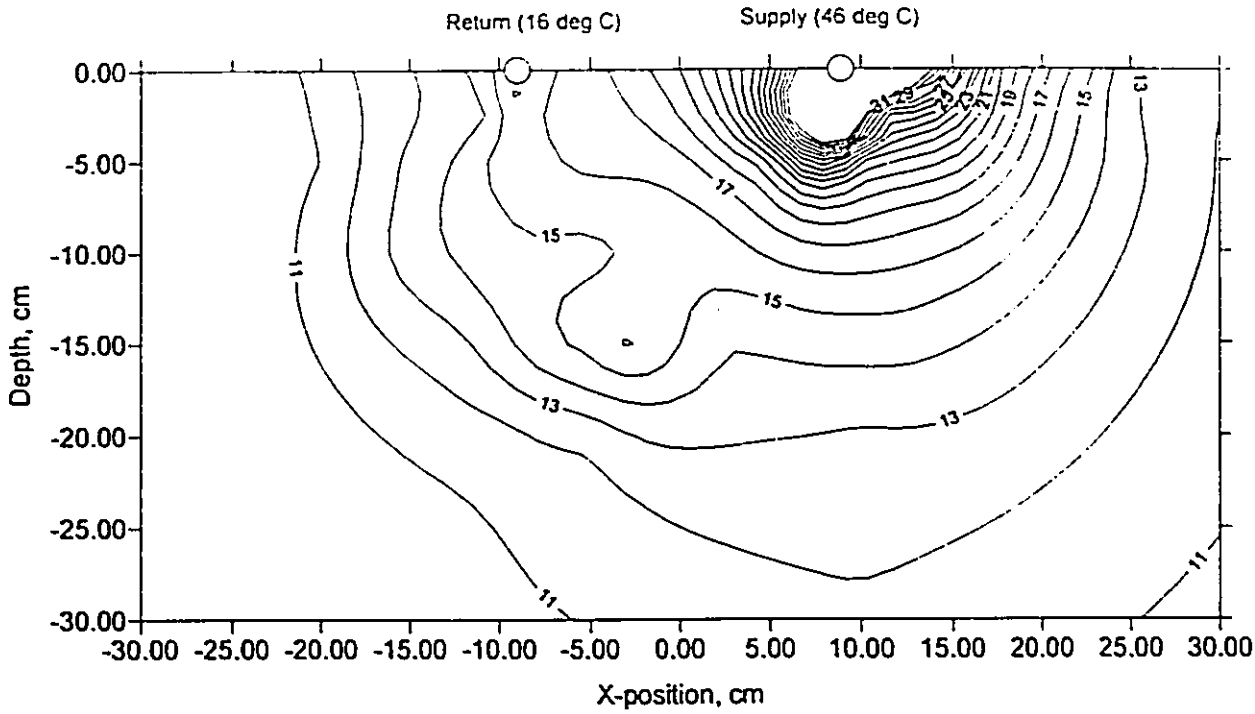

Figure 15: Isotherms from Experimental Data
1/2" System, Station A: March 6, 1995

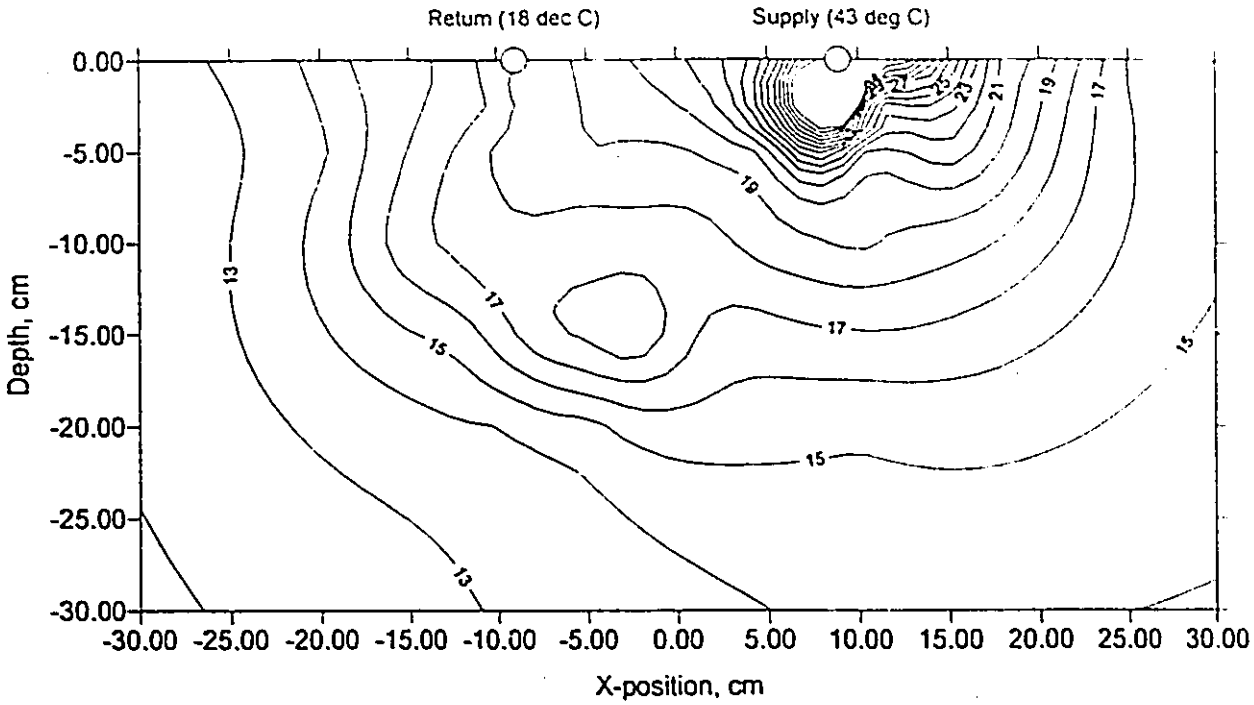

Figure 16: Isotherms from Experimental Data
1/2" System, Station A: April 3, 1995

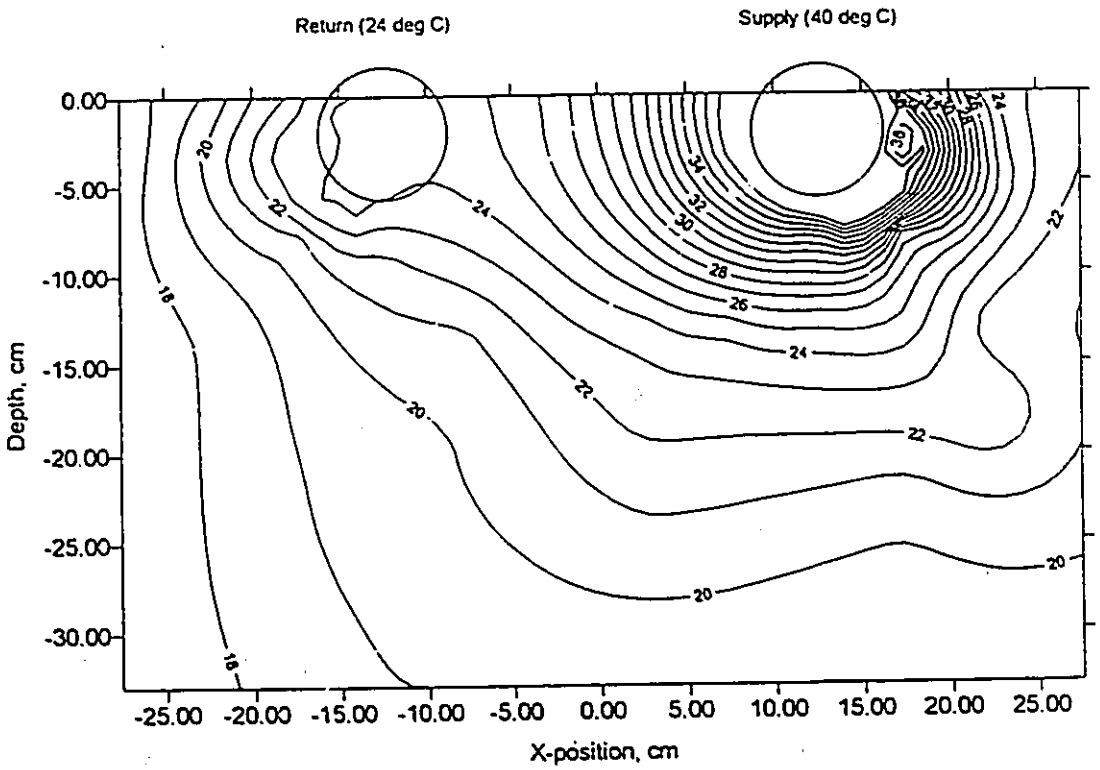

Figure 17: Isotherms from Experimental Data
3" System, Station A: December 26, 1994

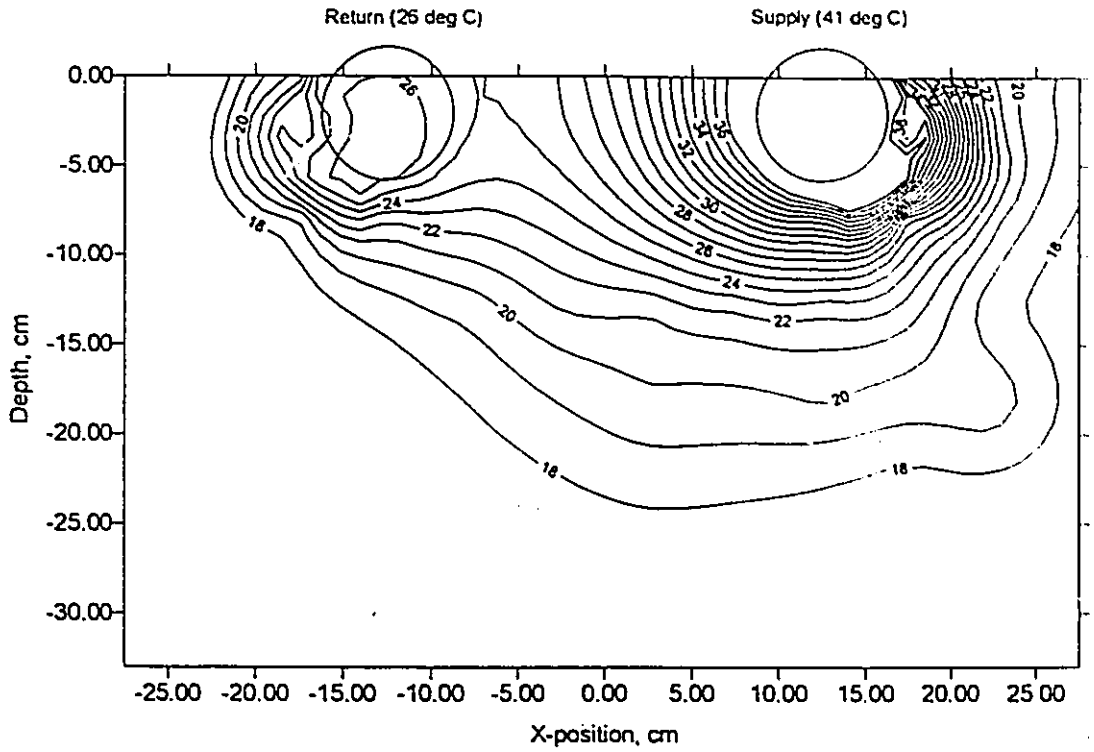

Figure 18: Isotherms from Experimental Data
3" System, Station A: February 16, 1995

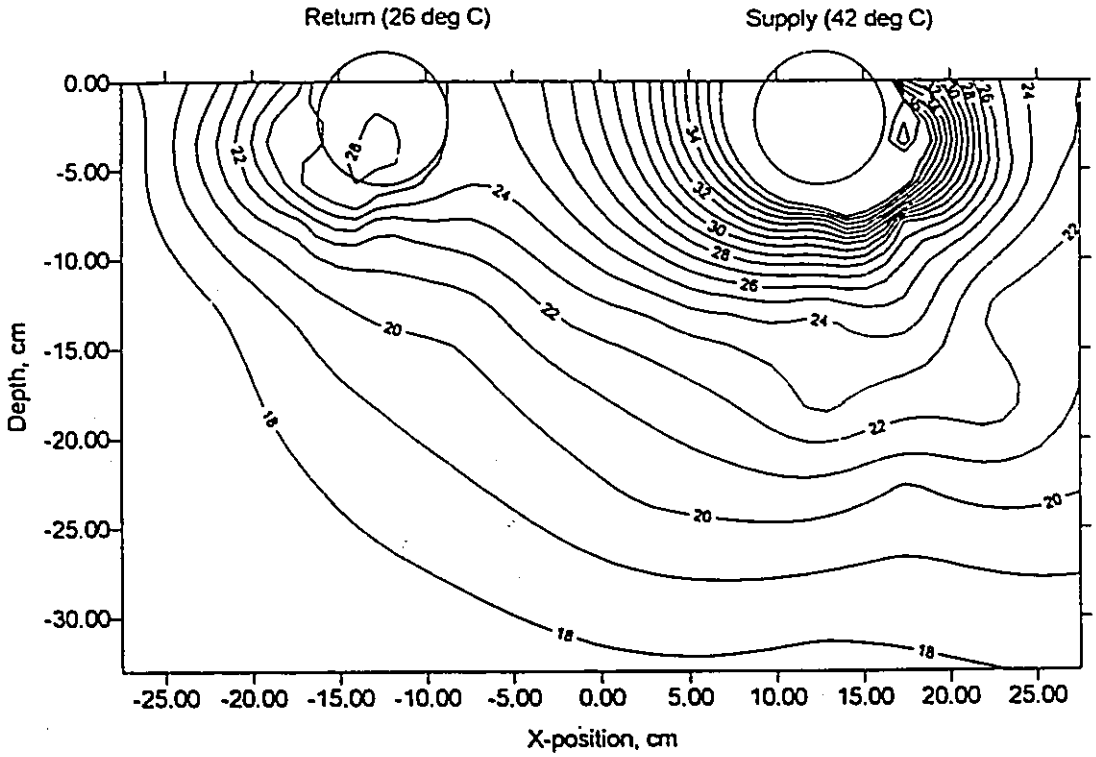

Figure 19: Isotherms from Experimental Data
3" System, Station A: April 3, 1995

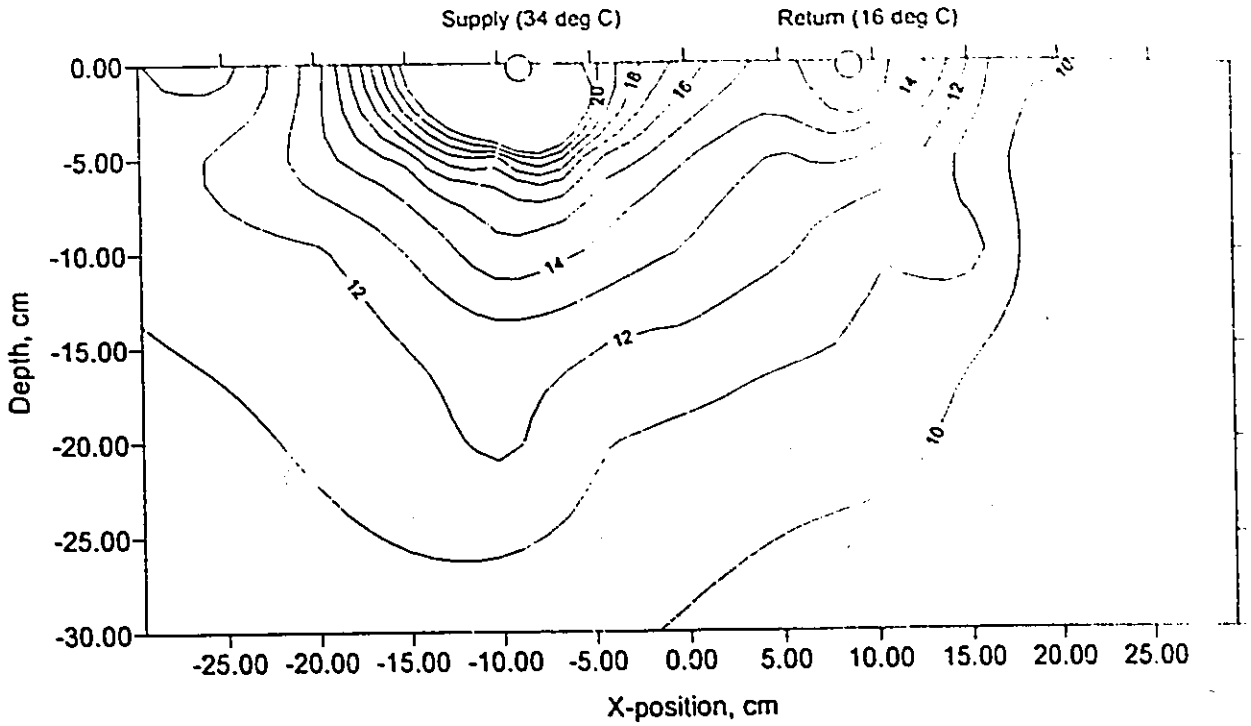

Figure 20: Isotherms from Experimental Data
1/2" System, Station B: March 6, 1995

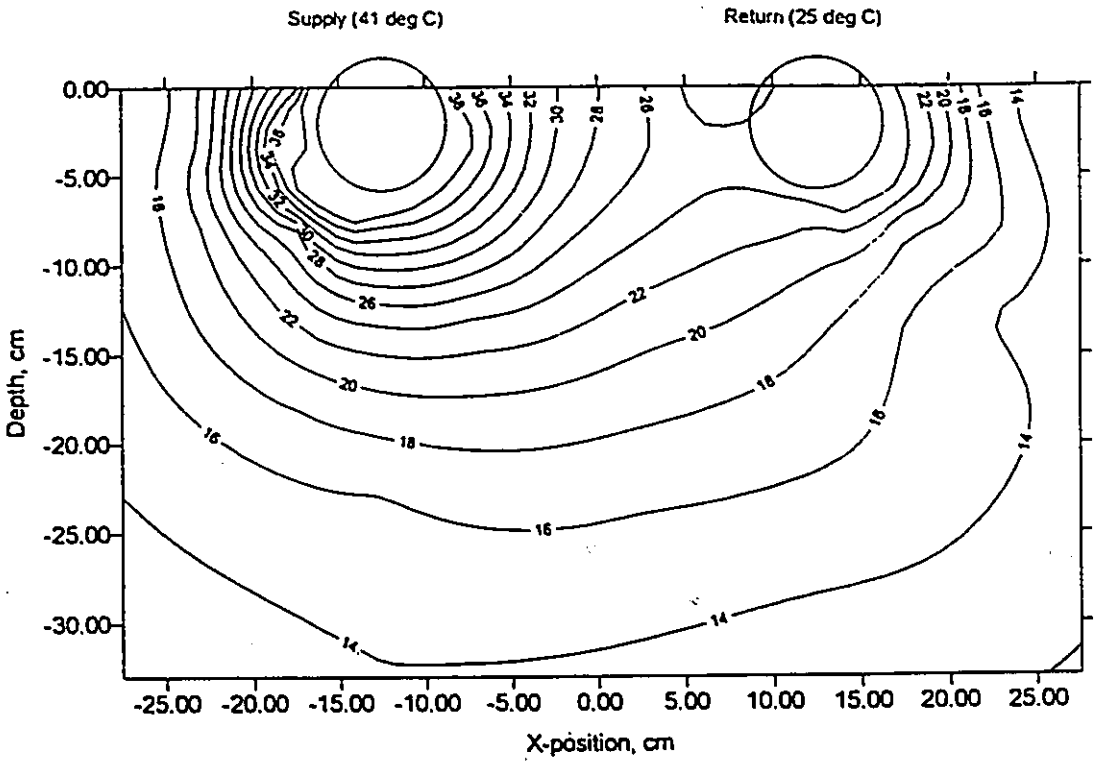

Figure 21: Isotherms from Experimental Data
3" System, Station B: March 6, 1995

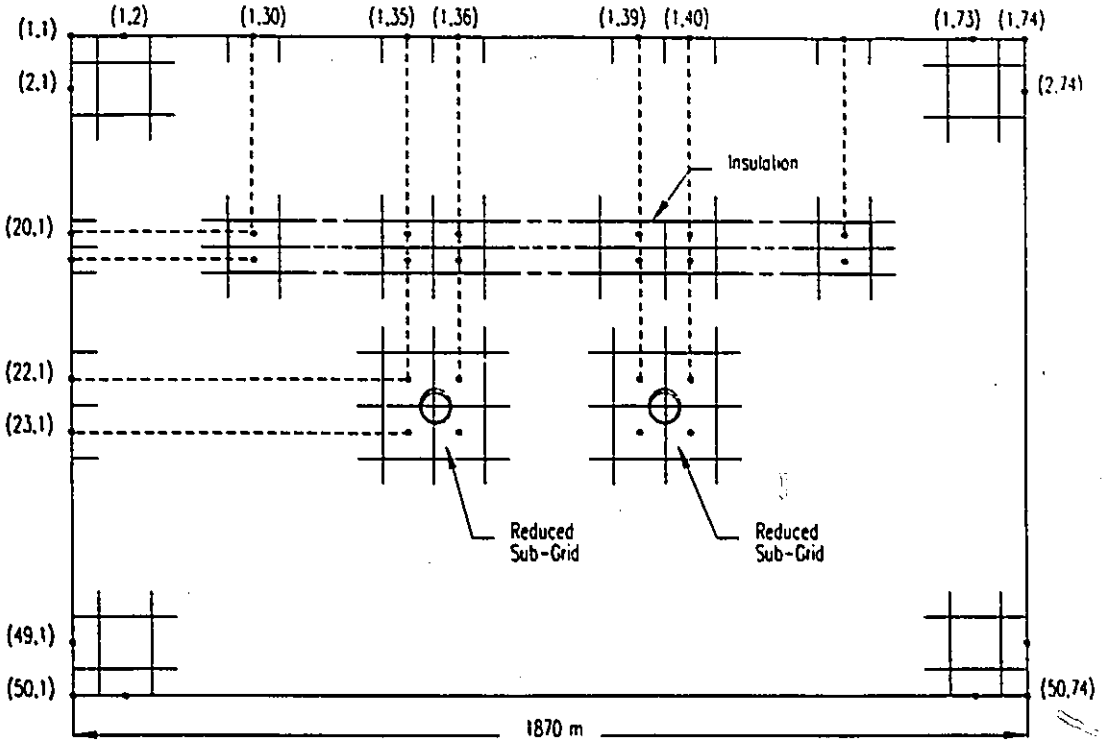

Figure 22: Model Grid for 1/2" Experimental System

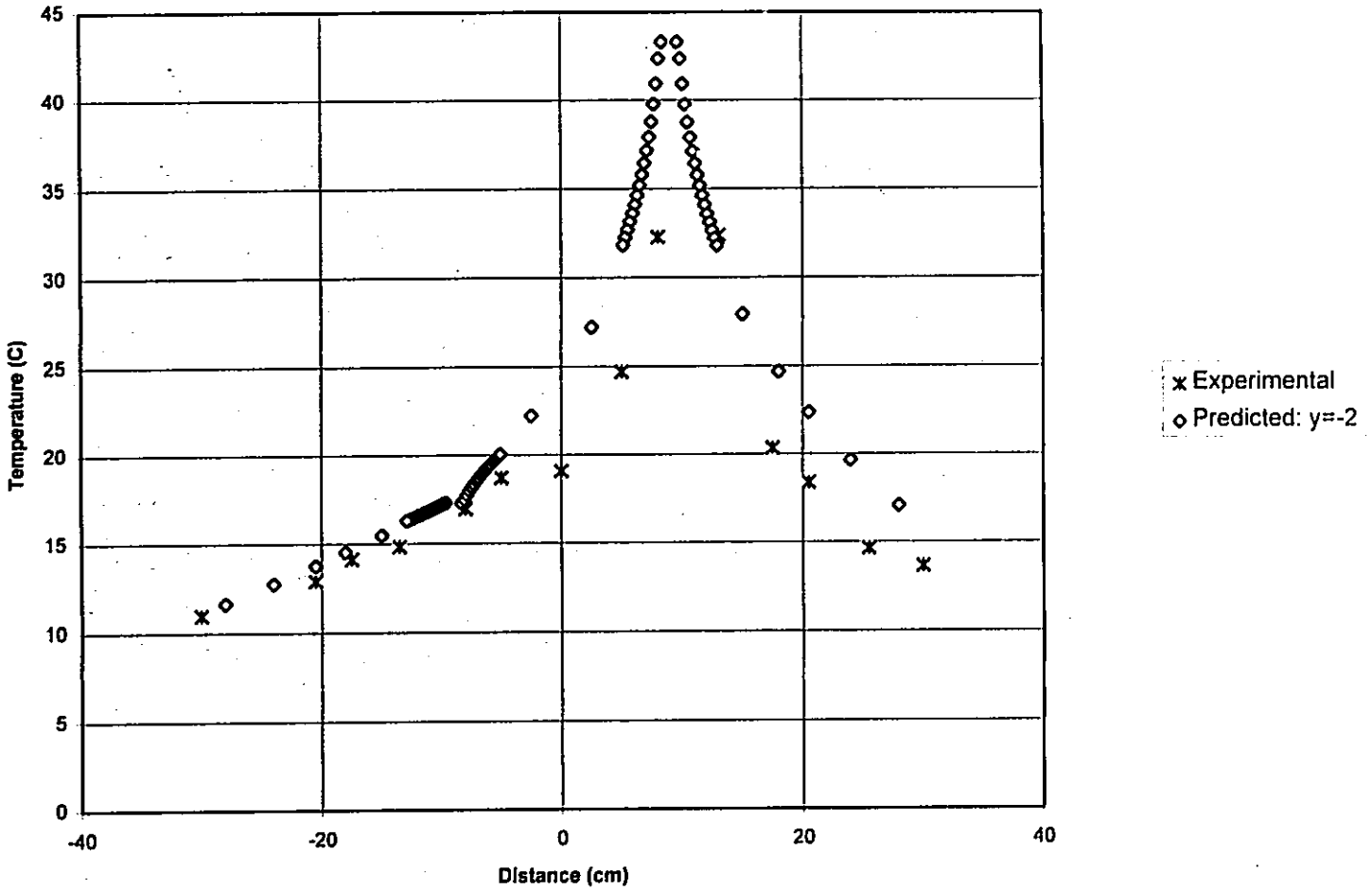

Figure 23: Comparison of Modeled and Experimental Temperatures: December 26, 1994, 1/2" System Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 1 degC, Treturn= 17.3 degC,Tsupply= 43.3 degC)

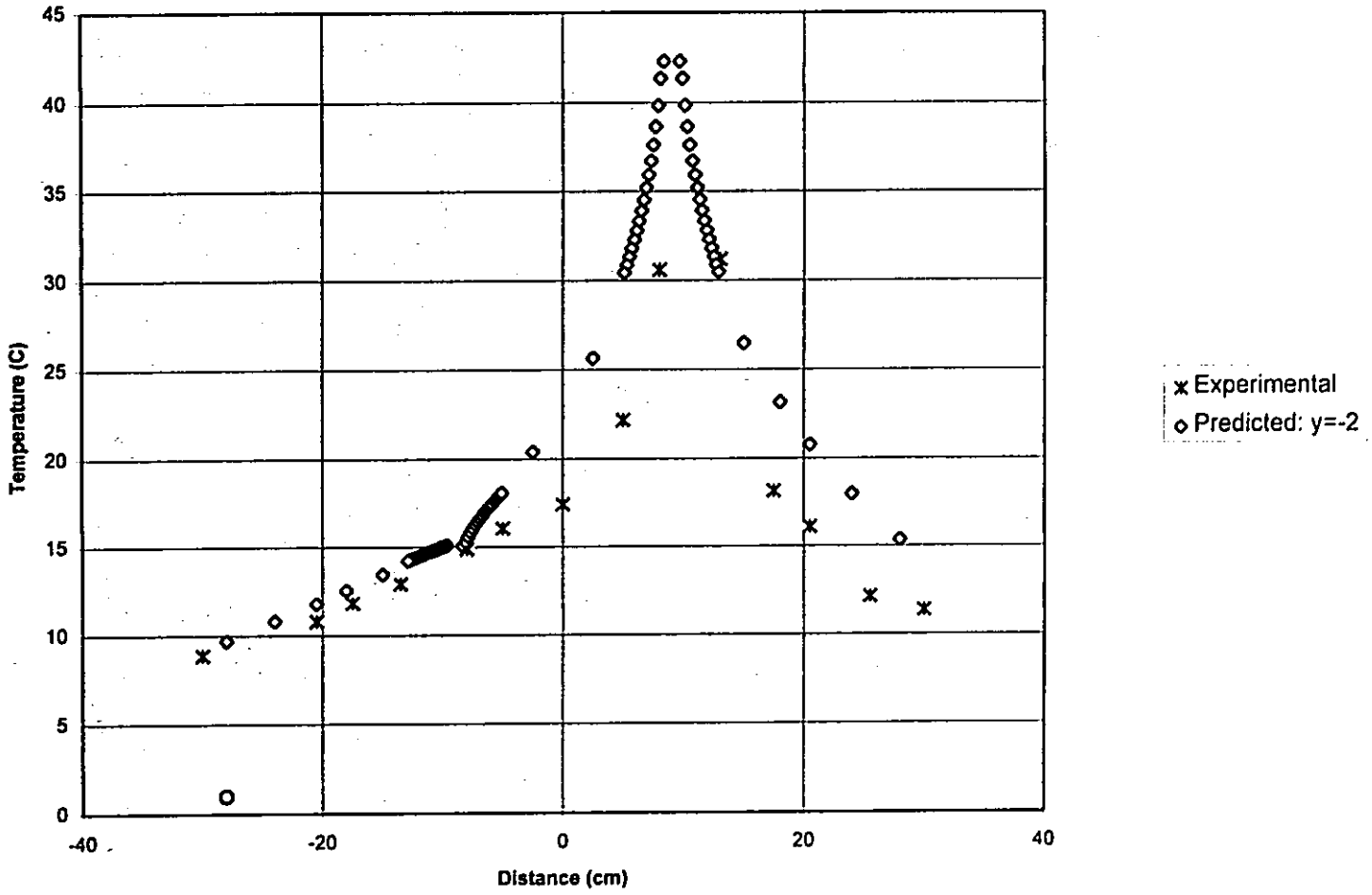

Figure 24: Comparison of Modeled and Experimental Temperatures: February 16, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= -1 degC, Treturn= 15.1 degC, Tsupply= 42.3 degC)

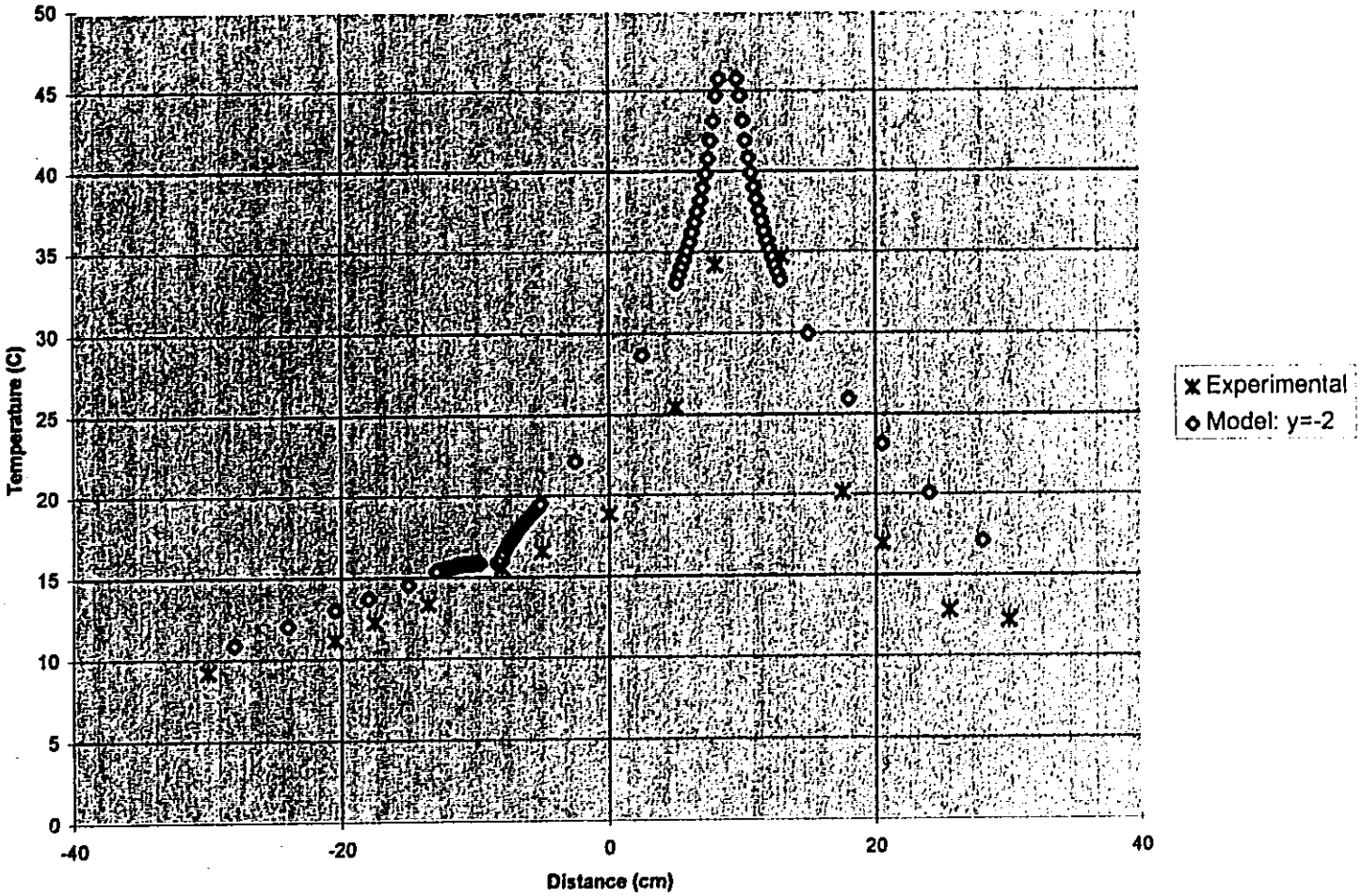

Figure 25: Comparison of Modeled and Experimental Temperatures: March 6, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 0.28 degC, Treturn= 15.87 degC, Tsupply= 45.76) degC

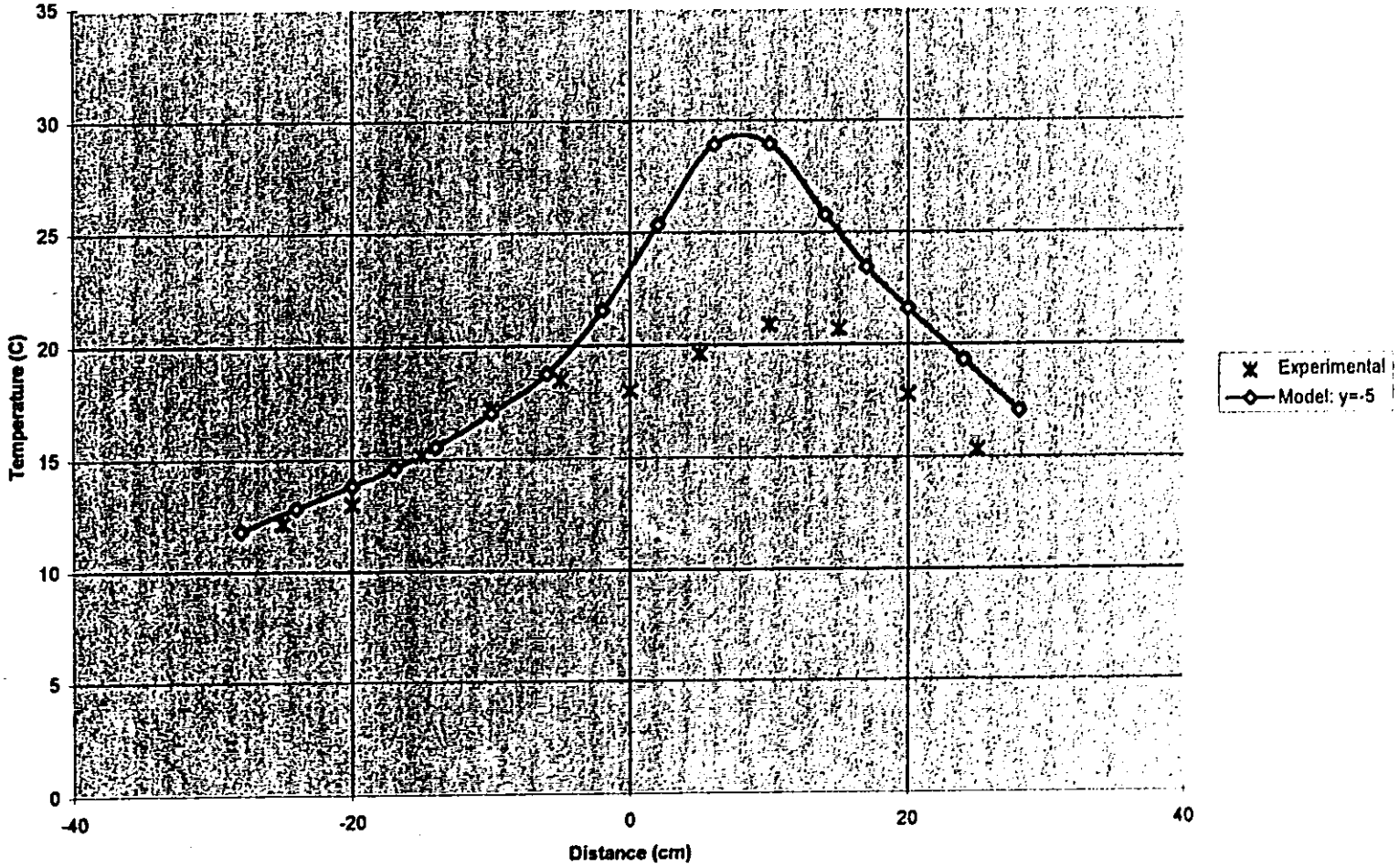

Figure 26: Comparison of Modeled and Experimental Temperatures: December 26, 1994, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 1 degC, Treturn= 17.3 degC, Tsupply= 43.3 degC)

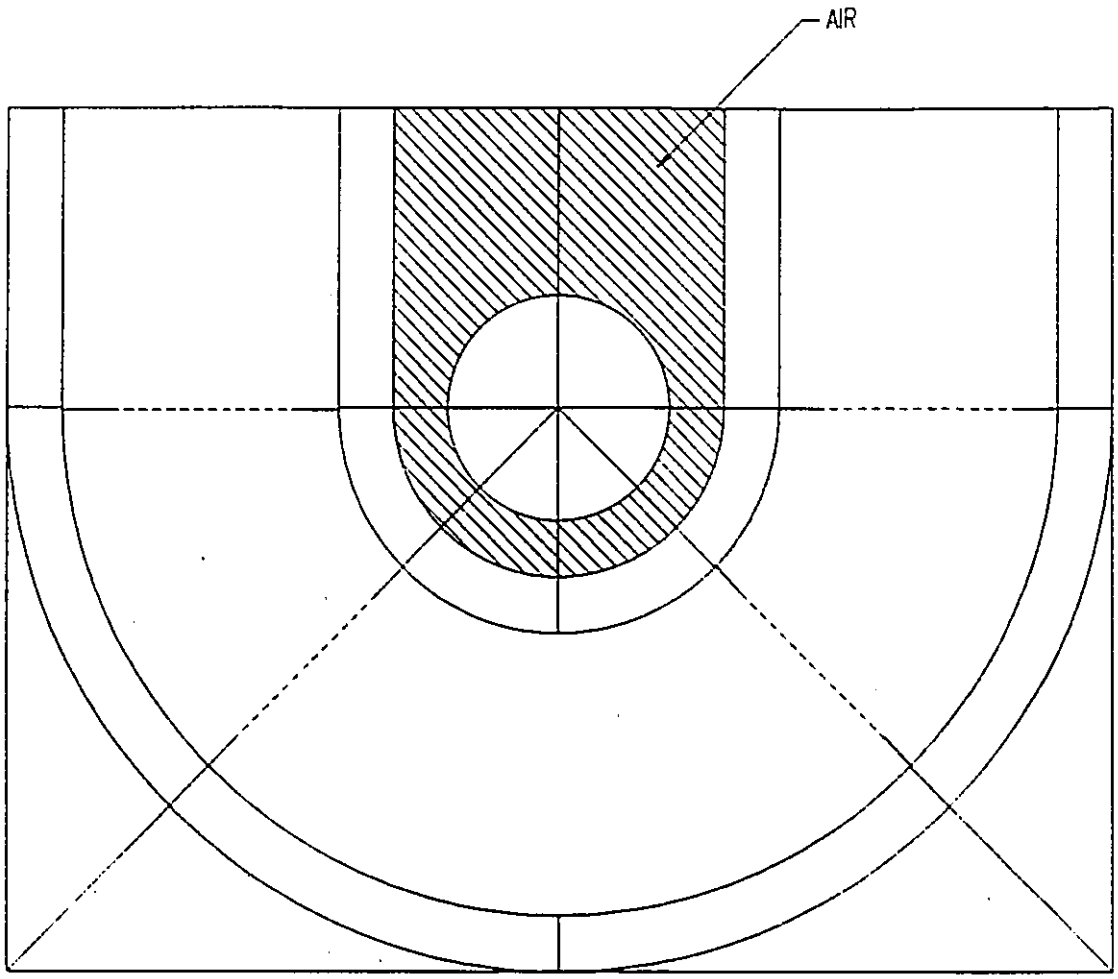

Figure 27: Air Gap Imposed Around Pipe: Reduced Sub-Grid Geometry

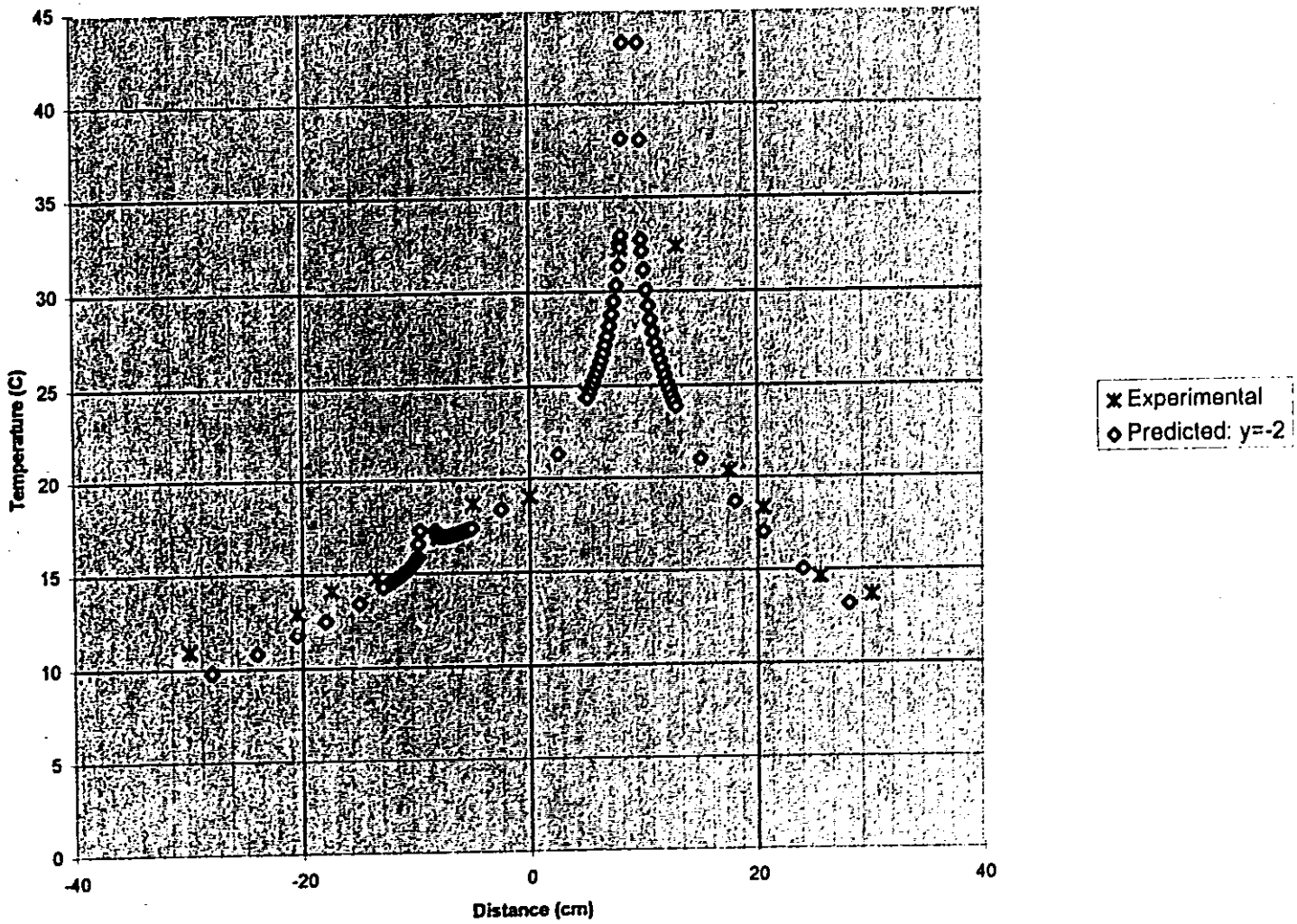

Figure 28: Comparison of Experimental and Modeled Temperatures: 0.035 cm Air Gap Around Modeled Pipes: December 26, 1994: 1/2" System. Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 1 degC, Treturn= 17.3 degC, Tsupply= 43.3 degC)

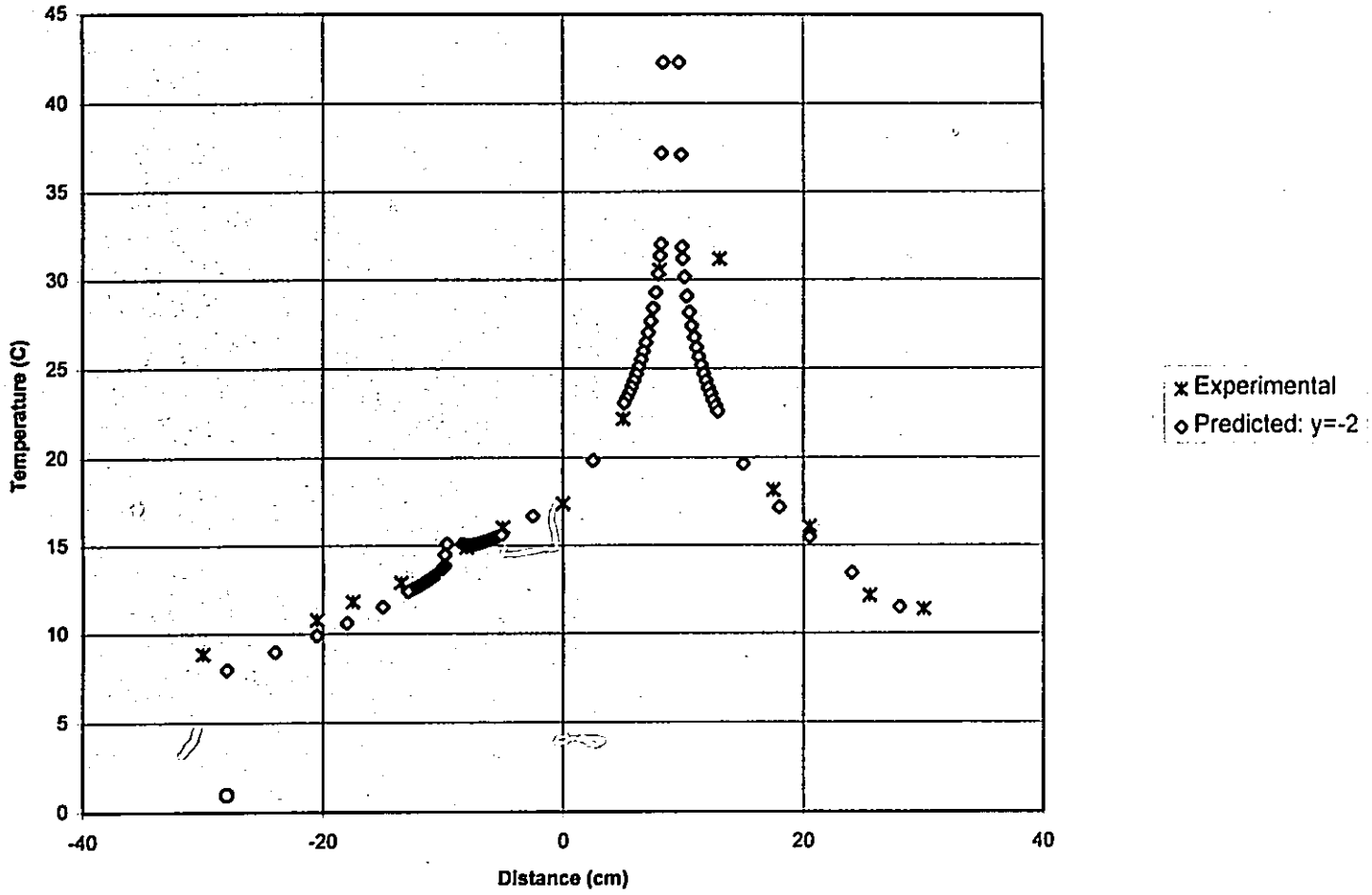

Figure 29: Comparison of Experimental and Modeled Temperatures: 0.035 cm Air Gap Around Modeled Pipes: February 16, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= -1 degC, Treturn= 15.1 degC, Tsupply= 42.3 degC)

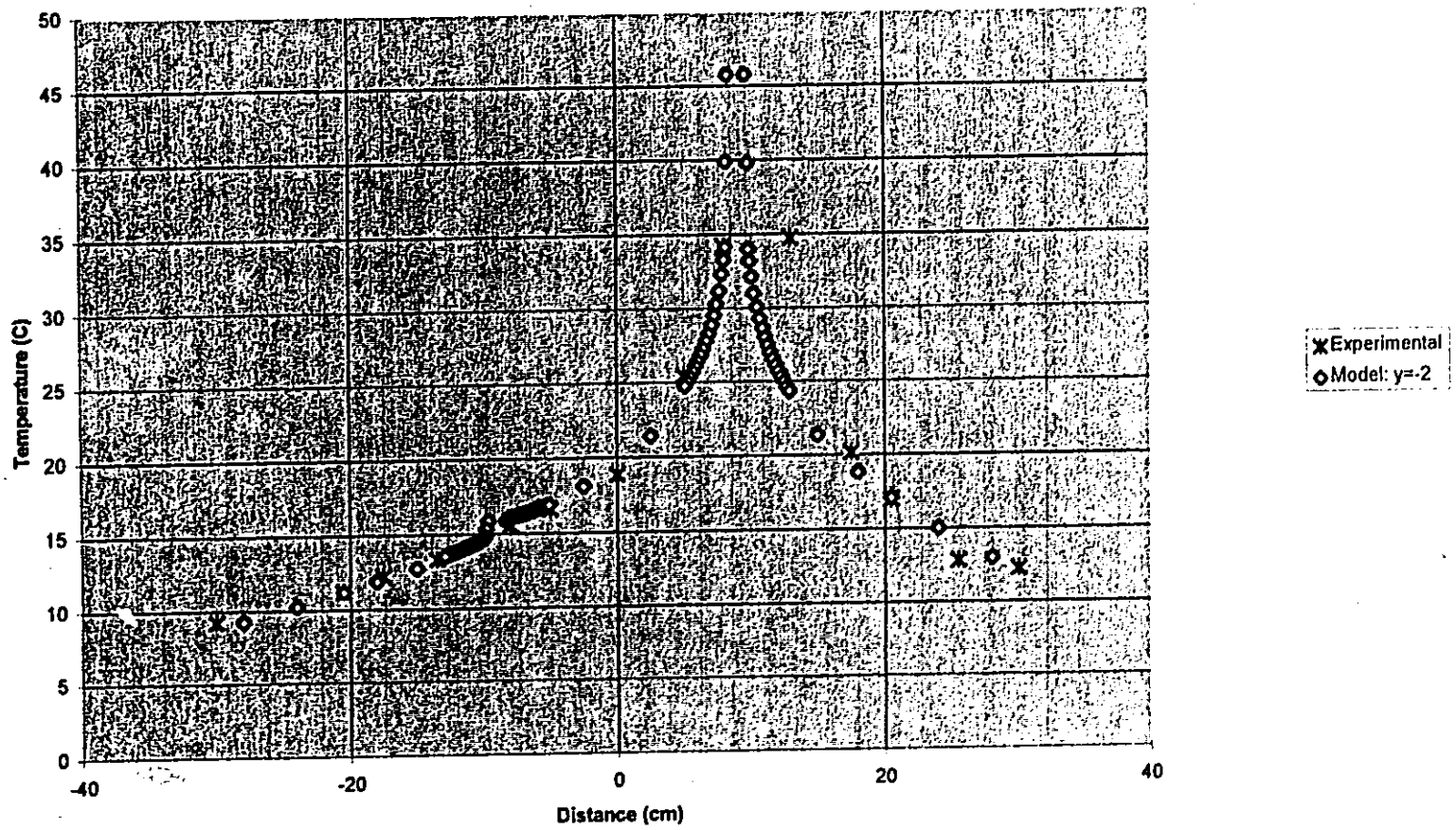

Figure 30: Comparison of Experimental and Modeled Temperatures: 0.035 cm Air Gap Around Modeled Pipes: March 6, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 0.28 degC, Treturn= 15.87 degC, Tsupply= 45.76) degC

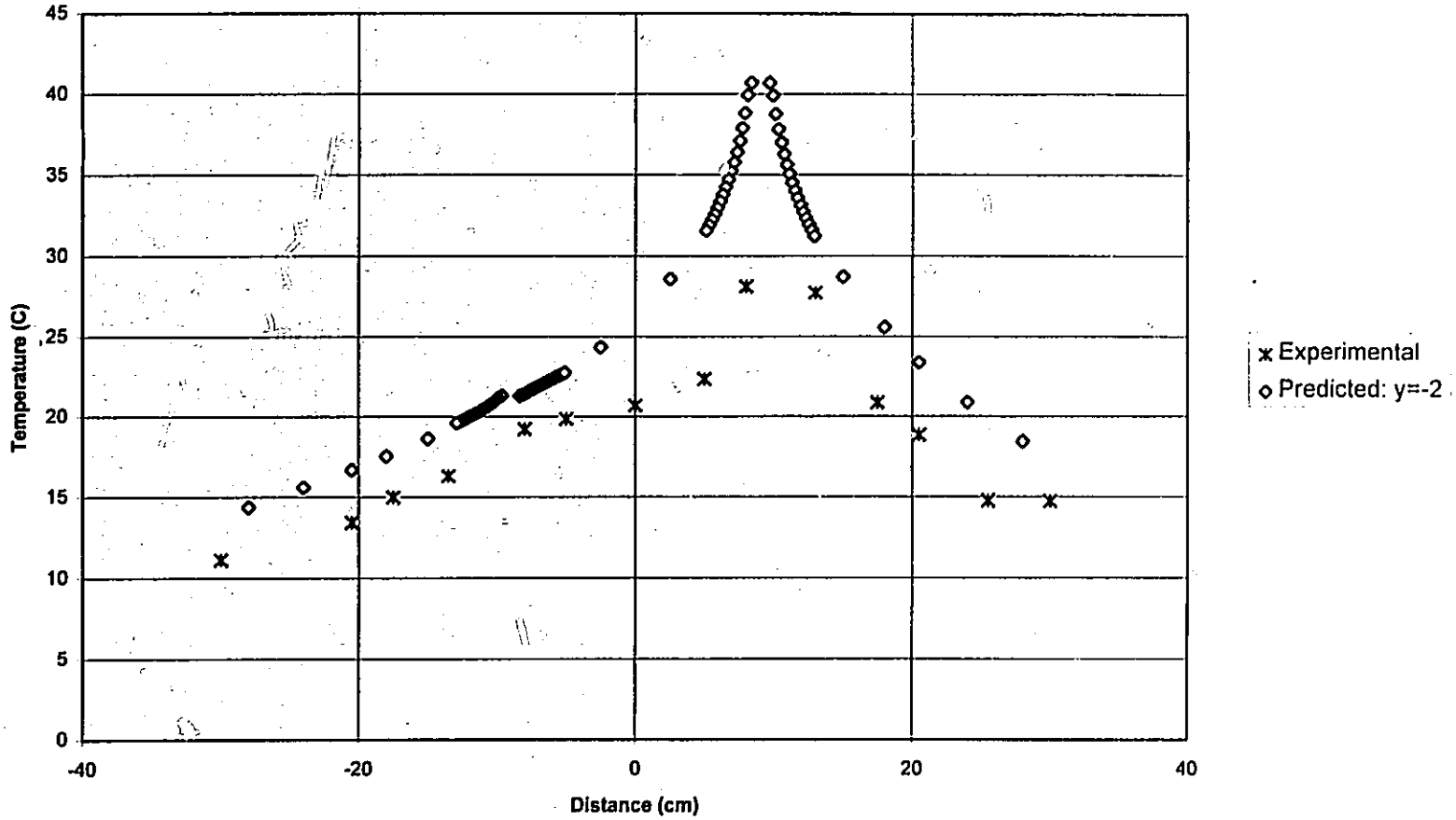

Figure 31: Comparison Of Experimental and Modeled Temperatures: March 22, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 4.5 degC, Treturn= 21.3 degC, Tsupply= 40.72 degC)

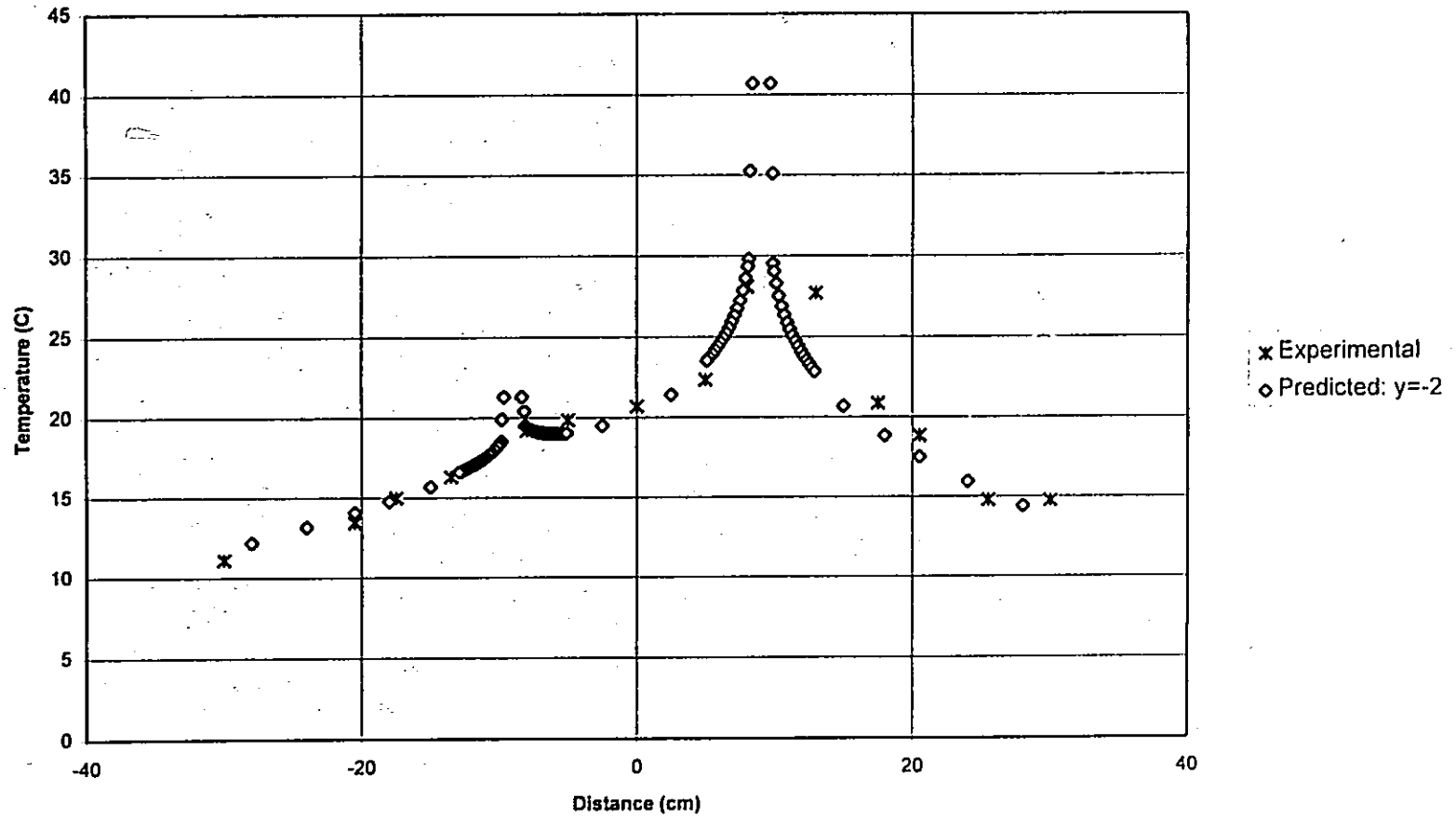

Figure 32: Comparison of Experimental and Modeled Temperatures: 0.035 cm Air Gap Around Modeled Pipes: March 22, 1995, 1/2" System, Station A ("Supply" pipe at +10 cm, "Return" pipe at -10 cm; Tgr= 4.5 degC, Treturn= 21.3 degC, Tsupply= 40.72 degC)

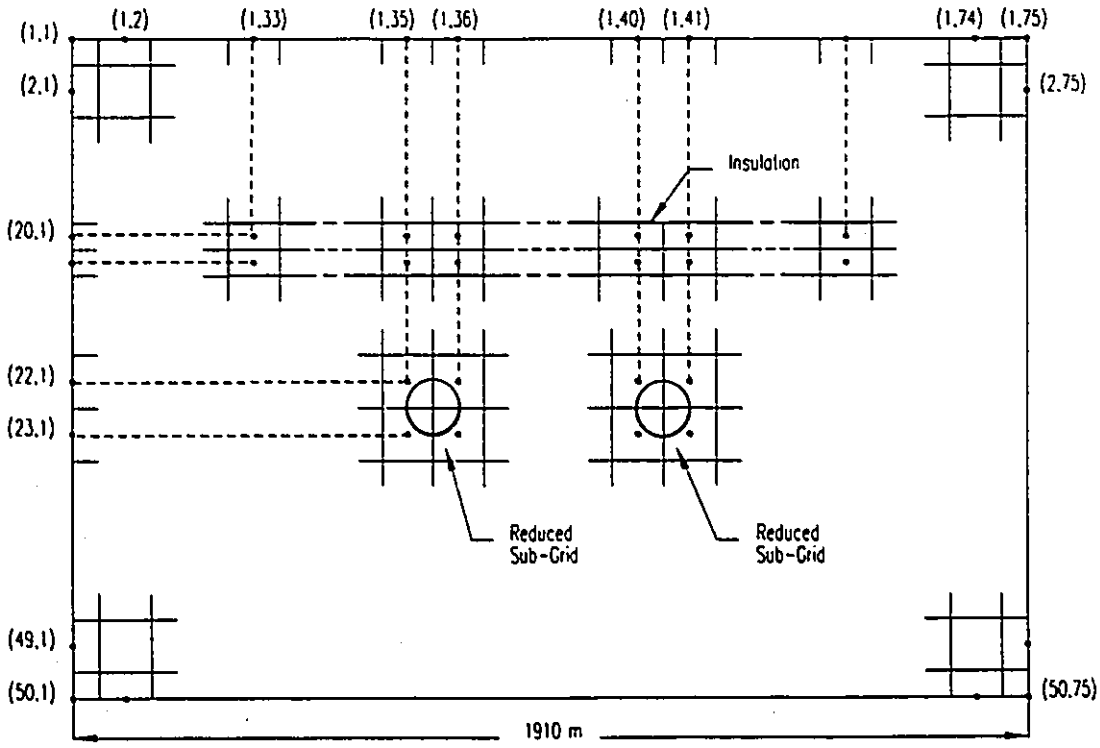

Figure 33: Model Grid for 3" Experimental System

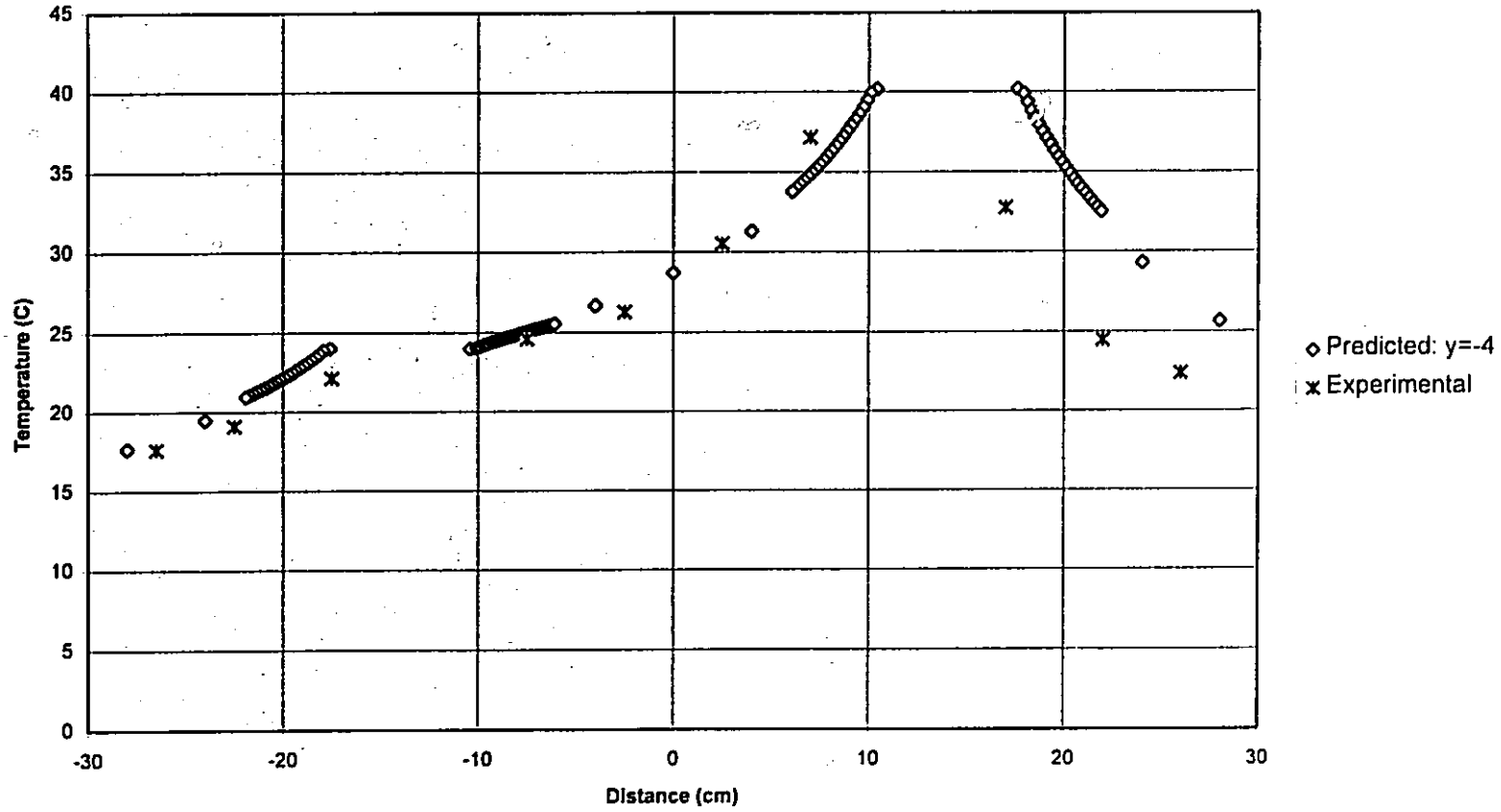

Figure 34: Comparison of Experimental and Modeled Temperatures: December 28, 1994, 3" System, Station A ("Supply" pipe at +14 cm, "Return" pipe at -14 cm; Tgr = 1 degC, Treturn = 24 degC, Tsupply = 40.2 degC)

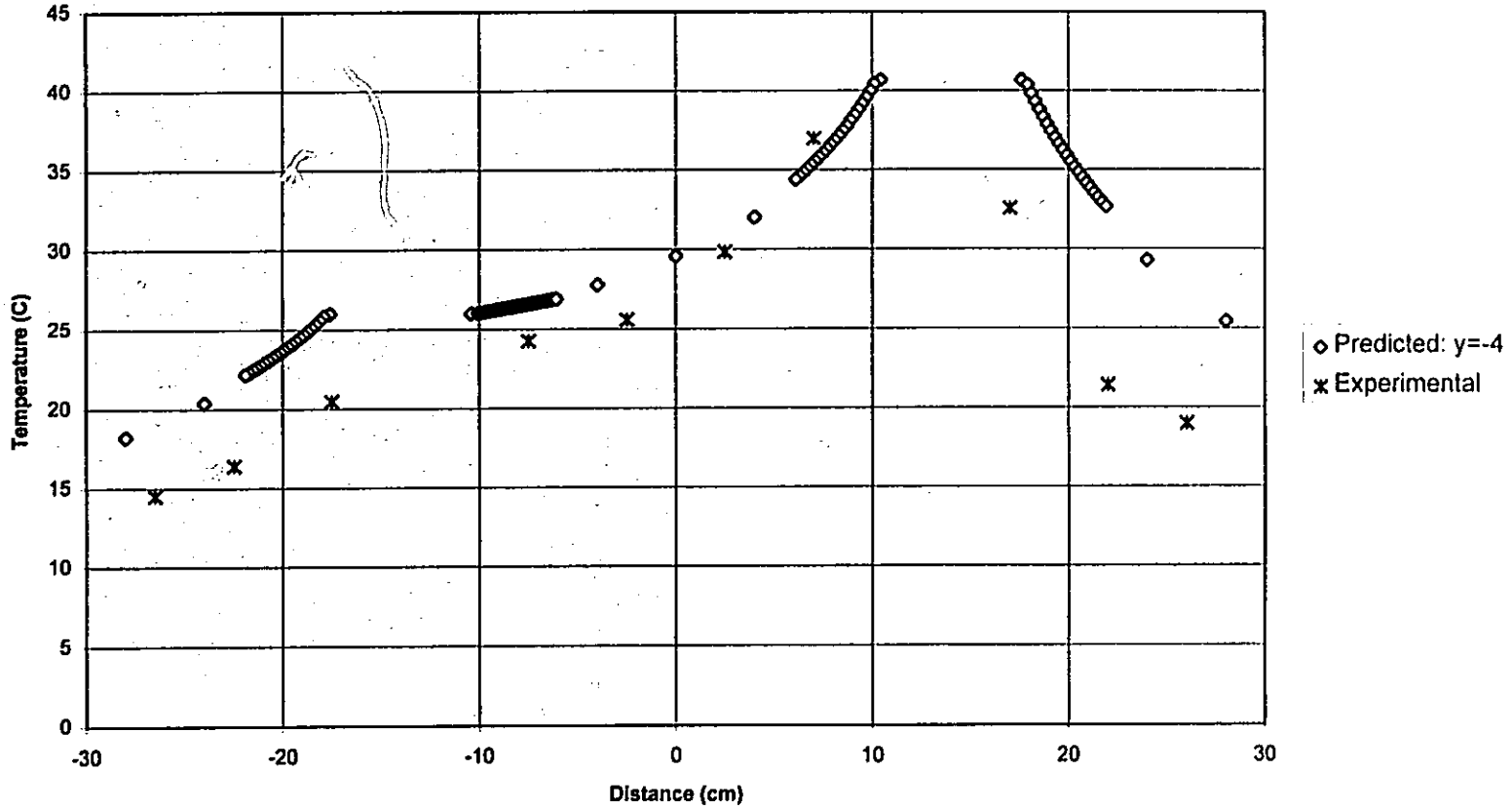

Figure 35: Comparison of Experimental and Modeled Tempertaures: February 16, 1995, 3" System, Station A
("Supply" pipe at +14 cm, "Return" pipe at -14 cm;Tgr = -1 degC,Treturn = 26 degC, Tsupply = 40.7 degC)

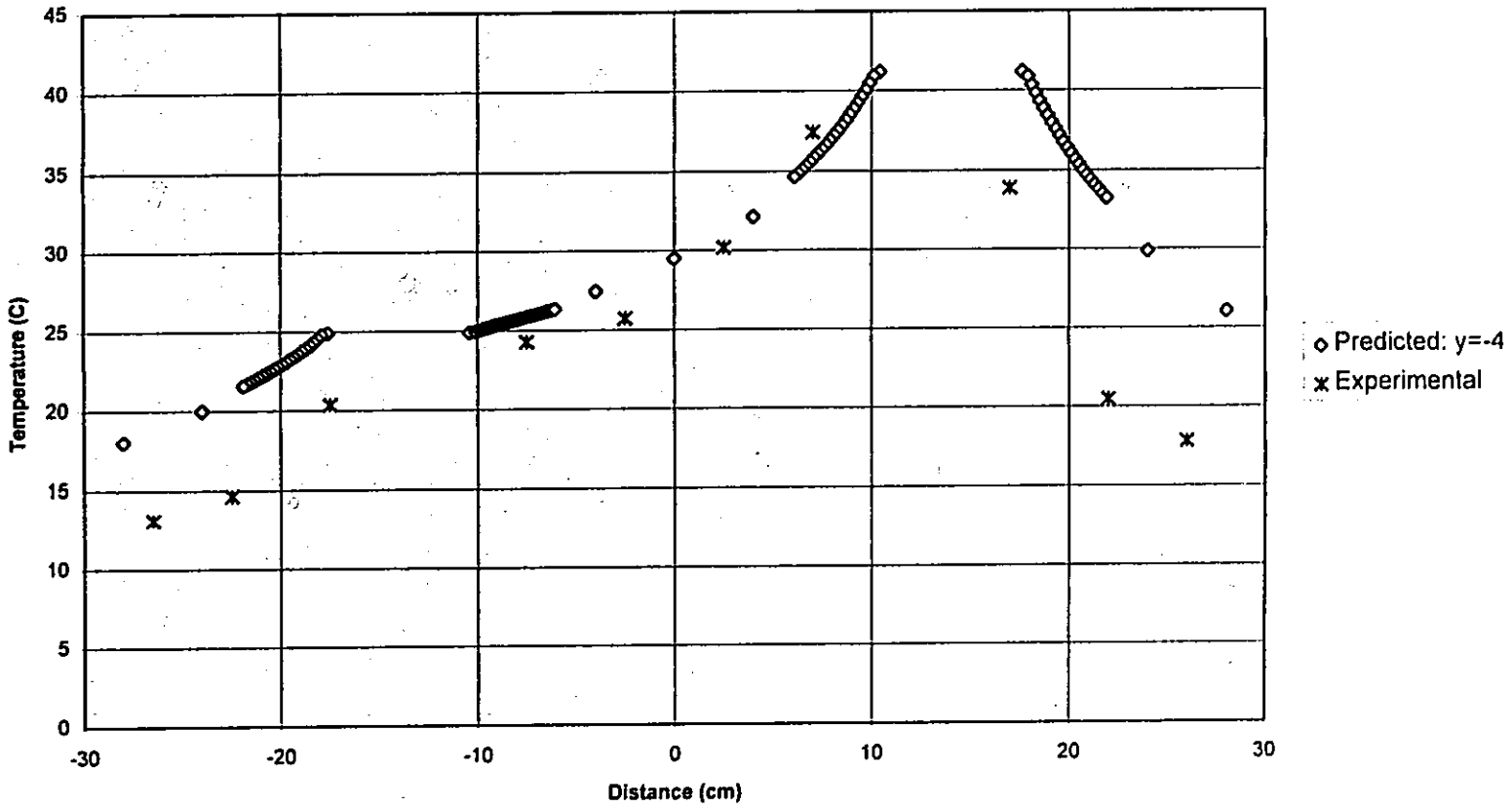

Figure 36: Comparison of Experimental and Modeled Temperatures: March 6, 1995, 3" System, Station A
("Supply" pipe at +14 cm, "Return" pipe at -14 cm;Tgr = 0.28 degC, Treturn = 24.9 degC, Tsupply = 41.2 degC)

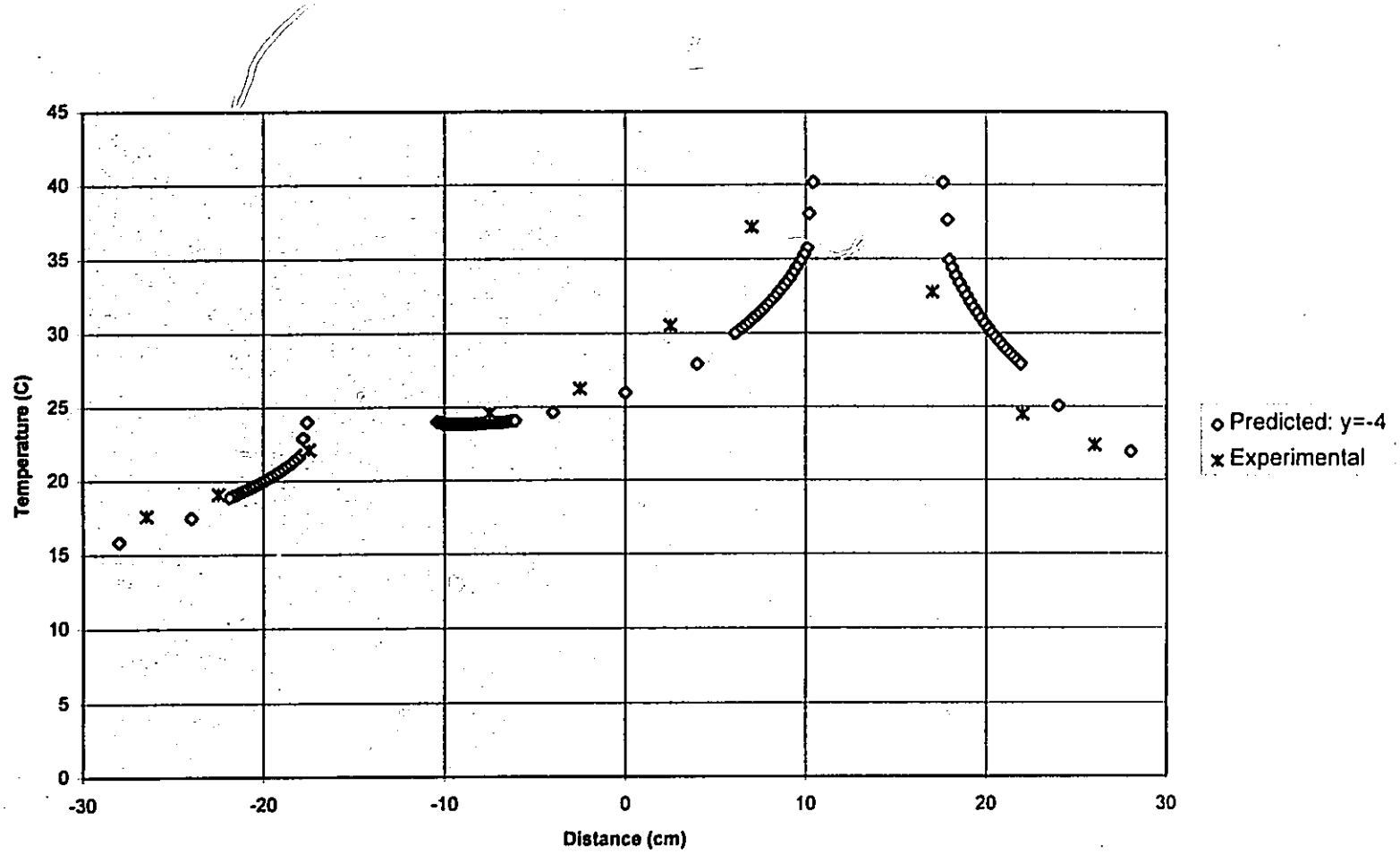

Figure 37: Comparison of Experimental and Modeled Temperatures: 0.05 cm Air Gap Around Modeled Pipes: December 26, 1994, 3" System, Station A ("Supply" pipe at +14 cm, "Return" pipe at -14 cm;Tgr = 1 degC,Treturn = 24 degC, Tsupply = 40.2 degC)

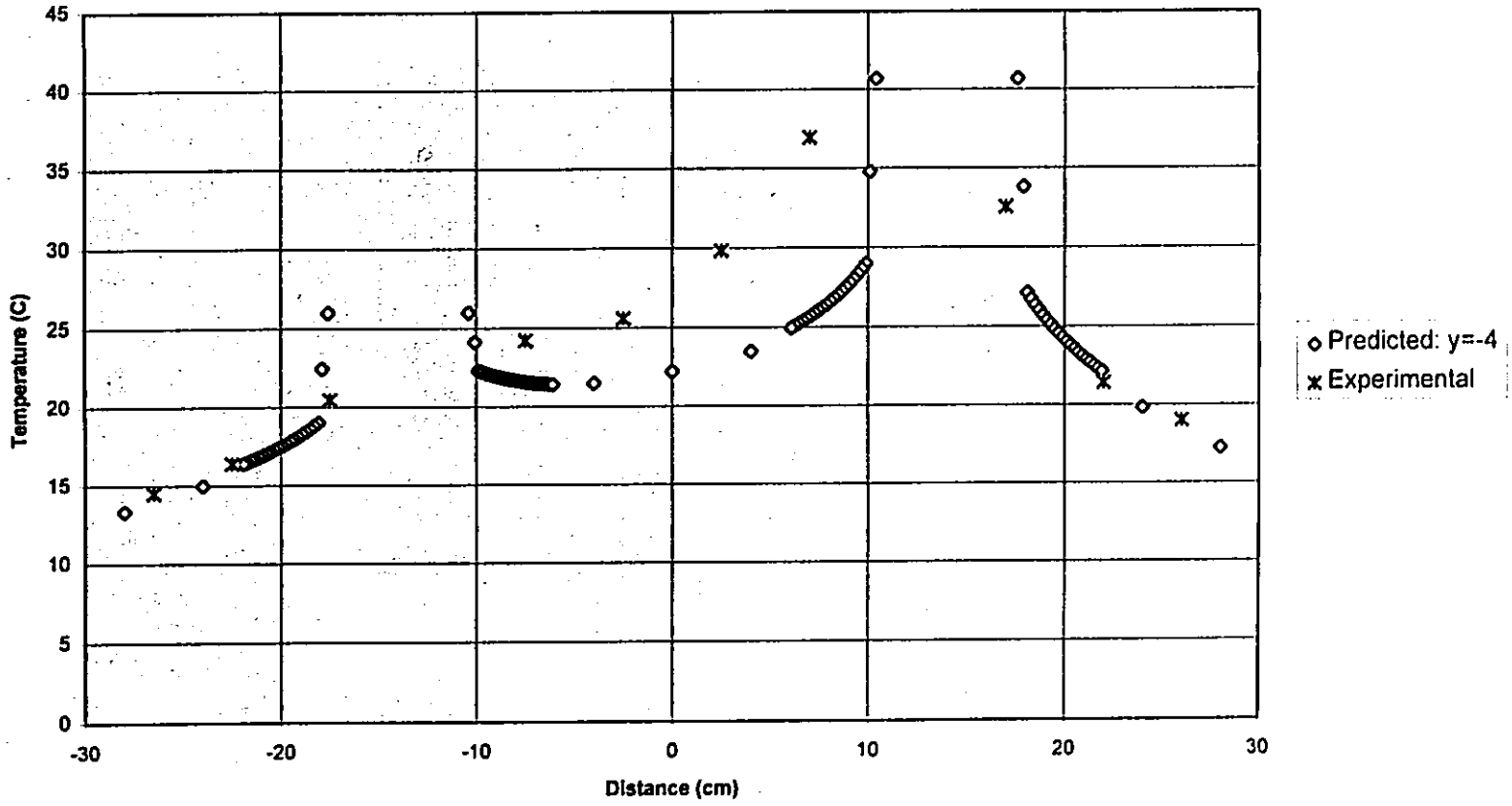

Figure 38: Comparison of Experimental and Modeled Temperatures: 0.20 cm Air Gap Around Modeled Pipes: February 16, 1995, 3" System, Station A ("Supply" pipe at +14 cm, "Return" pipe at -14 cm;Tgr = -1 degC, Treturn = 26 degC,Tsupply = 40.7 degC)

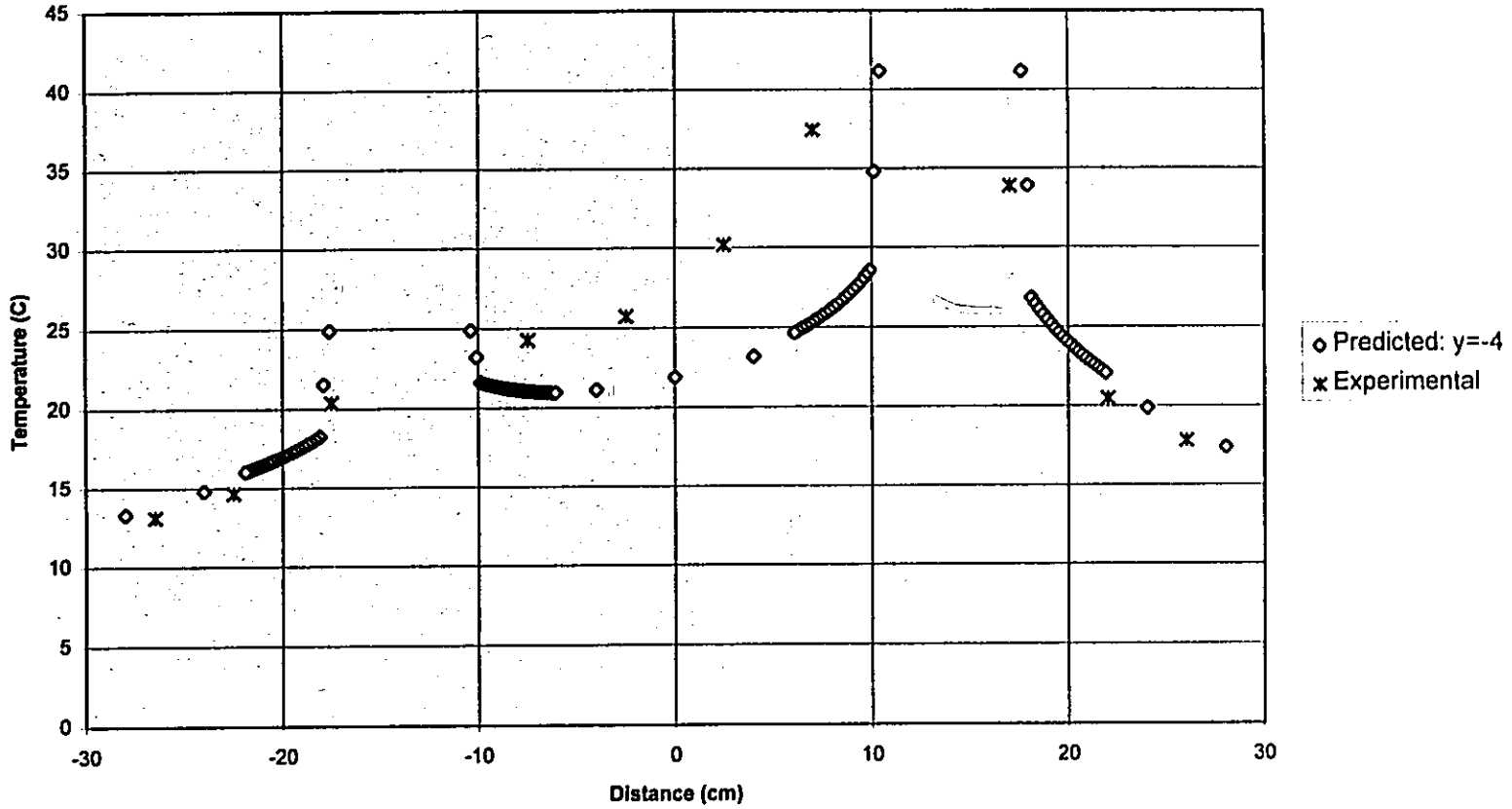

Figure 39: Comparison of Experimental and Modeled Temperatures: 0.20 cm Air Gap Around Modeled Pipes: March 6, 1995, 3" System, Station A ("Supply" pipe at +14 cm, "Return" pipe at -14 cm;Tgr = 0.28 degC,Treturn = 24.9 degC,Tsupply = 41.2 degC)

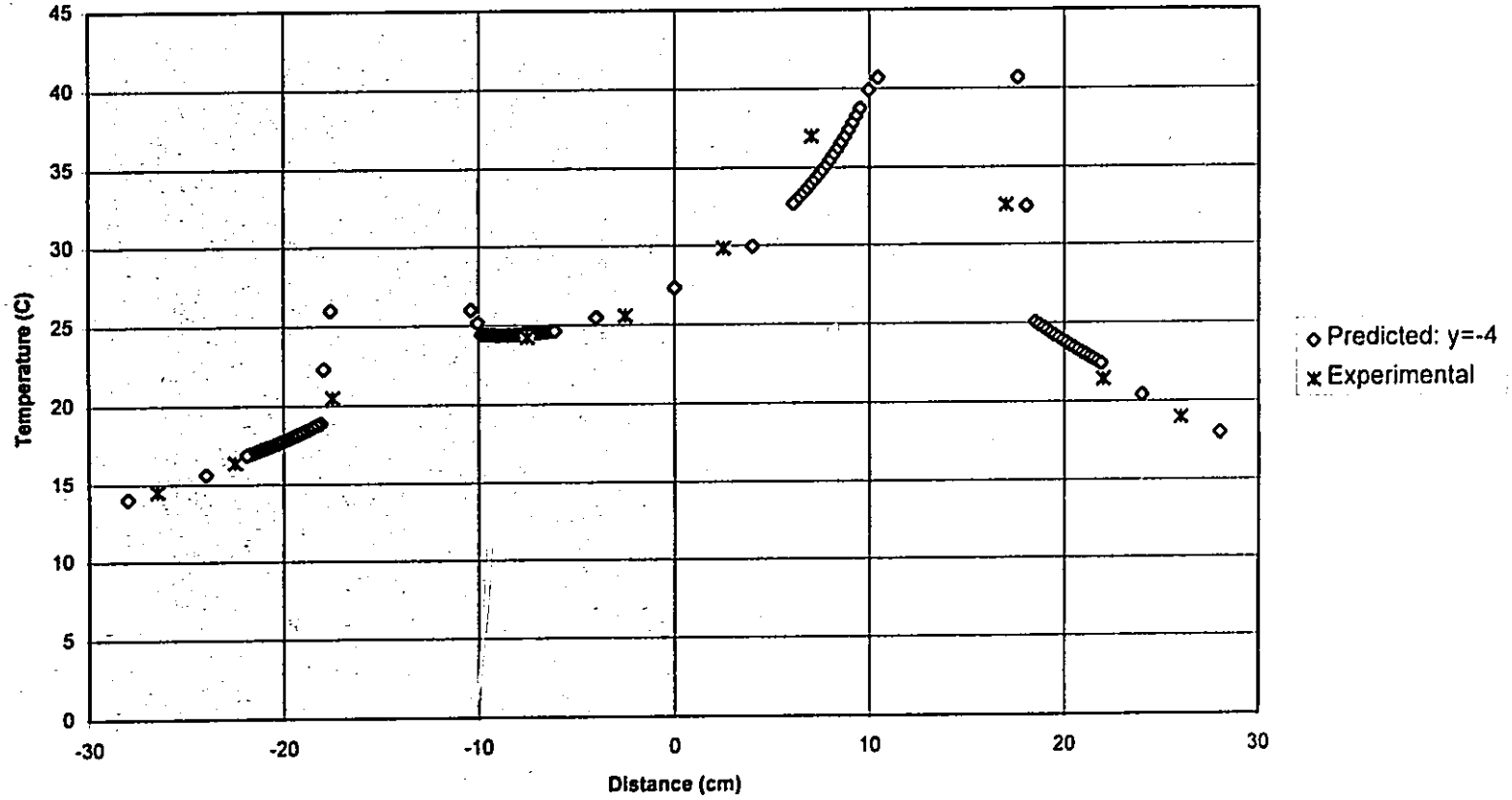

Figure 40: Comparison of Experimental and Modeled Temperatures: 0.30 cm Air Gap Around Return Pipe and 0.50 cm Around Right Half of Supply Pipe: December 26, 1994, 3" System, Station A (Tgr = -1 degC, Treturn = 26 degC,Tsupply = 40.7 degC)

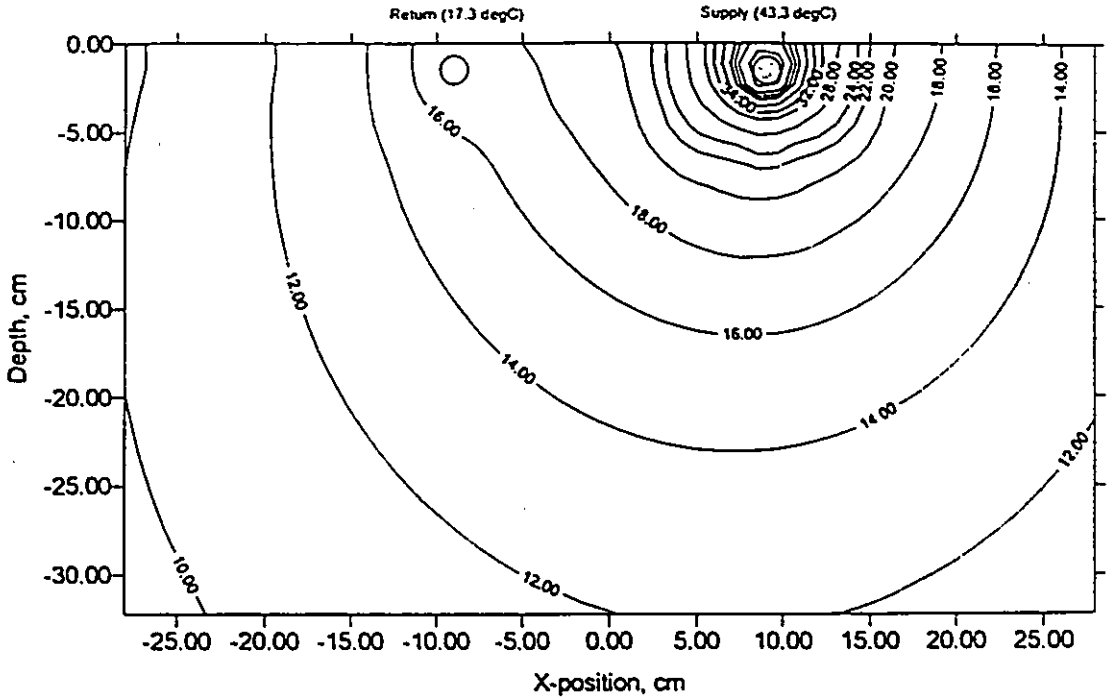

Figure 41: Isotherms from Modeled Data
1/2" System, Station A: December 26, 1994
0.035 cm Air Gap Imposed Around Pipes

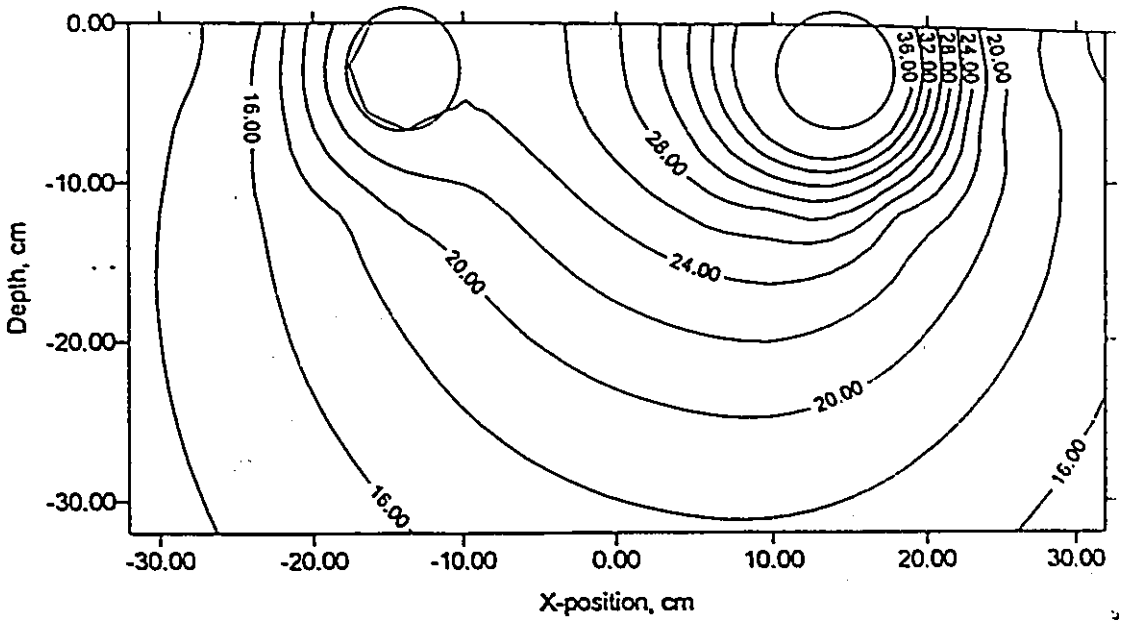

Figure 42: Isotherms from Model Data: 3" System, Station A
December 26, 1994. Air Gaps: 0.30 cm Around Return,
0.50 cm Around Right Half of Supply

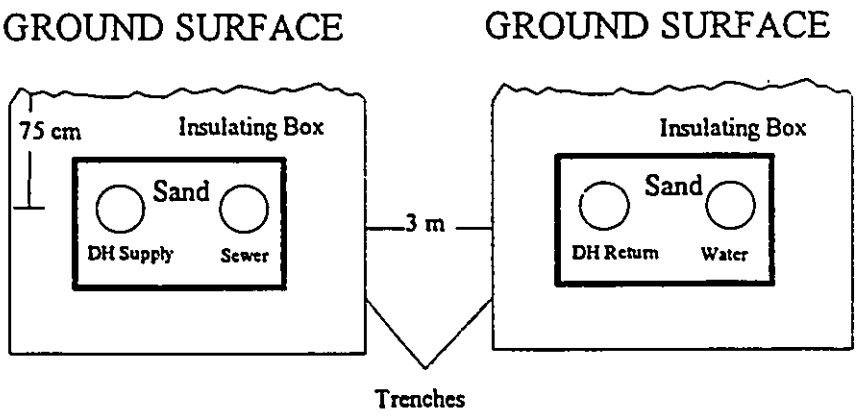

Figure 43: Trench Configuration for Integrated Services: Davis Inlet

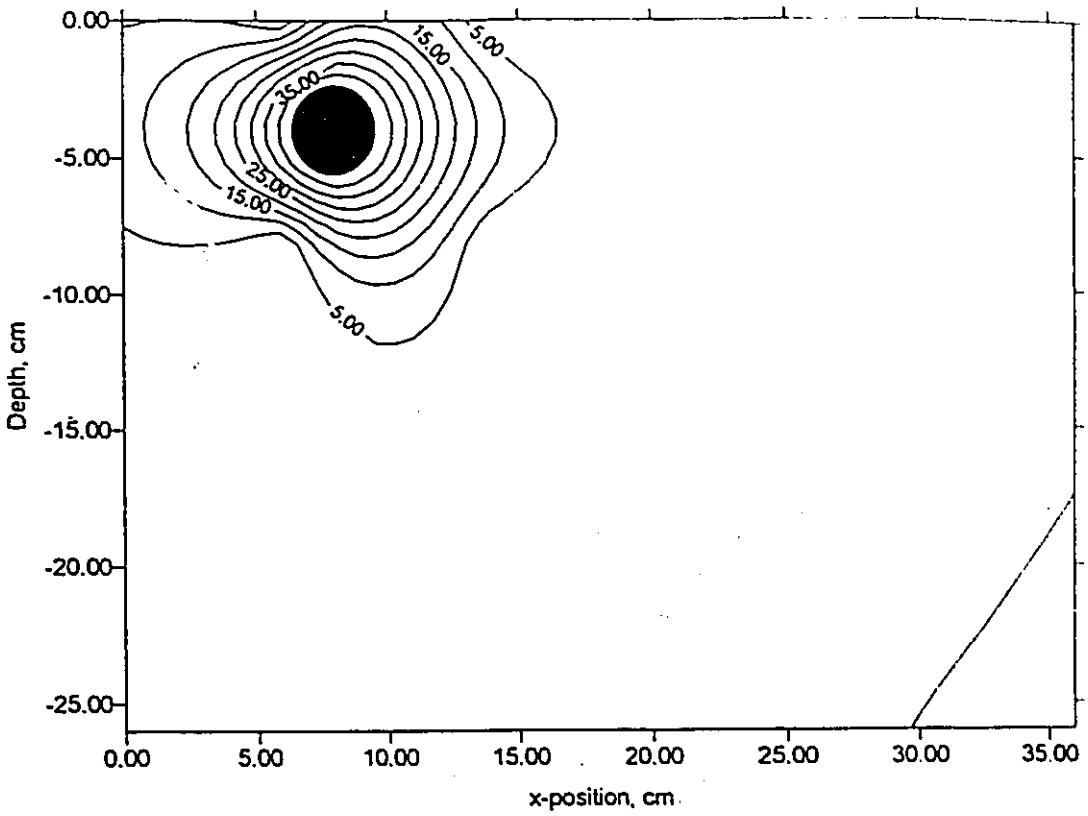

Figure 44: Isotherms from Predicted Data for Davis Inlet Design: Return DH Pipe (Water Pipe to be Placed Beside DH Pipe)

6.0 CONCLUSIONS

This paper has outlined the development of a numerical model which predicts the heat losses from, and the thermal field surrounding, buried pipelines in any configuration and under a wide variety of ground conditions. Verification of this model was presented, the results of which led to the conclusion that the model is reliable and can be used with confidence to design integrated heating, water and sewer services.

Accuracy of the model was demonstrated through comparison with an analytical solution. The predicted heat flow was within 1% of the analytical solution. The analytical solution was used as a benchmark to determine reasonable guidelines for designing a grid; these were given in Section 3.5.2 along with detailed specifications for a typical grid (Table 1). The model was proven to conserve energy throughout the calculation domain through comparison of the total heat flow at various points in the grid for numerous simulations of different configurations.

Further verification of the model was undertaken through comparison of predicted temperatures with experimental soil temperatures, taken around experimental pipe loops. The model predicted the experimental temperatures very well. The anomalies found between the experimental and predicted temperatures can be explained by the air gap model, which simulated settling and deficiencies in backfilling.

It was found that the use of an insulation board placed above the buried pipes, does create a warm thermal field between the pipes. The effectiveness of burying insulation boards around the pipes was further illustrated in Section 5.4 in the design for Davis Inlet.

In using the model, there are two main areas of uncertainty: selection of soil thermal conductivity, and the allowance to be made for settling, or poor compaction, of the sand via an air gap. The soil thermal conductivity affects the heat loss from the pipe(s) but not the thermal field. Imposing an air gap around the pipe(s) affects the thermal field as well as the heat loss.

The accuracy with which the model predicted the experimental temperatures and the consistency in trends exhibited by those predicted temperatures over a five month period, both in constant and in changing conditions, demonstrate the reliability of the model. The model can, therefore, be used with confidence to facilitate the design of integrated heating, water and sewer services. The primary motivation for this research stemmed from the financial and technical limitations associated with the installation of traditional municipal services in many northern communities. Integrated technologies, using waste heat from district heating pipes as freeze protection for municipal lines, all buried within one shallow trench, can provide an economically attractive, technically feasible alternative to these traditional, cost intensive services. The numerical model whose development and accuracy have been demonstrated in this report, will make this technology realizable.

For future work, it is recommended that the model be tested for new insulation geometries: that is, experiments should be done with new insulation geometries and the thermal field created tested against predicted temperatures. Some such insulation geometries might include a box of insulation entirely surrounding the pipe, an inverted U shape (insulation on the top and sides), or an inverted U with a piece of insulation separating the hot from the cool pipe. It is also recommended that more space be left between the pipes and the insulation board so that compaction of the sand around the pipes is better. This would minimize the air gap effect seen in the results of the present research.

7.0 REFERENCES

Agapkin, B.M., B.L. Krivoshein, B.A. Krasovitskii and V.P. Radchenko, "Nonstationary Thermal Regimes of Pipelines", Power Engineering (New York)(English Translation of Izvestiya Akademii Nauk SSSR, Energetica i Transport) **21**, 158-165 (1983).

"ASHRAE Handbook 1981 Fundamentals", American Society of Heating, Refrigerating and Air Conditioning Engineers, Inc., Atlanta, Georgia (1982).

Babus'Haq, R.F., S.D. Probert, M.J. Shilston and A. Talati,"Suggested Design Improvements Concerning District Heating Pipeline Configurations", Applied Energy **17**, 77-96 (1984).

Babus'Haq, R.F., S.D. Probert and M.J. Shilston, "Natural Convection Across Cavities: Design Advice", Applied Energy **20**, 161-188 (1985).

Babus'Haq, R.F., S.D. Probert and M.J. Shilston, "Improved Configurations for District-Cooling Pipelines", Applied Energy **16**, 67-76 (1984).

Battara, V., O. Mariani and M. Vigianni, " Heat Exchange in Buried Pipelines Considering Seasonal Temperature Oscillations", Heat and Technology **5**, 84-100 (1987).

Bau, Haim H. and S.S. Sadhal, "Heat Losses From a Fluid Flowing in a Buried Pipe", Int. J. Heat Mass Transfer **25**, 1621-1629 (1982).

Bøhm, Benny, "Efficiency of District Heating Networks Derived From Instantaneous Heat Loss Measurements", Official Proceedings of the International District Heating and Cooling Association, Danvers, Massachusetts, June 13-17 (1992).

Coulter, D.M., *Location Studies for Slurry Pipelines - The Effect of Ground Surface Temperatures*, U.S. Dept of Commerce, Nat. Tech. Info. Service PB-287 336 (U.S. National Science Foundation Technical Memorandum 14, Cryogenic Fuel Transmission Project) (1976).

Davis Engineering & Associates Limited and R.J. Burnside & Associates Limited, "Feasibility Study - Municipal Servicing - Little Sango Pond", Davis Engineering & Associates Limited and R.J. Burnside & Associates Limited, Newfoundland (1994).

Dyachuck, R.P., V.E. Kornilov and A.V. Furman, "Thermal Design of Buried Pipelines", Power Engineering (New York)(English Translation of Izvestiya Akademii Nauk SSSR, Energetica i Transport) **19**, 34-38 (1981).

Eckert, E.R.G. and Drake, R.M., "Analysis of Heat and Mass Transfer", McGraw-Hill, New York, (1972).

Hadvig, Sven, "Transmission of Heat Using Hot Water Pipes", Lectures for course held at Thermal Energy Storage Meeting, Ispra, Italy, p. 301-349 (1982).

Holman, J.P., "Heat Transfer - Seventh Edition", McGraw-Hill, Toronto (1990).

Hooker, R.P. and W.E. Brigham, "Temperature and Heat Transfer Along Buried Liquids Pipelines", Journal of Petroleum Technology **30**, 747-749 (1978).

Huovilainen, R., "Factors Affecting District Heating Line Heat Losses", ASHRAE Winter Meeting, Chicago, Jan. 1985, Paper CH-85-10 No. 2 (also ASHRAE Transactions **91 Pt. 1**, (1985)).

Kersten, M.S., "Final Report: Laboratory Research for the Determination of the Thermal Properties of Soils", US Army CRREL, St. Paul, Minnesota (1949).

Krivoshein, B.L. and V.M. Agapkin, "Unsteady Heat Losses of Underground Pipelines", Journal of Engineering Physics **33**, 972-977 (1977) (translation of Inzhenero-Fizicheskii Zhurnal **33**, 339-345 (1976)).

Krivoshein, B.L. and V.N. Novakovskii, "Transient Thermal Processes in Long Gas Pipelines", Heat Transfer - Sov. Res. (USSR) **7**, 134-144 (1975).

Kusuda, T., "Heat Transfer Studies of Underground Chilled Water and Heat Distribution Systems", Symposium on Underground Heat and Chilled Water Distribution Systems, Washington, D.C., pp. 18-41 (1973).

MacDavid, Jacob H., "Buried Steam Line Temperature and Heat Loss Calculation", Proceedings of the SPIE - The International Society for Optical Engineering **1467**, 11-17 (1991).

Macuga, W., Fenwell Electronics Incorporated, Personal Communication (1994).

Naylor, D., H.M. Badr and J.T. Tarasuk, "Experimental and Numerical Study of Natural Convection Between Two Eccentric Tubes", Int. J. Heat Mass Transfer **32**, 171-181 (1989).

Novakovskii, V.N., "An Analytic Method for Calculating Nonstationary Heat exchange in Soil with a Pipeline", Power Engineering (New York)(English Translation of Izvestiya Akademii Nauk SSSR, Energetica i Transport) **23**, 139-146 (1985).

Patankar, S.V., "Numerical Heat Transfer and Fluid Flow", Hemisphere, New York (1980).

Phetteplace, Gary, US Army Cold Regions Research and Engineering Laboratory, Personal Communication (1994).

Phetteplace, Gary, "Measurement of Heat Losses From Hot Water Distribution Systems", Proc. 83rd Annual Conf. International District Heating and Cooling Assoc., Danvers, Mass., U.S.A., p. 301-315 (1991).

Phetteplace, Gary E., Martin J. Kryska and David L. Carbee, *Field Measurements of Heat Losses From Three Types of Heat Distribution Systems*, U.S. Army Cold Regions Research and Engineering Laboratory, Hanover, N H, USA, Report CRREL-SR--91-19 (1991).

Phetteplace, Gary and Vernon Meyer, *Piping for Thermal Distribution Systems*, U.S. Army Cold Regions Research and Engineering Laboratory, Hanover, N H, USA, CRREL Internal Report 1059 (1990).

Phetteplace, G., P. Richmond and N. Humiston, "Thermal Analysis of a Shallow Utilidor", Proceedings of the International District Heating and Cooling Association, Washington, D.C., pp 61-69 (1986).

Schwerdtfeger, Peter, *The Measurement of Heat Flow in the Ground and the Theory of Heat Flux Meters*, Cold Regions Research and Engineering Laboratory, Hanover, N H, USA, Report CRREL-TR-232 (1970).

Shapiro, H.N. and M.J. Moran, "Simultaneous Heat and Mass Transfer in Soil with Application to Waste Heat Utilization", 6th Int. Heat Transfer Conference, Toronto, Ont, pp. 19-24 (1978).

8.0 APPENDICES

APPENDIX A: DISCRETIZATION EQUATIONS FOR BOUNDARY NODES OF MAIN GRID

Note: Upper Case subscripts refer to nodes; lower case subscripts refer to the interface between nodes (between the node in question and the central node, P, around which the neighbor equations are being derived).

A.1 Side Nodes (bottom and sides of grid): Zero Heat Flux Boundaries (see Figure A.1)

At the left side of the grid, a zero heat flux condition requires that $T_W = T_P$, resulting in

$$a_P T_P = a_E T_E + a_S T_S + a_N T_N$$

where

$$a_E = \frac{k_e \Delta y_P}{\delta x_e}$$

$$a_S = \frac{k_s \Delta x_P}{2\delta y_s}$$

$$a_N = \frac{k_n \Delta x_P}{2\delta y_n}$$

$$a_P T_P = a_E T_E + a_S T_S + a_N T_N$$

Bottom and right side nodes analogous

A.2 Bottom Corner Nodes: Zero Heat Flux Boundaries (see Figure A.2)

For the bottom left node,

$$a_P T_P = a_E T_E + a_N T_N$$

where

$$a_E = \frac{k_e \Delta y_P}{2\delta x_e}$$

$$a_N = \frac{k_n \Delta x_P}{2\delta y_n}$$

$$a_P T_P = a_E T_E + a_N T_N$$

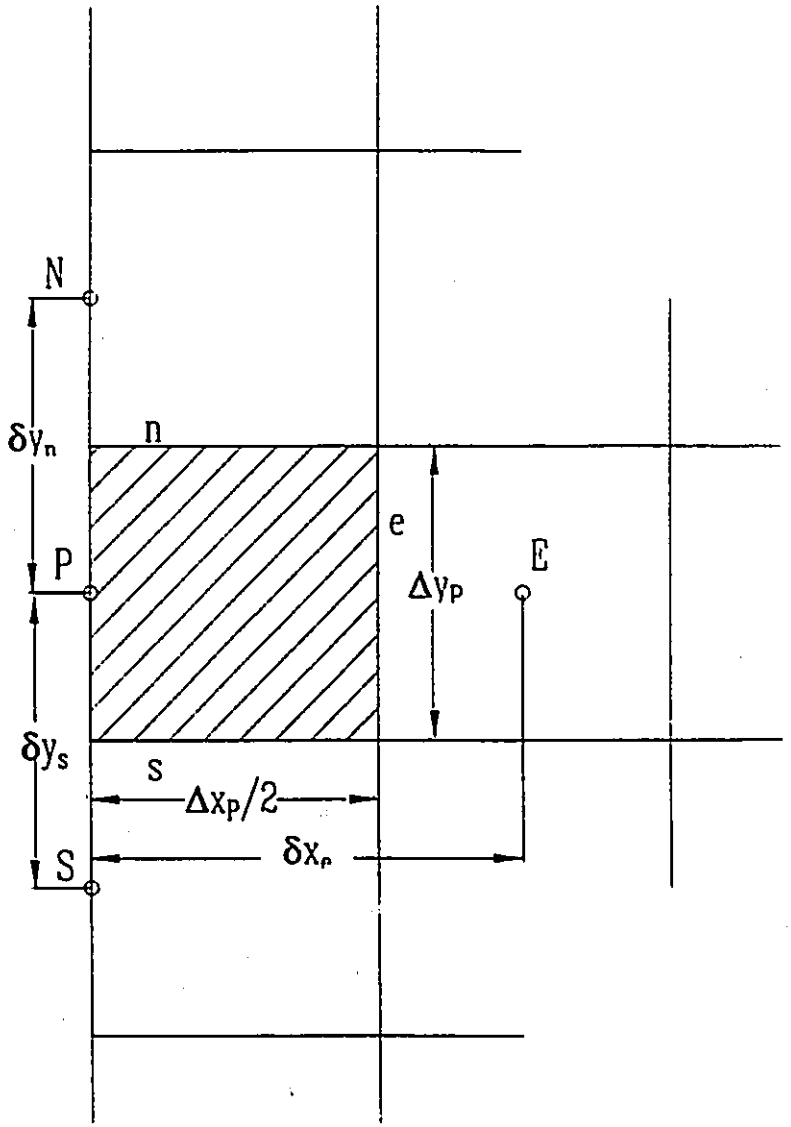

Figure A.1: Side Boundary Node: Main Grid

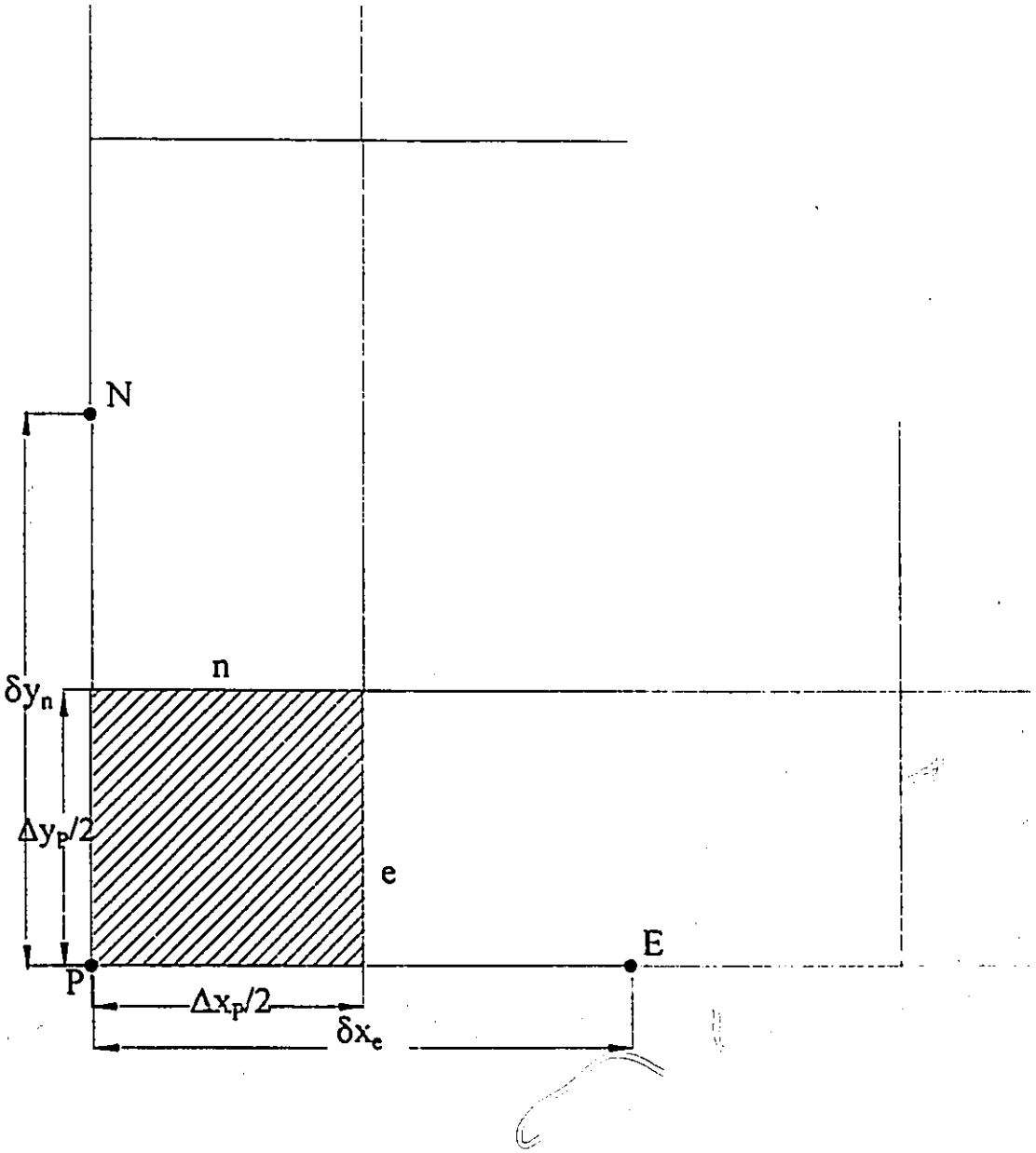

Figure A.2: Bottom Corner Boundary Node: Main Grid

APPENDIX B: DISCRETIZATION EQUATIONS FOR SUB-GRID NODES

Note: Upper Case subscripts refer to nodes; lower case subscripts refer to the interface between nodes (between the node in question and the central node, P, around which the neighbor equations are being derived). Also, $\Delta\Theta$ and $\delta\Theta$ are always equal to $\pi/4$ owing to the equal spacing of the cylindrical sub-grid radial lines. The assumption has been made
that the straight-line distance between nodes of irregular control volume geometry is the same as the distance component normal to the control volume interface.

B.1 Cylindrical Sub-Grid Boundary Nodes Bordering onto Main Grid Nodes (see Figure B.1)

Subscripts: "PSG" = node "P" in Cylindrical Section of Sub-Grid
"WSG" = "West" of node "P" in Cylindrical Section of Sub-Grid
"SSG" = "South" of node "P" in Cylindrical Section of Sub-Grid
"MG" = Main Grid neighboring node

$$a_{PSG}T_{PSG} = a_{WSG}T_{WSG} + a_{SSG}T_{SSG} + b$$

where

$$a_{WSG} = \frac{k_{wsg}\Delta r_{WSG}}{r_{WSG}(\delta\Theta)}$$

$$a_{SSG} = \frac{k_{ssg}\left(r_{SSG} + \frac{\Delta r_{SSG}}{2}\right)\Delta\Theta}{(\delta r)}$$

$$a_{PSG} = a_{WSG} + a_{SSG}$$

$$b = \frac{k_{psg-mg}\Delta y_{MG}}{\delta r_{(psg-mg)}}\left(T_{PSG} - T_{MG}\right)$$

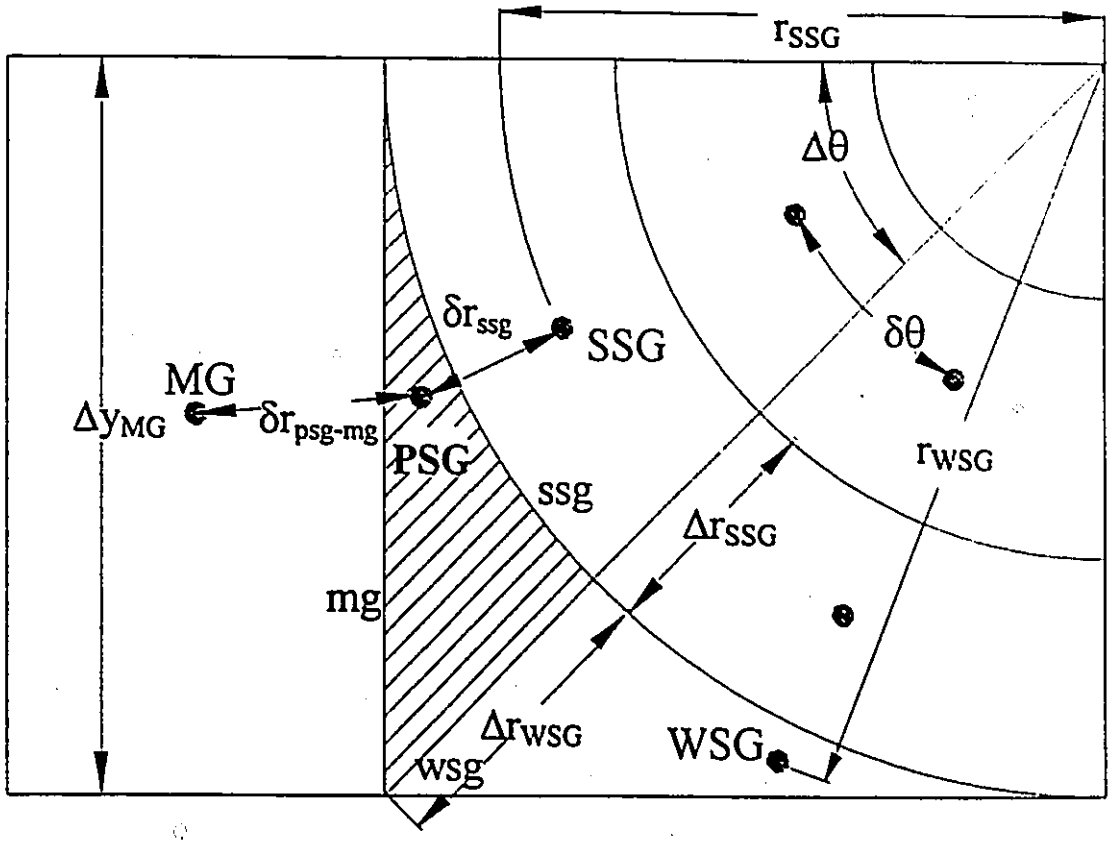

Figure B.1: Boundary Cylindrical Sub-Grid Node Bordering On Main Grid Node

B.2 Reduced Sub-Grid: Cylindrical Sub-Grid Node Bordering Onto One Rectangular Subgrid Node (see Figure B.2)

Subscripts: "PC" = node "P" in Cylindrical section
"WC" = "West" of node "P" in Cylindrical Section of Sub-Grid
"SC" = "South" of node "P" in Cylindrical Section of Sub-Grid
"NC" = "North" of node "P" in Cylindrical Section of Sub-Grid
"ER" = "East" of node "P" in Rectangular Section of Sub-Grid

$$a_{PC}T_{PC} = a_{WC}T_{WC} + a_{SC}T_{SC} + a_{NC}T_{NC} + a_{ER}T_{ER}$$

where

$$a_{WC} = \frac{k_{wc}\Delta r_{PC}}{r_{WC}(\delta\Theta)}$$

$$a_{SC} = \frac{k_{sc}\left(r_{SC} + \frac{\Delta r_{SC}}{2}\right)\Delta\Theta}{(\delta r)_{sc}}$$

$$a_{NC} = \frac{k_{nc}\left(r_{NC} - \frac{\Delta r_{NC}}{2}\right)\Delta\Theta}{(\delta r)_{n}}$$

$$a_{ER} = \frac{k_{er}\Delta r_{PC}}{\left(r_{PC}\frac{\pi}{8}\right) + \frac{\Delta y_{ER}}{2}}$$

$$a_{PC} = a_{WC} + a_{SC} + a_{NC} + a_{ER}$$

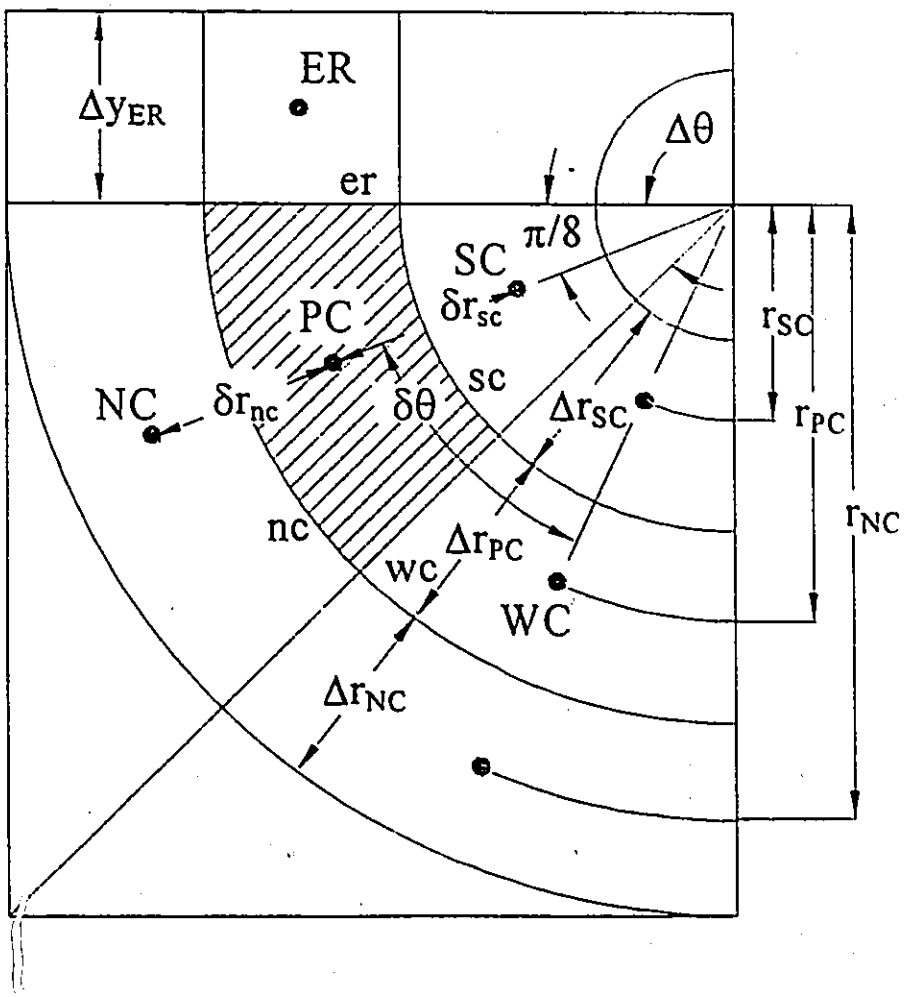

Figure B.2: Interior Cylindrical Sub-Grid Node Bordering On Rectangular Reduced Sub-Grid Node

B.3 Reduced Sub-Grid: Rectangular Sub-Grid Node Bordering Onto One Main Grid Node , Two Sub-Grid Rectangular Nodes and One Sub-Grid Cylindrical Node (see Figure B.3)

Subscripts: "PR" = Node "P" of Rectangular Section Of Sub-Grid
"WC" = "West" of Node "P" in Cylindrical Section of Sub-Grid
"SR" = "South" of Node "P" in Rectangular Section of Sub-Grid
"NR" = "North" of Node "P" in Rectangular Section of Sub-Grid
"EMG" = Main Grid Node, "East" of Node "P"

$$a_{PR}T_{PR} = a_{WC}T_{WC} + a_{SR}T_{SR} + a_{NR}T_{NR} + b$$

where

$$a_{WC} = \frac{k_{wc}\Delta r_{WC}}{\left(r_{WC}\frac{\pi}{8}\right) + \left(\frac{\Delta y_{PR}}{2}\right)}$$

$$a_{SR} = \frac{k_{sr}\Delta y_{PR}}{(\delta x)_{sr}}$$

$$a_{NR} = \frac{k_{nr}\Delta y_{PR}}{(\delta x)_{nr}}$$

$$b = a_{EMG}(T_{PR} - T_{EMG}) = \frac{k_{emg}\Delta x_{PR}}{\left(\frac{\Delta y_{PR}}{2}\right) + \left(\frac{\Delta y_{EMG}}{2}\right)}(T_{PR} - T_{EMG})$$

$$a_{PR} = a_{WC} + a_{SR} + a_{NR}$$

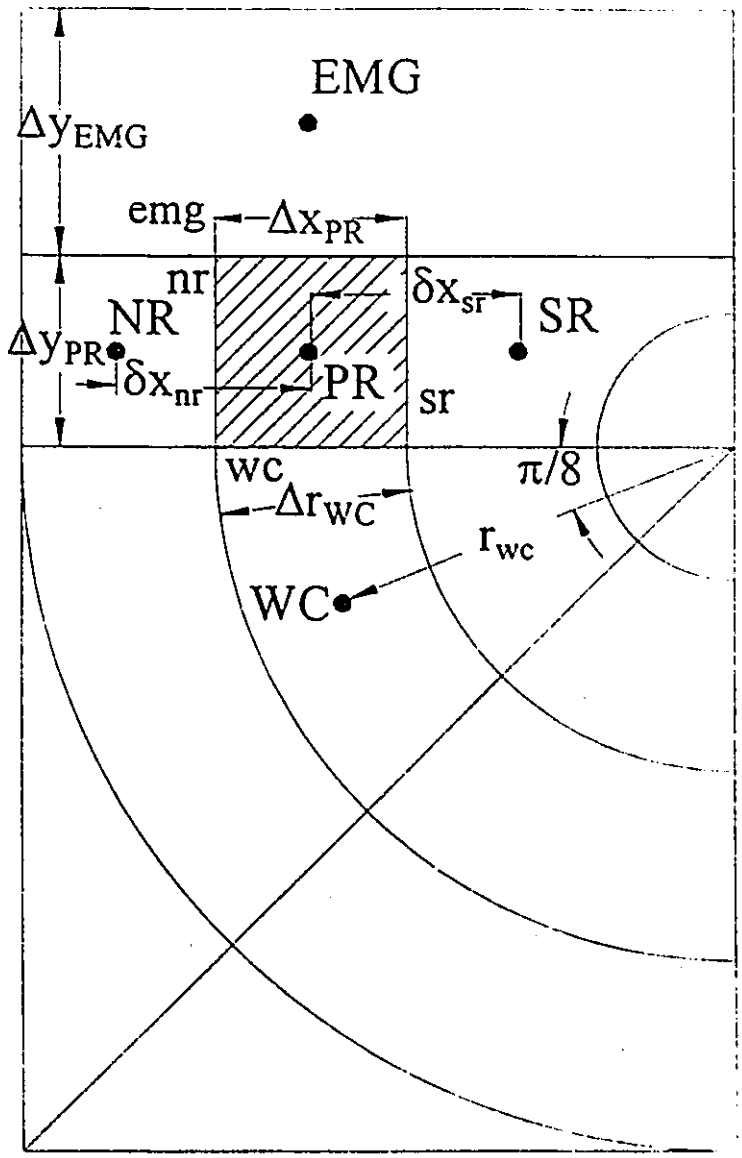

Figure B.3: Rectangular Reduced Sub-Grid Node Bordering On Main Grid Node

B.4 Reduced Sub-Grid: First Rectangular Sub-Grid Node Out From Pipe: Bordering Nodes are One Node Inside Pipe Wall, Two Rectangular Sub-Grid Nodes and One Main Grid Node (see Figure B.4)

Subscripts: "PR" = Node "P" of Rectangular Section Of Sub-Grid
"WR"= "West" of Node "P" in Rectangular Section of Sub-Grid
"SPI" = "South" of Node "P" On Inside of Pipe Wall
"NR" = "North" of Node "P" in Rectangular Section of Sub-Grid
"MG"= The Main Grid Node Next To Node "P"
"EC" = "East" of Node "P" in Cylindrical Section of Sub-Grid

$$a_{PR}T_{PR} = a_{WR}T_{WR} + a_{SPI}T_{SPI} + a_{NR}T_{NR} + a_{EC}T_{EC} + a_{MG}T_{MG}$$

where

$$a_{WR} = \frac{k_{we}\left(\Delta y_{PR} - r_{pipe}\right)}{\delta x_{wr}}$$

$$a_{SPI} = \frac{k_{pi}\left(r_{pipe}\frac{\pi}{2}\right)}{\delta_{SPI-PR}}$$

$$a_{NR} = \frac{k_{nr}\Delta y_{PR}}{(\delta x)_{nr}}$$

$$a_{EC} = \frac{k_{ec}\Delta r_{EC}}{\left(r_{EC}\frac{\pi}{8}\right) + \frac{\Delta y_{PR}}{2}}$$

$$a_{MG} = \frac{k_{mg}\Delta x_{PR}}{\left(\frac{\Delta y_{PR}}{2}\right) + \left(\frac{\Delta y_{MG}}{2}\right)}$$

$$a_{PR} = a_{WR} + a_{SPI} + a_{NR} + a_{EC} + a_{MG}$$

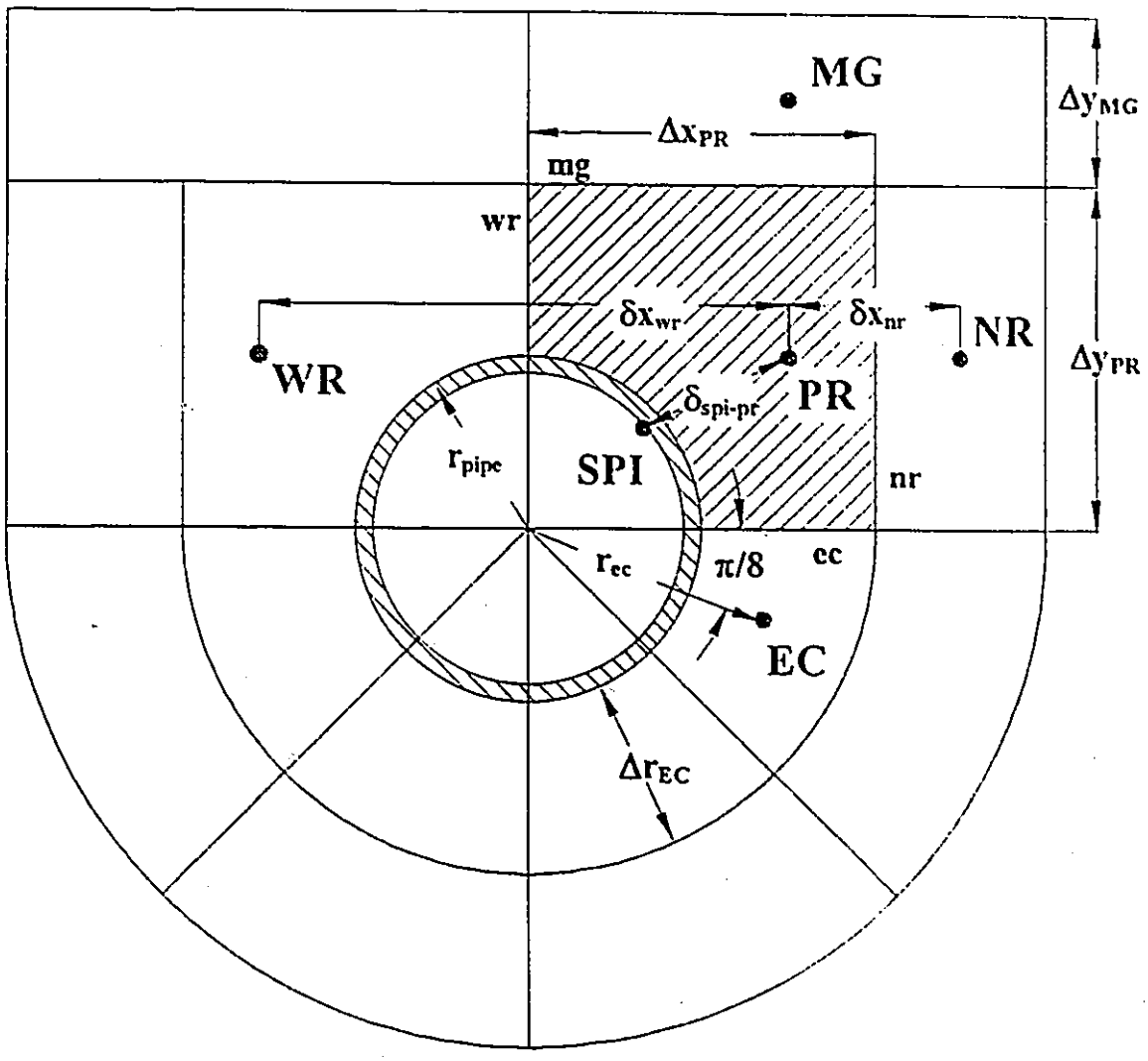

Figure B.4: Rectangular Reduced Sub-Grid Node Bordering On Pipe

APPENDIX C:

THERMISTOR LAYOUT IN LEXAN GRIDS:

MEASURING STATIONS A AND B FOR 1/2" AND 3" SYSTEMS

Table C.1 1/2" System, Measuring Station A: Thermistor's x-y coordinates in Lexan Grid

Thermistor #	X position	Y position
0	5	-5
1	10	-5
2	15	-5
3	20	-5
4	25	-5
5	-15	-10
6	-10	-10
7	-5	-10
8	0	-10
9	5	-10
10	10	-10
11	15	-10
12	-12.5	-15
13	-7.5	-15
14	-2.5	-15
15	2.5	-15
16	7.5	-15
17	12.5	-15
18	-10	-20
19	-5	-20
20	5	-20
21	10	-20
22	-10	-30
23	10	-30
24	-30	0
27	-20.5	0
28	-17.5	0
29	-13.5	0
31	-5	0
32	0	0
33	5	0
35	13	0
36	-8	0
37	17.5	0
38	20.5	0
39	25.5	0
40	30	0
41	8	0
42	10.5	-2
43	-10.5	-2
44	-25	-5
45	-20	-5
46	-15	-5
47	-10	-5
48	-5	-5
49	0	-5

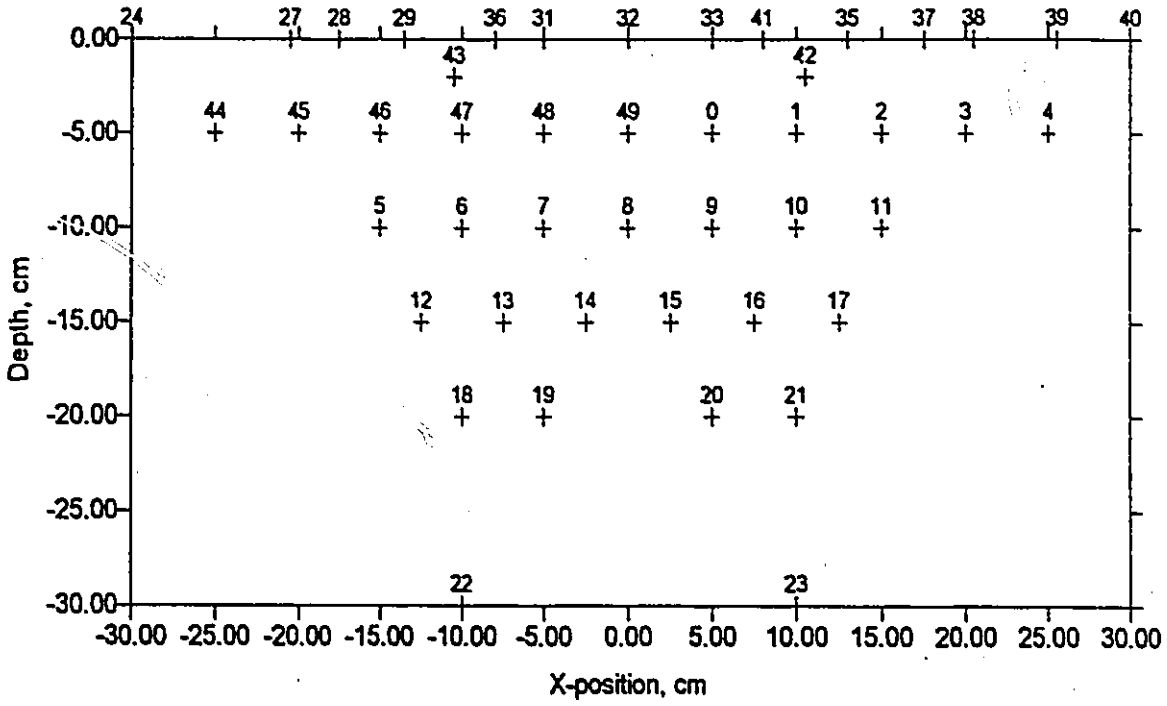

Figure C.1: 1/2" System, Measuring Station A:
Thermistor Layout in Lexan Grid

Table C.2 1/2" System, Measuring Station B: Thermistor's x-y Coordinates in Lexan Grid

Thermistor #	X position	Y position
50	20	-5
51	25	-5
52	29.5	-5
53	-20	-10
54	-15	-10
55	-10	-10
56	-5	-10
57	0	-10
58	5	-10
59	10	-10
60	15	-10
61	20	-10
62	-13	-15
63	-8	-15
64	-3	-15
65	0	-15
66	3	-15
67	8	-15
68	13	-15
69	-10	-20
70	-5	-20
71	5	-20
72	10	-20
73	-10	-30
74	10	-30
75	-25	0
76	-20	0
77	-17.5	0
78	-13.5	0
79	-8	0
80	-5	0
81	0	0
82	5	0
83	8	0
84	13	0
85	17.5	0
86	20	0
87	25	0
88	-10	-3
89	10	-3
90	-29.5	-5
91	-25	-5
92	-20	-5
93	-15	-5
94	-10	-5
95	-5	-5
96	0	-5
97	5	-5
98	10	-5
99	15	-5

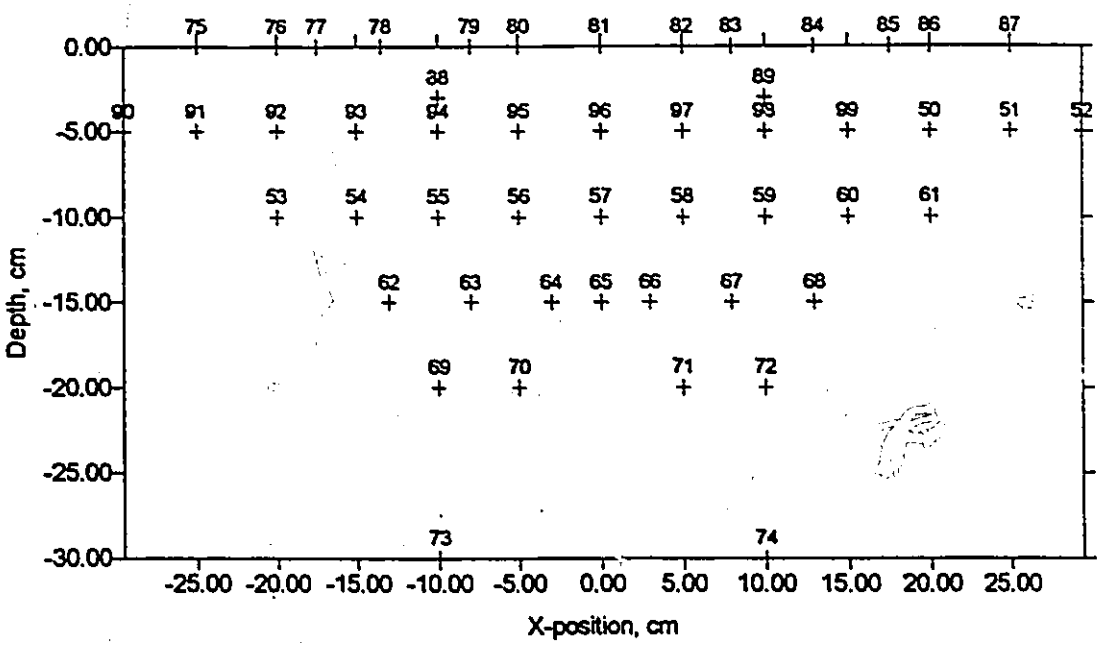

Figure C.2: 1/2" System, Measuring Station B:
Thermistor Layout in Lexan Grid

Table C.3 3" System, Measuring Station A: Thermistor's x-y Coordinates in Lexan Grid

Thermistor #	X position	Y position
200	-26.5	0
201	-22.5	0
202	-17.5	0
203	-7.5	0
204	-2.5	0
205	2.5	0
206	7	0
207	17	0
208	22	0
209	26	0
210	-27.5	-3
211	-22.5	-3
212	-17.5	-3
213	-7.5	-3
214	-2.5	-3
215	2.5	-3
216	7.5	-3
217	17.5	-3
218	22.5	-3
219	27.5	-3
220	-22.5	-8
221	-17.5	-8
222	-12.5	-8
223	-7.5	-8
224	-2.5	-8
225	2.5	-8
226	7.5	-8
227	12.5	-8
228	17.5	-8
229	22.5	-8
230	-22.5	-13
231	-17.5	-13
232	-12.5	-13
233	-7.5	-13
234	-2.5	-13
235	2.5	-13
236	7.5	-13
237	12.5	-13
238	17.5	-13
239	22.5	-13
240	-17.5	-18
241	-7.5	-18
242	2.5	-18
243	12.5	-18
244	22.5	-18
245	-12.5	-23
246	2.5	-23
247	17.5	-23
248	-12.5	-33
249	12.5	-33

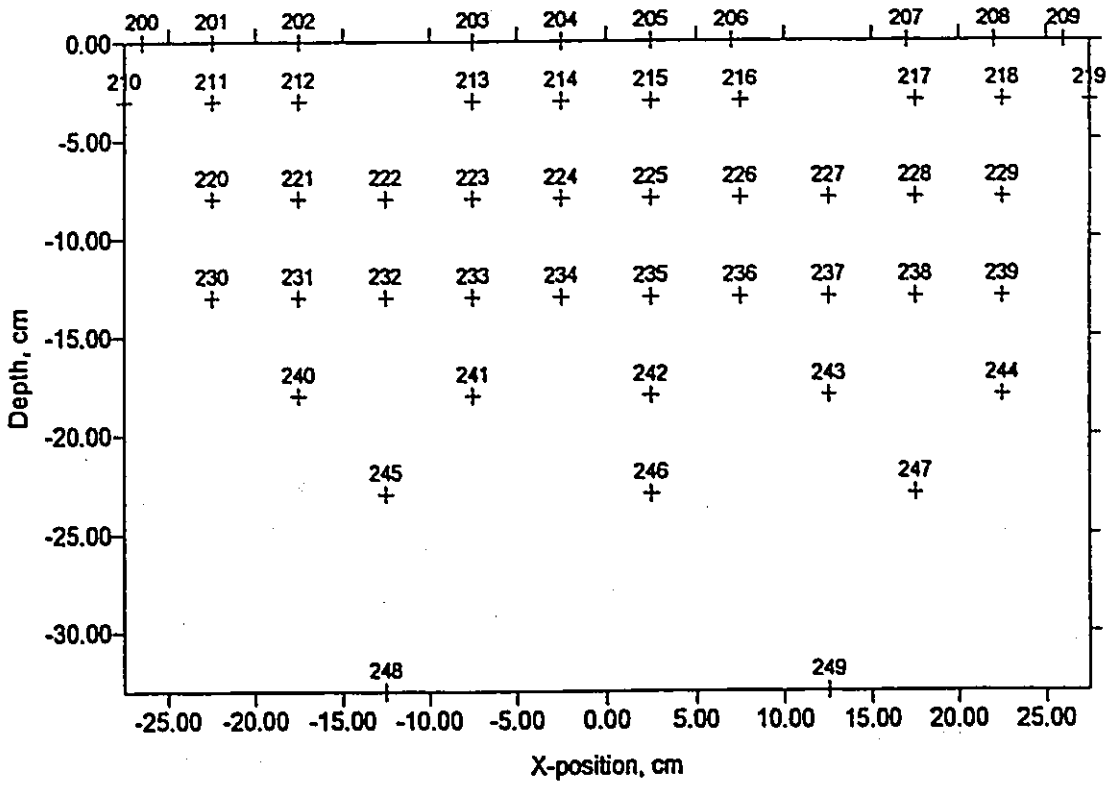

Figure C.3: 3" System, Measuring Station A: Thermistor Layout in Lexan Grid

Table C.4 3" System, Measuring Station B: Thermistor's x-y Coordinates in Lexan Grid

Thermistor #	X position	Y position
150	-26.5	0
151	-22.5	0
152	-17.5	0
153	-7.5	0
154	-2.5	0
155	2.5	0
156	7	0
157	17	0
158	22	0
159	26	0
160	-27.5	-3
161	-22.5	-3
162	-17.5	-3
163	-7.5	-3
164	-2.5	-3
165	2.5	-3
166	7.5	-3
167	17.5	-3
168	22.5	-3
169	27.5	-3
170	-22.5	-8
171	-17.5	-8
172	-12.5	-8
173	-7.5	-8
174	-2.5	-8
175	2.5	-8
176	7.5	-8
177	12.5	-8
178	17.5	-8
179	22.5	-8
180	-22.5	-13
181	-17.5	-13
182	-12.5	-13
183	-7.5	-13
184	-2.5	-13
185	2.5	-13
186	7.5	-13
187	12.5	-13
188	17.5	-13
189	22.5	-13
190	-17.5	-18
191	-7.5	-18
192	2.5	-18
193	12.5	-18
194	22.5	-18
195	-12.5	-23
196	2.5	-23
197	17.5	-23
198	-12.5	-33
199	12.5	-33

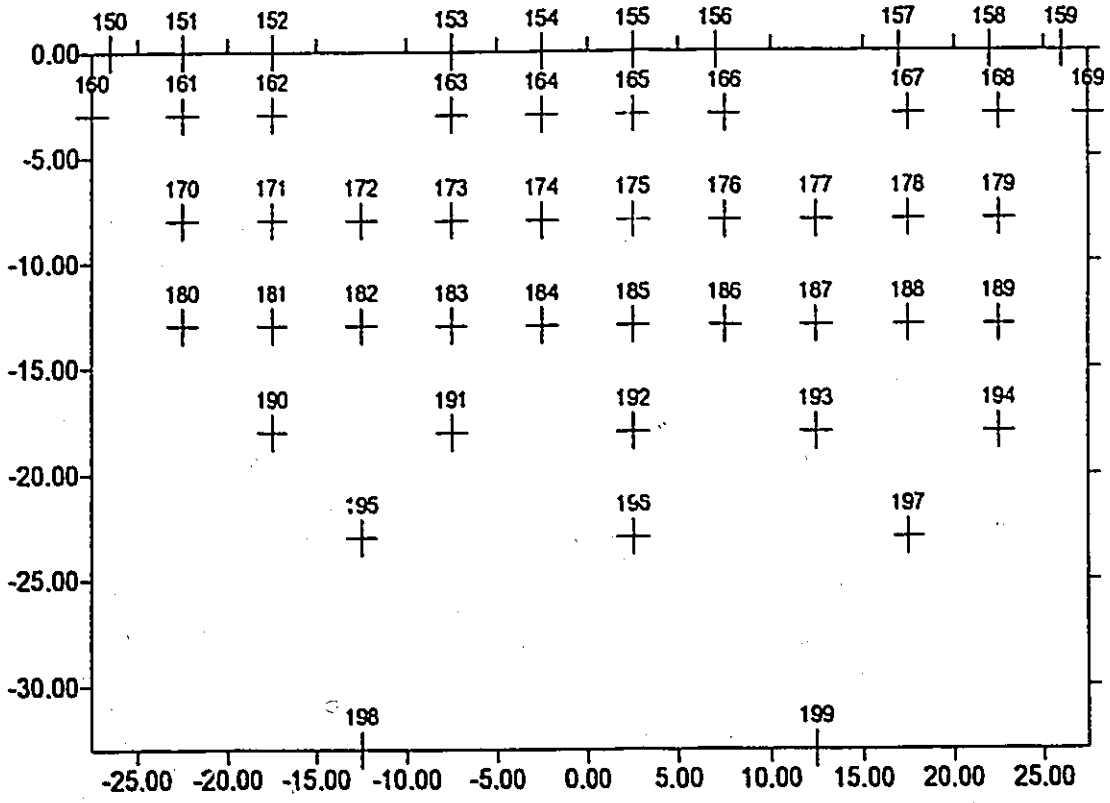

Figure C.4: 3" System, Measuring Station B:
Thermistor Layout in Lexan Grid

APPENDIX D:

FORTRAN PROGRAM LISTING:

"UNDERGROUND PIPELINE HEAT TRANSFER PROGRAM"

```
C  UNDERGROUND PIPELINE THERMAL PROGRAM
C
C
      INCLUDE 'FGRAPH.FI'
      PROGRAM PIPES
C ***************************************************************
C Variables:

C     aa (v1)          =   the horizontal distance from the final subgrid
C                          node on a radial line to the center of the main
C                          grid node (ie. lined up with adjacent main grid
C                          node)
C     A,B,C,D,P,Q      =   coefficients for the TDMA equations used in
C     (v2)                 the line by line solution algorithm for main grid
C     AS,BS,CS,DS,     =   coefficients for the TDMA equations used in
C     PS,QS (v3)           the line by line solution algorithm for subgrids
C     BEDSIDE (v1)     =   Distance from pipe centerline to border of
C                          square embedded subgrid
C     cc(v1)           =   Vertical distance from 1st node along radial lines
C                          5&6 to centerline level - used in calculation of DEL
C     DEL (v2)         =   distance between 1st and 2nd node of lines 5 & 6
C                          of reduced subgrid (lines 5&6 are rectangular geom)
C     DELOLD           =   difference between nodal temperatures calculated
C                          between successive iterations
C     DELR (v3)        =   distance between nodes in cylindrical subgrid
C     DELTAR (v3)      =   distance between n and s interfaces of control
C                          volumes in cylindrical subgrid
C     DELTARF(v1)      =   E or W interfacial area across which heat flows in
C                          final subgrid node (along a radial line)
C     DELTATH          =   difference in the two radial lines which are the
C                          e and w borders of a c.v. in cylndrical subgrid
C     DELTAX (v1)      =   distance between e and w interfaces
C     DELTAXHS (v2)    =   distance between n and s interfaces of rectangular
C                          nodes along lines 5&6 of reduced subgrids
C     DELTAY (v1)      =   distance between n and s interfaces
C     DELTAYHS (v2)    =   distance between e and w interfaces of rectangular
C                          nodes along lines 5&6 of reduced subgrids
C     DELTHETA         =   difference in angle betweentwo adjacent nodes
C                          lying along radial lines in cylindrical subgrid
C     DELX (v1)        =   distance between nodes in the x-direction
C     DELXHSG (v2)     =   distance between nodes along lines 5&6 of reduced
C                          subgrids
C     DELY (v1)        =   distance between nodes in the y-direction
C     DELYHSG (v2)     =   vertical distance between nodes along lines 5&6 of
C                          reduced subgrids and adjacent main grid nodes
C     DUMTLX (v1)      =   x main grid nodal coordinate of the top left
C                          corner node of a subgrid (will be a dummy node
C                          in main grid sweep)
C     DUMTLY,DUMBLX,   =   x,y coordinates of the other main grid dummy
C     DUMBLY,DUMTRX,       nodes making up the subgrid (total of four nodes
C     DUMTRY,DUMBRX,       per subgrid)
C     DUMBRY
C     e (v1)           =   Horizontal distance from 1st node along radial
C                          lines
C     EMSPACE (v2)     =   vector of the distances between nodes in subgrid
C     FEHSG (v3)       =  East interface parameter (used in calculation of
C                          interface conductivity) for lines 5&6 of reduced SG
C     FEMG (v1)        =  East interface parameter (used in calculation of
C                          interface conductivity) for MG nodes
C     FNSG (v3)        =  North interface parameter (used in calculation of
C                          interface conductivity) for SG nodes
C     FNMG (v1)        =  North interface parameter (used in calculation of
C                          interface conductivity) for MG nodes
C     h (v1)           =   vertical distance from subgrid node to side of
C                          subgrid (which is also the interface of the
C                          adjacent main grid node)
C     hh (v1)          =   Height of pipe center to the bottom of the
C                          insulation board (height of top 2 MG nodes of HSG)
```

```
C     IDIAM (v1)      =   inside diameter of a pipe
C     IRADIUS         =   Inside Radius of pipe
C     Kg              =   Thermal conductivity of sand
C     Kreg (v1)       =   thermal conductivity of region of different k
C                         than soil
C     M               =   number of nodes along y-axis in the main grid
C     MAXITER         =   Maximum number of iterations (parameter)
C     N               =   number of nodes along x-axis in the main grid
C     NEM (v1)        =   number of nodes along any radial line in subgrid
C     NFSG            =   number of full subgrids
C     NHSG            =   number of half (reduced) subgrids
C     NPIPE           =   number of pipes in the system
C     ODIAM (v1)      =   outside diameter of a pipe
C     ORADIUS         =   Outside Radius of pipe
C     R               =   radius to a node in cylindrical subgrid
C     RADINS          =   radius to the outside of the insulation
C     REGNODX (v1)    =   x-coord. of top left corner MG node of a region of
C                         differing thermal conductivity than soil
C                         (ie. insulation slab)
C     REGNODY (v1)    =   y-coord. of top left corner MG node of a region of
C                         differing thermal conductivity than soil
C                         (ie. insulation slab)
C     REGNUM          =   Number of rectangular regions of different thermal
C                         conductivity than soil (ie. insulation slabs)
C     REGNUMX (v1)    =   Number of nodes spanned in the x-direction by a
C                         region of differing thermal conductivity than soil
C     REGNUMY (v1)    =   Number of nodes spanned in the y-direction by a
C                         region of differing thermal conductivity than soil
C     s (v3)          =   Geometric Parameter used to calculate angle between
C                         boundary SG node and adjacent MG node
C     T (v2)          =   temperatures of main grid nodes
C     TFLUID (v1)     =   Wall temperature of pipe
C     Thet (v3)       =   Geometric Parameter used to calculate angle between
C                         boundary SG node and adjacent MG node
C     Tg              =   Temperature of ground surface
C     THICKINS (v1)   =   thickness of the insulation around a pipe
C     TKINSULN (v1)   =   thermal conductivity of insulation material
C     TKMG (v3)       =   thermal conductivity of main grid nodes
C     TKMGE (v2)      =   East interface conductivity - MG nodes
C     TKMGN (v2)      =   North interface conductivity - MG nodes
C     TKPIP (v1)      =   thermal conductivity of pipe
C     TKSG (v3)       =   thermal conductivity of subgrid nodes
C     TKSGE           =   thermal conductivities of east interfaces of
C                         subgrid nodes
C     TKSGN           =   thermal conductivities of north interfaces of
C                         subgrid nodes
C     TOLD(v2)        =   temperature of MG node from previous iteration
C     TS (v3)         =   temperatures of subgrid nodes
C     TSOLD (v3)      =   temperature of SG node from previous iteration
C     QN5 (v2)        =   Heat flux from a SG node of line 5 of reduced SG
C     QN6 (v2)        =   Heat flux from a SG node of line 6 of reduced SG
C     QN5T (v1)       =   Sum of heat flux from each SG node along line 5 of
C                         reduced SG - SG calculations
C     QN6T (v1)       =   Sum of heat flux from each SG node along line 6 of
C                         reduced SG - SG calculations
C     QN5WT (v2)      =   Heat flux from a SG node of line 5 of reduced SG
C                         (MG calculations - D term)
C     QN6WT (v2)      =   Heat flux from a SG node of line 6 of reduced SG
C                         (MG calculations - D term)
C     QN5WTT (v1)     =   Sum of heat flux from each SG node of line 5 of
C                         reduced SG (MG calculations - D term)
C     QN6WTT (v1)     =   Sum of heat flux from each SG node of line 6 of
C                         reduced SG (MG calculations - D term)
C     Z               =   distance from side of subgrid to adjacent node
C                         of main grid
C ================================================================
C
C
C
```

```
C
C
C

C  MAIN PROGRAM
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        REGNUM,REGNUMX,REGNUMY,REGNODX,REGNODY

      CHARACTER LINE*70
    1 OPEN(UNIT=1,FILE="INPUT.DAT")
      OPEN(UNIT=2,FILE="OUTPUT.DAT")
      OPEN(UNIT=3,FILE="PARAM.DAT")
      OPEN(UNIT=4,FILE="FIVSIX.DAT")
      OPEN(UNIT=5,FILE="INPUTA.DAT")
      OPEN(UNIT=6,FILE="INPUTB.DAT")
      OPEN(UNIT=7,FILE="INPUTC.DAT")
      OPEN(UNIT=8,FILE="PARAMET.DAT")
      OPEN(UNIT=9,FILE="TEMPMAIN.DAT")
      OPEN(UNIT=10,FILE="TEMPSUB.DAT")

      LINE= '------------------------------------------------------------
     +--'
  900 FORMAT(A70)
C ***************************************************************************
```

```
    2 CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'MAIN MENU'
      PRINT 900,LINE
      PRINT *,'1- DEFINE THE SYSTEM WITH COMPUTER ASSISTANCE'
      PRINT *,'2- SIMULATE A SYSTEM ON A FILE'
      PRINT *,'3- EXIT PROGRAM'
      PRINT 900,LINE
      READ *,MENU
C ================================================================
C SET-UP OF OVERALL GRID, with point 1,1 as the top left corner.  All
C nodes in the grid will be defined in relation to this 1,1 point.
C User defines vectors for spacing in the x and y directions, from which
C matrices of spacings b/w all nodes and all interfaces can be deduced:
C================================================================
      IF(MENU.EQ.1)THEN
        CALL ENTRGRID()
    7   FORMAT(A3)
C ================================================================
C ENTERING PIPE INFORMATION
C================================================================
        CALL ENTRPIPE()
C ================================================================
C CALCULATIONS FOR EMBEDDED GRIDS AROUND EACH PIPE
C================================================================
        CALL ENTRFSG()
        CALL ENTRHSG()
c        CALL ALGSPEC(CONVCRIT,MAXITER)
c        CALL WRTPARAM()
        ELSE
        IF(MENU.EQ.2)THEN
          CALL RDPARAM()
          ELSE
          IF(MENU.EQ.3)THEN
             STOP
          ENDIF
        ENDIF
        ENDIF
      IF(MENU.GT.3.OR.MENU.LT.1)GOTO 2
C================================================================
C TDMA CONSTANT PARAMETER CALCULATIONS FOR SUBGRID NODES AND MAIN GRID NODES
C----------------------------------------------------------------
C TDMA PARAMETER CALCULATION FOR ALL SUBGRID NODES - PREP FOR LINE BY
C LINE SOLUTION ALGORITHM: CONSTANT PARAMETER CALCULATION ONLY (A,B,C AND
C FIRST NODES)
C================================================================
      CALL FSGCSTP()
      CALL HSGCSTP()
C----------------------------------------------------------------
C TDMA PARAMETER CALCULATION FOR ALL MAIN GRID NODES - PREP FOR LINE BY
C LINE SOLUTION ALGORITHM: CONSTANT COEFFICIENTS ONLY (A,B,C)
C----------------------------------------------------------------
      CALL MGCSTP()

C================================================================
C ALGORITHM LOOP: MAIN AND SUBGRIDS ARE CONVERGED TOGETHER
C                -includes calculation of remaining TDMA parameters which
C                 change with each sweep due to dependence on neighbour
C                 temperatures
C                -one sweep through main grid, followed by one sweep
C                 through each subgrid.  Convergence is checked (set by
C                 maximum error value) then sweep through main grid starts
C                 again, etc. until convergence is reached
C================================================================
      if(menu.eq.1)then
        CALL ALGSPEC(CONVCRIT,MAXITER)
        CALL WRTPARAM()
      endif
```

```
      MCOUNTER=0
      IFLAG=0
      DO 8001 WHILE(IFLAG.EQ.0)
        DELOLD=0.
        MCOUNTER=MCOUNTER+1
        IF(MCOUNTER.GE.MAXITER)THEN
          IFLAG=1
        ENDIF
C-------------------------- ------------------------------

C SWEEP THROUGH MAIN GRID
C-------------------------------------------------------------
        CALL MGALG()
C-------------------------------------------------------------------------
C SWEEP THROUGH ALL SUBGRIDS
C-------------------------------------------------------------------------
        CALL FSGALG()
        CALL HSGALG()
C-------------------------------------------------------------------------
C CHECK FOR CONVERGENCE - CHECK MAXIMUM ERROR OF ALL NODES (SUB AND MAIN
C                         GRIDS) - ie MAX DIFFERENCE IN TEMP BETWEEN LAST
C                         AND PRESENT ITERATIONS

C-------------------------------------------------------------------------
        IF(DELOLD.LT.CONVCRIT)THEN
          IFLAG=1
        ENDIF
 8001 CONTINUE

C=========================================================================
C END LOOP
C=========================================================================
      DO 9003 IC=1,NPIPE
        T(DUMTRX(IC),DUMTRY(IC))=0.
        T(DUMTLX(IC),DUMTLY(IC))=0.
        T(DUMBRX(IC),DUMBRY(IC))=0.
        T(DUMBLX(IC),DUMBLY(IC))=0.
 9003 CONTINUE
      PRINT *,'ITERATION # ',MCOUNTER
      WRITE (1,*)'ITERATION # ',MCOUNTER
      WRITE (1,9002) ((T(J,I), J=1,80),I=1,M)
 9002 FORMAT (1X,80(F7.3,1X),/)
      WRITE (5,992) ((T(J,I), J=26,50), I=1,M)
  992 FORMAT(1X,25(F7.3,1X),/)
      WRITE (6,993) ((T(J,I), J=51,79), I=1,M)
  993 FORMAT(1X,29(F7.3,1X),/)
      WRITE (7,994) ((T(J,I), J=80,104), I=1,M)
  994 FORMAT(1X,25(F7.3,1X),/)

      WRITE (2,999) (((TS(IC,I,J), J=1,8), I=1,NEM(IC)),IC=1,NPIPE)
  999 FORMAT(8(F10.3,','),/)

C WRITING RESULTS TO TEMPMAIN.DAT AND TEMPSUB.DAT
      WRITE (10,999) (((TS(IC,I,J), J=1,6), I=1,NEM(IC)),IC=1,NPIPE)
            DO 7077 I=1,N
      DO 7078 J=1,M
      write (9,989) I,',',J,',',T(i,j)
 7078 CONTINUE
 7077 CONTINUE

  989 format (I4,A1,I4,A1,F13.7)
C CLOSE ALL FILES TO RECORD INFO PERMANENTLY

      CLOSE(1)
      CLOSE(2)
      CLOSE(3)
      CLOSE(4)
      CLOSE(5)
      CLOSE(6)
```

```
      CLOSE(7)
      CLOSE(8)
      CLOSE(9)
      CLOSE(10)

C RESET VALUES
      DO 110 I=1,MAX
        DO 120 J=1,MAX
          T(I,J)=0.
          TOLD(I,J)=0.
  120   CONTINUE
  110 CONTINUE

      DO 130 I=1,MAXR
        DO 140 J=1,MAXC
          DO 150 K=1,MAXP
            TS(K,J,I)=0.
            TSOLD(K,J,I)=0.
  150     CONTINUE
  140   CONTINUE
  130 CONTINUE

      GOTO 1
      END
C -------------------------------------------------------------------
```

C SUBROUTINES

```
C ===================================================================
C TITLE: ENTRGRID()
C
C DESCRIPTION:
C
C     This subroutine allows the user to define the grid
C     (calculation domain)
C -------------------------------------------------------------------
      SUBROUTINE ENTRGRID()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
```

```
     +           MAXC,MAXR)

      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z

      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


C ==========================================================================
C SET-UP OF OVERALL GRID, with point 1,1 as the top left corner.  All
C nodes in the grid will be defined in relation to this 1,1 point.
C User defines vectors for spacing in the x and y directions, from which
C matrices of spacings b/w all nodes and all interfaces can be deduced:
C==========================================================================
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'Enter the number of nodes in the x-direction.'
      READ *,N
      CALL CLEARSCREEN($GCLEARSCREEN)
      DO 100 I=1,N-1
         PRINT 1000,'Enter the distance (x-direction) between node',I
         PRINT 1005,'and node',I+1
         IF(I.EQ.1)THEN
            PRINT *
            PRINT *,'Note: the interface between nodes 1 and 2 must'
            PRINT *,'lie midway between the nodes.'
            PRINT *
            ELSEIF(I.EQ.N-1)THEN
               PRINT *
               PRINT *,'Note: the interface between the second last and'
               PRINT *,'last node must lie midway between the nodes.'
               PRINT *
         ENDIF
         READ *,DELX(I)
  100 CONTINUE
 1000 FORMAT(A46,I3)
 1005 FORMAT(A9,I3)

C SET UP DELTAX VECTOR:
C DELTAX is the distance between control volume faces e and w

C For nodes along line x=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAX(1)=DELX(1)/2.
C For second node (distance b/w interfaces on either side of node 2,J):
         DELTAX(2)=DELX(1)
C For interior nodes:
         DO 135 I=3,N-1
            DELTAX(I)=(DELX(I-1)-(DELTAX(I-1)/2.))*2.
  135    CONTINUE
C For last node (distance from e interface of (N-1)th node to Nth node)
         DELTAX(N)=DELX(N-1)-(DELTAX(N-1)/2.)

134      IF(DELX(N-1).NE.DELTAX(N-1))THEN
            PRINT *
            PRINT *,'ERROR: The interface between the second last and'
            PRINT *,'last node must lie midway between the nodes, that'
```

```
            PRINT *,'is, the distance between the second last and last'
            PRINT *,'node must be equal to delta x of the second last'
            PRINT *,'node.'
            PRINT *,'Please re-enter the distance between the second'
            PRINT *,'last and last node on the x-axis:'
            PRINT *
            READ *,DELX(N-1)
            GOTO 134
         ENDIF
C ------------------------------------------------------------------
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'Enter the number of nodes in the y-direction.'
      READ *,M
      CALL CLEARSCREEN($GCLEARSCREEN)
      DO 105 J=1,M-1
         PRINT 1010,'Enter the distance (y-direction) between node',J
         PRINT 1015,'and node',J+1
         IF(J.EQ.1)THEN
            PRINT *
            PRINT *,'Note: the interface between nodes 1 and 2 must'
            PRINT *,'lie midway between the nodes.'
            PRINT *
         ELSEIF(J.EQ.M-1)THEN
            PRINT *
            PRINT *,'Note: the interface between the second last and'
            PRINT *,'last node must lie midway between the nodes.'
            PRINT *
         ENDIF
         READ *,DELY(J)
  105 CONTINUE
 1010 FORMAT(A46,I3)
 1015 FORMAT(A9,I3)
C SET UP DELTAY VECTOR:
C DELTAY is the distance b/w control volume faces n and s

C For nodes along line y=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAY(1)=DELY(1)/2.
C For second node (distance b/w interfaces on either side of node I,2)

         DELTAY(2)=DELY(1)
C For interior nodes:
         DO 155 J=3,M-1
            DELTAY(J)=(DELY(J-1)-(DELTAY(J-1)/2.))*2.
  155    CONTINUE
C For last node (dist. from s interface of (N-1)th node to Nth node)

         DELTAY(M)=DELY(M-1)-(DELTAY(M-1)/2.)

154      IF(DELY(M-1).NE.DELTAY(M-1))THEN
            PRINT *
            PRINT *,'ERROR: The interface between the second last and'
            PRINT *,'last node must lie midway between the nodes, that'
            PRINT *,'is, the distance between the second last and last'
            PRINT *,'node must be equal to delta y of the second last'
            PRINT *,'node.'
            PRINT *,'Please re-enter the distance between the second'
            PRINT *,'last and last node along the y-axis:'
            PRINT *
            READ *,DELY(M-1)
            GOTO 154
         ENDIF
C--------------------------------------------------------------------

C TEMPERATURES along the top row of the grid are known - these are ground
C surface temperatures-input by the user (for now-all nodes will be same)
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'GROUND'
      PRINT *,'-----------------------------------------------------
     +-------'
```

```
      PRINT *,'Enter the ground surface temperature in degrees Celsius:'
      READ *,Tg
      DO 170 I=1,N
         T(I,1)=Tg
  170 CONTINUE
C -----------------------------------------------------------------
C Thermal conductivity of the ground
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'GROUND'
      PRINT *,'-------------------------------------------------------
     +-------'
      PRINT *,'What is the thermal conductivity of the ground'
      READ *, Kg
      DO 180 I=1,N
         DO 175 J=1,M
            TKMG(I,J)=Kg
  175    CONTINUE
  180 CONTINUE
C Redefine nodes in regions of other thermal conductivities - ie insulation
C boards:
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'Enter the number of rectangular regions of different'
      PRINT *,'thermal conductivity than the soil:'
      READ *,REGNUM
      IF(REGNUM.GT.0)THEN
       DO 171 I=1,REGNUM
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REGION:',I
         PRINT *,'-------------------------------------------------------
     +----------'
         PRINT *,'Enter the number of nodes spanned in the x-direction'
         PRINT 1016,'by region',I
         READ *,REGNUMX(I)
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REGION:',I
         PRINT *,'-------------------------------------------------------
     +----------'
         PRINT *,'Enter the number of nodes spanned in the y-direction'
         PRINT 1016,'by region',I
         READ *,REGNUMY(I)
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REGION:',I
         PRINT *,'-------------------------------------------------------
     +----------'
         PRINT *,'Enter the x-coordinate of the top left corner node'
         PRINT 1016,'of region',I
         READ *,REGNODX(I)
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REGION:',I
         PRINT *,'-------------------------------------------------------
     +----------'
         PRINT *,'Enter the y-coordinate of the top left corner node'
         PRINT 1016,'of region',I
         READ *,REGNODY(I)
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REGION:',I
         PRINT *,'-------------------------------------------------------
     +----------'
         PRINT 1017,'What is the thermal conductivity of region',I,'?'
         READ *,KREG(I)
         DO 172 J=REGNODX(I),(REGNODX(I)+(REGNUMX(I)-1))
            DO 173 L=REGNODY(I),(REGNODY(I)+(REGNUMY(I)-1))
               TKMG(J,L)=KREG(I)
  173       CONTINUE
  172    CONTINUE
  171 CONTINUE
 1016 FORMAT(A10,I3)
 1017 FORMAT(A44,I3,A2)
      ENDIF
```

```
C Set up interface thermal conductivities for all main grid nodes:
C-----------------------------------------------------------------------
C ALL NORTH COEFFICIENTS:
C FNMG is the f required for calculating the north interface conductivity
      FNMG(2)=DELTAY(1)/DELY(1)
      DO 156 J=3,M
        FNMG(J)=(DELTAY(J-1)/2.)/DELY(J-1)
  156 CONTINUE
      DO 157 I=1,N
         DO 158 J=2,M
            TKMGN(I,J)=(((1-FNMG(J))/TKMG(I,J))+(FNMG(J)/TKMG(I,J-1)))**
     +(-1)
  158    CONTINUE
  157 CONTINUE
C-----------------------------------------------------------------------
C ALL EAST COEFFICIENTS:
      FEMG(N-1)=DELTAX(N)/DELX(N-1)
      DO 159 I=1,N-2
         FEMG(I)=(DELTAX(I+1)/2.)/DELX(I)
  159 CONTINUE
      DO 176 I=1,N-1
         DO 177 J=1,M
            TKMGE(I,J)=(((1-FEMG(I))/TKMG(I,J))+(FEMG(I)/TKMG(I+1,J)))**
     +(-1)
  177    CONTINUE
  176 CONTINUE
    7 FORMAT(A3)
      END
C-----------------------------------------------------------------------
```

```
C ======================================================================
C TITLE: ENTRPIPE()
C
C DESCRIPTION:
C
C     This subroutine allows the user to enter all information
C     about the pipes - geometries
C
C ----------------------------------------------------------------------
      SUBROUTINE ENTRPIPE()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
```

```
     +           UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +           EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +           TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +           MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +           MAXC,MAXR)

      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z

      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +         regnum,regnumx,regnumy,regnodx,regnody


C ================================================================
C ENTERING PIPE INFORMATION
C================================================================
C DEFINING THE NUMBER OF PIPES
      CALL CLEARSCREEN($GCLEARSCREEN)
 1025 PRINT *, 'How many pipes are there in this system?'
      READ (*,610,ERR=1025) NPIPE
  610 FORMAT (I3)
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'How many of these pipes have reduced subgrids due to'
      PRINT *,'insulation board?'
      READ *,NHSG
      NFSG=NPIPE-NHSG
      IF(NFSG.GT.0.AND.NHSG.GT.0)THEN
        CALL CLEARSCREEN($GCLEARSCREEN)
        PRINT *,'SUBGRIDS'
        PRINT *,'-------------------------------------------------------
     +---------'
        PRINT *,'When entering information for subgrids, please number'
        PRINT 1031,'the full subgrids from 1 to',NFSG
        PRINT 1029,'and the reduced subgrids from',NFSG+1,'to',NPIPE
        READ *,DUMMY
 1029   FORMAT (A29,I3,A3,I3)
 1031   FORMAT (A27,I3)
      ENDIF
C ----------------------------------------------------------------
C LOOP - ENTERING INFORMATION ABOUT EACH PIPE:
      DO 185 ICOUNT=1,NPIPE
         PRINT 1030,'PIPE:',ICOUNT
 1030    FORMAT (A6,I3)
C ----------------------------------------------------------------
C Loop for entering the radius of each pipe in the system - inside
C and outside radii
         GOTO 415
  412    PRINT *
         PRINT *,'THE OUTSIDE DIAMETER MUST BE GREATER THAN THE INSIDE D
     +IAMETER'
         PRINT *,'PRESS <ENTER> TO CONTINUE'
         READ 2045, DUMMY
 2045    FORMAT(A1)
  415    CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'PIPE:',ICOUNT
         PRINT *,'-------------------------------------------------------
```

```
     +----------'
          PRINT 1035,'Enter the inside diameter of pipe',ICOUNT,'(meters)
     +'
          READ (*,*,ERR=415) IDIAM(ICOUNT)
          PRINT *
          IRADIUS(ICOUNT)=IDIAM(ICOUNT)/2.
  416     CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'PIPE:',ICOUNT
          PRINT *,'-----------------------------------------------------
     +----------'
          PRINT 1035,'Enter the outside diameter of pipe',ICOUNT,'(meters
     +)'
          READ (*,*,ERR=416) ODIAM(ICOUNT)
          ORADIUS(ICOUNT)=ODIAM(ICOUNT)/2.
          IF(ORADIUS(ICOUNT).LT.IRADIUS(ICOUNT))GOTO 412
  720     FORMAT (A21,I3)
  730     FORMAT (F8.4,A2)
 1035     FORMAT (A35,I3,A8)
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'PIPE:',ICOUNT
          PRINT *,'-----------------------------------------------------
     +----------'
          PRINT *,'Enter the thermal conductivity of the material of'
          PRINT 1040,'pipe',ICOUNT
 1040     FORMAT(A5,I3)
          READ *,TKPIP(ICOUNT)
C ---------------------------------------------------------------------
C Entering the insulation thickness (if any) on pipe
C (casing is left out because it is too small for the cylindrical grid
C  system and its relative contribution to resistance is negligible)
       IF(ICOUNT.GT.NFSG)THEN
          PRINT *
          PRINT *,'Note: Pipes with reduced grids cannot have insulation'
          PRINT *
          read (*,4)
    4  format(a3)
          THICKINS(ICOUNT)=0.0
          TKINSULN(ICOUNT)=0.0
          RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
       ELSE
  417     CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'PIPE:',ICOUNT
          PRINT *,'-----------------------------------------------------
     +----------'
          PRINT *,'Enter the thickness of the insulation around pipe (met
     +ers)'
          PRINT 1045,ICOUNT,'(meters)'
          READ (*,*,ERR=417) THICKINS(ICOUNT)
          RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
          IF(THICKINS(ICOUNT).EQ.0.0)THEN
             TKINSULN(ICOUNT)=0.0
          ENDIF
          PRINT *
 1045     FORMAT (I3,A8)
       ENDIF

C ---------------------------------------------------------------------
C Entering the thermal conductivities of the insulation material around
C the pipe
          IF(THICKINS(ICOUNT).GT.0.0)THEN
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'PIPE:',ICOUNT
          PRINT *,'-----------------------------------------------------
     +----------'
  419     PRINT *,'Enter the thermal conductivity of the insulation aroun
     +d'
          PRINT 1050,'pipe',ICOUNT,'IN W/mC'
          READ (*,*,ERR=419) TKINSULN(ICOUNT)
 1050     FORMAT (A5,I3,A8)
```

```
          ENDIF
C -----------------------------------------------------------------
C Enter the fluid temperature in each pipe
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'PIPE:',ICOUNT
          PRINT *,'-------------------------------------------------
     +----------'
          PRINT 1055,'What is the fluid temperature of pipe',ICOUNT,' (DEG.
     +C)'
          READ *,TFLUID(ICOUNT)
 1055     FORMAT(A38,I3,A8)
  185 CONTINUE
      END
C -----------------------------------------------------------------




C ==================================================================
C TITLE: ENTRFSG()
C
C DESCRIPTION:
C
C     This subroutine allows the user to define the FULL subgrids
C
C -----------------------------------------------------------------
      SUBROUTINE ENTRFSG()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z
```

```
      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


C ======================================================================
C CALCULATIONS FOR FULL EMBEDDED GRIDS AROUND EACH PIPE
C======================================================================
    7 FORMAT(A3)
C LOCATING SUBGRIDS
      DO 190 ICOUN=1,NFSG
        GOTO 525
         CALL CLEARSCREEN($GCLEARSCREEN)
  524    PRINT *
         PRINT *,'THE SIDE OF THE EMBEDDED GRID MUST BE LARGER THAN THE'
         PRINT *,'PIPE RADIUS:'
         PRINT *
  525    CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'EMBEDDED GRID OF PIPE',ICOUN
         PRINT *,'-----------------------------------------------------
     +----------'
         PRINT 1070,'Enter the distance from the center of pipe',ICOUN
         PRINT 1075,'to the side of the embedded grid around pipe',ICOUN
         PRINT *,'in meters - note: edges of subgrid must match up to '
         PRINT *,'interfaces of the 8 surrounding maingrid nodes'
         READ (*,*,ERR=525) BEDSIDE(ICOUN)
         IF(BEDSIDE(ICOUN).LT.ORADIUS(ICOUN))GOTO 524
 1070    FORMAT (A43,I3)
 1075    FORMAT(A45,I3)


C---------------------------------------------------------------------

         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'EMBEDDED GRIDS'
         PRINT *,'-----------------------------------------------------
     +----------'
         PRINT *,'Enter the main grid nodal coordinates of the top left'
         PRINT 1080,'corner node of the subgrid around pipe',ICOUN
 1080    FORMAT(A40,I3)

C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep
         PRINT *
         PRINT *,'Enter the x nodal coordinate:'
         READ *,DUMTLX(ICOUN)
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'EMBEDDED GRIDS'
         PRINT *,'-----------------------------------------------------
     +----------'
         PRINT *,'Enter the main grid nodal coordinates of the top left'
         PRINT 1080,'corner node of the subgrid around pipe',ICOUN
         PRINT *
         PRINT *,'Enter the y nodal coordinate:'
         READ *,DUMTLY(ICOUN)
         DUMBLX(ICOUN)=DUMTLX(ICOUN)
         DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
         DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
         DUMTRY(ICOUN)=DUMTLY(ICOUN)
         DUMBRX(ICOUN)=DUMTRX(ICOUN)
         DUMBRY(ICOUN)=DUMBLY(ICOUN)
  190 CONTINUE


C---------------------------------------------------------------------
```

```
C RULES FOR USER INFO INPUT:

C       -all embedded grids are square and take up 4 main grid nodes
C       -the center of pipe must be on the interfaces of main grid nodes -
C        at the meeting of interfaces of 4 main grid nodes
C       -sides of subgrid must coincide with interfaces of surrounding
C        eight main grid nodes
C       -the last circle must be tangent to all four sides
C       -first node on a radial line is ON the INNER pipe wall
C       -last node on a radial line is past the final circle, between
C        the last circle and the side of the subgrid
C       -outside radius of pipe is first interface circle - will be the
C        west interface of the second node
C       -if there is insulation, its outer boundary MUST coincide with
C        an interface circle (ie. must BE an interface between nodes
C       -outermost circle which touches four sides of sub-grid is an
C        interface circle not a nodal circle
C       -grid must be set up so that all nodes are midway b/w interfaces
C       -no subgrid may be at the edge of the main grid - there must be at
C        least two main grid nodes between the subgrid edge and the main
C        grid edge
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
C LOOP: DIMENSION CALCULATIONS FOR FULL SUBGRID NODES
C-------------------------------------------------------------------------

      PI=3.14159265359
      DO 195 ICOUNTER=1,NFSG


C SET UP SPACING FOR SUBGRID NODES:
C---------------------------------
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'Enter the number of nodes along a radial line of the'
         PRINT 1085,'sub-grid for pipe',ICOUNTER
         READ *,NEM(ICOUNTER)
 1085 FORMAT(A18,I3)

C Vector of spacing between sub-grid nodes along any radial line:
         CALL CLEARSCREEN($GCLEARSCREEN)
         DO 200 I=1,NEM(ICOUNTER)-2
            PRINT 1086,'Enter the distance between node',I,' and node',
     + I+1
 1086    FORMAT(A32,I5,A9,I5)
            READ *,EMSPACE(ICOUNTER,I)
  200    CONTINUE


C Set up vector of r's - radius to each node:
         R(ICOUNTER,1)=IRADIUS(ICOUNTER)
         DO 205 I=2,NEM(ICOUNTER)-1
            R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  205    CONTINUE
         R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                    (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))

C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
         DO 206 J=1,8
           DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                      R(ICOUNTER,NEM(ICOUNTER)-1)
  206 CONTINUE

C-------------------------------------------------------------------------
C DELR,DELTAR,DELTHETA,DELTATHETA - SUBGRIDS:

C Set up DELR for rest of sub-grid nodes:
```

```
         DO 215 I=1,NEM(ICOUNTER)-2
            DO 210 J=1,8
               DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
      WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
  210       CONTINUE
  215    CONTINUE

C DELR for last node - is distance between final subgrid node and
C adjacent main grid node

C     Z=distance from side of subgrid to adjacent node of main grid

         Z(ICOUNTER,1)=(DELTAX(DUMTRX(ICOUNTER)+1.))/2.
         Z(ICOUNTER,2)=(DELTAY(DUMTRY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,3)=(DELTAY(DUMTLY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,4)=(DELTAX(DUMTLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,5)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,6)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,7)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,8)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.

C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
         h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +               R(ICOUNTER,NEM(ICOUNTER)))

C     aa=the horizontal distance from the subgrid node to the center of
C        the main grid node (ie. lined up with adjacent main grid node
         aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +                *DSIN(PI/8.)))
         DO 220 J=1,8
         DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
     +                                    NTER))**2.)+(aa(ICOUNTER)**2.))
  220 CONTINUE

C DELTHETA are the same for all nodes - e and w DELTHETA are the same
C but note that for the last nodes there are only three sides to the
C control volume, so each node equations will include EITHER w or e
         DELTHETA=PI/4.

C DELTATH for all nodes is same:
         DELTATH=PI/4.

C DELTAR for 1st circle of nodes: (similar scheme to deltax above)
         DO 245 J=1,8
            DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
  245    CONTINUE

C DELTAR for second circle of nodes:
         DO 250 J=1,8
            DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                           1,J))*2.)
  250    CONTINUE

C DELTAR for interior nodes up to 2nd last node:
         DO 260 I=3,NEM(ICOUNTER)-1
            DO 255 J=1,8
               DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                             ICOUNTER,I-1,J)/2.))*2.
  255       CONTINUE
  260    CONTINUE


C DELTAR for last node of sub-grid
         DO 265 J=1,8
            DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                       (PI/8.)))-(BEDSIDE(ICOUNTER)))
  265    CONTINUE
```

```
C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID
C Thermal Conductivities AT subgrid nodes:
C     For 1st circle of nodes on inside of pipe:
         DO 270 J=1,8
            TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  270    CONTINUE
         DO 280 I=2,NEM(ICOUNTER)
            DO 275 J=1,8
               IF(R(ICOUNTER,I).LT.RADINS(ICOUNTER))THEN
                  TKSG(ICOUNTER,I,J)=TKINSULN(ICOUNTER)
               ELSE
                  TKSG(ICOUNTER,I,J)=Kg
               ENDIF
  275       CONTINUE
  280    CONTINUE

C-----------------------------------------------------------------------

C Setting up for thermal conductivities at interfaces of subgrid nodes:
C North interface conductivity for all FSG nodes is calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be the nodal conductivity itself.


      DO 266 I=1,NEM(ICOUNTER)-1
         DO 267 J=1,8
            FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                        ,I,J)
            TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                           ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                           I+1,J)))**(-1)
         WRITE (8,*) 'TKSGN(',ICOUNTER,I,J,')=',TKSGN(ICOUNTER,I,J)
  267    CONTINUE
  266 CONTINUE


C Find angle b/w final SG node and MG node:
      DO 264 J=1,8
         THET(ICOUNTER,NEM(ICOUNTER),J)=DASIN(aa(ICOUNTER)/DELR(ICOUNTER
     +                                 ,NEM(ICOUNTER),J))
         s(ICOUNTER,NEM(ICOUNTER),J)=(h(ICOUNTER)/DCOS(THET(ICOUNTER,
     +                               NEM(ICOUNTER),J)))
  264 CONTINUE


C Interface conductivity between final subgrid node and main grid node,
C (conductivity of north interface of final node):
         DO 286 J=1,8
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                  -s(ICOUNTER,NEM(ICOUNTER),J))/(DELR(ICOUNTER,NEM
     +                  (ICOUNTER),J))
  286 CONTINUE

          TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)
     +                                /TKMG(DUMTRX(ICOUNTER)+1,DUMTRY(ICOU
     +                                NTER))))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                                TKMG(DUMTRX(ICOUNTER),DUMTRY(ICOU
     +                                NTER)-1)))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                                TKMG(DUMTLX(ICOUNTER),DUMTLY(ICOUNTE
```

```
     +                                  R)-1)))**(-1)
           TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                 ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                 ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                                 TKMG(DUMTLX(ICOUNTER)-1,DUMTLY(ICOU
     +                                 NTER)))**(-1)
           TKSGN(ICOUNTER,NEM(ICOUNTER),5)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER),5))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                 ,5))+(FNSG(ICOUNTER,NEM(ICOUNTER),5)/
     +                                  TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                                  NTER)))**(-1)
           TKSGN(ICOUNTER,NEM(ICOUNTER),6)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER),6))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                 ,6))+(FNSG(ICOUNTER,NEM(ICOUNTER),6)/
     +                                  TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOU
     +                                  NTER)+1)))**(-1)
           TKSGN(ICOUNTER,NEM(ICOUNTER),7)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER),7))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                 ,7))+(FNSG(ICOUNTER,NEM(ICOUNTER),7)/
     +                                  TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOUNT
     +                                  ER)+1)))**(-1)
           TKSGN(ICOUNTER,NEM(ICOUNTER),8)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER),8))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                 ,8))+(FNSG(ICOUNTER,NEM(ICOUNTER),8)/
     +                                  TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                  NTER)))**(-1)
  195 CONTINUE
C END OF LOOP FOR INFO ABOUT SUBGRIDS
C-------------------------------------------------------------------------------
      END




C ==============================================================================
C TITLE: ENTRHSG()
C
C DESCRIPTION:
C
C     This subroutine allows the user to define the half subgrids
C
C ------------------------------------------------------------------------------
      SUBROUTINE ENTRHSG()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
```

```
     +           TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +           MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +           MAXC,MAXR)

      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


C ================================================================
C CALCULATIONS FOR REDUCED EMBEDDED GRIDS
C================================================================
      IF(NPIPE.GT.NFSG)THEN
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'INFORMATION FOR REDUCED SUB-GRID CALCULATIONS: PIPES'
         PRINT 1069,NFSG+1,'TO',NPIPE
      ENDIF
 1069 FORMAT(I3,A3,I3)
      PRINT *

C LOCATING SUBGRIDS

      DO 490 ICOUN=NFSG+1,NPIPE
    7 FORMAT (A3)
         PRINT *
         PRINT *
        GOTO 625
         PRINT *
  624    PRINT *,'THE SIDE OF THE EMBEDDED GRID MUST BE LARGER THAN THE'
         PRINT *,'PIPE RADIUS:'
         PRINT *
         PRINT *,'PRESS <ENTER> TO CONTINUE'
         READ 7,DUMMY
  625    CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REDUCED SUBGRID FOR PIPE',ICOUN
         PRINT *,'-------------------------------------------------
     +----------'
         PRINT 1070,'Enter the distance from the center of pipe',ICOUN
         PRINT 1075,'to the side of the embedded grid around pipe',ICOUN
         PRINT *,'in meters - note: edges of subgrid must match up to '
         PRINT *,'interfaces of the 8 surrounding maingrid nodes'
         READ (*,*,ERR=625) BEDSIDE(ICOUN)
         IF(BEDSIDE(ICOUN).LT.ORADIUS(ICOUN))GOTO 624
 1070    FORMAT (A43,I3)
 1075    FORMAT(A45,I3)


C-----------------------------------------------------------------

         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REDUCED SUBGRIDS'
         PRINT *,'-------------------------------------------------
     +----------'
         PRINT *,'Enter the main grid nodal coordinates of the top left'
```

```
          PRINT 1080,'corner node of the subgrid around pipe',ICOUN
 1080     FORMAT(A39,I3)
          PRINT *
C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep
          PRINT *,'Enter the x nodal coordinate:'
          READ *,DUMTLX(ICOUN)
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'REDUCED SUBGRID FOR PIPE',ICOUN
          PRINT *,'-------------------------------------------------------
     +----------'
          PRINT *,'Enter the main grid nodal coordinates of the top left'
          PRINT 1080,'corner node of the subgrid around pipe',ICOUN
          PRINT *
          PRINT *,'Enter the y nodal coordinate:'
          READ *,DUMTLY(ICOUN)
          DUMBLX(ICOUN)=DUMTLX(ICOUN)
          DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
          DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
          DUMTRY(ICOUN)=DUMTLY(ICOUN)
          DUMBRX(ICOUN)=DUMTRX(ICOUN)
          DUMBRY(ICOUN)=DUMBLY(ICOUN)
  490 CONTINUE

C-------------------------------------------------------------------------

C RULES FOR USER INFO INPUT:

C        -all embedded grids are square and take up 4 main grid nodes
C        -the center of pipe must be on the interfaces of main grid nodes -
C         at the meeting of interfaces of 4 main grid nodes
C        -sides of subgrid must coincide with interfaces of surrounding
C         eight main grid nodes
C        -the last circle must be tangent to all four sides
C        -first node on a radial line is ON the INNER pipe wall
C        -last node on a radial line is past the final circle, between
C         the last circle and the side of the subgrid
C        -outside radius of pipe is first interface circle - will be the
C         west interface of the second node
C        -if there is insulation, its outer boundary MUST coincide with
C         an interface circle (ie. must BE an interface between nodes
C        -outermost circle which touches four sides of sub-grid is an
C         interface circle not a nodal circle
C        -grid must be set up so that all nodes are midway b/w interfaces
C        -no subgrid may be at the edge of the main grid - there must be at
C         least two main grid nodes between the subgrid edge and the main
C         grid edge
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
C LOOP: DIMENSION CALCULATIONS FOR REDUCED SUBGRID NODES
C-------------------------------------------------------------------------
       PI=3.14159265359
       DO 495 ICOUNTER=NFSG+1,NPIPE,1
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'REDUCED SUBGRID'
          PRINT *,'-------------------------------------------------------
     +----------'
          PRINT *,'Enter the number of nodes along one of the four'
          PRINT *,'radial lines of the half cylindrical sub-grid for'
          PRINT 4084,'pipe',ICOUNTER
          READ *,NEM(ICOUNTER)
          CALL CLEARSCREEN($GCLEARSCREEN)
          PRINT *,'REDUCED SUBGRID'
          PRINT *,'-------------------------------------------------------
     +----------'
          PRINT *,'Enter the height from the pipe center to the bottom'
          PRINT 4085,'of the insulation board (meters) for pipe',
     +ICOUNTER
```

```
         READ *,hh(ICOUNTER)
 4084 FORMAT(A5,I3)
 4085 FORMAT(A43,I3)

C Vector of spacing between sub-grid nodes along any radial line:
         CALL CLEARSCREEN($GCLEARSCREEN)
         PRINT *,'REDUCED SUBGRID'
         PRINT *,'-------------------------------------------------
     +----------'
         DO 400 I=1,NEM(ICOUNTER)-2
            PRINT *,'Enter the distance between node',I,' and node',I+1
            PRINT *,'along a radial line of the half cylindrical grid:'
            READ *,EMSPACE(ICOUNTER,I)
 5103       FORMAT(A32,I5,A9,I5)
            PRINT *
  400    CONTINUE


C-------------------------------------------------------------------------
C R,DELR,DELTAR,DELTHETA,DELTATHETA - CYLINDRICAL SECTION OF SUBGRIDS:


C Set up vector of r's - radius to each node:
         R(ICOUNTER,1)=IRADIUS(ICOUNTER)
         DO 405 I=2,NEM(ICOUNTER)-1
            R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  405    CONTINUE
         R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                    (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))
C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
         DO 406 J=1,4
           DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                      R(ICOUNTER,NEM(ICOUNTER)-1)
  406 CONTINUE


C Set up DELR for rest of sub-grid nodes:
         DO 415 I=1,NEM(ICOUNTER)-2
            DO 410 J=1,4
               DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
      WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
  410       CONTINUE
  415    CONTINUE

C DELR for last node of radial lines - is distance between final subgrid
C node and adjacent main grid node - for the 4 final nodes along radial
C lines 1 and 4 (cylindrical sedtion of subgrid)

C     Z=distance from side of subgrid to adjacent node of main grid

C       (calculated for MG nodes beside cylindrical part of HSG)

         Z(ICOUNTER,1)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.
         Z(ICOUNTER,2)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,3)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,4)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
         h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +              R(ICOUNTER,NEM(ICOUNTER)))
C     aa=the horizontal distance from the subgrid node to the center of
C       the main grid node (ie. lined up with adjacent main grid node
         aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +                *DSIN(PI/8.)))

         DO 420 J=1,4
         DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
```

```
     +                                        NTER))**2.)+(aa(ICOUNTER)**2.))
  420 CONTINUE

C DELTHETA are the same for all nodes
         DELTHETA=PI/4.

C DELTATH for all nodes is same:
         DELTATH=PI/4.

C DELTAR for 1st circle of nodes:
         DO 445 J=1,4
            DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
  445    CONTINUE
C DELTAR for second circle of nodes:
         DO 450 J=1,4
            DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                                1,J))*2.)
  450    CONTINUE
C DELTAR for interior nodes up to 2nd last node:
         DO 460 I=3,NEM(ICOUNTER)-1
            DO 455 J=1,4
               DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                                  ICOUNTER,I-1,J)/2.))*2.
  455       CONTINUE
  460    CONTINUE
C DELTAR for last node of sub-grid
         DO 465 J=1,4
            DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                        (PI/8.)))-(BEDSIDE(ICOUNTER)))
  465    CONTINUE


C-------------------------------------------------------------------------
C SET UP GEOMETRIES FOR RECTANGULAR NODES OF HSG: there will be NEM-1
C nodes, and they will be on lines 5 and 6:
C-------------------------------------------------------------------------

C e is the horizontal distance from the 1st node (inside pipe) to the
C second node.  c is the vertical distance from the 1st node to the side
C of the cylindrical grid (pipe centerline level) - they are used to
C calculate the distance from the 1st to the 2nd node along lines 5&6
      cc(ICOUNTER)=R(ICOUNTER,1)*DSIN(PI/4.)
      e(ICOUNTER)=R(ICOUNTER,2)-(R(ICOUNTER,1)*DCOS(PI/4.))

C DEL for the 2 first nodes in the rectangular section of the subgrid
C (distance between 1st and 2nd nodes)
      DEL(ICOUNTER,5)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
     +                     (ICOUNTER))**2)
      PRINT *,'DEL(ICOUNTER,5)=',DEL(ICOUNTER,5)
      read 7,dummy
      WRITE (8,*) 'DEL(ICOUNTER,5)=',DEL(ICOUNTER,5)
      DEL(ICOUNTER,6)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
     +                     (ICOUNTER))**2)
      PRINT *,'DEL(ICOUNTER,6)=',DEL(ICOUNTER,6)
      read 7,dummy
C DELTAXHS, DELTAYHS, DELXHSG, DELYHSG - for rectangular subgrid nodes
C -will be same for lines 5 and 6
            DO 467 I=2,NEM(ICOUNTER)-1
               DELTAXHS(ICOUNTER,I)=(DELTAR(ICOUNTER,I,4))
               DELTAYHS(ICOUNTER,I)=hh(ICOUNTER)
  467       CONTINUE
            DO 468 I=2,NEM(ICOUNTER)-2
               DELXHSG(ICOUNTER,I)=R(ICOUNTER,I+1)-R(ICOUNTER,I)

  468       CONTINUE
C DELYHSG - will be same for lines 5 and 6 (due to spacing rules)
         DO 469 I=2,NEM(ICOUNTER)-1
            DELYHSG(ICOUNTER,I)=(hh(ICOUNTER)/2.)+(DELTAY(DUMTLY(ICOUNT
     +                           ER)-1)/2.)
```

```
  469     CONTINUE
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID CYLINDRICAL NODES
C-----------------------------------------------------------------------


C Thermal Conductivities AT subgrid nodes:
C----------------------------------------
C     For 1st circle of nodes on inside of pipe:
          DO 470 J=1,4
             TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  470     CONTINUE
C k for remaining 2 nodes inside pipe wall:
          TKSG(ICOUNTER,1,5)=TKPIP(ICOUNTER)
          TKSG(ICOUNTER,1,6)=TKPIP(ICOUNTER)
C ======TO INSERT RING OF AIR AROUND PIPE CHANGE Kg TO KAIR===
          DO 473 J=1,4
             TKSG(2,2,J)=Kg
             TKSG(1,2,J)=Kg
             PRINT *
  473     CONTINUE
          PRINT *
          PRINT *
C=================================================
C     For rest of nodes:
C ***NOTE: No insulation allowed in reduced subgrids!***
          DO 480 I=3,NEM(ICOUNTER)
             DO 475 J=1,4
                TKSG(ICOUNTER,I,J)=Kg
  475        CONTINUE
  480     CONTINUE
C=======TO INSERT AIR SPACE ABOVE PIPE - BETWEEN PIPE AND INSULN
C       BOARD - CHANGE kg TO kAIR=========  ==================

          DO 477 J=5,6
             TKSG(1,2,J)=Kg
             TKSG(2,2,J)=Kg
  477     CONTINUE

          DO 481 I=3,NEM(ICOUNTER)-1
             DO 476 J=5,6
                TKSG(1,I,J)=Kg
                TKSG(2,I,J)=Kg
  476        CONTINUE
  481     CONTINUE
C=======================================================================

C-----------------------------------------------------------------------

C Thermal conductivities at interfaces of subgrid nodes:
C-----------------------------------------------------------------------

C HSG radial lines 1 to 4:
C------------------------
C North interface conductivity for HSG nodes along lines 1-4 are calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be the nodal conductivity
       DO 478 I=1,NEM(ICOUNTER)-1
          DO 479 J=1,4
             FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                                ,I,J)
             TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                               ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                               I+1,J)))**(-1)
          WRITE (8,*) 'TKSGN(',ICOUNTER,I,J,')=',TKSGN(ICOUNTER,I,J)
  479     CONTINUE
  478  CONTINUE
```

```
      print *, 'TKSGN(',ICOUNTER,1,1,')=',TKSGN(ICOUNTER,1,1)
      read 7,dummy

C For HSG lines 5 and 6:
C-----------------------

C 1st nodes on lines 5&6:
      DO 482 J=5,6
         FNSG(ICOUNTER,1,J)=(DEL(ICOUNTER,J)-DELTAR(ICOUNTER,1,4))/DEL
     +                        (ICOUNTER,J)
         TKSGN(ICOUNTER,1,J)=(((1-FNSG(ICOUNTER,1,J))/TKSG(ICOUNTER,1,J)
     +                       )+(FNSG(ICOUNTER,1,J)/TKSG(ICOUNTER,2,J)))
     +                       **(-1)
      WRITE (8,*) 'TKSGN(',ICOUNTER,1,J,')=',TKSGN(ICOUNTER,1,J)
      print *, 'TKSGN(',ICOUNTER,1,J,')=',TKSGN(ICOUNTER,1,J)
      read 7,dummy
  482 CONTINUE
C North interface conductivities for rest of nodes along lines 5&6 -
C rectangular formulation from Patankar


C For nodes 2 to NEM-2:
      DO 474 I=2,NEM(ICOUNTER)-2
         DO 487 J=5,6
            FNSG(ICOUNTER,I,J)=(DELTAXHS(ICOUNTER,I+1)/2.)/DELXHSG
     +                              (ICOUNTER,I)

            TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,
     +                           I,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER
     +                           ,I+1,J)))**(-1)
  487    CONTINUE
  474 CONTINUE

C For final node NEM-1: (borders onto MG node)
      FNSG(ICOUNTER,NEM(ICOUNTER)-1,5)=((DELTAX(DUMTLX(ICOUNTER)-1)/2.)
     +                                 /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                 -1)/2.)+(DELTAX(DUMTLX(ICOUNTER)-1
     +                                 )/2.)))

      FNSG(ICOUNTER,NEM(ICOUNTER)-1,6)=((DELTAX(DUMTRX(ICOUNTER)+1)/2.)
     +                                 /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                 -1)/2.)+(DELTAX(DUMTRX(ICOUNTER)+1
     +                                 )/2.)))


      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,5)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                 )-1,5))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                 )-1,5))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                 ER)-1,5)/TKMG(DUMTLX(ICOUNTER)-1,
     +                                 DUMTLY(ICOUNTER))))**(-1)


      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,6)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                 )-1,6))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                 )-1,6))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                 ER)-1,6)/TKMG(DUMTRX(ICOUNTER)+1,
     +                                 DUMTRY(ICOUNTER))))**(-1)


C East interface for nodes 2 to NEM-1 along lines 5&6 of HSG: border
C onto MG nodes (from Patankar)
      DO 488 I=2,NEM(ICOUNTER)-1
         FEHSG(ICOUNTER,I,5)=(DELTAY(DUMTLY(ICOUNTER)-1)/2.)/DELYHSG
     +                        (ICOUNTER,I)
         FEHSG(ICOUNTER,I,6)=(DELTAY(DUMTRY(ICOUNTER)-1)/2.)/DELYHSG
     +                        (ICOUNTER,I)
         TKSGE(ICOUNTER,I,5)=(((1-FEHSG(ICOUNTER,I,5))/TKSG(ICOUNTER,I,5
     +                        ))+(FEHSG(ICOUNTER,I,5)/TKMG(DUMTLX(ICOUNTE
     +                        R),DUMTLY(ICOUNTER)-1)))**(-1)
```

```
          TKSGE(ICOUNTER,I,6)=(((1-FEHSG(ICOUNTER,I,6))/TKSG(ICOUNTER,I,6
     +                         ))+(FEHSG(ICOUNTER,I,6)/TKMG(DUMTRX(ICOUNTE
     +                         R),DUMTRY(ICOUNTER)-1)))**(-1)
  488 CONTINUE
C Interface conductivity between final subgrid nodes along radial lines 1-4
C and main grid node:

C Find angle b/w final SG node and MG node:
      DO 494 J=1,4
         THET(ICOUNTER,NEM(ICOUNTER),J)=DASIN(aa(ICOUNTER)/DELR(ICOUNTER
     +                                  ,NEM(ICOUNTER),J))
         s(ICOUNTER,NEM(ICOUNTER),J)=(h(ICOUNTER)/DCOS(PI/8.))
  494 CONTINUE


C Interface conductivity between final subgrid node and main grid node,
C (conductivity of north interface of final node):
         DO 496 J=1,4
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                  -s(ICOUNTER,NEM(ICOUNTER),J))/(DELR(ICOUNTER,NEM
     +                  (ICOUNTER),J))
  496 CONTINUE


          TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)/
     +                                TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                NTER))))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                               TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOU
     +                               NTER)+1)))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                               TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOUNTE
     +                               R)+1)))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                               TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                               NTER))))**(-1)

C-------------------------------------------------------------------------
  495 CONTINUE
C END OF LOOP FOR INFO ABOUT REDUCED SUBGRIDS
C-------------------------------------------------------------------------
      END




C ***************************************************************
C TITLE: FSGCSTP()
C
C DESCRIPTION:
C
C     This subroutine calculates the sub-grid constant parameters
C     for TDMA for FULL SUBGRIDS
C
C ---------------------------------------------------------------
```

```
      SUBROUTINE FSGCSTP()
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


C-----------------------------------------------------------------------
C TDMA PARAMETER CALCULATION FOR ALL SUBGRID NODES - PREP FOR LINE BY
C LINE SOLUTION ALGORITHM: CONSTANT PARAMETER CALCULATION ONLY (A,B,C AND
C FIRST NODES)
C-----------------------------------------------------------------------
    7 FORMAT(A3)
      DO 5999 ICOUNTER=1,NFSG
C FIRST nodes along any TDMA - ie. first circle nodes

         DO 6000 J=1,8
            TS(ICOUNTER,1,J)=TFLUID(ICOUNTER)
            QS(ICOUNTER,1,J)=TS(ICOUNTER,1,J)
            PS(ICOUNTER,1,J)=0.0
 6000    CONTINUE

C INTERIOR nodes:

         DO 6005 I=2,NEM(ICOUNTER)-1
            DO 6010 J=1,8
            BS(ICOUNTER,I,J)=(((R(ICOUNTER,I)+(DELTAR(ICOUNTER,I,J)/2.))
     +                        *TKSGN(ICOUNTER,I,J)*DELTATH)
```

```
     +                          /(DELR(ICOUNTER,I,J)))
           CS(ICOUNTER,I,J)=(((R(ICOUNTER,I)-(DELTAR(ICOUNTER,I,J)/2.))
     +                          *TKSGN(ICOUNTER,I-1,J)*DELTATH)
     +                          /(DELR(ICOUNTER,I-1,J)))
           AS(ICOUNTER,I,J)=BS(ICOUNTER,I,J)+CS(ICOUNTER,I,J)
     +                          +((TKSG(ICOUNTER,I,J)*DELTAR(ICOUNTER,I,J))
     +                          /(R(ICOUNTER,I)*DELTHETA))
     +                          +((TKSG(ICOUNTER,I,J)*DELTAR(ICOUN
     +                          TER,I,J))/(R(ICOUNTER,I)*DELTHETA))
 6010       CONTINUE
 6005    CONTINUE


C-------------------------------------------------------------------------
C For final nodes of subgrid, control volumes have irregular geometry
C with only three sides.  Interface to e (or w) neighbour is the distance
C between the final circle in the subgrid and the corner of the subrid:
C Since this distance will be the same for all 8 final nodes, node any of
C the 8 end nodes may be used to calculate this distance - value is called
C 'DELTARF' and is the same for all 8 end nodes of the subgrid

         DELTARF(ICOUNTER)=((BEDSIDE(ICOUNTER)/(DCOS(PI/4.)))-(R(I
     +                          COUNTER,NEM(ICOUNTER)-1)+(DELTAR(ICOUN
     +                          TER,NEM(ICOUNTER)-1,1)/2.)))
C-------------------------------------------------------------------------

C TDMA coefficients for final subgrid nodes:

         DO 6020 J=1,8
            BS(ICOUNTER,NEM(ICOUNTER),J)=0.0
            CS(ICOUNTER,NEM(ICOUNTER),J)=((TKSGN(ICOUNTER,NEM(ICOUNTER)-
     +                          1,J)*(R(ICOUNTER,NEM(ICOUNTER)-1)+DELTAR
     +                          (ICOUNTER,NEM(ICOUNTER)-1,J)/2.)*DELTATH)
     +                          /(DELR(ICOUNTER,NEM(ICOUNTER)-1,J)))

            AS(ICOUNTER,NEM(ICOUNTER),J)=((TKSG(ICOUNTER,NEM(ICOUNTER),J
     +                          )*DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOU
     +                          NTER))*DELTHETA))
     +                          +CS(ICOUNTER,NEM(ICOUNTER),J)
     +                          +((TKSGN(ICOUNTER,NEM(ICOUNTER),J)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,J))
 6020    CONTINUE
 5999 CONTINUE

C End of calculations of TDMA constant coefficients of full subgrid nodes
C-------------------------------------------------------------------------
      END




C ========================================================================
C TITLE: HSGCSTP()
C
C DESCRIPTION:
C
C     This subroutine calculates the sub-grid constant parameters
c     for TDMA
C
C ------------------------------------------------------------------------
      SUBROUTINE HSGCSTP()
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
```

```
     +           XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +           X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +           IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +           R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +           ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +           ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +           AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +           UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +           X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +           AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +           MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +           ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +           XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +           C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +           QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +           QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +           UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +           EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +           TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +           MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +           MAXC,MAXR)

      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z

      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


C-----------------------------------------------------------------------
C TDMA PARAMETER CALCULATION FOR ALL SUBGRID NODES - PREP FOR LINE BY
C LINE SOLUTION ALGORITHM: CONSTANT PARAMETER CALCULATION ONLY (A,B,C AND
C FIRST NODES)
C-----------------------------------------------------------------------
      DO 4999 ICOUNTER=NFSG+1,NPIPE
    7 FORMAT(A3)

C FIRST nodes along any TDMA - ie. first circle nodes

         DO 5000 J=1,6
            TS(ICOUNTER,1,J)=TFLUID(ICOUNTER)
            QS(ICOUNTER,1,J)=TS(ICOUNTER,1,J)
            PS(ICOUNTER,1,J)=0.0
 5000    CONTINUE

C INTERIOR CYLINDRICAL SECTION nodes:

         DO 5005 I=2,NEM (ICOUNTER)-1
            DO 5010 J=2,3
            BS(ICOUNTER,I,J)=(((R(ICOUNTER,I)+(DELTAR(ICOUNTER,I,J)/2.))
     +                        *TKSGN(ICOUNTER,I,J)*DELTATH)
     +                        /(DELR(ICOUNTER,I,J)))
            CS(ICOUNTER,I,J)=(((R(ICOUNTER,I)-(DELTAR(ICOUNTER,I,J)/2.))
     +                        *TKSGN(ICOUNTER,I-1,J)*DELTATH)
     +                        /(DELR(ICOUNTER,I-1,J)))
            AS(ICOUNTER,I,J)=BS(ICOUNTER,I,J)+CS(ICOUNTER,I,J)
```

```
     +                          +((TKSG(ICOUNTER,I,J)*DELTAR(ICOU
     +                          NTER,I,J))/(R(ICOUNTER,I)*DELTHETA))
     +                          +((TKSG(ICOUNTER,I,J)*DELTAR(ICOUN
     +                          TER,I,J))/(R(ICOUNTER,I)*DELTHETA))
 5010        CONTINUE
 5005    CONTINUE

C For radial line #1
         DO 5006 I=2,NEM (ICOUNTER)-1

            BS(ICOUNTER,I,1)=(((R(ICOUNTER,I)+(DELTAR(ICOUNTER,I,1)/2.))
     +                          *TKSGN(ICOUNTER,I,1)*DELTATH)
     +                          /(DELR(ICOUNTER,I,1)))
            CS(ICOUNTER,I,1)=(((R(ICOUNTER,I)-(DELTAR(ICOUNTER,I,1)/2.))
     +                          *TKSGN(ICOUNTER,I-1,1)*DELTATH)
     +                          /(DELR(ICOUNTER,I-1,1)))
            AS(ICOUNTER,I,1)=BS(ICOUNTER,I,1)+CS(ICOUNTER,I,1)
     +                          +((TKSG(ICOUNTER,I,1)*DELTAR(ICOU
     +                          NTER,I,1))/(R(ICOUNTER,I)*DELTHETA))
     +                          +((TKSG(ICOUNTER,I,1)*DELTAR(ICOU
     +                          NTER,I,1))/((R(ICOUNTER,I)*PI/8.)+(hh
     +                          (ICOUNTER)/2.)))
 5006    CONTINUE

C For radial line #4
         DO 5007 I=2,NEM (ICOUNTER)-1

            BS(ICOUNTER,I,4)=(((R(ICOUNTER,I)+(DELTAR(ICOUNTER,I,4)/2.))
     +                          *TKSGN(ICOUNTER,I,4)*DELTATH)
     +                          /(DELR(ICOUNTER,I,4)))
            CS(ICOUNTER,I,4)=(((R(ICOUNTER,I)-(DELTAR(ICOUNTER,I,4)/2.))
     +                          *TKSGN(ICOUNTER,I-1,4)*DELTATH)
     +                          /(DELR(ICOUNTER,I-1,4)))
            AS(ICOUNTER,I,4)=BS(ICOUNTER,I,4)+CS(ICOUNTER,I,4)
     +                          +((TKSG(ICOUNTER,I,4)*DELTAR(ICOU
     +                          NTER,I,4))/(R(ICOUNTER,I)*DELTHETA))
     +                          +((TKSG(ICOUNTER,I,4)*DELTAR(ICOU
     +                          NTER,I,4))/((R(ICOUNTER,I)*PI/8.)+(hh
     +                          (ICOUNTER)/2.)))
 5007    CONTINUE


C-------------------------------------------------------------------------
C For final nodes of cylindrical section of subgrid, control volumes have
C irregular geometry with only three sides.Interface to e (or w) neighbour
C is the distance between the final circle in the subgrid ar  he corner
C of the subrid:
C Since this distance will be the same for all 4 final nodes, node any of
C the 4 end nodes may be used to calculate this distance - value is called
C 'DELTARF' and is the same for all 4 end nodes of the subgrid

         DELTARF(ICOUNTER)=((BEDSIDE(ICOUNTER)/(DCOS(PI/4.)))-(R(I
     +                          COUNTER,NEM(ICOUNTER)-1)+(DELTAR(ICOUN
     +                          TER,NEM(ICOUNTER)-1,1)/2.)))

C-------------------------------------------------------------------------

C TDMA coefficients for final subgrid nodes (cylindrical section):

         DO 5020 J=1,4
            BS(ICOUNTER,NEM(ICOUNTER),J)=0.0
            CS(ICOUNTER,NEM(ICOUNTER),J)=((TKSGN(ICOUNTER,NEM(ICOUNTER)-
     +                          1,J)*(R(ICOUNTER,NEM(ICOUNTER)-1)
     +                          +DELTAR(ICOUNTER,NEM(ICOUNTER)-1,J)/
     +                          2.)*DELTATH)/(DELR(ICOUNTER,NEM(ICOUNTER
     +                          )-1,J)))

            AS(ICOUNTER,NEM(ICOUNTER),J)=((TKSG(ICOUNTER,NEM(ICOUNT
     +                          ER),J)*DELTARF(ICOUNTER))/(R(ICOUNTER
```

```
     +                          ,NEM(ICOUNTER))*DELTHETA))
     +                          +CS(ICOUNTER,NEM(ICOUNTER),J)
     +                          +((TKSGN(ICOUNTER,NEM(ICOUNTER),J)*BEDSI
     +                          DE(ICOUNTER))/(DELR(ICOUNTER,NEM(ICOUNTER
     +                          ),J)))
 5020    CONTINUE


C-------------------------------------------------------
C SET UP PARAMETERS FOR LINES 5&6: RECTANGULAR SECTION
C-------------------------------------------------------

C Equations for 2nd node:
         DO 5021 J=5,6

            BS(ICOUNTER,2,J)=(TKSGN(ICOUNTER,2,J)*hh(ICOUNTER))/(DELXHSG
     +                      (ICOUNTER,2))
            CS(ICOUNTER,2,J)=((TKSGN(ICOUNTER,1,J)*(R(ICOUNTER,2)-(DELTA
     +                      R(ICOUNTER,2,4)/2.))*(PI/2.))/(DEL(ICOUNTER
     +                      ,J)))
            AS(ICOUNTER,2,J)=BS(ICOUNTER,2,J)+CS(ICOUNTER,2,J)
     +                      +((TKSG(ICOUNTER,2,J)*(hh(ICOUNTER)-ORADIUS
     +                      (ICOUNTER)))/(2*R(ICOUNTER,2)))
     +                      +((TKSG(ICOUNTER,2,J)*DELTAR(ICOUNTER,2,4))
     +                      /((R(ICOUNTER,2)*(PI/8.))+(hh(ICOUNTER)/2.)
     +                      ))
     +                      +((TKSGE(ICOUNTER,2,J)*(R(ICOUNTER,2)+(
     +                      DELTAR(ICOUNTER,2,4)/2.)))/(DELYHSG(ICOUNT
     +                      ER,2)))
 5021    CONTINUE

C Equations for interior rectangular nodes:
         DO 5023 I=3, NEM(ICOUNTER)-2
            DO 5022 J=5,6
               BS(ICOUNTER,I,J)=(TKSGN(ICOUNTER,I,J)*hh(ICOUNTER))/
     +                          DELXHSG(ICOUNTER,I)
               CS(ICOUNTER,I,J)=(TKSGN(ICOUNTER,I-1,J)*hh(ICOUNTER))/
     +                          DELXHSG(ICOUNTER,I-1)
               AS(ICOUNTER,I,J)=BS(ICOUNTER,I,J)+CS(ICOUNTER,I,J)
     +                          +((TKSG(ICOUNTER,I,J)*DELTAXHS(ICOUNTER
     +                          ,I))/((R(ICOUNTER,I)*(PI/8.))+(hh(ICO
     +                          UNTER)/2.)))
     +                          +((TKSGE(ICOUNTER,I,J)*DELTAXHS(ICOUNTER
     +                          ,I))/(DELYHSG(ICOUNTER,I)))
 5022       CONTINUE
 5023    CONTINUE

C Equations for final rectangular subgrid nodes (NEM-1):
C FOR LINE 5
            BS(ICOUNTER,NEM(ICOUNTER)-1,5)=0.0
            CS(ICOUNTER,NEM(ICOUNTER)-1,5)=(TKSGN(ICOUNTER,NEM(ICOUNTER)
     +                      -2,5)*hh(ICOUNTER))/DELXHSG(ICOUNTER,NEM
     +                      (ICOUNTER)-2)

            AS(ICOUNTER,NEM(ICOUNTER)-1,5)=CS(ICOUNTER,NEM(ICOUNTER)-1,
     +                      5)+((TKSGN(ICOUNTER,NEM(ICOUNTER)-1,5)*hh
     +                      (ICOUNTER))/((DELTAXHS(ICOUNTER,NEM(ICOUNTER
     +                      )-1)/2.)+(DELTAX(DUMTLX(ICOUNTER)-1)/2.)))
     +                      +((TKSGE(ICOUNTER,NEM(ICOUNTER)-1,5)*DELTAXH
     +                      S(ICOUNTER,NEM(ICOUNTER)-1))/(DELYHSG(ICOUNT
     +                      ER,NEM(ICOUNTER)-1)))
     +                      +((TKSG(ICOUNTER,NEM(ICOUNTER)-1,5)*DELTAXHS
     +                      (ICOUNTER,NEM(ICOUNTER)-1))/((R(ICOUNTER,NEM
     +                      (ICOUNTER)-1)*(PI/8.))+(hh(ICOUNTER)/2.)))

C FOR LINE 6
            BS(ICOUNTER,NEM(ICOUNTER)-1,6)=0.0
            CS(ICOUNTER,NEM(ICOUNTER)-1,6)=(TKSGN(ICOUNTER,NEM(ICOUNTER)
```

```
     +                        -2,6)*hh(ICOUNTER))/DELXHSG(ICOUNTER,NEM
     +                        (ICOUNTER)-2)

          AS(ICOUNTER,NEM(ICOUNTER)-1,6)=CS(ICOUNTER,NEM(ICOUNTER)-1,
     +                        6)+((TKSGN(ICOUNTER,NEM(ICOUNTER)-1,6)*hh
     +                        (ICOUNTER))/((DELTAXHS(ICOUNTER,NEM(ICOUNTER
     +                        )-1)/2.)+(DELTAX(DUMTRX(ICOUNTER)+1)/2.)))
     +                        +((TKSGE(ICOUNTER,NEM(ICOUNTER)-1,6)*DELTAXH
     +                        S(ICOUNTER,NEM(ICOUNTER)-1))/(DELYHSG(ICOUNT
     +                        ER,NEM(ICOUNTER)-1)))
     +                        +((TKSG(ICOUNTER,NEM(ICOUNTER)-1,6)*DELTAXHS
     +                        (ICOUNTER,NEM(ICOUNTER)-1))/((R(ICOUNTER,NEM
     +                        (ICOUNTER)-1)*(PI/8.))+(hh(ICOUNTER)/2.)))

 4999 CONTINUE


C End of calculations of TDMA constant coefficients of all subgrid nodes
C-------------------------------------------------------------------------
      END




C ===========================================================
C TITLE: MGCSTP()
C
C DESCRIPTION:
C
C     This subroutine calculates the main grid constant parameters
c     for TDMA
C
C -------------------------------------------------------------
      SUBROUTINE MGCSTP()
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
```

```
     +          DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +          DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +          FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +          NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +          REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +          ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +          TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z

      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +          regnum,regnumx,regnumy,regnodx,regnody


C-------------------------------------------------------------------------
C TDMA PARAMETER CALCULATION FOR ALL MAIN GRID NODES - PREP FOR LINE BY
C LINE SOLUTION ALGORITHM: CONSTANT COEFFICIENTS ONLY (A,B,C)
C-------------------------------------------------------------------------
    7 FORMAT(A3)
C equations for first nodes on constant y=1 line (ie. TDMA parameters
C for all nodes along y=1 line - 1st node in each TDMA line)
      DO 8000 I=1,N
         Q(I,1)=T(I,1)
         P(I,1)=0.0
 8000 CONTINUE

C EQUATIONS FOR FIRST TDMA LINE - LEFT HAND SIDE ROW FROM (1,1) to
C (1,M) - boundary condition is q=0 on side nodes:

      DO 8005 J=2,M-1
         B(1,J)=((TKMGN(1,J+1)*DELTAX(1))/(DELY(J)))
         C(1,J)=((TKMGN(1,J)*DELTAX(1))/(DELY(J-1)))
         A(1,J)=B(1,J)+C(1,J)+((TKMGE(1,J)*DELTAY(J))/(DELX(1)))
 8005  CONTINUE

C Equation for bottom corner node - for first TDMA line along x=1
      B(1,M)=0.0
      C(1,M)=((TKMGN(1,M)*DELTAX(1))/(DELY(M-1)))
      A(1,M)=C(1,M)+((TKMGE(1,M)*DELTAY(M))/(DELX(1)))


C-------------------------------------------------------
C EQUATIONS FOR INTERIOR NODES - ie. LINES I=2 THROUGH I=N-1

C Interior nodes (J=2,M-1)
      DO 8020 I=2,N-1
         DO 8025 J=2,M-1
            B(I,J)=((TKMGN(I,J+1)*DELTAX(I))/(DELY(J)))
            C(I,J)=((TKMGN(I,J)*DELTAX(I))/(DELY(J-1)))
            A(I,J)=B(I,J)+C(I,J)+((TKMGE(I,J)*DELTAY(J))/(DELX(I)))+
     +              ((TKMGE(I-1,J)*DELTAY(J))/(DELX(I-1)))
 8025 CONTINUE

C Bottom node
      B(I,M)=0.0
      C(I,M)=((TKMGN(I,M)*DELTAX(I))/DELY(M-1))
      A(I,M)=C(I,M)+((TKMGE(I,M)*DELTAY(M))/(DELX(I)))+((TKMGE(I-1,M)
     +       *DELTAY(M))/(DELX(I-1)))
 8020 CONTINUE

C-------------------------------------------------------------------------
C CHANGE COEFFICIENTS FOR NODES AROUND SUBGRIDS
C (will all be interior nodes - rule)
C-------------------------------------------------------------------------

C----------------------
C AROUND FULL SUBGRIDS:
C----------------------
      DO 8026 IC=1,NFSG
```

```
C Beside top right corner dummy node - beside subgrid node 1

      A(DUMTRX(IC)+1,DUMTRY(IC))=((TKMGE(DUMTRX(IC)+1,DUMTRY(IC))*
     +                DELTAY(DUMTRY(IC)))/(DELX(DUMTRX(IC)+1)))
     +                +((TKMGN(DUMTRX(IC)+1,DUMTRY(IC)+1)*DELTAX(DUMTR
     +                X(IC)+1))/(DELY(DUMTRY(IC))))
     +                +((TKMGN(DUMTRX(IC)+1,DUMTRY(IC))*DELTAX
     +                (DUMTRX(IC)+1))/(DELY(DUMTRY(IC)-1)))
     +                +((TKSGN(IC,NEM(IC),1)*DELTAY(DUMTRY(IC)
     +                ))/DELR(IC,NEM(IC),1))

C Beside bottom right corner node - beside subgrid node 8

      A(DUMBRX(IC)+1,DUMBRY(IC))=((TKMGE(DUMBRX(IC)+1,DUMBRY(IC))*
     +                DELTAY(DUMBRY(IC)))/(DELX(DUMBRX(IC)+1)))
     +                +((TKMGN(DUMBRX(IC)+1,DUMBRY(IC)+1)*DELTAX(DUMBR
     +                X(IC)+1))/(DELY(DUMBRY(IC))))
     +                +((TKMGN(DUMBRX(IC)+1,DUMBRY(IC))*DELTAX
     +                (DUMBRX(IC)+1))/(DELY(DUMBRY(IC)-1)))
     +                +((TKSGN(IC,NEM(IC),8)*DELTAY(DUMBRY(IC)
     +                ))/DELR(IC,NEM(IC),8))

C Above top right dummy node - above subgrid node 2

      B(DUMTRX(IC),DUMTRY(IC)-1)=0.0

      A(DUMTRX(IC),DUMTRY(IC)-1)=((TKMGE(DUMTRX(IC),DUMTRY(IC)-1)*
     +                DELTAY(DUMTRY(IC)-1))/(DELX(DUMTRX(IC))))
     +                +((TKMGE(DUMTRX(IC)-1,DUMTRY(IC)-1)*DELTAY(DUMTR
     +                Y(IC)-1))/(DELX(DUMTRX(IC)-1)))
     +                +((TKMGN(DUMTRX(IC),DUMTRY(IC)-1)*DELTAX(D
     +                UMTRX(IC)))/(DELY(DUMTRY(IC)-2)))
     +                +((TKSGN(IC,NEM(IC),2)*DELTAX(DUMTRX(IC)))/
     +                DELR(IC,NEM(IC),2))

C Above top left dummy node - above subgrid node 3

      B(DUMTLX(IC),DUMTLY(IC)-1)=0.0

      A(DUMTLX(IC),DUMTLY(IC)-1)=((TKMGE(DUMTLX(IC),DUMTLY(IC)-1)*
     +                DELTAY(DUMTLY(IC)-1))/(DELX(DUMTLX(IC))))
     +                +((TKMGE(DUMTLX(IC)-1,DUMTLY(IC)-1)*DELTAY(DUMTL
     +                Y(IC)-1))/(DELX(DUMTLX(IC)-1)))
     +                +((TKMGN(DUMTLX(IC),DUMTLY(IC)-1)*DELTAX(DUMTLX
     +                (IC)))/(DELY(DUMTLY(IC)-2)))
     +                +((TKSGN(IC,NEM(IC),3)*DELTAX(DUMTLX(IC)))/
     +                DELR(IC,NEM(IC),3))

C Beside top left dummy node - beside subgrid node 4

      A(DUMTLX(IC)-1,DUMTLY(IC))=((TKMGN(DUMTLX(IC)-1,DUMTLY(IC)+1)*
     +                DELTAX(DUMTLX(IC)-1))/(DELY(DUMTLY(IC))))
     +                +((TKMGN(DUMTLX(IC)-1,DUMTLY(IC))*DELTAX(DUMTLX
     +                (IC)-1))/(DELY(DUMTLY(IC)-1)))
     +                +((TKMGE(DUMTLX(IC)-2,DUMTLY(IC))*DELTAY
     +                (DUMTLY(IC)))/(DELX(DUMTLX(IC)-2)))
     +                +((TKSGN(IC,NEM(IC),4)*DELTAY(DUMTLY(IC)))/
     +                DELR(IC,NEM(IC),4))

C Beside bottom left dummy node - beside subgrid node 5

      A(DUMBLX(IC)-1,DUMBLY(IC))=((TKMGN(DUMBLX(IC)-1,DUMBLY(IC)+1)*
     +                DELTAX(DUMBLX(IC)-1))/(DELY(DUMBLY(IC))))
     +                +((TKMGN(DUMBLX(IC)-1,DUMBLY(IC))*DELTAX(DUMBLX
     +                (IC)-1))/(DELY(DUMBLY(IC)-1)))
     +                +((TKMGE(DUMBLX(IC)-2,DUMBLY(IC))*DELTAY(DUMBLY
     +                (IC)))/(DELX(DUMBLX(IC)-2)))
     +                +((TKSGN(IC,NEM(IC),5)*DELTAY(DUMBLY(IC)))/
```

```
     +                    DELR(IC,NEM(IC),5))

C Below bottom left dummy node - below subgrid node 6

          C(DUMBLX(IC),DUMBLY(IC)+1)=0.0

          A(DUMBLX(IC),DUMBLY(IC)+1)=((TKMGE(DUMBLX(IC),DUMBLY(IC)+1)*
     +                    DELTAY(DUMBLY(IC)+1))/(DELX(DUMBLX(IC))))
     +                    +((TKMGE(DUMBLX(IC)-1,DUMBLY(IC)+1)*DELTAY(DUMBL
     +                    Y(IC)+1))/(DELX(DUMBLX(IC)-1)))
     +                    +((TKMGN(DUMBLX(IC),DUMBLY(IC)+2)*DELTAX(DUMBLX
     +                    (IC)))/(DELY(DUMBLY(IC)+1)))
     +                    +((TKSGN(IC,NEM(IC),6)*DELTAX(DUMBLX(IC)))/
     +                    DELR(IC,NEM(IC),6))

C Below bottom right dummy node - below subgrid node 7

          C(DUMBRX(IC),DUMBRY(IC)+1)=0.0


          A(DUMBRX(IC),DUMBRY(IC)+1)=((TKMGE(DUMBRX(IC),DUMBRY(IC)+1)*
     +                    DELTAY(DUMBRY(IC)+1))/(DELX(DUMBRX(IC))))
     +                    +((TKMGE(DUMBRX(IC)-1,DUMBRY(IC)+1)*DELTAY(DUMBR
     +                    Y(IC)+1))/(DELX(DUMBRX(IC)-1)))
     +                    +((TKMGN(DUMBRX(IC),DUMBRY(IC)+2)*DELTAX(DUMBRX
     +                    (IC)))/(DELY(DUMBRY(IC)+1)))
     +                    +((TKSGN(IC,NEM(IC),7)*DELTAX(DUMBRX(IC)))/
     +                    DELR(IC,NEM(IC),7))


 8026 CONTINUE


C-------------------------
C AROUND REDUCED SUBGRIDS:
C-------------------------
      DO 8027 IC=NFSG+1,NPIPE

C Beside top right corner dummy node - beside final rectangular subgrid
C node on line 6

          A(DUMTRX(IC)+1,DUMTRY(IC))=((TKMGE(DUMTRX(IC)+1,DUMTRY(IC))*
     +                    DELTAY(DUMTRY(IC)))/(DELX(DUMTRX(IC)+1)))
     +                    +((TKMGN(DUMTRX(IC)+1,DUMTRY(IC)+1)*DELTAX(DUMTR
     +                    X(IC)+1))/(DELY(DUMTRY(IC))))
     +                    +((TKMGN(DUMTRX(IC)+1,DUMTRY(IC))*DELTAX
     +                    (DUMTRX(IC)+1))/(DELY(DUMTRY(IC)-1)))
     +                    +((TKSGN(IC,NEM(IC)-1,6)*DELTAY(DUMTRY(IC)
     +                    ))/((DELTAXHS(IC,NEM(IC)-1)/2.)+(DELTAX(DUMTRX
     +                    (IC)+1)/2.)))

C Beside top left corner dummy node - beside final rectangular subgrid
C node on line 5

          A(DUMTLX(IC)-1,DUMTLY(IC))=((TKMGN(DUMTLX(IC)-1,DUMTLY(IC)+1)*
     +                    DELTAX(DUMTLX(IC)-1))/(DELY(DUMTLY(IC))))
     +                    +((TKMGN(DUMTLX(IC)-1,DUMTLY(IC))*DELTAX(DUMTLX
     +                    (IC)-1))/(DELY(DUMTLY(IC)-1)))
     +                    +((TKMGE(DUMTLX(IC)-2,DUMTLY(IC))*DELTAY
     +                    (DUMTLY(IC)))/(DELX(DUMTLX(IC)-2)))
     +                    +((TKSGN(IC,NEM(IC)-1,5)*DELTAY(DUMTLY(IC)))/
     +                    ((DELTAXHS(IC,NEM(IC)-1)/2.)+(DELTAX(DUMTLX
     +                    (IC)-1)/2.)))

C Above top right corner dummy node - above rectangular subgrid
C nodes along line 6

          B(DUMTRX(IC),DUMTRY(IC)-1)=0.0
```

```
C The contribution from the SG nodes under the insulation board to
C the node in the insulations board (MG node) are a sum of the q's
C from the SG nodes

C First redefine DELTAXHS for node 2:

      DELTAXHS(IC,2)=(R(IC,2)+(DELTAR(IC,2,4)/2.))

         DO 2 I=2,NEM(IC)-1
            QN5(IC,I)=(TKSGE(IC,I,5)*DELTAXHS(IC,I))/DELYHSG(IC,I)
            QN6(IC,I)=(TKSGE(IC,I,6)*DELTAXHS(IC,I))/DELYHSG(IC,I)
            QN5T(IC)=QN5T(IC)+QN5(IC,I)
            QN6T(IC)=QN6T(IC)+QN6(IC,I)
    2    CONTINUE


         A(DUMTRX(IC),DUMTRY(IC)-1)=((TKMGE(DUMTRX(IC),DUMTRY(IC)-1)*
     +                   DELTAY(DUMTRY(IC)-1))/(DELX(DUMTRX(IC))))
     +                   +((TKMGE(DUMTRX(IC)-1,DUMTRY(IC)-1)*DELTAY(DUMTR
     +                   Y(IC)-1))/(DELX(DUMTRX(IC)-1)))
     +                   +((TKMGN(DUMTRX(IC),DUMTRY(IC)-1)*DELTAX(D
     +                   UMTRX(IC)))/(DELY(DUMTRY(IC)-2)))
     +                   +QN6T(IC)

C Above top left corner dummy node - above rectangular subgrid
C nodes along line 5

         B(DUMTLX(IC),DUMTLY(IC)-1)=0.0

         A(DUMTLX(IC),DUMTLY(IC)-1)=((TKMGE(DUMTLX(IC),DUMTLY(IC)-1)*
     +                   DELTAY(DUMTLY(IC)-1))/(DELX(DUMTLX(IC))))
     +                   +((TKMGE(DUMTLX(IC)-1,DUMTLY(IC)-1)*DELTAY(DUMTL
     +                   Y(IC)-1))/(DELX(DUMTLX(IC)-1)))
     +                   +((TKMGN(DUMTLX(IC),DUMTLY(IC)-1)*DELTAX(DUMTLX
     +                   (IC)))/(DELY(DUMTLY(IC)-2)))
     +                   +QN5T(IC)


C Beside bottom right corner node - beside final subgrid node of
C cylindrical section on radial line 1

         A(DUMBRX(IC)+1,DUMBRY(IC))=((TKMGE(DUMBRX(IC)+1,DUMBRY(IC))*
     +                   DELTAY(DUMBRY(IC)))/(DELX(DUMBRX(IC)+1)))
     +                   +((TKMGN(DUMBRX(IC)+1,DUMBRY(IC)+1)*DELTAX(DUMBR
     +                   X(IC)+1))/(DELY(DUMBRY(IC))))
     +                   +((TKMGN(DUMBRX(IC)+1,DUMBRY(IC))*DELTAX
     +                   (DUMBRX(IC)+1))/(DELY(DUMBRY(IC)-1)))
     +                   +((TKSGN(IC,NEM(IC),1)*DELTAY(DUMBRY(IC)
     +                   ))/DELR(IC,NEM(IC),1))

C Beside bottom left corner node - beside final subgrid node of
C cylindrical section on radial line 4

         A(DUMBLX(IC)-1,DUMBLY(IC))=((TKMGN(DUMBLX(IC)-1,DUMBLY(IC)+1)*
     +                   DELTAX(DUMBLX(IC)-1))/(DELY(DUMBLY(IC))))
     +                   +((TKMGN(DUMBLX(IC)-1,DUMBLY(IC))*DELTAX(DUMBLX
     +                   (IC)-1))/(DELY(DUMBLY(IC)-1)))
     +                   +((TKMGE(DUMBLX(IC)-2,DUMBLY(IC))*DELTAY(DUMBLY
     +                   (IC)))/(DELX(DUMBLX(IC)-2)))
     +                   +((TKSGN(IC,NEM(IC),4)*DELTAY(DUMBLY(IC)))/
     +                   DELR(IC,NEM(IC),4))

C Below bottom left corner node - below final subgrid node of
C cylindrical section on radial line 3

         C(DUMBLX(IC),DUMBLY(IC)+1)=0.0

         A(DUMBLX(IC),DUMBLY(IC)+1)=((TKMGE(DUMBLX(IC),DUMBLY(IC)+1)*
     +                   DELTAY(DUMBLY(IC)+1))/(DELX(DUMBLX(IC))))
```

```
     +                    +((TKMGE(DUMBLX(IC)-1,DUMBLY(IC)+1)*DELTAY(DUMBL
     +                    Y(IC)+1))/(DELX(DUMBLX(IC)-1)))
     +                    +((TKMGN(DUMBLX(IC),DUMBLY(IC)+2)*DELTAX(DUMBLX
     +                    (IC)))/(DELY(DUMBLY(IC)+1)))
     +                    +((TKSGN(IC,NEM(IC),3)*DELTAX(DUMBLX(IC)))/
     +                    DELR(IC,NEM(IC),3))

C Below bottom right corner node - below final subgrid node of
C cylindrical section on radial line 2

         C(DUMBRX(IC),DUMBRY(IC)+1)=0.0


         A(DUMBRX(IC),DUMBRY(IC)+1)=((TKMGE(DUMBRX(IC),DUMBRY(IC)+1)*
     +                    DELTAY(DUMBRY(IC)+1))/(DELX(DUMBRX(IC))))
     +                    +((TKMGE(DUMBRX(IC)-1,DUMBRY(IC)+1)*DELTAY(DUMBR
     +                    Y(IC)+1))/(DELX(DUMBRX(IC)-1)))
     +                    +((TKMGN(DUMBRX(IC),DUMBRY(IC)+2)*DELTAX(DUMBRX
     +                    (IC)))/(DELY(DUMBRY(IC)+1)))
     +                    +((TKSGN(IC,NEM(IC),2)*DELTAX(DUMBRX(IC)))/
     +                    DELR(IC,NEM(IC),2))


 8027 CONTINUE
C-------------------------------------------------------------------------


C EQUATIONS FOR LAST ROW (along x=N)

C Interior side nodes:
      DO 8035 J=2,M-1
         B(N,J)=((TKMGN(N,J+1)*DELTAX(N))/(DELY(J)))
         C(N,J)=((TKMGN(N,J)*DELTAX(N))/(DELY(J-1)))
         A(N,J)=B(N,J)+C(N,J)+((TKMGE(N-1,J)*DELTAY(J))/(DELX(N-1)))
 8035 CONTINUE

C Equation for bottom corner node - for last TDMA line along x=N
      B(N,M)=0.0
      C(N,M)=((TKMGN(N,M)*DELTAX(N))/(DELY(M-1)))
      A(N,M)=C(N,M)+((TKMGE(N-1,M)*DELTAY(M))/(DELX(N-1)))

C-------------------------------------------------------------------------


      END




C ***********************************************************************
C TITLE: ALGSPEC(CONVCRIT,MAXITER)
C
C DESCRIPTION:
C
C     This subroutine allows the user to define the algorith criteria:
C     convergenge criteria and maximum number of iterations
C
C ---------------------------------------------------------------------
      SUBROUTINE ALGSPEC(CONVCRIT,MAXITER)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      CALL CLEARSCREEN($GCLEARSCREEN)
      PRINT *,'Enter the convergence criteria for the solution algorithm
     +'
      READ *,CONVCRIT
      CALL CLEARSCREEN($GCLEARSCREEN)
```

```
      PRINT *,'Enter the maximum number of iterations of the algorithm:'
      READ *,MAXITER
      END




C ======================================================================
C TITLE: MGALG()
C
C DESCRIPTION:
C
C     This subroutine sweeps the main grid with TDMA.
C
C ----------------------------------------------------------------------
      SUBROUTINE MGALG()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody


      DELOLD=0.
      MCOUNTER=MCOUNTER+1
      IF(MCOUNTER.GE.MAXITER)THEN
```

```
         IFLAG=1
      ENDIF

C-------------------------------------------------------------

C BEGIN SWEEP THROUGH MAIN GRID
C-------------------------------------------------------------

C TDMA for 1st row (x=1)
      DO 8006 J=2,M-1
         D(1,J)=((TKMGE(1,J)*DELTAY(J))/(DELX(1)))*T(2,J)
         P(1,J)=(B(1,J)/(A(1,J)-(C(1,J)*P(1,J-1))))
         Q(1,J)=((D(1,J)+(C(1,J)*Q(1,J-1)))/(A(1,J)-(C(1,J)*P(1,J-1))))
 8006  CONTINUE

C Equation for bottom corner node - for first TDMA line along x=1
      D(1,M)=((TKMGE(1,M)*DELTAY(M))/(DELX(1)))*T(2,M)
      P(1,M)=0.0
      Q(1,M)=((D(1,M)+(C(1,M)*Q(1,M-1)))/(A(1,M)-(C(1,M)*
     +          P(1,M-1))))
      T(1,M)=Q(1,M)

C ALGORITHM
      DO 8011 J=M-1,2,-1
         TOLD(1,J)=T(1,J)
         T(1,J)=(P(1,J)*T(1,J+1))+Q(1,J))

         IF(DABS(T(1,J)-TOLD(1,J)).GT.DELOLD)THEN
            DELOLD=DABS(T(1,J)-TOLD(1,J))
         ENDIF

 8011 CONTINUE

C-------------------------------------------------------------------------
C TDMA FOR INTERIOR NODES - ie. LINES I=2 THROUGH I=N-1
C-------------------------------------------------------------------------
C INTERIOR NODES (J=2,M-1)
      DO 8021 I=2,N-1
         DO 8024 J=2,M-1

C Calculate D's for column:
            D(I,J)=((((TKMGE(I,J)*DELTAY(J))/(DELX(I)))*T(I+1,J))+
     +              (((TKMGE(I-1,J)*DELTAY(J))/(DELX(I-1)))*T(I-1,J)))
C            PRINT *,'D(',I,J,')=',D(I,J)
C Change D's if they are surrounding a subgrid:
C-----------------------------------------
            DO 8025 IC=1,NPIPE
      IF(I.EQ.(DUMTRX(IC)+1).AND.J.EQ.DUMTRY(IC).AND.IC.LE.NFSG)THEN
C FSG:Beside top right corner dummy node - beside subgrid node 1
         D(DUMTRX(IC)+1,DUMTRY(IC))=(((TKMGE(DUMTRX(IC)+1,DUMTRY(IC))*
     +                   DELTAY(DUMTRY(IC)))/(DELX(DUMTRX(IC)+1)))*
     +                   T(DUMTRX(IC)+2,DUMTRY(IC)))
     +                   +(((TKSGN(IC,NEM(IC),1)*DELTAY(DUMTRY(IC)
     +                   ))/DELR(IC,NEM(IC),1))*(TS(IC,NEM(IC),1)))
      ENDIF
      IF(I.EQ.(DUMTRX(IC)+1).AND.J.EQ.DUMTRY(IC).AND.IC.GT.NFSG)THEN

C HSG:Beside top right corner dummy node - beside final rectangular
C subgrid node on line 6

         D(DUMTRX(IC)+1,DUMTRY(IC))=(((TKMGE(DUMTRX(IC)+1,DUMTRY(IC))*
     +                   DELTAY(DUMTRY(IC)))/(DELX(DUMTRX(IC)+1)))*
     +                   T(DUMTRX(IC)+2,DUMTRY(IC)))
     +                   +(((TKSGN(IC,NEM(IC)-1,6)*DELTAY(DUMTRY(IC)
     +                   ))/((DELTAXHS(IC,NEM(IC)-1)/2.)+(DELTAX(DUMTRX
     +                   (IC)+1)/2.)))*(TS(IC,NEM(IC)-1,6)))
      ENDIF
```

```
      IF(I.EQ.(DUMBRX(IC)+1).AND.J.EQ.DUMBRY(IC).AND.IC.LE.NFSG)THEN
C FSG:Beside bottom right corner node - beside subgrid node 8
         D(DUMBRX(IC)+1,DUMBRY(IC))=(((TKMGE(DUMBRX(IC)+1,DUMBRY(IC))*
     +                    DELTAY(DUMBRY(IC)))/(DELX(DUMBRX(IC)+1)))*
     +                    T(DUMBRX(IC)+2,DUMBRY(IC)))
     +                    +(((TKSGN(IC,NEM(IC),8)*DELTAY(DUMBRY(IC)
     +                    ))/DELR(IC,NEM(IC),8))*(TS(IC,NEM(IC),8)))
       ENDIF

      IF(I.EQ.(DUMBRX(IC)+1).AND.J.EQ.DUMBRY(IC).AND.IC.GT.NFSG)THEN

C HSG:Beside bottom right corner node - beside final subgrid node
C of cylindrical section on radial line 1

         D(DUMBRX(IC)+1,DUMBRY(IC))=(((TKMGE(DUMBRX(IC)+1,DUMBRY(IC))*
     +                    DELTAY(DUMBRY(IC)))/(DELX(DUMBRX(IC)+1)))*
     +                    T(DUMBRX(IC)+2,DUMBRY(IC)))
     +                    +(((TKSGN(IC,NEM(IC),1)*DELTAY(DUMBRY(IC)
     +                    ))/DELR(IC,NEM(IC),1))*(TS(IC,NEM(IC),1)))
       ENDIF



      IF(I.EQ.DUMTRX(IC).AND.J.EQ.(DUMTRY(IC)-1).AND.IC.LE.NFSG)THEN

C FSG:Above top right dummy node - above subgrid node 2

         D(DUMTRX(IC),DUMTRY(IC)-1)=(((TKMGE(DUMTRX(IC),DUMTRY(IC)-1)*
     +                    DELTAY(DUMTRY(IC)-1))/(DELX(DUMTRX(IC))))*T
     +                    (DUMTRX(IC)+1,DUMTRY(IC)-1))
     +                    +(((TKMGE(DUMTRX(IC)-1,DUMTRY(IC)-1)*DELTAY(DUMT
     +                    RY(IC)-1))/(DELX(DUMTRX(IC)-1)))*T(DUMTRX(IC)-1,
     +                    DUMTRY(IC)-1))
     +                    +(((TKSGN(IC,NEM(IC),2)*DELTAX(DUMTRX(IC)))/
     +                    DELR(IC,NEM(IC),2))*(TS(IC,NEM(IC),2)))
       ENDIF


      IF(I.EQ.DUMTRX(IC).AND.J.EQ.(DUMTRY(IC)-1).AND.IC.GT.NFSG)THEN

C        HSG:Above top right corner dummy node - above rectangular subgrid
C        nodes along line 6
C        First redefine DELTAXHS for node 2:


         DELTAXHS(IC,2)=(R(IC,2)+(DELTAR(IC,2,4)/2.))
C        The contribution from the SG nodes under the insulation board to
C        the node in the insulations board (MG node) are a sum of the q's
C        from the SG nodes - for the D term, TS is included
         QN6WTT(IC)=0.
            DO 3 K=2,NEM(IC)-1
              QN6WT(IC,K)=(((TKSGE(IC,K,6)*DELTAXHS(IC,K))/DELYHSG(IC,K)
     +                    )*TS(IC,K,6))
              QN6WTT(IC)=QN6WTT(IC)+QN6WT(IC,K)
    3       CONTINUE
         D(DUMTRX(IC),DUMTRY(IC)-1)=(((TKMGE(DUMTRX(IC),DUMTRY(IC)-1)*
     +                    DELTAY(DUMTRY(IC)-1))/(DELX(DUMTRX(IC))))*T
     +                    (DUMTRX(IC)+1,DUMTRY(IC)-1))
     +                    +(((TKMGE(DUMTRX(IC)-1,DUMTRY(IC)-1)*DELTAY(DUMT
     +                    RY(IC)-1))/(DELX(DUMTRX(IC)-1)))*T(DUMTRX(IC)-1,
     +                    DUMTRY(IC)-1))
     +                    +QN6WTT(IC)
       ENDIF
       IF(I.EQ.DUMTLX(IC).AND.J.EQ.(DUMTLY(IC)-1).AND.IC.LE.NFSG)THEN
C         FSG:Above top left dummy node - above subgrid node 3
          D(DUMTLX(IC),DUMTLY(IC)-1)=(((TKMGE(DUMTLX(IC),DUMTLY(IC)-1)*
     +                    DELTAY(DUMTLY(IC)-1))/(DELX(DUMTLX(IC))))*T
     +                    (DUMTLX(IC)+1,DUMTLY(IC)-1))
     +                    +(((TKMGE(DUMTLX(IC)-1,DUMTLY(IC)-1)*DELTAY(DUMT
```

```
     +                    LY(IC)-1))/(DELX(DUMTLX(IC)-1)))*T(DUMTLX(IC)-1,
     +                    DUMTLY(IC)-1))
     +                    +(((TKSGN(IC,NEM(IC),3)*DELTAX(DUMTLX(IC)))/
     +                    DELR(IC,NEM(IC),3))*(TS(IC,NEM(IC),3)))
      ENDIF


      IF(I.EQ.DUMTLX(IC).AND.J.EQ.(DUMTLY(IC)-1).AND.IC.GT.NFSG)THEN

C        HSG:Above top left corner dummy node - above rectangular subgrid
C        nodes along line 5

C        First redefine DELTAXHS for node 2:

         DELTAXHS(IC,2)=(R(IC,2)+(DELTAR(IC,2,4)/2.))
C        The contribution from the SG nodes under the insulation board to
C        the node in the insulations board (MG node) are a sum of the q's
C        from the SG nodes - for the D term, TS is included
         QN5WTT(IC)=0.
            DO 4 K=2,NEM(IC)-1
              QN5WT(IC,K)=(((TKSGE(IC,K,5)*DELTAXHS(IC,K))/DELYHSG(IC,K)
     +                    )*TS(IC,K,5))
              QN5WTT(IC)=QN5WTT(IC)+QN5WT(IC,K)
    4       CONTINUE
         D(DUMTLX(IC),DUMTLY(IC)-1)=(((TKMGE(DUMTLX(IC),DUMTLY(IC)-1)*
     +                   DELTAY(DUMTLY(IC)-1))/(DELX(DUMTLX(IC))))*T
     +                   (DUMTLX(IC)+1,DUMTLY(IC)-1))
     +                   +(((TKMGE(DUMTLX(IC)-1,DUMTLY(IC)-1)*DELTAY(DUMT
     +                   LY(IC)-1))/(DELX(DUMTLX(IC)-1)))*T(DUMTLX(IC)-1,
     +                   DUMTLY(IC)-1))
     +                   +QN5WTT(IC)
      ENDIF


      IF(I.EQ.(DUMTLX(IC)-1).AND.J.EQ.DUMTLY(IC).AND.IC.LE.NFSG)THEN

C        FSG:Beside top left dummy node - beside subgrid node 4

         D(DUMTLX(IC)-1,DUMTLY(IC))=(((TKMGE(DUMTLX(IC)-2,DUMTLY(IC))*
     +                   DELTAY(DUMTLY(IC)))/(DELX(DUMTLX(IC)-2)))*
     +                   T(DUMTLX(IC)-2,DUMTLY(IC)))
     +                   +(((TKSGN(IC,NEM(IC),4)*DELTAY(DUMTLY(IC)))/
     +                   DELR(IC,NEM(IC),4))*(TS(IC,NEM(IC),4)))
      ENDIF


      IF(I.EQ.(DUMTLX(IC)-1).AND.J.EQ.DUMTLY(IC).AND.IC.GT.NFSG)THEN

C        HSG:Beside top left dummy node - beside final rectangular subgrid
C        node on line 5

         D(DUMTLX(IC)-1,DUMTLY(IC))=(((TKMGE(DUMTLX(IC)-2,DUMTLY(IC))*
     +                   DELTAY(DUMTLY(IC)))/(DELX(DUMTLX(IC)-2)))*
     +                   T(DUMTLX(IC)-2,DUMTLY(IC)))
     +                   +(((TKSGN(IC,NEM(IC)-1,5)*DELTAY(DUMTLY(IC)))/
     +                   ((DELTAXHS(IC,NEM(IC)-1)/2.)+(DELTAX(DUMTLX
     +                   (IC)-1)/2.)))*(TS(IC,NEM(IC)-1,5)))
      ENDIF


      IF(I.EQ.(DUMBLX(IC)-1).AND.J.EQ.DUMBLY(IC).AND.IC.LE.NFSG)THEN

C        FSG:Beside bottom left dummy node - beside subgrid node 5

         D(DUMBLX(IC)-1,DUMBLY(IC))=(((TKMGE(DUMBLX(IC)-2,DUMBLY(IC))*
     +                   DELTAY(DUMBLY(IC)))/(DELX(DUMBLX(IC)-2)))*T
     +                   (DUMBLX(IC)-2.DUMBLY(IC)))
     +                   +(((TKSGN(IC,NEM(IC),5)*DELTAY(DUMBLY(IC)))/
```

```
     +                    DELR(IC,NEM(IC),5))*(TS(IC,NEM(IC),5)))
      ENDIF


      IF(I.EQ.(DUMBLX(IC)-1).AND.J.EQ.DUMBLY(IC).AND.IC.GT.NFSG)THEN

C        HSG:Beside bottom left corner node - beside final subgrid node of
C        cylindrical section on radial line 4

         D(DUMBLX(IC)-1,DUMBLY(IC))=(((TKMGE(DUMBLX(IC)-2,DUMBLY(IC))*
     +                   DELTAY(DUMBLY(IC)))/(DELX(DUMBLX(IC)-2)))*T
     +                   (DUMBLX(IC)-2,DUMBLY(IC)))
     +                   +(((TKSGN(IC,NEM(IC),4)*DELTAY(DUMBLY(IC)))/
     +                   DELR(IC,NEM(IC),4))*(TS(IC,NEM(IC),4)))
      ENDIF


      IF(I.EQ.DUMBLX(IC).AND.J.EQ.(DUMBLY(IC)+1).AND.IC.LE.NFSG)THEN

C        FSG:Below bottom left dummy node - below subgrid node 6

         D(DUMBLX(IC),DUMBLY(IC)+1)=(((TKMGE(DUMBLX(IC),DUMBLY(IC)+1)*
     +                   DELTAY(DUMBLY(IC)+1))/(DELX(DUMBLX(IC))))*T
     +                   (DUMBLX(IC)+1,DUMBLY(IC)+1))
     +                   +(((TKMGE(DUMBLX(IC)-1,DUMBLY(IC)+1)*DELTAY(DUMB
     +                   LY(IC)+1))/(DELX(DUMBLX(IC)-1)))*T(DUMBLX(IC)-1,
     +                   DUMBLY(IC)+1))
     +                   +(((TKSGN(IC,NEM(IC),6)*DELTAX(DUMBLX(IC)))/
     +                   DELR(IC,NEM(IC),6))*(TS(IC,NEM(IC),6)))
      ENDIF


      IF(I.EQ.DUMBLX(IC).AND.J.EQ.(DUMBLY(IC)+1).AND.IC.GT.NFSG)THEN

C        HSG:Below bottom left corner node - below final subgrid node of
C        cylindrical section on radial line 3

         D(DUMBLX(IC),DUMBLY(IC)+1)=(((TKMGE(DUMBLX(IC),DUMBLY(IC)+1)*
     +                   DELTAY(DUMBLY(IC)+1))/(DELX(DUMBLX(IC))))*T
     +                   (DUMBLX(IC)+1,DUMBLY(IC)+1))
     +                   +(((TKMGE(DUMBLX(IC)-1,DUMBLY(IC)+1)*DELTAY(DUMB
     +                   LY(IC)+1))/(DELX(DUMBLX(IC)-1)))*T(DUMBLX(IC)-1,
     +                   DUMBLY(IC)+1))
     +                   +(((TKSGN(IC,NEM(IC),3)*DELTAX(DUMBLX(IC)))/
     +                   DELR(IC,NEM(IC),3))*(TS(IC,NEM(IC),3)))
      ENDIF


      IF(I.EQ.DUMBRX(IC).AND.J.EQ.(DUMBRY(IC)+1).AND.IC.LE.NFSG)THEN

C        FSG:Below bottom right dummy node - below subgrid node 7

         D(DUMBRX(IC),DUMBRY(IC)+1)=(((TKMGE(DUMBRX(IC),DUMBRY(IC)+1)*
     +                   DELTAY(DUMBRY(IC)+1))/(DELX(DUMBRX(IC))))*T
     +                   (DUMBRX(IC)+1,DUMBRY(IC)+1))
     +                   +(((TKMGE(DUMBRX(IC)-1,DUMBRY(IC)+1)*DELTAY(DUMB
     +                   RY(IC)+1))/(DELX(DUMBRX(IC)-1)))*T(DUMBRX(IC)-1,
     +                   DUMBRY(IC)+1))
     +                   +(((TKSGN(IC,NEM(IC),7)*DELTAX(DUMBRX(IC)))/
     +                   DELR(IC,NEM(IC),7))*(TS(IC,NEM(IC),7)))
      ENDIF


      IF(I.EQ.DUMBRX(IC).AND.J.EQ.(DUMBRY(IC)+1).AND.IC.GT.NFSG)THEN

C        HSG:Below bottom right corner node - below final subgrid node of
C        cylindrical section on radial line 2

         D(DUMBRX(IC),DUMBRY(IC)+1)=(((TKMGE(DUMBRX(IC),DUMBRY(IC)+1)*
```

```
     +                    DELTAY(DUMBRY(IC)+1))/(DELX(DUMBRX(IC))))*T
     +                    (DUMBRX(IC)+1,DUMBRY(IC)+1))
     +                    +(((TKMGE(DUMBRX(IC)-1,DUMBRY(IC)+1)*DELTAY(DUMB
     +                    RY(IC)+1))/(DELX(DUMBRX(IC)-1)))*T(DUMBRX(IC)-1,
     +                    DUMBRY(IC)+1))
     +                    +(((TKSGN(IC,NEM(IC),2)*DELTAX(DUMBRX(IC)))/
     +                    DELR(IC,NEM(IC),2))*(TS(IC,NEM(IC),2)))
      ENDIF

 8025          CONTINUE


C Calculate P,Q for interior nodes of column I:
         P(I,J)=(B(I,J)/(A(I,J)-(C(I,J)*P(I,J-1))))
         Q(I,J)=((D(I,J)+(C(I,J)*Q(I,J-1)))/(A(I,J)-(C(I,J)*P(I,J-1))))
 8024 CONTINUE
C Calculate D,P,Q for bottom node of column I:
         D(I,M)=(((TKMGE(I,M)*DELTAY(M))/(DELX(I)))*T(I+1,M))+
     +        (((TKMGE(I-1,M)*DELTAY(M))/(DELX(I-1)))*T(I-1,M))
         P(I,M)=0.0
         Q(I,M)=((D(I,M)+(C(I,M)*Q(I,M-1)))/(A(I,M)-(C(I,M)*
     +           P(I,M-1))))
         T(I,M)=Q(I,M)


C Carry out TDMA for column I:
         DO 8031 J=M-1,2,-1
            TOLD(I,J)=T(I,J)
            T(I,J)=((P(I,J)*T(I,J+1))+Q(I,J))
            IF(DABS(T(I,J)-TOLD(I,J)).GT.DELOLD)THEN
               DELOLD=DABS(T(I,J)-TOLD(I,J))
            ENDIF
 8031    CONTINUE

 8021    CONTINUE

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

C TDMA FOR LAST ROW (along x=N)

C Interior side nodes
      DO 8036 J=2,M-1
         D(N,J)=((TKMGE(N-1,J)*DELTAY(J))/(DELX(N-1)))*T(N-1,J)
         P(N,J)=(B(N,J)/(A(N,J)-(C(N,J)*P(N,J-1))))
         Q(N,J)=((D(N,J)+(C(N,J)*Q(N,J-1)))/(A(N,J)-(C(N,J)
     +             *P(N,J-1))))
 8036 CONTINUE

C Equation for bottom corner node - for last TDMA line along x=N
      D(N,M)=((TKMGE(N-1,M)*DELTAY(M))/(DELX(N-1)))*T(N-1,M)
      P(N,M)=0.0
      Q(N,M)=((D(N,M)+(C(N,M)*Q(N,M-1)))/(A(N,M)-
     +          (C(N,M)*P(N,M-1))))
      T(N,M)=Q(N,M)

C ALGORITHM
      DO 8040 J=M-1,2,-1
         TOLD(N,J)=T(N,J)
         T(N,J)=((P(N,J)*T(N,J+1))+Q(N,J))
         IF(DABS(T(N,J)-TOLD(N,J)).GT.DELOLD)THEN
            DELOLD=DABS(T(N,J)-TOLD(N,J))
         ENDIF
 8040 CONTINUE



C-------------------------------------------------------------------------
C END OF SWEEP THROUGH MAIN GRID
```

```
C-----------------------------------------------------------------------
      END




C ======================================================== ==============
C TITLE: FSGALG()
C
C DESCRIPTION:
C
C     This subroutine changes coefficients for TDMA for nodes around
c     subgrids.
C
C -------------------------------------------------------------------
      SUBROUTINE FSGALG()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody

C-----------------------------------------------------------------------
C BEGIN SWEEP THROUGH ALL FULL SUBGRIDS
C-----------------------------------------------------------------------
```

```
      DO 7000 ICOUNTER=1,NFSG
         DO 7002 J=1,8
C Interior nodes along radial line J:
C-------------------------------------
            DO 7001 I=2,NEM(ICOUNTER)-1

C DS is calculated separately due to the "J+1" term involved-at the final
C node (node 8) J+1 is actually 1 not 9

       IF(J.EQ.1)THEN
                  DS(ICOUNTER,I,1)=(((TKSG(ICOUNTER,I,1)*DELTAR(ICOUNTER
     +                  ,I,1))/(R(ICOUNTER,I)*DELTHETA))
     +                  *TS(ICOUNTER,I,2))+(((TKSG(ICOUNTER,I,1)
     +                  *DELTAR(ICOUNTER,I,1))/(R(ICOUNTER,I)*DELT
     +                  HETA))*TS(ICOUNTER,I,8))
       ELSEIF(J.EQ.2.OR.J.EQ.3.OR.J.EQ.4.OR.J.EQ.5.OR.J.EQ.6.OR.J.EQ.7)TH
     +EN
                  DS(ICOUNTER,I,J)=(((TKSG(ICOUNTER,I,J)*DELTAR(ICOUNTER
     +                   ,I,J))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUNTER
     +                   ,I,J+1))+(((TKSG(ICOUNTER,I,J)*DELTAR(ICOUN
     +                   TER,I,J))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUN
     +                   TER,I,J-1))
       ELSEIF(J.EQ.8)THEN
            DS(ICOUNTER,I,8)=(((TKSG(ICOUNTER,I,8)*DELTAR(ICOUNTER
     +                  ,I,8))/(R(ICOUNTER,I)*DELTHETA))
     +                  *TS(ICOUNTER,I,1))+(((TKSG(ICOUNTER,I,8)
     +                  *DELTAR(ICOUNTER,I,8))/(R(ICOUNTER,I)*DELT
     +                  HETA))*TS(ICOUNTER,I,7))
       ENDIF

C PS,QS, are calculated for radial line J:
            PS(ICOUNTER,I,J)=(BS(ICOUNTER,I,J)/(AS(ICOUNTER,I,J)-(CS(
     +                  ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
            QS(ICOUNTER,I,J)=((DS(ICOUNTER,I,J)+(CS(ICOUNTER,I,J)*QS(
     +                  ICOUNTER,I-1,J)))/(AS(ICOUNTER,I,J)-(CS(
     +                  ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
 7001       CONTINUE


C DS,PS,QS for final node along radial line J:
C---------------------------------------------

C DS is calculated for each node as it includes the temp. of adjacent
C main grid nodes which are defined specifically  (DS = aW (OR aE) + q)

       IF(J.EQ.1)THEN
          DS(ICOUNTER,NEM(ICOUNTER),1)=(((TKSG(ICOUNTER,NEM(ICOUNTER),1)*
     +                  DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                  ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                  ,2))+
     +                  (((TKSGN(ICOUNTER,NEM(ICOUNTER),1)*BEDSI
     +                  DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                  ,1))*(T(DUMTRX(ICOUNTER)+1,DUMTRY(ICOUNT
     +                  ER))))
       ELSEIF(J.EQ.2)THEN
          DS(ICOUNTER,NEM(ICOUNTER),2)=(((TKSG(ICOUNTER,NEM(ICOUNTER),2)*
     +                  DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                  ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                  ,1))+
     +                  (((TKSGN(ICOUNTER,NEM(ICOUNTER),2)*BEDSI
     +                  DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                  ,2))*(T(DUMTRX(ICOUNTER),DUMTRY(ICOUNTER)
     +                  -1)))
       ELSEIF(J.EQ.3)THEN
          DS(ICOUNTER,NEM(ICOUNTER),3)=(((TKSG(ICOUNTER,NEM(ICOUNTER),3)*
     +                  DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                  ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                  ,4))+
     +                  (((TKSGN(ICOUNTER,NEM(ICOUNTER),3)*BEDSI
```

```
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,3))*(T(DUMTLX(ICOUNTER),DUMTLY(ICOUNTER)
     +                          -1)))
      ELSEIF(J.EQ.4)THEN
         DS(ICOUNTER,NEM(ICOUNTER),4)=(((TKSG(ICOUNTER,NEM(ICOUNTER),4)*
     +                          DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                          ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                          ,3))+
     +                          (((TKSGN(ICOUNTER,NEM(ICOUNTER),4)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,4))*(T(DUMTLX(ICOUNTER)-1,DUMTLY(ICOUNTE
     +                          R))))
      ELSEIF(J.EQ.5)THEN
         DS(ICOUNTER,NEM(ICOUNTER),5)=(((TKSG(ICOUNTER,NEM(ICOUNTER),5)*
     +                          DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                          ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                          ,6))+
     +                          (((TKSGN(ICOUNTER,NEM(ICOUNTER),5)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,5))*(T(DUMBLX(ICOUNTER)-1,DUMBLY(ICOUNTE
     +                          R))))
      ELSEIF(J.EQ.6)THEN
         DS(ICOUNTER,NEM(ICOUNTER),6)=(((TKSG(ICOUNTER,NEM(ICOUNTER),6)*
     +                          DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                          ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                          ,5))+
     +                          (((TKSGN(ICOUNTER,NEM(ICOUNTER),6)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,6))*(T(DUMBLX(ICOUNTER),DUMBLY(ICOUNTER)
     +                          +1)))
      ELSEIF(J.EQ.7)THEN
         DS(ICOUNTER,NEM(ICOUNTER),7)=(((TKSG(ICOUNTER,NEM(ICOUNTER),7)*
     +                          DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                          ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                          ,8))+
     +                          (((TKSGN(ICOUNTER,NEM(ICOUNTER),7)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,7))*(T(DUMBRX(ICOUNTER),DUMBRY(ICOUNTER)
     +                          +1)))
      ELSEIF(J.EQ.8)THEN
         DS(ICOUNTER,NEM(ICOUNTER),8)=(((TKSG(ICOUNTER,NEM(ICOUNTER),8)*
     +                          DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                          ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                          ,7))+
     +                          (((TKSGN(ICOUNTER,NEM(ICOUNTER),8)*BEDSI
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,8))*(T(DUMBRX(ICOUNTER)+1,DUMBRY(ICOUNTE
     +                          R))))
      ENDIF

C Calculate PS,QS for final node along radial line I:
         PS(ICOUNTER,NEM(ICOUNTER),J)=(BS(ICOUNTER,NEM(ICOUNTER),J)/
     +                           (AS(ICOUNTER,NEM(ICOUNTER),J)-(CS(ICO
     +                           UNTER,NEM(ICOUNTER),J)*PS(ICOUNTER,NEM
     +                           (ICOUNTER)-1,J))))
         QS(ICOUNTER,NEM(ICOUNTER),J)=((DS(ICOUNTER,NEM(ICOUNTER),J)+
     +                           (CS(ICOUNTER,NEM(ICOUNTER),J)*QS(ICOUNTER
     +                           ,NEM(ICOUNTER)-1,J)))/(AS(ICOUNTER,NEM
     +                           (ICOUNTER),J)-(CS(ICOUNTER,NEM(ICOUNTER),
     +                           J)*PS(ICOUNTER,NEM(ICOUNTER)-1,J))))

         TS(ICOUNTER,NEM(ICOUNTER),J)=QS(ICOUNTER,NEM(ICOUNTER),J)


C ALGORITHM:for radial line I:
C-------------------------------
      DO 7031 I=NEM(ICOUNTER)-1,2,-1
            TSOLD(ICOUNTER,I,J)=TS(ICOUNTER,I,J)
```

```
            TS(ICOUNTER,I,J)=((PS(ICOUNTER,I,J)*TS(ICOUNTER,I+1,J))+
     +                        QS(ICOUNTER,I,J))
            IF(DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J)).GT.DELOLD)THEN
               DELOLD=DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J))
            ENDIF
 7031 CONTINUE
C-------------------------------------------------------------------------
 7002    CONTINUE

 7000 CONTINUE

C-------------------------------------------------------------------------
C END OF SWEEP THROUGH EACH FULL SUBGRID
C-------------------------------------------------------------------------

      END




C ======================================================================
C TITLE: HSGALG()
C
C DESCRIPTION:
C
C     This subroutine changes coefficients for TDMA for nodes around
c     subgrids.
C
C ----------------------------------------------------------------------
      SUBROUTINE HSGALG()
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
```

```
     +          REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +          ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +          TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +          regnum,regnumx,regnumy,regnodx,regnody


C-------------------------------------------------------------------------
C BEGIN SWEEP THROUGH ALL HALF CYLINDRICAL SUBGRIDS
C-------------------------------------------------------------------------
      DO 7004 ICOUNTER=NFSG+1,NPIPE
         DO 7005 J=1,8

C CALCULATE DS
C-------------

         DO 7006 I=2,NEM(ICOUNTER)-1

C For radial line #1
      IF(J.EQ.1)THEN
            DS(ICOUNTER,I,1)=(((TKSG(ICOUNTER,I,1)*DELTAR(ICOUNTER
     +                        ,I,1))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUNTE
     +                        R,I,2))
     +                        +(((TKSG(ICOUNTER,I,1)*DELTAR(ICOU
     +                        NTER,I,1))/((R(ICOUNTER,I)*PI/8.)+(hh
     +                        (ICOUNTER)/2.)))*TS(ICOUNTER,I,6))
C INTERIOR nodes - for radial lines (cylindrical section of the subgrid):

      ELSEIF(J.EQ.2.OR.J.EQ.3)THEN
               DS(ICOUNTER,I,J)=(((TKSG(ICOUNTER,I,J)*DELTAR(ICOUNTER
     +                        ,I,J))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUNTER
     +                        ,I,J+1))+(((TKSG(ICOUNTER,I,J)*DELTAR(ICOUN
     +                        TER,I,J))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUN
     +                        TER,I,J-1))
C For radial line #4
      ELSEIF(J.EQ.4)THEN
            DS(ICOUNTER,I,4)=(((TKSG(ICOUNTER,I,4)*DELTAR(ICOUNTER
     +                        ,I,4))/(R(ICOUNTER,I)*DELTHETA))*TS(ICOUNTE
     +                        R,I,3))
     +                        +(((TKSG(ICOUNTER,I,4)*DELTAR(ICOU
     +                        NTER,I,4))/((R(ICOUNTER,I)*PI/8.)+(hh
     +                        (ICOUNTER)/2.)))*TS(ICOUNTER,I,5))
      ENDIF
 7006       CONTINUE


C DS FOR RECTANGULAR HSG NODES
C------------------------------

C DS FOR NODE 2 OF LINES 5 AND 6
      IF(J.EQ.5)THEN
         DS(ICOUNTER,2,5)=(((TKSG(ICOUNTER,2,5)*(hh(ICOUNTER)-ORADIUS
     +                     (ICOUNTER)))/(2*R(ICOUNTER,2)))*TS(ICOUNTER,
     +                     2,6))
     +                     +(((TKSG(ICOUNTER,2,5)*DELTAR(ICOUNTER,2,4))
     +                     /((R(ICOUNTER,2)*(PI/8.))+hh(ICOUNTER)/2.))*
     +                     TS(ICOUNTER,2,4))
     +                     +(((TKSGE(ICOUNTER,2,5)*(R(ICOUNTER,2)+(DELTAR
     +                     (ICOUNTER,2,4)/2.)))/(DELYHSG(ICOUNTER,2)))*
     +                     (T(DUMTLX(ICOUNTER),DUMTLY(ICOUNTER)-1)))
      ELSEIF(J.EQ.6)THEN
         DS(ICOUNTER,2,6)=(((TKSG(ICOUNTER,2,6)*(hh(ICOUNTER)-ORADIUS
     +                     (ICOUNTER)))/(2*R(ICOUNTER,2)))*TS(ICOUNTER,
     +                     2,5))
     +                     +(((TKSG(ICOUNTER,2,6)*DELTAR(ICOUNTER,2,1))
     +                     /((R(ICOUNTER,2)*(PI/8.))+hh(ICOUNTER)/2.))*
```

```
     +                    TS(ICOUNTER,2,1))
     +                    +(((TKSGE(ICOUNTER,2,6)*(R(ICOUNTER,2)+(DELTAR
     +                    (ICOUNTER,2,1)/2.)))/(DELYHSG(ICOUNTER,2)))*
     +                    (T(DUMTRX(ICOUNTER),DUMTRY(ICOUNTER)-1)))
      ENDIF

C DS FOR INTERIOR RECTANGULAR NODES:
      IF(J.EQ.5)THEN
         DO 70014 I=3,NEM(ICOUNTER)-2
            DS(ICOUNTER,I,5)=(((TKSG(ICOUNTER,I,5)*DELTAXHS(ICOUNTER,I))
     +                       /((R(ICOUNTER,I)*(PI/8.))+(hh(ICO
     +                       UNTER)/2.)))*TS(ICOUNTER,I,4))
     +                       +(((TKSGE(ICOUNTER,I,5)*DELTAXHS(ICOUNTER
     +                       ,I))/(DELYHSG(ICOUNTER,I)))*(T(DUMTLX(ICOU
     +                       NTER),DUMTLY(ICOUNTER)-1)))
70014    CONTINUE

      ELSEIF(J.EQ.6)THEN
         DO 70015 I=3,NEM(ICOUNTER)-2
            DS(ICOUNTER,I,6)=(((TKSG(ICOUNTER,I,6)*DELTAXHS(ICOUNTER,I))
     +                       /((R(ICOUNTER,I)*(PI/8.))+(hh(ICO
     +                       UNTER)/2.)))*TS(ICOUNTER,I,1))
     +                       +(((TKSGE(ICOUNTER,I,6)*DELTAXHS(ICOUNTER
     +                       ,I))/(DELYHSG(ICOUNTER,I)))*(T(DUMTRX(ICOU
     +                       NTER),DUMTRY(ICOUNTER)-1)))
70015    CONTINUE
      ENDIF


C DS FOR FINAL NODES
C--------------------

C Cylindrical section: radial lines 1-4:
C DS is calculated for each node as it includes the temp. cf adjacent
C main grid nodes which are defined specifically  (DS = aW (OR aE) + q)

      IF(J.EQ.1)THEN
         DS(ICOUNTER,NEM(ICOUNTER),1)=(((TKSG(ICOUNTER,NEM(ICOUNTER),1)*
     +                           DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                           ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                           ,2))+
     +                           (((TKSGN(ICOUNTER,NEM(ICOUNTER),1)*BEDSI
     +                           DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                           ,1))*(T(DUMBRX(ICOUNTER)+1,DUMBRY(ICOUNT
     +                           ER))))
      ELSEIF(J.EQ.2)THEN
         DS(ICOUNTER,NEM(ICOUNTER),2)=(((TKSG(ICOUNTER,NEM(ICOUNTER),2)*
     +                           DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                           ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                           ,1))+
     +                           (((TKSGN(ICOUNTER,NEM(ICOUNTER),2)*BEDSI
     +                           DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                           ,2))*(T(DUMBRX(ICOUNTER),DUMBRY(ICOUNTER)
     +                           +1)))
      ELSEIF(J.EQ.3)THEN
         DS(ICOUNTER,NEM(ICOUNTER),3)=(((TKSG(ICOUNTER,NEM(ICOUNTER),3)*
     +                           DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                           ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                           ,4))+
     +                           (((TKSGN(ICOUNTER,NEM(ICOUNTER),3)*BEDSI
     +                           DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                           ,3))*(T(DUMBLX(ICOUNTER),DUMBLY(ICOUNTER)
     +                           +1)))
      ELSEIF(J.EQ.4)THEN
         DS(ICOUNTER,NEM(ICOUNTER),4)=(((TKSG(ICOUNTER,NEM(ICOUNTER),4)*
     +                           DELTARF(ICOUNTER))/(R(ICOUNTER,NEM(ICOUNT
     +                           ER))*DELTHETA))*TS(ICOUNTER,NEM(ICOUNTER)
     +                           ,3))+
     +                           (((TKSGN(ICOUNTER,NEM(ICOUNTER),4)*BEDSI
```

```
     +                          DE(ICOUNTER))/DELR(ICOUNTER,NEM(ICOUNTER)
     +                          ,4))*(T(DUMBLX(ICOUNTER)-1,DUMBLY(ICOUNTE
     +                          R))))
C Rectangular section:lines 5&6:
      ELSEIF(J.EQ.5)THEN
         DS(ICOUNTER,NEM(ICOUNTER)-1,5)=(((TKSGN(ICOUNTER,NEM(ICOUNTER)-
     +                        1,5)*hh(ICOUNTER))/((DELTAXHS(ICOUNTER,NEM
     +                        (ICOUNTER)-1)/2.)+(DELTAX(DUMTLX(ICOUNTER)-1
     +                        )/2.)))*(T(DUMTLX(ICOUNTER)-1,DUMTLY(ICOUNTE
     +                        R))))
     +                        +(((TKSGE(ICOUNTER,NEM(ICOUNTER)-1,5)*DELTAX
     +                        HS(ICOUNTER,NEM(ICOUNTER)-1))/(DELYHSG(ICOUN
     +                        TER,NEM(ICOUNTER)-1)))*(T(DUMTLX(ICOUNTER),
     +                        DUMTLY(ICOUNTER)-1)))
     +                        +(((TKSG(ICOUNTER,NEM(ICOUNTER)-1,5)*DELTAXH
     +                        S(ICOUNTER,NEM(ICOUNTER)-1))/((R(ICOUNTER,
     +                        NEM(ICOUNTER)-1)*(PI/8.))+(hh(ICOUNTER)/2.)
     +                        ))*(TS(ICOUNTER,NEM(ICOUNTER)-1,4)))
      ELSEIF(J.EQ.6)THEN
         DS(ICOUNTER,NEM(ICOUNTER)-1,6)=(((TKSGN(ICOUNTER,NEM(ICOUNTER)-
     +                        1,6)*hh(ICOUNTER))/((DELTAXHS(ICOUNTER,NEM
     +                        (ICOUNTER)-1)/2.)+(DELTAX(DUMTRX(ICOUNTER)+1
     +                        )/2.)))*(T(DUMTRX(ICOUNTER)+1,DUMTRY(ICOUNTE
     +                        R))))
     +                        +(((TKSGE(ICOUNTER,NEM(ICOUNTER)-1,6)*DELTAX
     +                        HS(ICOUNTER,NEM(ICOUNTER)-1))/(DELYHSG(ICOUN
     +                        TER,NEM(ICOUNTER)-1)))*(T(DUMTRX(ICOUNTER),
     +                        DUMTRY(ICOUNTER)-1)))
     +                        +(((TKSG(ICOUNTER,NEM(ICOUNTER)-1,6)*DELTAXH
     +                        S(ICOUNTER,NEM(ICOUNTER)-1))/((R(ICOUNTER,
     +                        NEM(ICOUNTER)-1)*(PI/8.))+(hh(ICOUNTER)/2.)
     +                        ))*(TS(ICOUNTER,NEM(ICOUNTER)-1,1)))
      ENDIF

C PS,QS FOR NODES ALONG LINE J:
C------------------------------

C PS, QS FOR CYLINDRICAL LINE NODES:
      IF(J.EQ.1.OR.J.EQ.2.OR.J.EQ.3.OR.J.EQ.4)THEN
         DO 70016 I=2,NEM(ICOUNTER)
            PS(ICOUNTER,I,J)=(BS(ICOUNTER,I,J)/(AS(ICOUNTER,I,J)-(CS(
     +                       ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
            QS(ICOUNTER,I,J)=((DS(ICOUNTER,I,J)+(CS(ICOUNTER,I,J)*QS(
     +                       ICOUNTER,I-1,J)))/(AS(ICOUNTER,I,J)-(CS(
     +                       ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
70016    CONTINUE

C PS,QS FOR RECTANGULAR LINE NODES:
      ELSEIF(J.EQ.5.OR.J.EQ.6)THEN
         DO 7007 I=2,NEM(ICOUNTER)-1
           PS(ICOUNTER,I,J)=(BS(ICOUNTER,I,J)/(AS(ICOUNTER,I,J)-(CS(
     +                       ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
           QS(ICOUNTER,I,J)=((DS(ICOUNTER,I,J)+(CS(ICOUNTER,I,J)*QS(
     +                       ICOUNTER,I-1,J)))/(AS(ICOUNTER,I,J)-(CS(
     +                       ICOUNTER,I,J)*PS(ICOUNTER,I-1,J))))
 7007       CONTINUE
      ENDIF

      IF(J.EQ.1.OR.J.EQ.2.OR.J.EQ.3.OR.J.EQ.4)THEN
         TS(ICOUNTER,NEM(ICOUNTER),J)=QS(ICOUNTER,NEM(ICOUNTER),J)
      ELSEIF(J.EQ.5.OR.J.EQ.6)THEN
         TS(ICOUNTER,NEM(ICOUNTER)-1,J)=QS(ICOUNTER,NEM(ICOUNTER)-1,J)
      ENDIF
C-----------------------------------------------------------------

C ALGORITHM:
      IF(J.EQ.1.OR.J.EQ.2.OR.J.EQ.3.OR.J.EQ.4)THEN
         DO 7030 I=NEM(ICOUNTER)-1,2,-1
            TSOLD(ICOUNTER,I,J)=TS(ICOUNTER,I,J)
```

```
            TS(ICOUNTER,I,J)=((PS(ICOUNTER,I,J)*TS(ICOUNTER,I+1,J))+
     +                        QS(ICOUNTER,I,J))
            IF(DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J)).GT.DELOLD)THEN
               DELOLD=DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J))
            ENDIF
 7030     CONTINUE

      ELSEIF(J.EQ.5.OR.J.EQ.6)THEN
         DO 7033 I=NEM(ICOUNTER)-2,2,-1
            TSOLD(ICOUNTER,I,J)=TS(ICOUNTER,I,J)
            TS(ICOUNTER,I,J)=((PS(ICOUNTER,I,J)*TS(ICOUNTER,I+1,J))+
     +                        QS(ICOUNTER,I,J))
            IF(DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J)).GT.DELOLD)THEN
               DELOLD=DABS(TS(ICOUNTER,I,J)-TSOLD(ICOUNTER,I,J))
            ENDIF
 7033     CONTINUE
      ENDIF
C-----------------------------------------------------------------
 7005     CONTINUE
 7004 CONTINUE


C-------------------------------------------------------------------------
C END OF SWEEP THROUGH EACH SUBGRID
C-------------------------------------------------------------------------
      END



C ======================================================================
C TITLE: RDPARAM()
C
C DESCRIPTION:
C
C     This subroutine reads all parameters from a file
C
C ---------------------------------------------------------------------
      SUBROUTINE RDPARAM()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)

      EXTERNAL COMMENTS
```

```
      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z

      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody
```

```
C ======================================================================
C SET-UP OF OVERALL GRID, with point 1,1 as the top left corner.  All
C nodes in the grid will be defined in relation to this 1,1 point.
C User defines vectors for spacing in the x and y directions, from which
C matrices of spacings b/w all nodes and all interfaces can be deduced:
C======================================================================
C       CALL CLEARSCREEN($GCLEARSCREEN)
C----------------------------------------------------------------------
      CALL COMMENTS()
      READ (3,8000) N
      DO 100 I=1,N-1
         CALL COMMENTS()
         READ (3,8010) DELX(I)
  100 CONTINUE
 1000 FORMAT(A46,I3)
 1005 FORMAT(A9,I3)
 8000 FORMAT(I5)
 8010 FORMAT(F13.7)
C SET UP DELTAX VECTOR:
C DELTAX is the distance between control volume faces e and w

C For nodes along line x=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAX(1)=DELX(1)/2.
C For second node (distance b/w interfaces on either side of node 2,J):
         DELTAX(2)=DELX(1)
C For interior nodes:
         DO 135 I=3,N-1
            DELTAX(I)=(DELX(I-1)-(DELTAX(I-1)/2.))*2.
  135    CONTINUE
C For last node (distance from e interface of (N-1)th node to Nth node)
         DELTAX(N)=DELX(N-1)-(DELTAX(N-1)/2.)
C ----------------------------------------------------------------------
      CALL COMMENTS()
      READ (3,8000) M
      DO 105 J=1,M-1
      CALL COMMENTS()
         READ (3,8010) DELY(J)
  105 CONTINUE
 1010 FORMAT(A46,I3)
 1015 FORMAT(A9,I3)

C SET UP DELTAY VECTOR:
C DELTAY is the distance b/w control volume faces n and s

C For nodes along line y=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAY(1)=DELY(1)/2.
C For second node (distance b/w interfaces on either side of node I,2)
```

```
         DELTAY(2)=DELY(1)
C For interior nodes:
         DO 155 J=3,M-1
            DELTAY(J)=(DELY(J-1)-(DELTAY(J-1)/2.))*2.
  155    CONTINUE
C For last node (dist. from s interface of (N-1)th node to Nth node)

         DELTAY(M)=DELY(M-1)-(DELTAY(M-1)/2.)
C-----------------------------------------------------------------------

C TEMPERATURES along the top row of the grid are known - these are ground
C surface temperatures-input by the user (for now-all nodes will be same)
      CALL COMMENTS()
      READ (3,8010) Tg
      DO 170 I=1,N
         T(I,1)=Tg
  170 CONTINUE

C ---------------------------------------------------------------------
C Thermal conductivity of the ground
      CALL COMMENTS()
      READ (3,8010)  Kg
      DO 180 I=1,N
         DO 175 J=1,M
            TKMG(I,J)=Kg
  175    CONTINUE
  180 CONTINUE
C Redefine nodes in regions of other thermal conductivities - ie insulation
C boards:
      CALL COMMENTS()
      READ (3,8000) REGNUM
      IF(REGNUM.GT.0)THEN
       DO 171 I=1,REGNUM
      CALL COMMENTS()
         READ (3,8000) REGNUMX(I)
      CALL COMMENTS()
         READ (3,8000) REGNUMY(I)
      CALL COMMENTS()
         READ (3,8000) REGNODX(I)
      CALL COMMENTS()
         READ (3,8000) REGNODY(I)
      CALL COMMENTS()
         READ (3,8010) KREG(I)
         DO 172 J=REGNODX(I),(REGNODX(I)+(REGNUMX(I)-1))
            DO 173 L=REGNODY(I),(REGNODY(I)+(REGNUMY(I)-1))
               TKMG(J,L)=KREG(I)
  173       CONTINUE
  172    CONTINUE
  171 CONTINUE

 1016 FORMAT(A10,I3)
 1017 FORMAT(A44,I3,A2)
      ENDIF

C Set up interface thermal conductivities for all main grid nodes:
C---------------------------------------------------------------------
C ALL NORTH COEFFICIENTS:
C FNMG is the f required for calculating the north interface conductivity
      FNMG(2)=DELTAY(1)/DELY(1)
      DO 156 J=3,M
        FNMG(J)=(DELTAY(J-1)/2.)/DELY(J-1)
  156 CONTINUE
      DO 157 I=1,N
         DO 158 J=2,M
            TKMGN(I,J)=(((1-FNMG(J))/TKMG(I,J))+(FNMG(J)/TKMG(I,J-1)))**
     +(-1)
  158    CONTINUE
  157 CONTINUE
```

```
C-----------------------------------------------------------------------
C ALL EAST COEFFICIENTS:
      FEMG(N-1)=DELTAX(N)/DELX(N-1)
      DO 159 I=1,N-2
         FEMG(I)=(DELTAX(I+1)/2.)/DELX(I)
  159 CONTINUE
      DO 176 I=1,N-1
         DO 177 J=1,M
            TKMGE(I,J)=(((1-FEMG(I))/TKMG(I,J))+(FEMG(I)/TKMG(I+1,J)))**
     +(-1)
  177    CONTINUE
  176 CONTINUE
C =====================================================================
C DEFINING THE NUMBER OF PIPES
      CALL COMMENTS()
      READ (3,8000) NPIPE
  610 FORMAT (I3)
      CALL COMMENTS()
      READ (3,8000) NHSG
      NFSG=NPIPE-NHSG
C ---------------------------------------------------------------------
C LOOP - ENTERING INFORMATION ABOUT EACH PIPE:
      DO 185 ICOUNT=1,NPIPE

C ---------------------------------------------------------------------
C Loop for entering the radius of each pipe in the system - inside
C and outside radii
         CALL COMMENTS()
         READ (3,8010)  IDIAM(ICOUNT)
         IRADIUS(ICOUNT)=IDIAM(ICOUNT)/2.
         CALL COMMENTS()
         READ (3,8010)  ODIAM(ICOUNT)
         ORADIUS(ICOUNT)=ODIAM(ICOUNT)/2.
         CALL COMMENTS()
         READ (3,8010) TKPIP(ICOUNT)

C ---------------------------------------------------------------------
C Entering the insulation thickness (if any) on pipe
C (casing is left out because it is too small for the cylindrical grid
C  system and its relative contribution to resistance is negligible)
C      CALL COMMENTS()
C         READ (3,8010)  THICKINS(ICOUNT)
C         RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
C        IF(THICKINS(ICOUNT).EQ.0.0)THEN
C           TKINSULN(ICOUNT)=0.0
C         ENDIF
C
C ---------------------------------------------------------------------
C Entering the insulation thickness (if any) on pipe
C (casing is left out because it is too small for the cylindrical grid
C  system and its relative contribution to resistance is negligible)
      IF(ICOUNT.GT.NFSG)THEN
C          PRINT *,'Note: Pipes with reduced grids cannot have insulation'
         THICKINS(ICOUNT)=0.0
         TKINSULN(ICOUNT)=0.0
         RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
C          GOTO 411
      ELSE
 2200 FORMAT (A41,I5,A9)
         CALL COMMENTS()
         READ (3,8010) THICKINS(ICOUNT)
         RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
         IF(THICKINS(ICOUNT).EQ.0.0)THEN
            TKINSULN(ICOUNT)=0.0
         ENDIF
      ENDIF
 2210    FORMAT (I5,A8)
C ---------------------------------------------------------------------
```

```
C Entering the thermal conductivities of the insulation material around
C the pipe
         IF(THICKINS(ICOUNT).GT.0.0)THEN
         CALL COMMENTS()
         READ (3,8010)  TKINSULN(ICOUNT)
         ENDIF
C -------------------------------------------------------------------
C Enter the fluid temperature in each pipe
      CALL COMMENTS()
         READ (3,8010) TFLUID(ICOUNT)
C -------------------------------------------------------------------
  185 CONTINUE


c FULL SUBGRIDS
C ===================================================================
C LOCATING SUBGRIDS
      DO 190 ICOUN=1,NFSG
      CALL COMMENTS()
         READ (3,8010)  BEDSIDE(ICOUN)
C-------------------------------------------------------------------

C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep
         CALL COMMENTS()
         READ (3,8000) DUMTLX(ICOUN)
         CALL COMMENTS()
         READ (3,8000) DUMTLY(ICOUN)
         DUMBLX(ICOUN)=DUMTLX(ICOUN)
         DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
         DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
         DUMTRY(ICOUN)=DUMTLY(ICOUN)
         DUMBRX(ICOUN)=DUMTRX(ICOUN)
         DUMBRY(ICOUN)=DUMBLY(ICOUN)
  190 CONTINUE
      PI=3.14159265359
      DO 195 ICOUNTER=1,NFSG
C SET UP SPACING FOR SUBGRID NODES:
C---------------------------------

         CALL COMMENTS()
         READ (3,8000) NEM(ICOUNTER)
 1085 FORMAT(A18,I3)
C Vector of spacing between sub-grid nodes along any radial line:
         DO 200 I=1,NEM(ICOUNTER)-2
            CALL COMMENTS()
            READ (3,8010) EMSPACE(ICOUNTER,I)
  200    CONTINUE

C Set up vector of r's - radius to each node:
         R(ICOUNTER,1)=IRADIUS(ICOUNTER)
         DO 205 I=2,NEM(ICOUNTER)-1
            R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  205    CONTINUE
         R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                    (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))

C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
         DO 206 J=1,8
           DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                      R(ICOUNTER,NEM(ICOUNTER)-1)
  206 CONTINUE

C-------------------------------------------------------------------------
C DELR,DELTAR,DELTHETA,DELTATHETA - SUBGRIDS:

C Set up DELR for rest of sub-grid nodes:
```

```
      DO 215 I=1,NEM(ICOUNTER)-2
         DO 210 J=1,8
            DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
      WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
210      CONTINUE
215   CONTINUE


C DELR for last node - is distance between final subgrid node and
C adjacent main grid node

C     Z=distance from side of subgrid to adjacent node of main grid

         Z(ICOUNTER,1)=(DELTAX(DUMTRX(ICOUNTER)+1.))/2.
         Z(ICOUNTER,2)=(DELTAY(DUMTRY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,3)=(DELTAY(DUMTLY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,4)=(DELTAX(DUMTLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,5)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,6)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,7)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,8)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.

C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
         h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +               R(ICOUNTER,NEM(ICOUNTER)))

C     aa=the horizontal distance from the subgrid node to the center of
C        the main grid node (ie. lined up with adjacent main grid node
         aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +                 *DSIN(PI/8.)))
         DO 220 J=1,8
         DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
     +                                    NTER))**2.)+(aa(ICOUNTER)**2.))
220   CONTINUE

C DELTHETA are the same for all nodes - e and w DELTHETA are the same
C but note that for the last nodes there are only three sides to the
C control volume, so each node equations will include EITHER w or e
         DELTHETA=PI/4.

C DELTATH for all nodes is same:
         DELTATH=PI/4.

C DELTAR for 1st circle of nodes: (similar scheme to deltax above)
         DO 245 J=1,8
            DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
245      CONTINUE

C DELTAR for second circle of nodes:
         DO 250 J=1,8
            DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                              1,J))*2.)
250      CONTINUE

C DELTAR for interior nodes up to 2nd last node:
         DO 260 I=3,NEM(ICOUNTER)-1
            DO 255 J=1,8
               DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                                 ICOUNTER,I-1,J)/2.))*2.
255         CONTINUE
260      CONTINUE


C DELTAR for last node of sub-grid
         DO 265 J=1,8
            DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                        (PI/8.)))-(BEDSIDE(ICOUNTER)))
265      CONTINUE
```

```
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID
C Thermal Conductivities AT subgrid nodes:

C     For 1st circle of nodes on inside of pipe:
         DO 270 J=1,8
            TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  270    CONTINUE
         DO 280 I=2,NEM(ICOUNTER)
            DO 275 J=1,8
               IF(R(ICOUNTER,I).LT.RADINS(ICOUNTER))THEN
                  TKSG(ICOUNTER,I,J)=TKINSULN(ICOUNTER)
               ELSE
                  TKSG(ICOUNTER,I,J)=Kg
               ENDIF
  275       CONTINUE


  280    CONTINUE


C-----------------------------------------------------------------------

C Setting up for thermal conductivities at interfaces of subgrid nodes:
C North interface conductivity for all FSG nodes is calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be either nodal conductivity
      DO 266 I=1,NEM(ICOUNTER)-1
         DO 267 J=1,8
            FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                        ,I,J)
            TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                         ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                         I+1,J)))**(-1)
         WRITE (8,*) 'TKSGN(',ICOUNTER,I,J,')=',TKSGN(ICOUNTER,I,J)
  267    CONTINUE
  266 CONTINUE

C Find angle b/w final SG node and MG node:
      DO 264 J=1,8
         THET(ICOUNTER,NEM(ICOUNTER),J)=DASIN(aa(ICOUNTER)/
     +                                  DELR(ICOUNTER,NEM(ICOUNTER),J))
         s(ICOUNTER,NEM(ICOUNTER),J)=(h(ICOUNTER)/DCOS(THET(ICOUNTER,
     +                               NEM(ICOUNTER),J)))
  264 CONTINUE


C Interface conductivity between final subgrid node and main grid node,
C (conductivity of north interface of final node):
         DO 286 J=1,8
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                  -s(ICOUNTER,NEM(ICOUNTER),J))/(DELR(ICOUNTER,NEM
     +                  (ICOUNTER),J))
  286 CONTINUE


          TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)
     +                                /TKMG(DUMTRX(ICOUNTER)+1,DUMTRY(ICOU
     +                                NTER))))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                                TKMG(DUMTRX(ICOUNTER),DUMTRY(ICOU
     +                                NTER)-1)))**(-1)
```

```
      TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                                TKMG(DUMTLX(ICOUNTER),DUMTLY(ICOUNTE
     +                                R)-1)))**(-1)
      TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                               TKMG(DUMTLX(ICOUNTER)-1,DUMTLY(ICOU
     +                               NTER))))**(-1)
      TKSGN(ICOUNTER,NEM(ICOUNTER),5)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),5))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,5))+(FNSG(ICOUNTER,NEM(ICOUNTER),5)/
     +                                TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                                NTER))))**(-1)
      TKSGN(ICOUNTER,NEM(ICOUNTER),6)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),6))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,6))+(FNSG(ICOUNTER,NEM(ICOUNTER),6)/
     +                                TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOU
     +                                NTER)+1)))**(-1)
      TKSGN(ICOUNTER,NEM(ICOUNTER),7)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),7))/TKSG(ICOUNTER,NEM(ICGUNTER)
     +                               ,7))+(FNSG(ICOUNTER,NEM(ICOUNTER),7)/
     +                                TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOUNT
     +                                ER)+1)))**(-1)
      TKSGN(ICOUNTER,NEM(ICOUNTER),8)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),8))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,8))+(FNSG(ICOUNTER,NEM(ICOUNTER),8)/
     +                                TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                NTER))))**(-1)
C-----------------------------------------------------------------------
  195 CONTINUE

C HALF SUBGRIDS
C ==============================================================================
C CALCULATIONS FOR REDUCED EMBEDDED GRIDS
C LOCATING SUBGRIDS
      DO 490 ICOUN=NFSG+1,NPIPE
         CALL COMMENTS()
         READ (3,8010)  BEDSIDE(ICOUN)

C-----------------------------------------------------------------------

C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep
         CALL COMMENTS()
         READ (3,8000) DUMTLX(ICOUN)
         CALL COMMENTS()
         READ (3,8000) DUMTLY(ICOUN)
         DUMBLX(ICOUN)=DUMTLX(ICOUN)
         DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
         DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
         DUMTRY(ICOUN)=DUMTLY(ICOUN)
         DUMBRX(ICOUN)=DUMTRX(ICOUN)
         DUMBRY(ICOUN)=DUMBLY(ICOUN)
  490 CONTINUE
      PI=3.14159265359
      DO 495 ICOUNTER=NFSG+1,NPIPE,1
         CALL COMMENTS()
         READ (3,8000) NEM(ICOUNTER)
         CALL COMMENTS()
         READ (3,8010) hh(ICOUNTER)
C Vector of spacing between sub-grid nodes along any radial line:
         DO 400 I=1,NEM(ICOUNTER)-2
            CALL COMMENTS()
            READ (3,8010) EMSPACE(ICOUNTER,I)
  400    CONTINUE

C-----------------------------------------------------------------------
```

```
C R,DELR,DELTAR,DELTHETA,DELTATHETA - CYLINDRICAL SECTION OF SUBGRIDS:
C Set up vector of r's - radius to each node:
         R(ICOUNTER,1)=IRADIUS(ICOUNTER)
         DO 405 I=2,NEM(ICOUNTER)-1
            R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  405    CONTINUE
         R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                    (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))
C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
         DO 406 J=1,4
           DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                      R(ICOUNTER,NEM(ICOUNTER)-1)
  406 CONTINUE
C Set up DELR for rest of sub-grid nodes:
         DO 415 I=1,NEM(ICOUNTER)-2
            DO 410 J=1,4
               DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
               WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
  410       CONTINUE
  415    CONTINUE

C DELR for last node of radial lines - is distance between final subgrid
C node and adjacent main grid node - for the 4 final nodes along radial
C lines 1 and 4 (cylindrical sedtion of subgrid)
C     Z=distance from side of subgrid to adjacent node of main grid

C       (calculated for MG nodes beside cylindrical part of HSG)
         Z(ICOUNTER,1)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.
         Z(ICOUNTER,2)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,3)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,4)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
         h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                R(ICOUNTER,NEM(ICOUNTER)))
C     aa=the horizontal distance from the subgrid node to the center of
C        the main grid node (ie. lined up with adjacent main grid node
         aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +                 *DSIN(PI/8.)))
         DO 420 J=1,4
         DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
     +                                      NTER))**2.)+(aa(ICOUNTER)**2.))
  420 CONTINUE

C DELTHETA are the same for all nodes
         DELTHETA=PI/4.
C DELTATH for all nodes is same:
         DELTATH=PI/4.
C DELTAR for 1st circle of nodes:
         DO 445 J=1,4
            DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
  445    CONTINUE
C DELTAR for second circle of nodes:
         DO 450 J=1,4
            DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                              1,J))*2.)
  450    CONTINUE
C DELTAR for interior nodes up to 2nd last node:
         DO 460 I=3,NEM(ICOUNTER)-1
            DO 455 J=1,4
               DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                                 ICOUNTER,I-1,J)/2.))*2.
  455       CONTINUE
  460    CONTINUE

C DELTAR for last node of sub-grid
         DO 465 J=1,4
```

```
            DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                        (PI/8.)))-(BEDSIDE(ICOUNTER)))
  465     CONTINUE


C-----------------------------------------------------------------------
C SET UP GEOMETRIES FOR RECTANGULAR NODES OF HSG: there will be NEM-1
C nodes, and they will be on lines 5 and 6:
C-----------------------------------------------------------------------
C e is the horizontal distance from the 1st node (inside pipe) to the
C second node.  c is the vertical distance from the 1st node to the side
C of the cylindrical grid (pipe centerline level) - they are used to
C calculate the distance from the 1st to the 2nd node along lines 5&6
       cc(ICOUNTER)=R(ICOUNTER,1)*DSIN(PI/4.)
       e(ICOUNTER)=R(ICOUNTER,2)-(R(ICOUNTER,1)*DCOS(PI/4.))
C DEL for the 2 first nodes in the rectangular section of the subgrid
C (distance between 1st and 2nd nodes)
       DEL(ICOUNTER,5)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
      +                     (ICOUNTER))**2)
       WRITE (8,*) 'DEL(ICOUNTER,5)=',DEL(ICOUNTER,5)
       DEL(ICOUNTER,6)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
      +                     (ICOUNTER))**2)

C DELTAXHS, DELTAYHS, DELXHSG, DELYHSG - for rectangular subgrid nodes
C -will be same for lines 5 and 6
            DO 467 I=2,NEM(ICOUNTER)-1
               DELTAXHS(ICOUNTER,I)=(DELTAR(ICOUNTER,I,4))
               DELTAYHS(ICOUNTER,I)=hh(ICOUNTER)
  467       CONTINUE
            DO 468 I=2,NEM(ICOUNTER)-2
               DELXHSG(ICOUNTER,I)=R(ICOUNTER,I+1)-R(ICOUNTER,I)

  468       CONTINUE
C DELYHSG - will be same for lines 5 and 6 (due to spacing rules)
         DO 469 I=2,NEM(ICOUNTER)-1
            DELYHSG(ICOUNTER,I)=(hh(ICOUNTER)/2.)+(DELTAY(DUMTLY(ICOUNT
     +                           ER)-1)/2.)
  469    CONTINUE
C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID CYLINDRICAL NODES
C-----------------------------------------------------------------------

C Thermal Conductivities AT subgrid nodes:
C----------------------------------------
C     For 1st circle of nodes on inside of pipe:
         DO 470 J=1,4
            TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  470    CONTINUE
C k for remaining 2 nodes inside pipe wall:
         TKSG(ICOUNTER,1,5)=TKPIP(ICOUNTER)
         TKSG(ICOUNTER,1,6)=TKPIP(ICOUNTER)

C =======TO INSERT RING OF AIR AROUND PIPE CHANGE kg TO kAIR=
         DO 473 J=1,4
            TKSG(2,2,J)=Kg
            TKSG(1,2,J)=Kg
  473    CONTINUE
C=======================================================

C     For rest of nodes:
         DO 480 I=3,NEM(ICOUNTER)
            DO 475 J=1,4
               TKSG(ICOUNTER,I,J)=Kg
  475       CONTINUE
  480    CONTINUE

C========TO INSERT AIR SPACE ABOVE PIPE - BETWEEN PIPE AND INSULN
C       BOARD - CHANGE kg TO kAIR============================
         DO 477 J=5,6
```

```
            TKSG(1,2,J)=Kg
            TKSG(2,2,J)=Kg
  477     CONTINUE

          DO 481 I=3,NEM(ICOUNTER)-1
             DO 476 J=5,6
                TKSG(1,I,J)=Kg
                TKSG(2,I,J)=Kg
  476        CONTINUE
  481     CONTINUE

C=======================================================================



C-----------------------------------------------------------------------

C Thermal conductivities at interfaces of subgrid nodes:
C-----------------------------------------------------------------------

C HSG radial lines 1 to 4:
C-------------------------
C North interface conductivity for HSG nodes along lines 1-4 are calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be the nodal conductivity
       DO 478 I=1,NEM(ICOUNTER)-1
          DO 479 J=1,4
             FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                         ,I,J)
             TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                           ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                           I+1,J)))**(-1)
          WRITE (8,*) 'TKSGN(',ICOUNTER,I,J,')=',TKSGN(ICOUNTER,I,J)
  479     CONTINUE
  478  CONTINUE


C For HSG lines 5 and 6:
C-----------------------

C 1st nodes on lines 5&6:
       DO 482 J=5,6
          FNSG(ICOUNTER,1,J)=(DEL(ICOUNTER,J)-DELTAR(ICOUNTER,1,4))/DEL
     +                        (ICOUNTER,J)
          TKSGN(ICOUNTER,1,J)=(((1-FNSG(ICOUNTER,1,J))/TKSG(ICOUNTER,1,J)
     +                         )+(FNSG(ICOUNTER,1,J)/TKSG(ICOUNTER,2,J)))
     +                         **(-1)
          WRITE (8,*) 'TKSGN(',ICOUNTER,1,J,')=',TKSGN(ICOUNTER,1,J)
  482  CONTINUE
C North interface conductivities for rest of nodes along lines 5&6 -
C rectangular formulation from Patankar


C For nodes 2 to NEM-2:
       DO 474 I=2,NEM(ICOUNTER)-2
          DO 487 J=5,6
             FNSG(ICOUNTER,I,J)=(DELTAXHS(ICOUNTER,I+1)/2.)/DELXHSG
     +                          (ICOUNTER,I)
             TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,
     +                           I,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER
     +                           ,I+1,J)))**(-1)
  487     CONTINUE
  474  CONTINUE

C For final node NEM-1: (borders onto MG node)
       FNSG(ICOUNTER,NEM(ICOUNTER)-1,5)=((DELTAX(DUMTLX(ICOUNTER)-1)/2.)
     +                                  /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                  -1)/2.)+(DELTAX(DUMTLX(ICOUNTER)-1
```

```
     +                                  )/2.)))

      FNSG(ICOUNTEF,NEM(ICOUNTER)-1,6)=((DELTAX(DUMTRX(ICOUNTER)+1)/2.)
     +                                  /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                  -1)/2.)+(DELTAX(DUMTRX(ICOUNTER)+1
     +                                  )/2.)))


      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,5)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,5))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,5))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER)-1,5)/TKMG(DUMTLX(ICOUNTER)-1,
     +                                  DUMTLY(ICOUNTER)))**(-1)


      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,6)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,6))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,6))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER)-1,6)/TKMG(DUMTRX(ICOUNTER)+1,
     +                                  DUMTRY(ICOUNTER)))**(-1)


C East interface for nodes 2 to NEM-1 along lines 5&6 of HSG: border
C onto MG nodes (from Patankar)
      DO 488 I=2,NEM(ICOUNTER)-1
         FEHSG(ICOUNTER,I,5)=(DELTAY(DUMTLY(ICOUNTER)-1)/2.)/DELYHSG
     +                       (ICOUNTER,I)
         FEHSG(ICOUNTER,I,6)=(DELTAY(DUMTRY(ICOUNTER)-1)/2.)/DELYHSG
     +                       (ICOUNTER,I)
         TKSGE(ICOUNTER,I,5)=(((1-FEHSG(ICOUNTER,I,5))/TKSG(ICOUNTER,I,5
     +                       ))+(FEHSG(ICOUNTER,I,5)/TKMG(DUMTLX(ICOUNTE
     +                       R),DUMTLY(ICOUNTER)-1)))**(-1)
         TKSGE(ICOUNTER,I,6)=(((1-FEHSG(ICOUNTER,I,6))/TKSG(ICOUNTER,I,6
     +                       ))+(FEHSG(ICOUNTER,I,6)/TKMG(DUMTRX(ICOUNTE
     +                       R),DUMTRY(ICOUNTER)-1)))**(-1)
  488 CONTINUE
C Interface conductivity between final subgrid nodes along radial lines 1-4
C and main grid node:

C Find angle b/w final SG node and MG node:
      DO 494 J=1,4
         THET(ICOUNTER,NEM(ICOUNTER),J)=DASIN(aa(ICOUNTER)/DELR(ICOUNTER
     +                                  ,NEM(ICOUNTER),J))
         s(ICOUNTER,NEM(ICOUNTER),J)=(h(ICOUNTER)/DCOS(PI/8.))
  494 CONTINUE


C Interface conductivity between final subgrid node and main grid node,
C (conductivity of north interface of final node):
         DO 496 J=1,4
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                  -s(ICOUNTER,NEM(ICOUNTER),J))/(DELR(ICOUNTER,NEM
     +                  (ICOUNTER),J))
  496 CONTINUE

          TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)/
     +                                TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                NTER))))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                               TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOU
     +                               NTER)+1)))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                               TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOUNTE
```

```
     +                                R)+1)))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                                TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                                NTER))))**(-1)
  495 CONTINUE

C END OF LOOP FOR INFO ABOUT REDUCED SUBGRIDS
C-----------------------------------------------------------------------

      CALL COMMENTS()
      read (3,8010) CONVCRIT
      CALL COMMENTS()
      read (3,8000) MAXITER
      END


C ======================================================================
C TITLE: wrtPARAM()
C
C DESCRIPTION:
C
C     This subroutine writes all data to a file
C ----------------------------------------------------------------------
      SUBROUTINE WRTPARAM()

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      INCLUDE 'FGRAPH.FD'
      PARAMETER (MAX=150,MAXP=30,MAXC=50,MAXR=8,MAXREG=30,MAXIT=100000)

      DIMENSION TFLUID(MAXC),T(MAX,MAX),TOLD(MAX,MAX),TSOLD(MAXP,MAXC,MA
     +          XR),P(MAX,MAX),Q(MAX,MAX),A(MAX,MAX),B(MAX,MAX),C(MAX,MA
     +          X),D(MAX,MAX),ODIAM(MAXP),IDIAM(MAXP),ORADIUS(MAXP),IRAD
     +          IUS(MAXP),THICKINS(MAXP),RADINS(MAXP),TKSG(MAXP,MAXC,MAX
     +          R),TKMG(MAX,MAX),BEDSIDE(MAXP),TKPIP(MAXP),TKINSULN(MAXP
     +          ),TKSGN(MAXP,MAXC,MAXR),EMSPACE(MAXP,MAXC),DELTAX(MAX),D
     +          ELTAY(MAX),DELX(MAX),DELY(MAX),DELR(MAXP,MAXC,MAXR),DELT
     +          AR(MAXP,MAXC,MAXR),NEM(MAXP),Z(MAXP,MAXR),R(MAXP,MAXC),D
     +          UMTLX(MAXP),DUMTLY(MAXP),DUMBLX(MAXP),DUMBLY(MAXP),DUMTR
     +          X(MAXP),DUMTRY(MAXP),DUMBRX(MAXP),DUMBRY(MAXP),AS(MAXP,M
     +          AXC,MAXR),BS(MAXP,MAXC,MAXR),CS(MAXP,MAXC,MAXR),DS(MAXP,
     +          MAXC,MAXR),PS(MAXP,MAXC,MAXR),QS(MAXP,MAXC,MAXR),TS(MAXP
     +          ,MAXC,MAXR),h(MAXP),aa(MAXP),DELTARF(MAXP),cc(MAXP),e(MA
     +          XP),DEL(MAXP,MAXR),DELTAXHS(MAXP,MAXC),DELTAYHS(MAXP,MAX
     +          C),DELYHSG(MAXP,MAXC),DELXHSG(MAXP,MAXC),QN5(MAXP,MAXC),
     +          QN6(MAXP,MAXC),QN5T(MAXP),QN6T(MAXP),QN5WT(MAXP,MAXC),
     +          QN6WT(MAXP,MAXC),QN5WTT(MAXP),QN6WTT(MAXP),hh(MAXP),REGN
     +          UMX(MAXREG),REGNUMY(MAXREG),REGNODX(MAXREG),REGNODY(MAXR
     +          EG),KREG(MAXREG),FNMG(MAX),FEMG(MAX),TKMGE(MAX,MAX),
     +          TKMGN(MAX,MAX),TKSGE(MAXP,MAXC,MAXR),FNSG(MAXP,MAXC,
     +          MAXR),FEHSG(MAXP,MAXC,MAXR),THET(MAXP,MAXC,MAXR),s(MAXP,
     +          MAXC,MAXR)


      EXTERNAL COMMENTS

      COMMON  A,aa,AS,B,BS,BEDSIDE,cc,C,CONVCRIT,CS,D,DS,DEL,DELOLD,DELR
     +        ,DELTAR,DELTARF,DELTHETA,DELTATH,DELTAX,DELTAXHS,DELTAY,
     +        DELTAYHS,DELX,DELXHSG,DELY,DELYHSG,DUMBLX,DUMBLY,DUMBRX,
     +        DUMBRY,DUMTLX,DUMTLY,DUMTRX,DUMTRY,e,EMSPACE,FEHSG,FEMG,
     +        FNSG,FNMG,h,hh,IDIAM,IRADIUS,Kg,KREG,MAXITER,NEM,NFSG,
     +        NHSG,NPIPE,ODIAM,ORADIUS,P,PI,PS,Q,QS,RADINS,R,REGNODX,
     +        REGNODY,REGNUM,REGNUMX,REGNUMY,s,T,TFLUID,TG,THET,THICKINS
     +        ,TKINSULN,TKMG,TKMGE,TKMGN,TKPIP,TKSG,TKSGE,TKSGN,TOLD,TS,
     +        TSOLD,N,M,QN5,QN6,QN5T,QN6T,QN5WT,QN6WT,QN5WTT,QN6WTT,Z


      DOUBLE PRECISION Kg,KREG,IRADIUS,IDIAM
```

```
      INTEGER DUMTLX,DUMTLY,DUMBLX,DUMBLY,DUMTRX,DUMTRY,DUMBRX,DUMBRY,
     +        regnum,regnumx,regnumy,regnodx,regnody

C ===============================================================
C SET-UP OF OVERALL GRID, with point 1,1 as the top left corner.  All
C nodes in the grid will be defined in relation to this 1,1 point.
C User defines vectors for spacing in the x and y directions, from which
C matrices of spacings b/w all nodes and all interfaces can be deduced:
C===============================================================
C      CALL CLEARSCREEN($GCLEARSCREEN)
C---------------------------------------------------------------------
      REWIND(3)
      WRITE (3,8020) '/Number of nodes in the x-direction.'
      WRITE (3,8000) N
      DO 100 I=1,N-1
         WRITE (3,8030) '/Distance (x-direction) between node',I
         WRITE (3,8040) '/and node',I+1
         WRITE (3,8010) DELX(I)
  100 CONTINUE
 8000 FORMAT(I5)
 8010 FORMAT(F13.7)
 8020 FORMAT(A36)
 8030 FORMAT(A36,I3)
 8040 FORMAT(A9,I3)

C SET UP DELTAX VECTOR:
C DELTAX is the distance between control volume faces e and w

C For nodes along line x=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAX(1)=DELX(1)/2.
C For second node (distance b/w interfaces on either side of node 2,J):
         DELTAX(2)=DELX(1)
C For interior nodes:
         DO 110 I=3,N-1
            DELTAX(I)=(DELX(I-1)-(DELTAX(I-1)/2.))*2.
  110    CONTINUE
C For last node (distance from e interface of (N-1)th node to Nth node)
         DELTAX(N)=DELX(N-1)-(DELTAX(N-1)/2.)
C ---------------------------------------------------------------
         WRITE (3,8399) '/-----------------------------------------
     +-------------'
      WRITE (3,8020) '/Number of nodes in the y-direction.'
      WRITE (3,8000) M
      DO 120 J=1,M-1
         WRITE (3,8030) '/Distance (y-direction) between node',J
         WRITE (3,8040) '/and node',J+1
         WRITE (3,8010) DELY(J)
  120 CONTINUE
         WRITE (3,8399) '/-----------------------------------------
     +-------------'
C SET UP DELTAY VECTOR:
C DELTAY is the distance b/w control volume faces n and s

C For nodes along line y=1 (from node 1 to 1st interface-b/w 1 and 2)
         DELTAY(1)=DELY(1)/2.
C For second node (distance b/w interfaces on either side of node I,2)

         DELTAY(2)=DELY(1)
C For interior nodes:
         DO 130 J=3,M-1
            DELTAY(J)=(DELY(J-1)-(DELTAY(J-1)/2.))*2.
  130    CONTINUE
C For last node (dist. from s interface of (N-1)th node to Nth node)

         DELTAY(M)=DELY(M-1)-(DELTAY(M-1)/2.)
C---------------------------------------------------------------------

C TEMPERATURES along the top row of the grid are known - these are ground
```

```
C surface temperatures-input by the user (for now-all nodes will be same)
      WRITE (3,8050) '/Ground surface temperature in degrees Celsius:'
 8050 FORMAT(A47)
      WRITE (3,8010) Tg
      DO 140 I=1,N
         T(I,1)=Tg
  140 CONTINUE

C ---------------------------------------------------------------------
C Thermal conductivity of the ground
      WRITE (3,8060) '/Thermal conductivity of the ground'

 8060 FORMAT(A35)
      WRITE (3,8010) Kg
      DO 150 I=1,N
         DO 160 J=1,M
            TKMG(I,J)=Kg
  160    CONTINUE
  150 CONTINUE
C Redefine nodes in regions of other thermal conductivities - ie insulation
C boards:
         WRITE (3,8399) '/-----------------------------------------
     +-------------'
      WRITE (3,8070) '/Number of rectangular regions of different'
 8070 FORMAT(A43)
      WRITE (3,8080) '/thermal conductivity than the soil:'
 8080 FORMAT(A36)
      WRITE (3,8000) REGNUM
      IF(REGNUM.GT.0)THEN
       DO 170 I=1,REGNUM
         WRITE (3,8090) '/Number of nodes spanned in the x-direction'
 8090 FORMAT(A43)
         WRITE (3,8100) '/by region',I
 8100 FORMAT(A10,I3)
         WRITE (3,8000) REGNUMX(I)
         WRITE (3,8070)'/Number of nodes spanned in the y-direction'
         WRITE (3,8100) '/by region',I
         WRITE (3,8000) REGNUMY(I)
         WRITE (3,8110) '/X-coordinate of the top left corner node'
 8110    FORMAT(A41)
         WRITE (3,8100) '/of region',I
         WRITE (3,8000) REGNODX(I)
         WRITE (3,8110) '/Y-coordinate of the top left corner node'
         WRITE (3,8100) '/of region',I
         WRITE (3,8000) REGNODY(I)
         WRITE (3,8120) '/Thermal conductivity of region',I,'?'
 8120 FORMAT(A31,I3,A1)
         WRITE (3,8010) KREG(I)
         DO 180 J=REGNODX(I),(REGNODX(I)+(REGNUMX(I)-1))
            DO 190 L=REGNODY(I),(REGNODY(I)+(REGNUMY(I)-1))
               TKMG(J,L)=KREG(I)
  190       CONTINUE
  180    CONTINUE
         WRITE (3,8399) '/-----------------------------------------
     +-------------'

  170 CONTINUE

 1016 FORMAT(A10,I3)
 1017 FORMAT(A44,I3,A2)
      ENDIF

C Set up interface thermal conductivities for all main grid nodes:
C---------------------------------------------------------------------
C ALL NORTH COEFFICIENTS:
C FNMG is the f required for calculating the north interface conductivity
      FNMG(2)=DELTAY(1)/DELY(1)
      DO 200 J=3,M
        FNMG(J)=(DELTAY(J-1)/2.)/DELY(J-1)
```

```
  200 CONTINUE
      DO 210 I=1,N
         DO 220 J=2,M
            TKMGN(I,J)=(((1-FNMG(J))/TKMG(I,J))+(FNMG(J)/TKMG(I,J-1)))**
     +(-1)
  220    CONTINUE
  210 CONTINUE
C-----------------------------------------------------------------------
C ALL EAST COEFFICIENTS:

      FEMG(N-1)=DELTAX(N)/DELX(N-1)
      DO 230 I=1,N-2
         FEMG(I)=(DELTAX(I+1)/2.)/DELX(I)
  230 CONTINUE
      DO 240 I=1,N-1
         DO 250 J=1,M
            TKMGE(I,J)=(((1-FEMG(I))/TKMG(I,J))+(FEMG(I)/TKMG(I+1,J)))**
     +(-1)
  250    CONTINUE
  240 CONTINUE
C =====================================================================
C DEFINING THE NUMBER OF PIPES
      WRITE (3,8130) '/Pipes in this system'
 8130 FORMAT(A20)
      WRITE (3,8000) NPIPE
      WRITE (3,8140) '/Pipes having reduced subgrids due to'
 8140 FORMAT(A37)
      WRITE (3,8150) '/insulation board?'
 8150 FORMAT(A18)
      WRITE (3,8000) NHSG
      NFSG=NPIPE-NHSG
C ---------------------------------------------------------------------
C LOOP - ENTERING INFORMATION ABOUT EACH PIPE:
      DO 260 ICOUNT=1,NPIPE
         write (3,1036) '/-----------------------------------------
     +-------------'
         WRITE (3,1030) '/PIPE:',ICOUNT
 1036    format (A60)
 1030    FORMAT (A6,I3)
C ---------------------------------------------------------------------
C Loop for entering the radius of each pipe in the system - inside
C and outside radii
         WRITE (3,8160) '/Inside diameter of pipe',ICOUNT,'(meters)'
 8160 FORMAT(A24,I3,A8)
         WRITE (3,8010) IDIAM(ICOUNT)
         IRADIUS(ICOUNT)=IDIAM(ICOUNT)/2.
         WRITE (3,1035) '/Outside diameter of pipe',ICOUNT,'(meters)'
         WRITE (3,8010) ODIAM(ICOUNT)
         ORADIUS(ICOUNT)=ODIAM(ICOUNT)/2.
 1035    FORMAT (A25,I3,A8)
         WRITE (3,8170) '/Thermal conductivity of the material of'
 8170 FORMAT(A40)
         WRITE (3,8180) '/pipe',ICOUNT
 8180    FORMAT(A5,I3)
         WRITE (3,8010) TKPIP(ICOUNT)
C ---------------------------------------------------------------------
C Entering the insulation thickness (if any) on pipe
C (casing is left out because it is too small for the cylindrical grid
C  system and its relative contribution to resistance is negligible)
      IF(ICOUNT.GT.NFSG)THEN
c          PRINT *,'Note: Pipes with reduced grids cannot have insulation'
         THICKINS(ICOUNT)=0.0
         TKINSULN(ICOUNT)=0.0
         RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
C          GOTO 411
      ELSE
  417    WRITE (3,2200) '/Thickness of the insulation around pipe)',
     +   ICOUNT,' (meters)'
 2200 FORMAT (A41,I5,A9)
```

```
         WRITE (3,8010) THICKINS(ICOUNT)
         RADINS(ICOUNT)=(THICKINS(ICOUNT)+ORADIUS(ICOUNT))
         IF(THICKINS(ICOUNT).EQ.0.0)THEN
            TKINSULN(ICOUNT)=0.0
         ENDIF
      ENDIF
 2210    FORMAT (I5,A8)

C ---------------------------------------------------------------
C Entering the thermal conductivities of the insulation material around
C the pipe
         IF(THICKINS(ICOUNT).GT.0.0)THEN
         WRITE (3,8190) '/Thermal conductivity of the insulation around'
 8190    FORMAT(A46)
         WRITE (3,8200) '/pipe',ICOUNT,'IN W/mC'
         WRITE (3,8010) TKINSULN(ICOUNT)
 8200    FORMAT (A5,I3,A8)
         ENDIF
C ---------------------------------------------------------------
C Enter the fluid temperature in each pipe
         WRITE (3,8210) '/Fluid temperature of pipe',ICOUNT,'(DEG.C)'
         WRITE (3,8010) TFLUID(ICOUNT)
 8210    FORMAT(A26,I3,A8)
C ---------------------------------------------------------------
  260 CONTINUE
         write (3,1036) '/----------------------------------------
     +-------------'

C ===============================================================
C LOCATING SUBGRIDS
      DO 270 ICOUN=1,NFSG
         WRITE (3,8220) '/distance from the center of pipe',ICOUN
         WRITE (3,8230) '/to the side of the embedded grid around pipe',
     +ICOUN
         WRITE (3,8235) '/in meters'
         WRITE (3,8010) BEDSIDE(ICOUN)
 8235    FORMAT(A10)
 8220    FORMAT (A33,I3)
 8230    FORMAT(A45,I3)
C***PUT IN ERROR MESSAGE FOR: WHAT IF EMBEDDED GRIDS OVERLAP??
C---------------------------------------------------------------


         WRITE (3,8240) '/Main grid nodal coordinates of the top left'
 8240    FORMAT(A44)
         WRITE (3,8250) '/corner node of the subgrid around pipe',ICOUN
 8250    FORMAT(A39,I3)


C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep

         WRITE (3,8260) '/X nodal coordinate:'
 8260    FORMAT(A20)
         WRITE (3,8000) DUMTLX(ICOUN)
         WRITE (3,8260) '/Y nodal coordinate:'
         WRITE (3,8000) DUMTLY(ICOUN)
         DUMBLX(ICOUN)=DUMTLX(ICOUN)
         DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
         DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
         DUMTRY(ICOUN)=DUMTLY(ICOUN)
         DUMBRX(ICOUN)=DUMTRX(ICOUN)
         DUMBRY(ICOUN)=DUMBLY(ICOUN)
  270 CONTINUE
         write (3,1036) '/----------------------------------------
     +-------------'

      PI=3.14159265359
      DO 280 ICOUNTER=1,NFSG
```

```
C SET UP SPACING FOR SUBGRID NODES:
C--------------------------------
         WRITE (3,8270) '/Number of nodes along a radial line of the'
 8270    FORMAT(A43)
         WRITE (3,8280) '/sub-grid for pipe',ICOUNTER
 8280    FORMAT(A18,I3)
         WRITE (3,8000) NEM(ICOUNTER)
C Vector of spacing between sub-grid nodes along any radial line:
         DO 290 I=1,NEM(ICOUNTER)-2
            WRITE (3,8290) '/Distance between node',I,' and node',I+1
 8290       FORMAT(A22,I3,A9,I3)
            WRITE (3,8010) EMSPACE(ICOUNTER,I)
  290    CONTINUE
C Set up vector of r's - radius to each node:
         R(ICOUNTER,1)=IRADIUS(ICOUNTER)
         DO 300 I=2,NEM(ICOUNTER)-1
            R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  300    CONTINUE
         R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                      (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))
C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
         DO 310 J=1,8
           DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                      R(ICOUNTER,NEM(ICOUNTER)-1)
  310 CONTINUE
C-------------------------------------------------------------------------
C DELR,DELTAR,DELTHETA,DELTATHETA - SUBGRIDS:

C Set up DELR for rest of sub-grid nodes:
         DO 320 I=1,NEM(ICOUNTER)-2
            DO 330 J=1,8
               DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
               WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
  330       CONTINUE
  320    CONTINUE

C DELR for last node - is distance between final subgrid node and
C adjacent main grid node

C     Z=distance from side of subgrid to adjacent node of main grid

         Z(ICOUNTER,1)=(DELTAX(DUMTRX(ICOUNTER)+1.))/2.
         Z(ICOUNTER,2)=(DELTAY(DUMTRY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,3)=(DELTAY(DUMTLY(ICOUNTER)-1.))/2.
         Z(ICOUNTER,4)=(DELTAX(DUMTLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,5)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
         Z(ICOUNTER,6)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,7)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
         Z(ICOUNTER,8)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.

C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
         h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                R(ICOUNTER,NEM(ICOUNTER)))

C     aa=the horizontal distance from the subgrid node to the center of
C        the main grid node (ie. lined up with adjacent main grid node
         aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +                 *DSIN(PI/8.)))
         DO 340 J=1,8
         DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
     +                                    NTER))**2.)+(aa(ICOUNTER)**2.))
  340 CONTINUE

C DELTHETA are the same for all nodes - e and w DELTHETA are the same
C but note that for the last nodes there are only three sides to the
C control volume, so each node equations will include EITHER w or e
```

```
          DELTHETA=PI/4.

C DELTATH for all nodes is same:
          DELTATH=PI/4.

C DELTAR for 1st circle of nodes: (similar scheme to deltax above)
          DO 350 J=1,8
             DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
  350     CONTINUE

C DELTAR for second circle of nodes:
          DO 360 J=1,8
             DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                              1,J))*2.)
  360     CONTINUE

C DELTAR for interior nodes up to 2nd last node:
          DO 370 I=3,NEM(ICOUNTER)-1
             DO 380 J=1,8
                DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                                 ICOUNTER,I-1,J)/2.))*2.
  380        CONTINUE
  370     CONTINUE


C DELTAR for last node of sub-grid
          DO 390 J=1,8
             DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                        (PI/8.)))-(BEDSIDE(ICOUNTER)))
  390     CONTINUE

C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID

C Thermal Conductivities AT subgrid nodes:

C      For 1st circle of nodes on inside of pipe:
          DO 400 J=1,8
             TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  400     CONTINUE
          DO 410 I=2,NEM(ICOUNTER)
             DO 420 J=1,8
                IF(R(ICOUNTER,I).LT.RADINS(ICOUNTER))THEN
                   TKSG(ICOUNTER,I,J)=TKINSULN(ICOUNTER)
                ELSE
                   TKSG(ICOUNTER,I,J)=Kg
                ENDIF
  420        CONTINUE


  410     CONTINUE
C-----------------------------------------------------------------------

C Setting up for thermal conductivities at interfaces of subgrid nodes:
C North interface conductivity for all FSG nodes is calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be either nodal conductivity

       DO 430 I=1,NEM(ICOUNTER)-1
          DO 440 J=1,8
             FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                          ,I,J)
             TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                           ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                           I+1,J)))**(-1)
  440     CONTINUE
  430 CONTINUE
```

```
C Interface conductivity between final subgrid node and main grid node,
C (conductivity of north interface of final node):
         DO 450 J=1,8
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                    -(DELTAR(ICOUNTER,NEM(ICOUNTER),J)/2.))/(DELR
     +                    (ICOUNTER,NEM(ICOUNTER),J))
  450 CONTINUE

          TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                    ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)
     +                                    /TKMG(DUMTRX(ICOUNTER)+1,DUMTRY(ICOU
     +                                    NTER)))))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                                    TKMG(DUMTRX(ICOUNTER),DUMTRY(ICOU
     +                                    NTER)-1)))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                                    TKMG(DUMTLX(ICOUNTER),DUMTLY(ICOUNTE
     +                                    R)-1)))**(-1)
          TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                   ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                                   TKMG(DUMTLX(ICOUNTER)-1,DUMTLY(ICOU
     +                                   NTER)))))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),5)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),5))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,5))+(FNSG(ICOUNTER,NEM(ICOUNTER),5)/
     +                                    TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                                    NTER)))))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),6)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),6))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,6))+(FNSG(ICOUNTER,NEM(ICOUNTER),6)/
     +                                    TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOU
     +                                    NTER)+1)))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),7)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),7))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,7))+(FNSG(ICOUNTER,NEM(ICOUNTER),7)/
     +                                   TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOUNT
     +                                   ER)+1)))**(-1)

          TKSGN(ICOUNTER,NEM(ICOUNTER),8)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER),8))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                   ,8))+(FNSG(ICOUNTER,NEM(ICOUNTER),8)/
     +                                    TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                    NTER)))))**(-1)
C-------------------------------------------------------------------------
  280 CONTINUE

C ========================================================================
C CALCULATIONS FOR REDUCED EMBEDDED GRIDS
C LOCATING SUBGRIDS
      DO 460 ICOUN=NFSG+1,NPIPE
    7 FORMAT (A3)
        write (3,1036) '/-----------------------------------------
     +-------------'

        WRITE (3,8300) '/Distance from the center of pipe',ICOUN
 8300   FORMAT(A33,I3)
        WRITE (3,8310) '/to the side of the embedded grid around pipe',
     +   ICOUN
```

```
 8310     FORMAT(A45,I3)
          WRITE (3,8320) '/in meters - note: edges of subgrid must match
     +up'
 8320     FORMAT(A49)
          WRITE (3,8330) '/to interfaces of the 8 surrounding maingrid no
     +des'
 8330     FORMAT(A50)
          WRITE (3,8010) BEDSIDE(ICOUN)
C-----------------------------------------------------------------

          WRITE (3,8340) '/Main grid nodal coordinates of the top left'
 8340     FORMAT(A44)
          WRITE (3,8350) '/corner node of the subgrid around pipe',ICOUN
 8350     FORMAT(A39,I3)

C NOTE: each subgrid is comprised of 4 main grid nodes - these will be
C 'dummy' nodes for the main grid sweep
          WRITE (3,8360) '/X nodal coordinate:'
 8360     FORMAT(A20)
          WRITE (3,8000) DUMTLX(ICOUN)
          WRITE (3,8360) '/Y nodal coordinate:'
          WRITE (3,8000) DUMTLY(ICOUN)
          DUMBLX(ICOUN)=DUMTLX(ICOUN)
          DUMBLY(ICOUN)=DUMTLY(ICOUN)+1
          DUMTRX(ICOUN)=DUMTLX(ICOUN)+1
          DUMTRY(ICOUN)=DUMTLY(ICOUN)
          DUMBRX(ICOUN)=DUMTRX(ICOUN)
          DUMBRY(ICOUN)=DUMBLY(ICOUN)

  460 CONTINUE
      PI=3.14159265359
      DO 470 ICOUNTER=NFSG+1,NPIPE,1
          WRITE (3,8399) '/-----------------------------------------------
     +-------------'

          WRITE (3,8370) '/Number of nodes along one of the four'
 8370     FORMAT(A38)
          WRITE (3,8380) '/radial lines of the half cylindrical sub-grid
     +for'
 8380     FORMAT(A50)
          WRITE (3,8390) '/pipe',ICOUNTER
 8399     FORMAT(A60)
          WRITE (3,8000) NEM(ICOUNTER)
          WRITE (3,8400) '/Height from the pipe center to the bottom'
          WRITE (3,8410) '/of the insulation board (meters) for pipe',
     +ICOUNTER
          WRITE (3,8010) hh(ICOUNTER)
 8390 FORMAT(A5,I3)
 8400 FORMAT(A42)
 8410 FORMAT(A42,I3)
C Vector of spacing between sub-grid nodes along any radial line:
          WRITE (3,8399) '/-----------------------------------------------
     +-------------'

          DO 480 I=1,NEM(ICOUNTER)-2
             WRITE (3,8420) '/Distance between node',I,'and node',I+1
 8420        FORMAT(A22,I3,A8,I3)
             WRITE (3,8430)'/along a radial line of the half cylindrical
     +grid:'
 8430        FORMAT(A50)
             WRITE (3,8010) EMSPACE(ICOUNTER,I)
  480     CONTINUE
C-----------------------------------------------------------------------
C R,DELR,DELTAR,DELTHETA,DELTATHETA - CYLINDRICAL SECTION OF SUBGRIDS:
C Set up vector of r's - radius to each node:
          R(ICOUNTER,1)=IRADIUS(ICOUNTER)
          DO 490 I=2,NEM(ICOUNTER)-1
             R(ICOUNTER,I)=R(ICOUNTER,I-1)+EMSPACE(ICOUNTER,I-1)
  490     CONTINUE
```

```
        R(ICOUNTER,NEM(ICOUNTER))=(((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +                    (BEDSIDE(ICOUNTER)))/2.)+(BEDSIDE(ICOUNTER))
C Distance between 2nd last node and last node is calculated - last node
C is placed on the same radial line midway between the final circle and
C the side of the sub-grid
        DO 500 J=1,4
          DELR(ICOUNTER,NEM(ICOUNTER)-1,J)=R(ICOUNTER,NEM(ICOUNTER))-
     +                                     R(ICOUNTER,NEM(ICOUNTER)-1)
  500 CONTINUE
C Set up DELR for rest of sub-grid nodes:
        DO 510 I=1,NEM(ICOUNTER)-2
           DO 520 J=1,4
              DELR(ICOUNTER,I,J)=EMSPACE(ICOUNTER,I)
              WRITE (8,*) 'DELR(',ICOUNTER,I,J,')=',DELR(ICOUNTER,I,J)
  520      CONTINUE
  510   CONTINUE
C DELR for last node of radial lines - is distance between final subgrid
C node and adjacent main grid node - for the 4 final nodes along radial
C lines 1 and 4 (cylindrical sedtion of subgrid)

C     Z=distance from side of subgrid to adjacent node of main grid

C       (calculated for MG nodes beside cylindrical part of HSG)
        Z(ICOUNTER,1)=(DELTAX(DUMBRX(ICOUNTER)+1.))/2.
        Z(ICOUNTER,2)=(DELTAY(DUMBRY(ICOUNTER)+1.))/2.
        Z(ICOUNTER,3)=(DELTAY(DUMBLY(ICOUNTER)+1.))/2.
        Z(ICOUNTER,4)=(DELTAX(DUMBLX(ICOUNTER)-1.))/2.
C     h=vertical distance from subgrid node to side of subgrid (which
C       is also the interface of the adjacent main grid node
        h(ICOUNTER)=(DCOS(PI/8.))*((BEDSIDE(ICOUNTER)/DCOS(PI/8.))-
     +               R(ICOUNTER,NEM(ICOUNTER)))
C     aa=the horizontal distance from the subgrid node to the center of
C        the main grid node (ie. lined up with adjacent main grid node
        aa(ICOUNTER)=((BEDSIDE(ICOUNTER)/2.)-(R(ICOUNTER,NEM(ICOUNTER))
     +               *DSIN(PI/8.)))
        DO 530 J=1,4
        DELR(ICOUNTER,NEM(ICOUNTER),J)=DSQRT(((Z(ICOUNTER,J)+h(ICOU
     +                                    NTER))**2.)+(aa(ICOUNTER)**2.))
  530 CONTINUE

C DELTHETA are the same for all nodes
        DELTHETA=PI/4.
C DELTATH for all nodes is same:
        DELTATH=PI/4.
C DELTAR for 1st circle of nodes:
        DO 540 J=1,4
           DELTAR(ICOUNTER,1,J)=ORADIUS(ICOUNTER)-IRADIUS(ICOUNTER)
  540   CONTINUE
C DELTAR for second circle of nodes:
        DO 550 J=1,4
           DELTAR(ICOUNTER,2,J)=((DELR(ICOUNTER,1,J)-DELTAR(ICOUNTER,
     +                           1,J))*2.)
  550   CONTINUE
C DELTAR for interior nodes up to 2nd last node:
        DO 560 I=3,NEM(ICOUNTER)-1
           DO 570 J=1,4
              DELTAR(ICOUNTER,I,J)=(DELR(ICOUNTER,I-1,J)-(DELTAR(
     +                             ICOUNTER,I-1,J)/2.))*2.
  570      CONTINUE
  560   CONTINUE

C DELTAR for last node of sub-grid
        DO 580 J=1,4
           DELTAR(ICOUNTER,NEM(ICOUNTER),J)=((BEDSIDE(ICOUNTER)/(DCOS
     +                                      (PI/8.)))-(BEDSIDE(ICOUNTER)))
  580   CONTINUE
C-------------------------------------------------------------------------
C SET UP GEOMETRIES FOR RECTANGULAR NODES OF HSG: there will be NEM-1
C nodes, and they will be on lines 5 and 6:
```

```
C-----------------------------------------------------------------------
C e is the horizontal distance from the 1st node (inside pipe) to the
C second node.  c is the vertical distance from the 1st node to the side
C of the cylindrical grid (pipe centerline level) - they are used to
C calculate the distance from the 1st to the 2nd node along lines 5&6
      cc(ICOUNTER)=R(ICOUNTER,1)*DSIN(PI/4.)
      e(ICOUNTER)=R(ICOUNTER,2)-(R(ICOUNTER,1)*DCOS(PI/4.))
C DEL for the 2 first nodes in the rectangular section of the subgrid
C (distance between 1st and 2nd nodes)
      DEL(ICOUNTER,5)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
     +                    (ICOUNTER))**2)
      DEL(ICOUNTER,6)=DSQRT(e(ICOUNTER)**2+((hh(ICOUNTER)/2.)-cc
     +                   (ICOUNTER))**2)

C DELTAXHS, DELTAYHS, DELXHSG, DELYHSG - for rectangular subgrid nodes
C -will be same for lines 5 and 6
            DO 590 I=2,NEM(ICOUNTER)-1
               DELTAXHS(ICOUNTER,I)=(DELTAR(ICOUNTER,I,4))
               DELTAYHS(ICOUNTER,I)=hh(ICOUNTER)
  590       CONTINUE
            DO 600 I=2,NEM(ICOUNTER)-2
               DELXHSG(ICOUNTER,I)=R(ICOUNTER,I+1)-R(ICOUNTER,I)

  600       CONTINUE
C DELYHSG - will be same for lines 5 and 6 (due to spacing rules)
         DO 610 I=2,NEM(ICOUNTER)-1
            DELYHSG(ICOUNTER,I)=(hh(ICOUNTER)/2.)+(DELTAY(DUMTLY(ICOUNT
     +                           ER)-1)/2.)
  610    CONTINUE
C-----------------------------------------------------------------------
C THERMAL CONDUCTIVITIES - SUBGRID CYLINDRICAL NODES
C-----------------------------------------------------------------------

C Thermal Conductivities AT subgrid nodes:
C-----------------------------------------
C     For 1st circle of nodes on inside of pipe:
         DO 620 J=1,4
            TKSG(ICOUNTER,1,J)=TKPIP(ICOUNTER)
  620    CONTINUE
C k for remaining 2 nodes inside pipe wall:
         TKSG(ICOUNTER,1,5)=TKPIP(ICOUNTER)
         TKSG(ICOUNTER,1,6)=TKPIP(ICOUNTER)

C =======TO INSERT RING OF AIR AROUND PIPE CHANGE kg TO kAIR==
         DO 630 J=1,4
            TKSG(2,2,J)=Kg
            TKSG(1,2,J)=Kg
  630    CONTINUE
C=====================================================

C     For rest of nodes:
         DO 640 I=3,NEM(ICOUNTER)
            DO 650 J=1,4
               TKSG(ICOUNTER,I,J)=Kg
  650       CONTINUE
  640    CONTINUE
C========TO INSERT AIR SPACE ABOVE PIPE - BETWEEN PIPE AND INSULN
C       BOARD - CHANGE kg TO kAIR=====
         DO 659 J=5,6
            TKSG(1,2,J)=Kg
            TKSG(2,2,J)=Kg
  659    CONTINUE

         DO 660 I=3,NEM(ICOUNTER)-1
            DO 670 J=5,6
               TKSG(1,I,J)=Kg
               TKSG(2,I,J)=Kg
  670       CONTINUE
  660    CONTINUE
```

```
C=======================================================================


C-----------------------------------------------------------------------

C Thermal conductivities at interfaces of subgrid nodes:
C-----------------------------------------------------------------------

C HSG radial lines 1 to 4:
C-------------------------
C North interface conductivity for HSG nodes along lines 1-4 are calculated
C (formulation based on heat flux through cylindrical wall).  Conductivity
C will not vary in the east/west direction, therefore interface conductiv-
C ities will simply be the nodal conductivity
      DO 680 I=1,NEM(ICOUNTER)-1
         DO 690 J=1,4
            FNSG(ICOUNTER,I,J)=(DELTAR(ICOUNTER,I+1,J)/2.)/DELR(ICOUNTER
     +                          ,I,J)
            TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,I
     +                           ,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER,
     +                           I+1,J)))**(-1)
  690    CONTINUE
  680 CONTINUE

C For HSG lines 5 and 6:
C-----------------------

C 1st nodes on lines 5&6:
      DO 700 J=5,6
         FNSG(ICOUNTER,1,J)=(DEL(ICOUNTER,J)-DELTAR(ICOUNTER,1,4))/DEL
     +                      (ICOUNTER,J)
         TKSGN(ICOUNTER,1,J)=(((1-FNSG(ICOUNTER,1,J))/TKSG(ICOUNTER,1,J)
     +                        )+(FNSG(ICOUNTER,1,J)/TKSG(ICOUNTER,2,J)))
     +                        **(-1)
  700 CONTINUE
C North interface conductivities for rest of nodes along lines 5&6 -
C rectangular formulation from Patankar


C For nodes 2 to NEM-2:
      DO 710 I=2,NEM(ICOUNTER)-2
         DO 720 J=5,6
            FNSG(ICOUNTER,I,J)=(DELTAXHS(ICOUNTER,I+1)/2.)/DELXHSG
     +                         (ICOUNTER,I)
            TKSGN(ICOUNTER,I,J)=(((1-FNSG(ICOUNTER,I,J))/TKSG(ICOUNTER,
     +                           I,J))+(FNSG(ICOUNTER,I,J)/TKSG(ICOUNTER
     +                           ,I+1,J)))**(-1)
  720    CONTINUE
  710 CONTINUE

C For final node NEM-1: (borders onto MG node)
      FNSG(ICOUNTER,NEM(ICOUNTER)-1,5)=((DELTAX(DUMTLX(ICOUNTER)-1)/2.)
     +                                  /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                  -1)/2.)+(DELTAX(DUMTLX(ICOUNTER)-1
     +                                  )/2.)))

      FNSG(ICOUNTER,NEM(ICOUNTER)-1,6)=((DELTAX(DUMTRX(ICOUNTER)+1)/2.)
     +                                  /((DELTAXHS(ICOUNTER,NEM(ICOUNTER)
     +                                  -1)/2.)+(DELTAX(DUMTRX(ICOUNTER)+1
     +                                  )/2.)))


      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,5)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,5))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                  )-1,5))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                  ER)-1,5)/TKMG(DUMTLX(ICOUNTER)-1,
     +                                  DUMTLY(ICOUNTER))))**(-1)
```

```
      TKSGN(ICOUNTER,NEM(ICOUNTER)-1,6)=(((1-FNSG(ICOUNTER,NEM(ICOUNTER
     +                                    )-1,6))/TKSG(ICOUNTER,NEM(ICOUNTER
     +                                    )-1,6))+(FNSG(ICOUNTER,NEM(ICOUNT
     +                                    ER)-1,6)/TKMG(DUMTRX(ICOUNTER)+1,
     +                                    DUMTRY(ICOUNTER))))**(-1)


C East interface for nodes 2 to NEM-1 along lines 5&6 of HSG: border
C onto MG nodes (from Patankar)

      DO 730 I=2,NEM(ICOUNTER)-1
         FEHSG(ICOUNTER,I,5)=(DELTAY(DUMTLY(ICOUNTER)-1)/2.)/DELYHSG
     +                       (ICOUNTER,I)
         FEHSG(ICOUNTER,I,6)=(DELTAY(DUMTRY(ICOUNTER)-1)/2.)/DELYHSG
     +                       (ICOUNTER,I)
         TKSGE(ICOUNTER,I,5)=(((1-FEHSG(ICOUNTER,I,5))/TKSG(ICOUNTER,I,5
     +                       ))+(FEHSG(ICOUNTER,I,5)/TKMG(DUMTLX(ICOUNTE
     +                       R),DUMTLY(ICOUNTER)-1)))**(-1)
         TKSGE(ICOUNTER,I,6)=(((1-FEHSG(ICOUNTER,I,6))/TKSG(ICOUNTER,I,6
     +                       ))+(FEHSG(ICOUNTER,I,6)/TKMG(DUMTRX(ICOUNTE
     +                       R),DUMTRY(ICOUNTER)-1)))**(-1)
  730 CONTINUE

C Interface conductivity between final subgrid nodes along radial lines 1-4
C and main grid node:

         DO 750 J=1,4
          FNSG(ICOUNTER,NEM(ICOUNTER),J)=(DELR(ICOUNTER,NEM(ICOUNTER),J)
     +                 -(DELTAR(ICOUNTER,NEM(ICOUNTER),J)/2.))/(DELR
     +                 (ICOUNTER,NEM(ICOUNTER),J))
  750 CONTINUE
         TKSGN(ICOUNTER,NEM(ICOUNTER),1)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                                ER),1))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                                ,1))+(FNSG(ICOUNTER,NEM(ICOUNTER),1)/
     +                                TKMG(DUMBRX(ICOUNTER)+1,DUMBRY(ICOU
     +                                NTER))))**(-1)
         TKSGN(ICOUNTER,NEM(ICOUNTER),2)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),2))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,2))+(FNSG(ICOUNTER,NEM(ICOUNTER),2)/
     +                               TKMG(DUMBRX(ICOUNTER),DUMBRY(ICOU
     +                               NTER)+1)))**(-1)
         TKSGN(ICOUNTER,NEM(ICOUNTER),3)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),3))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,3))+(FNSG(ICOUNTER,NEM(ICOUNTER),3)/
     +                               TKMG(DUMBLX(ICOUNTER),DUMBLY(ICOUNTE
     +                               R)+1)))**(-1)
         TKSGN(ICOUNTER,NEM(ICOUNTER),4)=(((1.-FNSG(ICOUNTER,NEM(ICOUNT
     +                               ER),4))/TKSG(ICOUNTER,NEM(ICOUNTER)
     +                               ,4))+(FNSG(ICOUNTER,NEM(ICOUNTER),4)/
     +                               TKMG(DUMBLX(ICOUNTER)-1,DUMBLY(ICOU
     +                               NTER))))**(-1)
  470 CONTINUE


C END OF LOOP FOR INFO ABOUT REDUCED SUBGRIDS
C-------------------------------------------------------------------------


      WRITE (3,8004)'/Convergence criteria for the solution algorithm'
 8004 FORMAT(A48)
      WRITE (3,8010) CONVCRIT
      WRITE (3,8002)'/Maximum number of iterations of the algorithm:'
 8002   FORMAT(A47)
      WRITE (3,8000) MAXITER
```

```
      END



C ===================================================================
C  TITLE: COMMENTS()
C
C  DESCRIPTION:
C
C     This subroutine reads the comments from the file INPUT.DAT
C -------------------------------------------------------------------
      SUBROUTINE COMMENTS()
      CHARACTER COLUMN*1, DUMMY*70
C -------------------------------------------------------------------
      ICOMM=1
      DO 10 WHILE(ICOMM.EQ.1)
      READ(3,800) COLUMN
  800 FORMAT(A1,\)
      IF(COLUMN.EQ.'/')THEN
       READ (3,810) DUMMY
  810  FORMAT(A70)
       ICOMM=1
       ELSE
        ICOMM=0
      ENDIF
   10 CONTINUE
      END
```