EVALUATION OF HIGH TEMPERATURE SUPERCONDUCTIVE THERMAL BRIDGES FOR SPACE BORNE CRYOGENIC DETECTORS

A FINAL REPORT

FOR NASA CONTRACT NAG-1-1500

Submitted to NASA Langley Research Center
Hampton, VA

By

Elaine P. Scott
Associate Professor
Department of Mechanical Engineering
Virginia Tech
Blacksburg, Virginia 24061-0238

May, 1996

EVALUATION OF HIGH TEMPERATURE SUPERCONDUCTIVE THERMAL BRIDGES FOR SPACE BORNE CRYOGENIC DETECTORS

NASA CONTRACT NAG-1-1500
by
Elaine P. Scott, Associate Professor
Department of Mechanical Engineering, Virginia Tech
Blacksburg, Virginia 24061-0238

ABSTRACT

Infrared sensor satellites are used to monitor the conditions in the earth's upper atmosphere. In these systems, the electronic links connecting the cryogenically cooled infrared detectors to the significantly warmer amplification electronics act as thermal bridges and, consequently, the mission lifetimes of the satellites are limited due to cryogenic evaporation. High-temperature superconductor (HTS) materials have been proposed by researchers at the National Aeronautics and Space Administration Langley's Research Center (NASA-LaRC) as an alternative to the currently used manganin wires for electrical connection. The potential for using HTS films as thermal bridges has provided the motivation for the design and the analysis of a spaceflight experiment to evaluate the performance of this superconductive technology in the space environment. The initial efforts were focused on the preliminary design of the experimental system which allows for the quantitative comparison of superconductive leads with manganin leads, and on the thermal conduction modeling of the proposed system (see previous progress report - Scott and Lee, 1994). Most of the HTS materials were indicated to be potential replacements for the manganin wires. In the continuation of this multi-year research, the objectives of this study were to evaluate the sources of heat transfer on the thermal bridges that have been neglected in the preliminary conductive model and then to develop a methodology for the estimation of the thermal conductivities of the HTS thermal bridges in space.

The Joule heating created by the electrical current through the manganin wires was incorporated as a volumetric heat source into the manganin conductive model. The radiative heat source on the HTS thermal bridges was determined by performing a separate radiant interchange analysis within a high-Tc superconductor housing area. Both heat sources indicated no significant contribution on the cryogenic heat load, which validates the results obtained in the preliminary conduction model.

A methodology was presented for the estimation of the thermal conductivities of the individual HTS thermal bridge materials and the effective thermal conductivities of the composite HTS thermal bridges as functions of temperature. This methodology included a sensitivity analysis and the demonstration of the estimation procedure using simulated data with added random errors. The thermal conductivities could not be estimated as functions of temperature; thus the effective thermal conductivities of the HTS thermal bridges were analyzed as constants.

Table of Contents

List of Tables

List of Figures

List of Symbols

A_c	Cross-sectional area (m^2)
$\underline{A}$	Scalar used in the Box-Kanemasu method
b_i	Mean of the parameter estimates
$\boldsymbol{b}$	Estimated parameter vector containing the parameters β_i of the thermal conductivity models
c	Specific heat (J/kg-K)
D_{ij}	Distribution factor between surfaces i and j
$\Delta_g \boldsymbol{b}$	Vector defined in the Box-Kanemasu method
E	Weighted error
e_i	Error on each surface i
$\boldsymbol{G}$	Scalar used in the Box-Kanemasu method
$\boldsymbol{h}$	Scalar interpolation function used in the Box-Kanemasu method
H_c	Critical magnetic field (T)
I	Electrical current (A)
i_i	Incident intensity of radiation (W/m^2-str)
J_c	Critical current density (A/cm^2)
J	Current density (A/cm^2)
k	Thermal conductivity (W/m-K)
L	Length of a thermal bridge (m)
n	Total number of surfaces in the enclosure
$nrings$	number of horizontal rings in the substrate material
N	Number of surfaces where the net heat flux q is known
N_p	Number of temperature data points used the estimation procedure
N_i	Counter of the number of energy bundles emitted by surface i in the Monte-Carlo formulation
N_{ij}	Counter of the number of energy bundles emitted by surface i and absorbed by surface j in the Monte-Carlo formulation
$\boldsymbol{P}$	Vector defined in the modified Box-Kanemasu method
P_{elect}	Electrical power (W)
q''	Heat Flux (W/m^2)
$Q_{rad\text{-}bot}$	Radiative heat load on the bottom of the substrate (T=4 K) from the entire enclosure (W)
$Q_{rad\text{-}topbot}$	Radiative heat load on the bottom of the substrate (T=4 K) from the top of the substrate (T=80 K) (W)

$Q_{rad\text{-}tot}$	Radiative heat load on the entire substrate from the entire enclosure (W)
$Q_{rad\text{-}54}$	Radiative heat load on surface 4 from surface 5 (W)
R	Reflectivity ratio
R_{elect}	Electrical resistance (Ω)
R_{th}	Thermal resistance (K/W)
Rn	Random number used in the Monte-Carlo procedure
s	Standard deviation of measurement errors (K)
S	Least squares function
S_o	Sum of squares value at zero
S_α	Sum of squares value at α
S_{gen}	Volumetric electrical heat source (W/m^3)
T_c	Critical or transition temperature of a superconductor (K)
T	Calculated temperature vector (K)
T_0	Temperature at x=0 (K)
T_L	Temperature at x=L (K)
x	Position along the x axis (m)
X	Sensitivity coefficient matrix
$(X_\beta)^+$	Dimensionless sensitivity coefficient of the parameter β
y	Position along the y axis (m)
Y	Measured temperature vector (K)
z	Position along the z axis (m)

Greek

α	Scalar used in the Box-Kanemasu method
α	Absorptivity
α_e	Temperature coefficient per °C for the electrical resistivity
$\underline{\beta}$	Parameter vector containing the parameters β_i of the thermal conductivity models
$\beta_{T(x)}$	Estimate for the constant effective thermal conductivity of a HTS thermal bridge obtained using simulated data along the thermal bridge (W/m-K)
β_{T_L}	Estimate of the constant effective thermal conductivity of a HTS thermal bridge obtained using simulated data at the warm end of the thermal bridge (W/m-K)
$\beta_{trueT(x)}$	True value of the constant effective thermal conductivity of a HTS thermal bridge estimated using simulated data along the thermal bridge (W/m-K)

β_{trueT_L}	True value of the constant effective thermal conductivity of a HTS thermal bridge estimated using simulated data at the warm end of the thermal bridge (W/m-K)
δ_{ij}	Kronecker delta function
∇	Matrix derivative operator
ϵ	Emissivity
λ	Wavelength (m)
ϕ	Angle made with the surface tangent (rad)
ρ	Reflectivity
ρ_{elect}	Electrical resistivity (Ω.m)
σ	Stefan Boltzmann constant ($W/m^2\text{-}K^4$)
θ	Angle made with the surface normal (rad)

Superscripts

'	Directional
T	Transpose
+	Dimensionless

Subscripts

λ	Spectral
d	Diffuse
s	Specular
eff	effective

List of Abbreviations

AWG	American Wire Gauge
BCS	Bardeen, Cooper, and Schrieffer Theory
BF	HTS thermal bridge BSCCO/FSI
BSCCO	$BiSrCaCu_2O_x$
BY	HTS thermal bridge BSCCO/YSZ
FSI	Fused Silica
GREEN	Green Phase
HTS	High-Temperature Superconductive
IR	Infrared
MRI	Magnetic Resonance Imaging
NASA-LaRC	National Aeronautics and Space Administration's Langley Research Center
SAFIRE	Spectroscopy of the Atmosphere Using Far-Infrared Emissions
SIRTF	Shuttle Infrared Telescope Facility
SQUIDs	Superconducting Quantum Interference Devices
YBCO	$YBa_2Cu_3O_{7-x}$
YF	HTS thermal bridge YBCO/FSI
YG	HTS thermal bridge YBCO/GREEN
YSZ	Yttrium-stabilized zirconia
YY	HTS thermal bridge YBCO/YSZ

CHAPTER 1

Introduction

Infrared (IR) sensors are crucial instruments for monitoring the concentration of chemical radicals present in the earth's upper atmosphere. Their use in sensor satellites allows for the evaluation of both the present condition and future changes in the atmosphere. In these systems, the IR detectors are connected to the data acquisition and storage electronics by the means of an electrical link, as seen in Figure 1.1. The IR detectors require liquid helium cryogenic refrigeration (at ~ 4 K), whereas the electronics must be kept at a considerably warmer temperature (at ~ 80 K). This technology is therefore limited by the heat conducted through the electrical leads, which contributes to the rate of cryogen evaporation. Because the amount of cryogen available characterizes the mission lifetime of the satellites, the minimization of the nonparasitic heat load on the cryogen system by the electrical instrumentation acting as a thermal bridge is of critical interest. An efficient solution results in the use of an electrically conducting and thermally isolating link between the IR detectors and the electronics. In order to meet these conflicting requirements, high-temperature superconducting (HTS) materials have

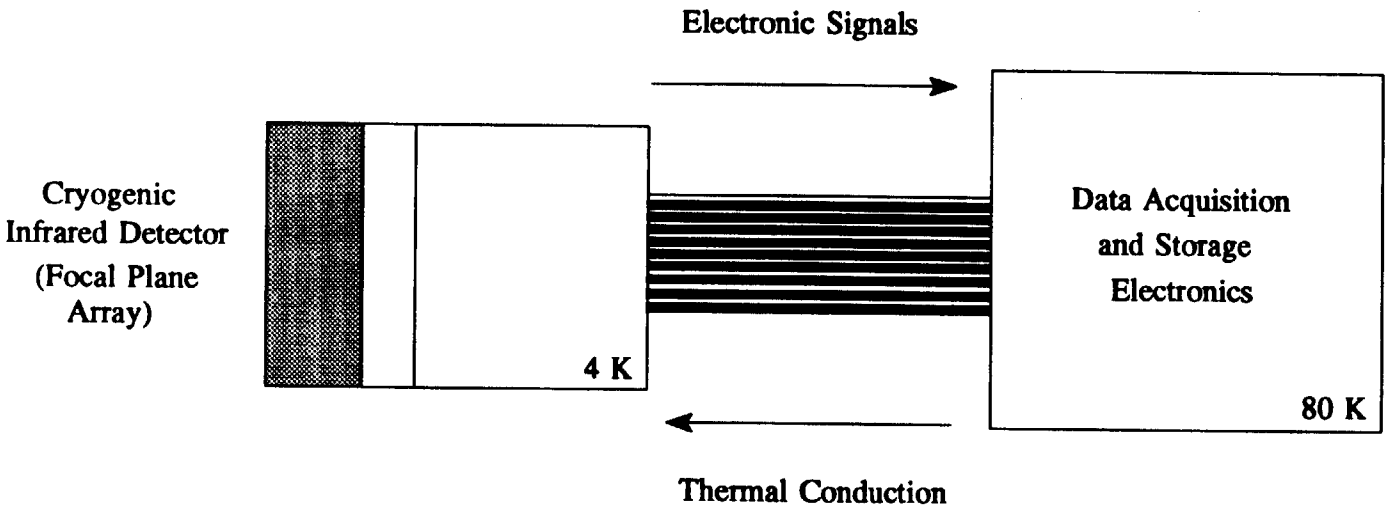


Figure 1.1. Schematic of Infrared Detector Sensor Satellite.

been proposed by researchers at the National Aeronautics and Space Administration's Langley Research Center (NASA-LaRC) as an alternative to the currently used manganin wires for electrical connection. Indeed, HTS materials have been shown in the cryogenic region to exhibit a lower thermal conductivity than manganin with a sufficiently high electrical conductivity for detector applications, where typical currents are on the order of 1 μA. An electronic link with such characteristics is therefore expected to improve the thermal isolation of IR detectors and to reduce the rate of cryogen evaporation, or in other words, to increase the lifetime of the satellites. Based on these theoretical considerations, NASA-LaRC has considered the incorporation of this technology of a HTS link in IR remote sensing platforms such as SAFIRE (Spectroscopy of the Atmosphere using Far Infrared Emission) and SIRTF (Space Infrared Telescope Facility).

Preliminary investigations of replacing manganin technology with HTS films technology have indicated a substantial reduction in thermal loss, translating into approximately 10-15 percent enhancement in mission lifetime (Wise et al., 1992). It is the potential for using HTS materials as thermal bridges in infrared sensor satellite systems that has provided the motivation for the design and the analysis of an experiment to evaluate the performance of this superconductive technology in the space environment. The anticipated space launch of this experiment is projected for 1998. The initial efforts in this multi-year research were focused on the preliminary design of the experimental system and on the thermal conduction modeling of the proposed system (Lee, 1994). In addition, preliminary structural analysis have been performed (Spencer, 1994). The experimental design completed by Lee allows for the quantitative comparison of the heat

load on the cryogen between different types of thermal bridges. Lee concentrated the thermal analysis of the bridges on the conduction due to the temperature gradient along the links, assuming that all other sources of heat transfer (radiation, electrical conduction) are negligible. The results of this conductive analysis indicate that the majority of the HTS materials displays heat loads on the cryogenic system under 15 percent compared to the 20 percent heat load by the currently used manganin wires. These results, based on the assumption that conduction is the dominant mode of heat transfer, show that most of the HTS materials are potential replacements for the manganin wires as electrical connections in infrared sensor satellite systems.

1.1 Goals and Objectives

The continuation of this on-going research focuses on the analysis of the space experimental design for High-T_c superconductive thermal bridges completed by Lee (1994). The overall objectives of this study can be divided into two major areas which are:

1) the evaluation of the sources of heat transfer on the thermal bridges that have been assumed negligible in the conductive analysis performed by Lee,

and

2) the development of a methodology for the estimation of the thermal conductivities of the HTS thermal bridges in the space environment.

The emphasis in the first area of interest is on determining whether or not the

sources of heat transfer neglected in the preliminary conductive mathematical model of the thermal bridges are significant sources of heat load on the cryogen. These sources include the electrical heat source generated by the electronic signals from the IR detectors, and the radiative exchange within the thermal bridge housing area of the experimental design. The electrical heat source was evaluated through the implementation of the Joule heating term into the conductive mathematical model of the thermal bridges. The specific objective was to verify that the Joule heating term affects neither the heat loads on the cryogen, nor the temperature distributions along the thermal bridges. In the assessment of the radiative heat source on the HTS thermal bridges, a radiant interchange analysis within a HTS housing area was conducted using the Monte-Carlo technique. The focal point in this analysis was to compare the radiative heat loads on the HTS thermal bridges with the respective conductive heat loads generated on the cryogen.

The second area of interest focuses on developing a methodology for the estimation of the thermal conductivities of the HTS thermal bridges in space. The overall significance of the capability to determine these thermal properties in the space environment is that it will enable the quantitative assessment of the performance of HTS materials as electronic leads in infrared sensor satellite systems. This capability will also allow the evaluation of various space effects, thus providing a future means of monitoring possible changes in the material due to the space environment. Both the thermal conductivities of the individual HTS thermal bridge materials and the effective thermal conductivities of the composite HTS thermal bridges were sought in this investigation. The temperature dependence in both cases was modeled with the use of polynomials in

temperature. The first specific objective was to conduct a sensitivity analysis on both thermal conductivity models and to determine which thermal conductivity parameter could be estimated for each model. The estimation procedure used in this study is a modification of the Gauss linearization method and is based on the minimization of a least-squares function with respect to the unknown parameters. This method also requires temperature measurements; simulation of experimental data was therefore necessitated as no actual temperature measurements are yet available for this research. The second specific objective was then to demonstrate the estimation procedure using simulated data with added random errors.

CHAPTER 2

Literature Review

This chapter provides a review of the literature for both the analysis of the radiative exchange within a space experimental design for high-temperature superconductive thermal bridges and the estimation of the thermal conductivities of these HTS thermal bridges.

The first section gives an introduction to superconductivity and discusses the possible applications for high-temperature superconductors. The following section summarizes the present state of knowledge pertaining to radiative exchange between surfaces and emphasizes the Monte-Carlo method, which was used to assess the radiative heat load on the HTS thermal bridges. The final section describes a minimization procedure based on the Gauss method which was used to develop a methodology for the estimation of the thermal conductivities of the HTS thermal bridges.

2.1 Superconductivity

After a brief history of superconductivity, this section presents the basic physical characteristics of the two types of superconductors. The aim is to show that superconductors are not only perfect conductors, but also diamagnetic materials. The section concludes with an overview of possible applications for high-temperature superconductors.

2.1.1 Brief History

Superconductivity was discovered in 1911 by Kamerlingh Onnes whose experiments showed that mercury becomes superconductive when cooled to liquid helium temperatures (Doss, 1989). Between 1911 and 1986 many more pure metals, alloys and doped semiconductors were found to have this property. But prior to 1986, the maximum observed transition temperature remained only at 23.21K for Nb_3Ge (Figure 2.1.1). The transition temperature, T_c, is defined as the temperature below which the superconductor is in the well-known "superconducting state" (Allen, 1969). Then in 1986 a new group of materials, known as high-temperature superconductors, was discovered by Bednorz and Müller (1986). In a very short period of time, the maximum critical temperatures were increased from 23K to 35K and then to 90K. Indeed, there are now materials that superconduct at 110-125K (Hunt, 1989), and the latest results obtained by Laguës et al. (1993) suggest possibilities of superconductivity at 250K.

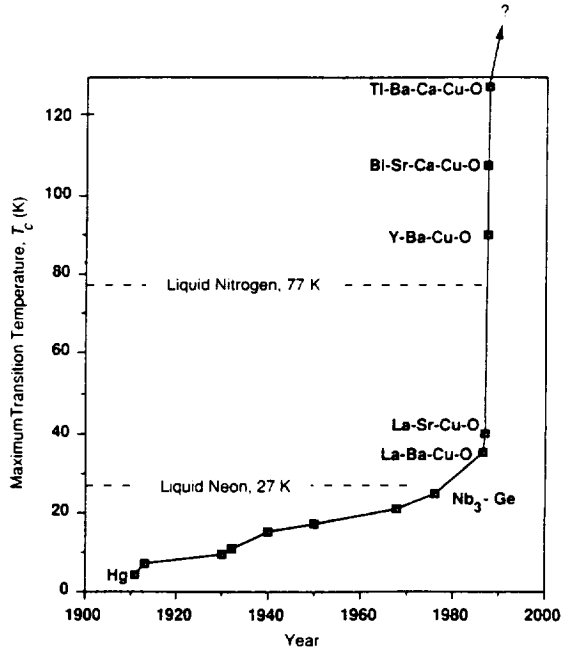


Figure 2.1.1. Maximum Transition Temperature of Superconductors vs. Year of Discovery (Doss, 1989).

2.1.2 The Two Types of Superconductors

Superconductors are generally divided into two types. Most pure elemental superconductors are Type I, whereas most alloy conventional superconductors and all HTS are Type II.

2.1.2.1 Type I Superconductors

In March 1987 at the famous meeting of the American Physical Society which became known as the "Woodstock of Physics", Brian Maple, professor of physics at the University of California, San Diego, said: "*the fascination of superconductivity is associated with the words perfect, infinite, and zero*" (Vidali, 1993). This summarizes

very well the properties of superconductors. Superconductivity actually possesses two outstanding features occuring suddenly at the critical temperature T_c. These features are the zero electrical resistance, or in other words the infinite conductance, and the expulsion of magnetic flux, also called diamagnetism. The second property, the diamagnetism, which is less apparent than the first but also important, was studied by Meissner and Oschsenfeld in 1933. The Meissner effect is the proof that superconductivity is more than perfect conductivity. Both superconductive properties have operating regions. Electric currents can be propagated without resistance if the current density is less than the critical current density J_c of the superconducting material. In the same logic, if the magnetic field applied is greater than the critical magnetic field, H_c, of the superconductor (Figure 2.1.2), its superconductivity is destroyed (Hunt, 1969). Consequently, each superconductor is associated with a specific critical temperature, magnetic field and current density (Bardeen, 1968).

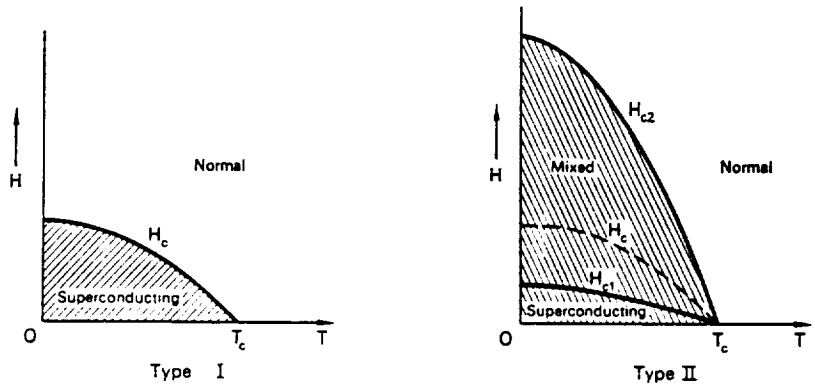


Figure 2.1.2. Phase Diagrams of Type I and Type II Superconductors (Rose-Innes and Rhoderick, 1969).

In Type I superconductors, a relationship between the critical current density and the critical magnetic field exits. This relationship can be derived from the equations describing the electrodynamics of the supercurrent, which are known as the *London equations* (Rose-Innes and Rhoderick, 1969).

But how does superconductivity occur in these materials? The theory developed by Bardeen, Cooper and Schrieffer (1957) (the BCS theory) gives a successful explanation for Type I materials. It is based on a coherent pairing of electrons which takes place with the help of the ions from the solid, under appropriate circumstances. The stronger the coupling, the higher the critical temperature. It should be noted that the critical temperatures predicted by the BCS theory cannot exceed 40 K.

As certain superconductors, especially alloys and impure metals, were noticed to behave differently than Type I superconductors, Abrikosov pointed out in 1957 that the apparent anomalous properties were the inherent features of another class of superconductors, known as Type II (Rose-Innes and Rhoderick, 1969).

2.1.2.2 Type II Superconductors

Type II materials behave like Type I materials for magnetic fields below a critical level H_{c1}, as shown by Figure 2.1.2. However, when the magnetic field is increased above H_{c1}, Type II superconductors allow the flux to penetrate, and the material is in the "mixed-state" until the magnetic field reaches the upper critical level H_{c2} (Rose-Innes and Rhoderick, 1969). In the "mixed state" ($H_{c1}<H<H_{c2}$), normal (nonsuperconducting) regions are microscopically mingled with superconducting regions and the material

remains superconducting. Since the superconductor's energy is not used in expelling the flux, the critical current density is generally much higher in Type II superconductors than in Type I.

Type II superconductors with a critical temperature above 25 K belong by convention to the "high-T_c" category and are called High-Temperature Superconductors (Doss, 1989). With its 40 K limit on the critical temperature, the BCS theory is unable to explain the critical temperatures obtained for some HTS Type II materials. As a result, some alternative theories, which are also based on electron pairs, have been proposed but so far no comprehensive theory has emerged.

The latest discovered HTS materials (YBCO, BSCCO and TlBaCaCuO) are ceramic oxides and have the brittle mechanical characteristics of ceramics. In addition, the superconductivity properties are highly anisotropic. The HTS material characteristics, especially the properties of the superconductors YBCO and BSCCO, and the HTS material structure have been described in detail by Lee (1994). Lee also discussed the processing of HTS films on substrate materials, the requirements of the different techniques, and possible applications for HTS materials. Indeed, it is important to distinguish between bulk HTS materials and their thin-film counterparts: HTS thin films generally have much better critical current densities (J_c can achieve 10^5-10^6 A/m^2) than for the same material in bulk form, although some other properties such as the critical temperature T_c are sometimes worse.

2.1.3 Applications of High-Temperature Superconductors

Prior to the discovery of the HTS materials, the zero electrical resistivity of superconductors cooled to liquid helium temperatures had been exploited for various electronic and large-scale applications. The Josephson effect discovered in 1962 led to the development of Josephson junction switches and Superconducting Quantum Interference Devices (SQUIDs). The routinely high-field applications are restricted to the use of superconducting magnets in particle accelerators in the study of high energy physics and to Magnetic Resonance Imaging (MRI) for medical diagnostic purposes (Geballe and Hulm, 1988). The major advantage of using HTS materials is their high critical temperatures which allows for cooling with liquid nitrogen rather than liquid helium, expensive and inconvenient to use. The greatest savings could be obtained in the field of microelectronics where the refrigeration cost is a major part of the system cost (Geballe and Hulm, 1988). Other possibilities for HTS materials are starting to be realized by industry. Lee (1994) discussed the near-term and longer-term projects where HTS materials are being strongly considered. More work, however, needs to be done in understanding and enhancing the restrictive processing requirements of HTS materials.

Space application of superconductivity is another excellent demonstration where the utilization of superconductors can vastly improve performances or perform tasks that were not previously feasible (Geballe and Hulm, 1988). The use of HTS ceramics for sensor leads for several sensing systems in millimeter and infrared ranges could diminish both the electrical losses and the thermal noise limitations. Consequently reduced cryogenic requirements, high frequency operations and lower power local oscillators could be

ensured (Krishen and Ignatiev, 1988).

High-temperature superconductors offer attractive advantages for a wide variety of both small- and large-scale applications. But it may take a long time before technologies using these new materials are ready to enter the market. However, if the critical temperature is increased to room temperature, as suggested by the latest experiments (Laguës et al., 1993), there would be an immediate opening for many incredible applications.

2.2 Numerical Methods Used for the Evaluation of Radiative Exchange within Enclosures

The determination of the radiative heat load on the HTS thermal bridges will enable us to conclude whether or not radiation affects the heat load on the cryogen. An alternate approach for determining this radiative heat load is to evaluate the radiative exchange within the housing chamber for the thermal bridges.

For many years, the analysis of radiation exchange between surface elements within enclosures has been conducted in various ways. In many practical engineering situations, the diffuse-gray enclosure approximation is made and the geometric configuration factor is computed. This approach is reasonable if the assumptions are well approximated or the directional spectral radiative properties are not available, and if high accuracy is not needed. When the enclosure consists of both diffuse and purely specular surfaces, the configuration factor is extended to the exchange factor (Eckert and Sparrow, 1961). These factors are based on geometry, and as the enclosure geometry becomes complex

they can become very difficult to evaluate. In addition, their use in the formulation of the interchange problems require the inversion of an often large matrix. Considering the limited amount of directional information these factors contain, the use of these tedious numerical techniques might not be worth while.

To account for mixed specular and diffuse reflection models for surfaces, the reflectivity, ρ, can be assumed to be the sum of two components and be expressed as,

$$\rho = \rho^d + \rho^s \ , \tag{2.2.1}$$

where ρ^d is the diffuse component of the reflectivity and ρ^s is the specular component of the reflectivity. This assumed behavior of the reflectivity is taken into account in the computation of the radiation distribution factors D_{ij} (Mahan and Eskin, 1984). This factor represents the fraction of diffusely emitted radiation from surface i absorbed by surface j due to direct radiation and to all directional diffuse and specular reflections. From its definition, one can see that the distribution factor is not strictly a geometrical factor and that it contains directional information. An efficient way for the distribution factors to be computed is through the Monte-Carlo method.

2.2.1 The Monte-Carlo Method

The Monte-Carlo method is a statistical numerical method used to compute the distribution factors. It models the radiative exchange process by following the life of discrete energy bundles from emission to absorption using the probabilistic interpretation of the surface properties. This method is very useful to treat complex geometries. It can also solve for directional and spectral-surface property variations. As a statistical method

it has the disadvantage of requiring the emission of a large number of energy bundles to converge and thus excessive computing time may be necessary. But once the distribution factors have been obtained, if the temperatures are known for all surfaces and wavelength intervals, calculation of the net heat flux (W/m^2) through surface i is then relatively simple using the following equation,

$$q_{i\,net} = \sum_{k=1}^{K} \sum_{j=1}^{n} \epsilon_{ik} e_b(\Delta_{\lambda_k}, T_j) \, (\delta_{ij} - D_{ijk}) \ , \tag{2.2.2}$$

where ϵ_{ik} is the emissivity of surface i in wavelength interval k, $e_b(\Delta_{\lambda_k}, T_j)$ is the emissive power of surface j in wavelength interval k, D_{ijk} is the distribution factor from surface i to surface j in wavelength interval k, and δ_{ij} is the Kronecker delta function. This formulation succeeds in eliminating the matrix inversion required when using the configuration or exchange factor. As the spatial and spectral resolutions increase, the solution becomes exact.

The Monte-Carlo method was applied in this study to evaluate the radiative heat source on the high-temperature superconductive thermal bridges. This technique was preferred because of its ability to treat complex geometries and to account for mixed specular and diffuse reflection models.

2.3 Minimization Methods Used for the Estimation of Thermal Properties

A key strategy to assess the feasibility of HTS-substrate combinations as electronic leads in infrared sensor satellite systems, is to estimate the thermal conductivities of the

samples in the space environment. An effective technique for the simultaneous estimation of thermal properties consists of the minimization of an objective function. The Gauss Linearization method based on the least squares function is an important method in this field.

2.3.1 The Gauss Linearization Method

The Gauss Linearization method involves the minimization of the sum of squares function S with respect to the unknown parameters. The sum of squares function given by Beck and Arnold (1977) is

$$S = [\boldsymbol{Y} - \boldsymbol{T}(\underline{\beta})]^T \boldsymbol{W} [\boldsymbol{Y} - \boldsymbol{T}(\underline{\beta})] , \tag{2.3.1}$$

where $\boldsymbol{Y}$ is a vector of measured temperatures, $\boldsymbol{T}$ is a vector of calculated temperatures as a function of $\underline{\beta}$, the true parameter vector, and $\boldsymbol{W}$ is a weighting matrix. The thermal properties are found using an iterative process which minimizes the sum of the square of the difference between the measured and the calculated temperatures by updating the thermal property values. First, the derivative of S with respect to $\underline{\beta}$ is set equal to zero. Then, using a Taylor series expansion, this expression is solved for $\boldsymbol{b}$, the estimated parameter vector for $\underline{\beta}$. This process provides a linear approximation to the nonlinear model. It requires that the first derivatives of $\boldsymbol{T}$ are continuous in $\underline{\beta}$ and that the higher derivatives are bounded. Beck and Arnold (1977) describe the Gauss method as being simple and practical for seeking minima which are reasonably well defined provided the initial estimates are in the neighborhood of the minimum. But in the case of poor initial guesses for the parameters or near-linear dependence between the parameter sensitivity

coefficients, oscillations and non-convergence can occur in the iterative process. In order to improve the Gauss estimation method, Box and Kanemasu (1972) suggested a small correction in the direction of the parameter variations. Bard (1970) modified the Box-Kanemasu method by including a check to ensure the continuous decrease of S, the sum of squares function, from one iteration to another. This is done by reducing the step by one-half if the function does not decrease.

The modified Box-Kanemasu method has been applied in a wide range of engineering areas. Scott and Saad (1993) employed the modified Box-Kanemasu method for the estimation of kinetic parameters associated with the curing of epoxy resin. They showed that inaccurate parameter estimations and in some cases non-convergence of the estimation process could result from linear dependence between the sensitivity coefficients. The use of the modified Box-Kanemasu method in cryosurgical applications by Scott and Scott (1993) allowed for the determination of the optimal time for cryogenic tumor treatment. This work also concluded that, when available, prior information should be included in the estimation process as it significantly improves accuracy.

Iterative least squares schemes similar to the modified Box-Kanemasu method exist and some are discussed by Beck and Arnold (1977). Jurkowsky et al. (1992) studied an optimization procedure enabling the simultaneous identification of thermal conductivity and thermal contact resistance without using interior sensors. They concluded that small sensitivity coefficients or the unbalance of the sensitivity matrix resulted in the instability of their estimation procedure.

The careful examination of the sensitivity coefficients therefore appears to be an

imperative step prior to the implementation of parameter estimation methodologies. Indeed, the sum of squares function S has no unique minimum if the sensitivity coefficients are correlated. Furthermore, the magnitudes of the dimensionless sensitivity coefficients are by convention limiting factors in the possibility of estimating parameters because they indicate the influence of each parameter in the mathematical model. In the analysis of sublimation-dehydration within a porous medium, Scott (1994) conducted a sensitivity study which examined the importance of the material properties on the solution. Scott was able to conclude for which parameter temperature provides the most information. The sensitivity study can also be applied for the design of optimal experiments. In this case the maximum magnitude values of the sensitivity coefficients are interpreted as criteria for the determination of optimal experimental parameters. Taktak et al. (1991) and Moncman (1994) employed this technique to design optimal experiments for the estimation of thermal properties of composite materials. In a one-dimensional experiment, they both determined the optimal heating time of the applied heat flux, the optimal temperature sensor location and the optimal experimental time. Moncman also studied these parameters for a two-dimensional experimental design.

The procedure used in this research to estimate the thermal conductivities of the HTS-substrate combinations is a modification of the Gauss Linearization method previously discussed. This method was chosen due to its capability to simultaneously estimate parameters and its accuracy in the final estimates.

CHAPTER 3

Heat Transfer Analysis of the Thermal Bridges

This chapter is devoted to the thermal analysis of different types of electrical leads in infrared sensor satellite systems. These electrical leads act as thermal bridges because they are submitted to a temperature gradient between the data acquisition unit at a temperature of 80 K and the cryogenic infrared detector at a temperature of 4 K. The analysis of the heat transfer in the thermal bridges was realized assuming spaceflight conditions.

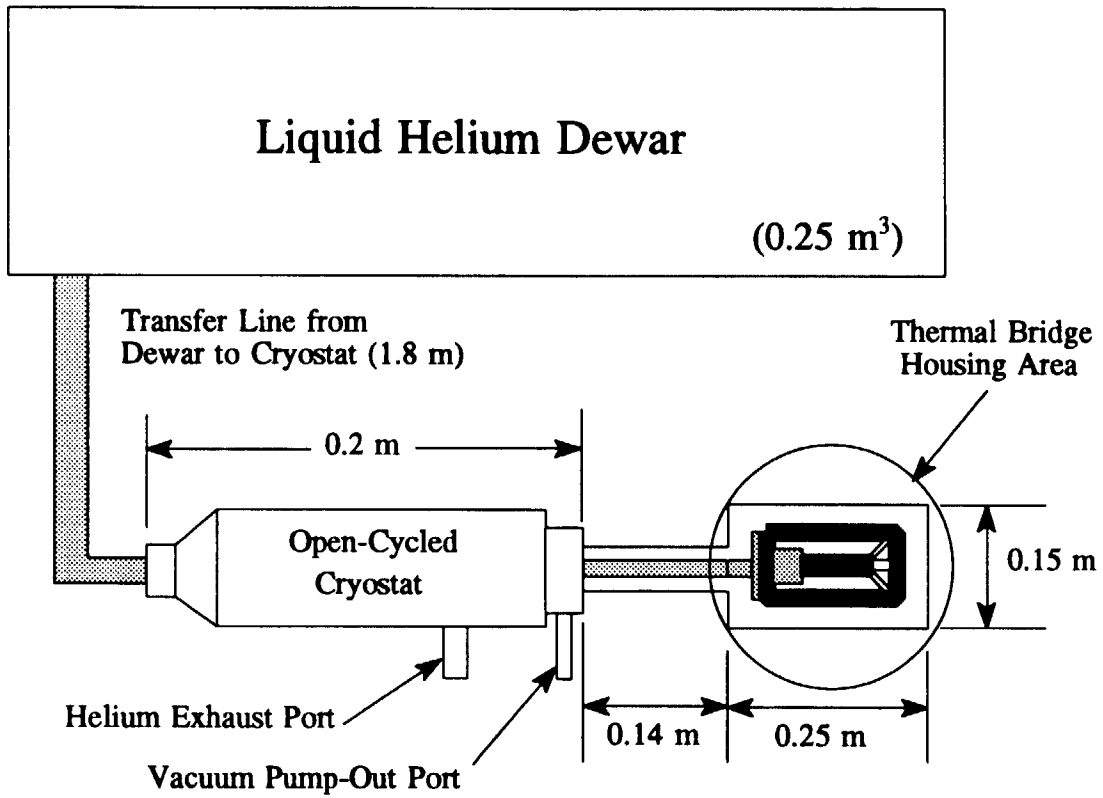
The first section provides a general description of the preliminary experimental design completed by Lee (1994). This design is characterized by identical chambers with only the thermal isolator material being different in each chamber; all sources of thermal transfer other than conduction are minimized. The different HTS-substrate combinations are presented at the end of the first section. The following section focuses on the determination of the conductive heat load supplied to the cryogen by each thermal bridge. As an analytical exact solution could not be found for such a complex conduction problem, a numerical scheme was used. The finite difference program ORTHO3D was

utilized to formulate the conductive mathematical models. In the final section, the assessment of the importance of the radiative heat load on the cryogenic heat load due to the use of HTS thermal bridges is presented. The Monte-Carlo technique was used to perform the analysis of the radiative exchange within the thermal bridge housing area.

3.1 Lee's Experimental Design

The experimental design completed by Lee (1994), shown in Figure 3.1.1, is comprised of three major components, which are the liquid helium dewar, the cryostat and the thermal bridge housing area. The cross-sectional view of the thermal bridge housing area (Figure 3.1.2) shows the three identical vacuum chambers containing three different types of electronic leads. Two chambers will house high-temperature superconductive leads and the third will contain manganin wires. The vacuum environment of 6.8×10^{-6} atm is produced to minimize convection. A steady heat flux, controlled by a heater, is applied at one end of the thermal bridges so that the temperature reaches approximately 80 K at this end (Figure 3.1.3). The other end is attached to a copper block in direct contact with a cryogenically cooled disk at an approximate temperature of 4 K. This temperature is maintained by a temperature controller. The heat loss due to each type of thermal isolator can be calculated by a simple conduction analysis along the copper blocks which contain the cold ends of the thermal isolators and the cold tip disk. The temperature at both ends of the thermal bridges are measured by temperature sensors.

The constraints of the spaceflight mission were taken into account. This preliminary



Note: Drawing is not to scale.
Dimensions are preliminary.

Figure 3.1.1. Preliminary Thermal Bridge Experimental Design Completed by Lee (1994).

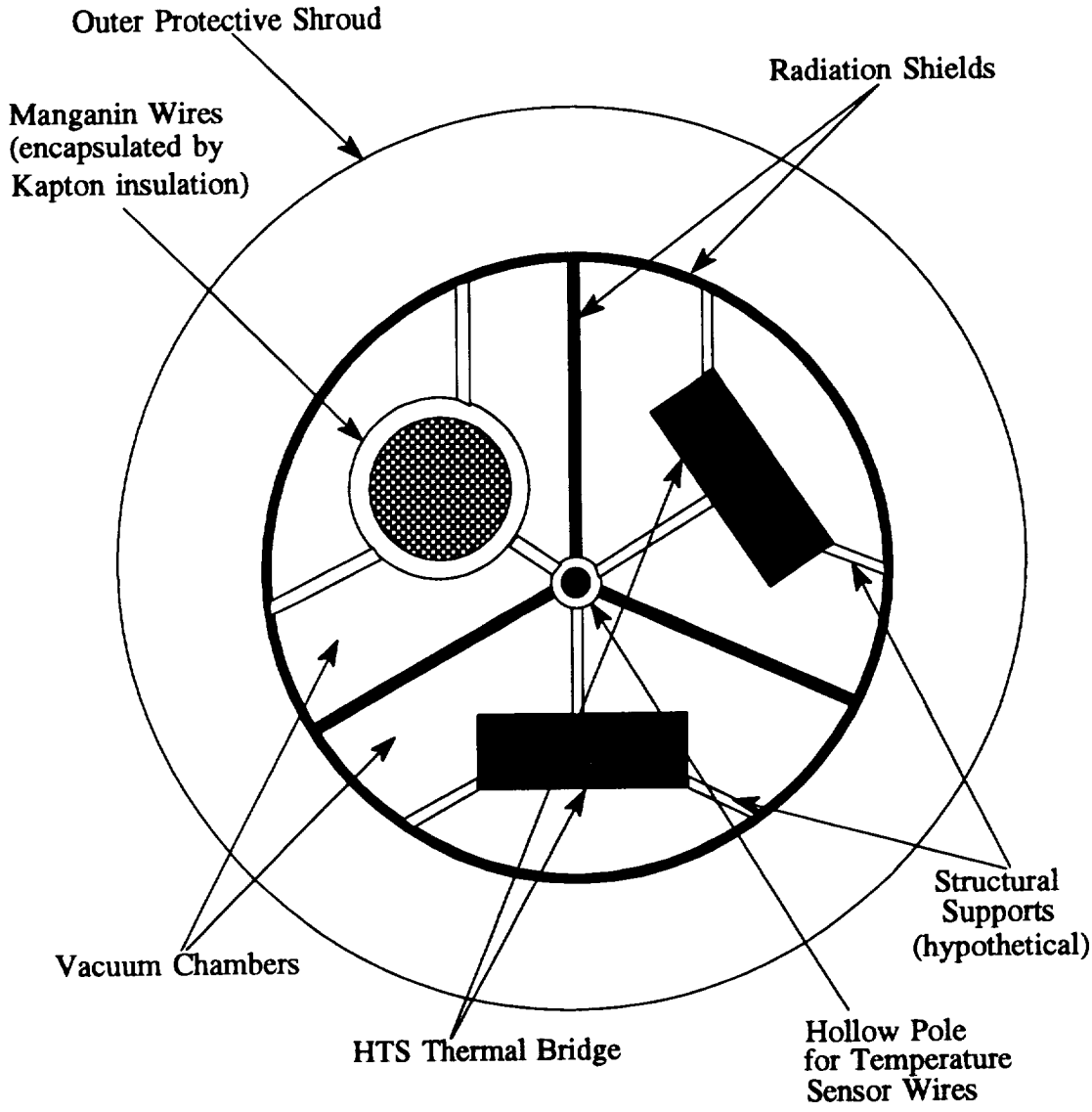


Figure 3.1.2. Cross-Sectional View of Thermal Bridge Housing Area.

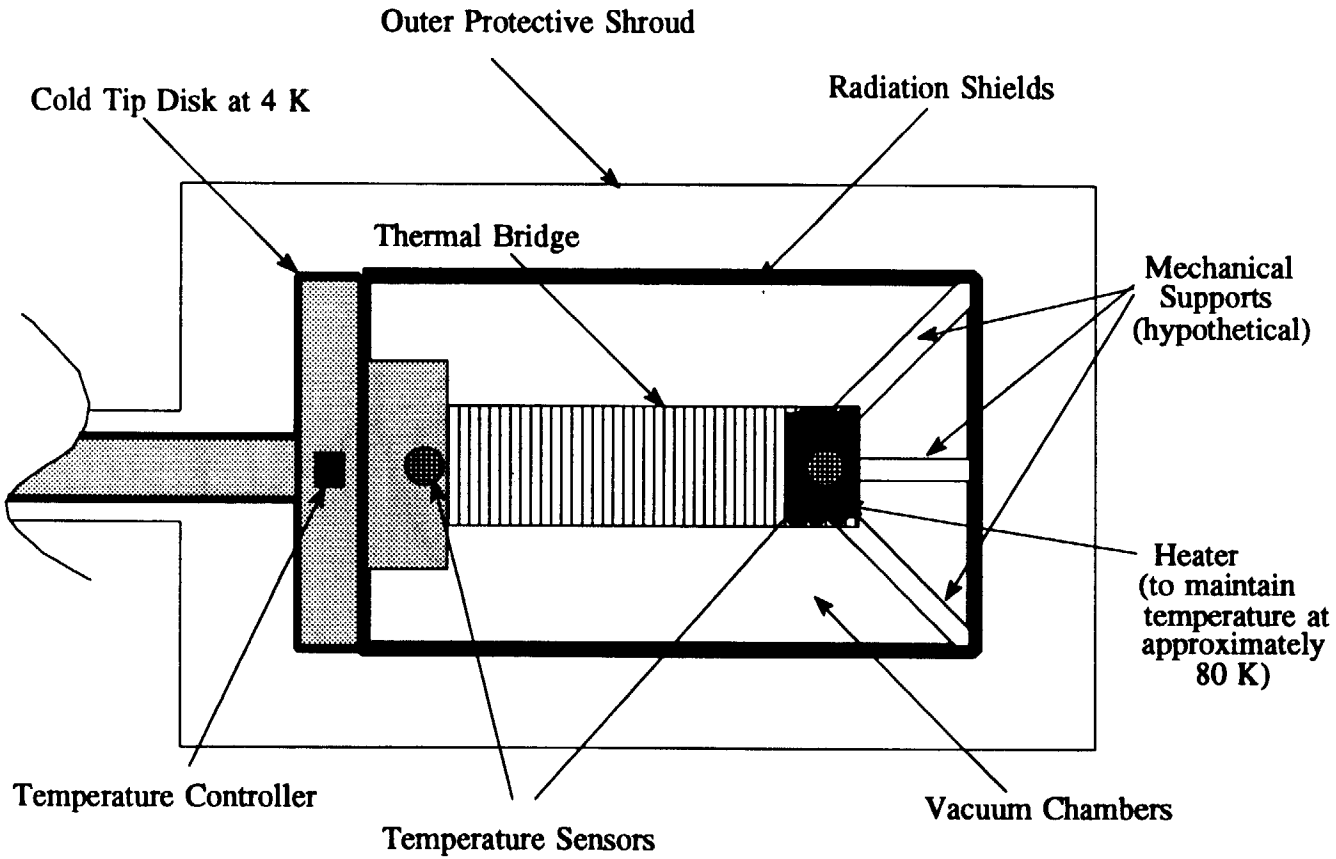
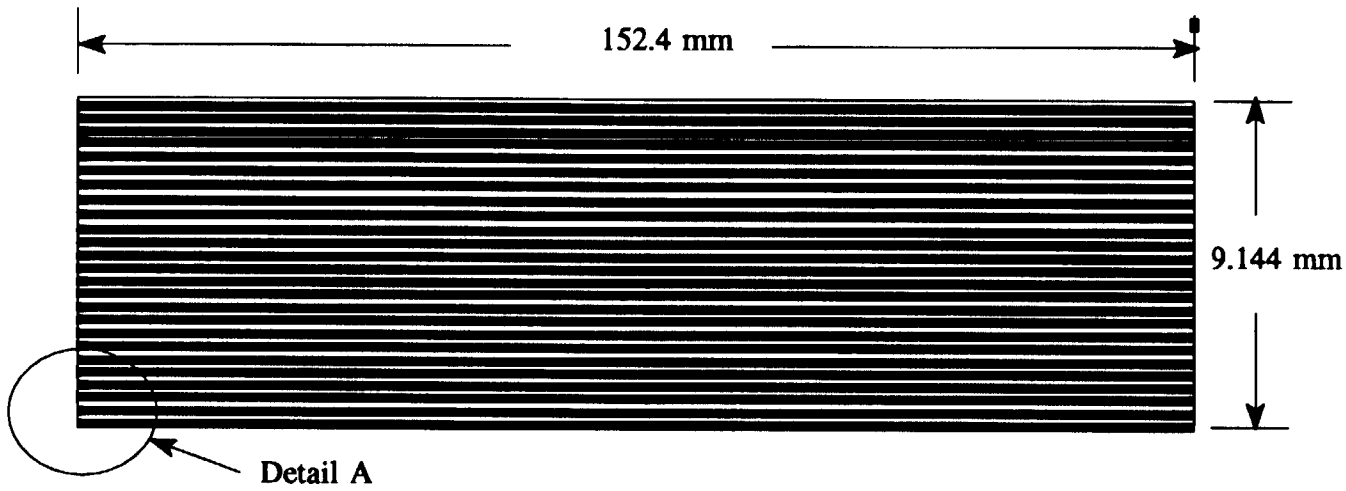


Figure 3.1.3. Detailed View of the Thermal Bridge Housing Area.

design respects the size, weight, power and budget limitations required. It will eventually need to meet the spaceflight launch conditions. The HTS thermal bridges must actually be supported to withstand the vibrational loads of 12.6 g rms associated with the launch. Following Spencer's work (1994), Ron Nottingham, an undergraduate student in the Mechanical Engineering Department at Virginia Tech, is presently studying several support mechanisms. These are designed to strengthen the HTS thermal bridges during launch while the bridges are expected to vibrate, but to have no contact with them in space, where they are presumed to be stable. As a result, in space, the heat transfer through the supports should not affect the heat transfer through the HTS thermal bridges.

Five possible HTS-substrate combinations are evaluated in this research. These are $YBa_2Cu_3O_{7-x}$ lines on Yttria-Stabilized Zirconia (YSZ, 10wt%, cubic); $BiSrCaCu_2O_x$ lines on YSZ; $YBa_2Cu_3O_{7-x}$ lines on Fused SIlica (FSI, with 3000 Å buffer layer of zirconia); $BiSrCaCu_2O_x$ lines on FSI (3000 Å buffer layer of zirconia); and $YBa_2Cu_3O_{7-x}$ lines on 211 Green Phase (GREEN). Only two of these five combinations will be selected for the final experimental design.

The detector leads must be able to transmit typical signal of 1 μA or less. Each thermal bridge contains a minimum of sixty detector leads. In the case of manganin, these are 40 AWG wires encased in a thin layer of Kapton for insulation (about 0.0254 mm thickness). In the case of HTS materials, there are sixty HTS leads printed onto a 152.4 mm long by 9.144 mm wide by 0.1524 mm thick substrate, as shown by Figure 3.1.4. Each HTS lead is 0.0508 mm wide by 0.0508 mm thick. The spacing between the leads is 0.1016 mm. The details of the HTS dimensions are provided in Figure 3.1.5.



Legend: High Temperature Superconducting Leads

Substrate Material

Notes: Not to Scale.
60 HTS leads are printed on a substrate.

Figure 3.1.4. High Temperature Superconducting Leads Printed on Substrate (Thermal Bridge).

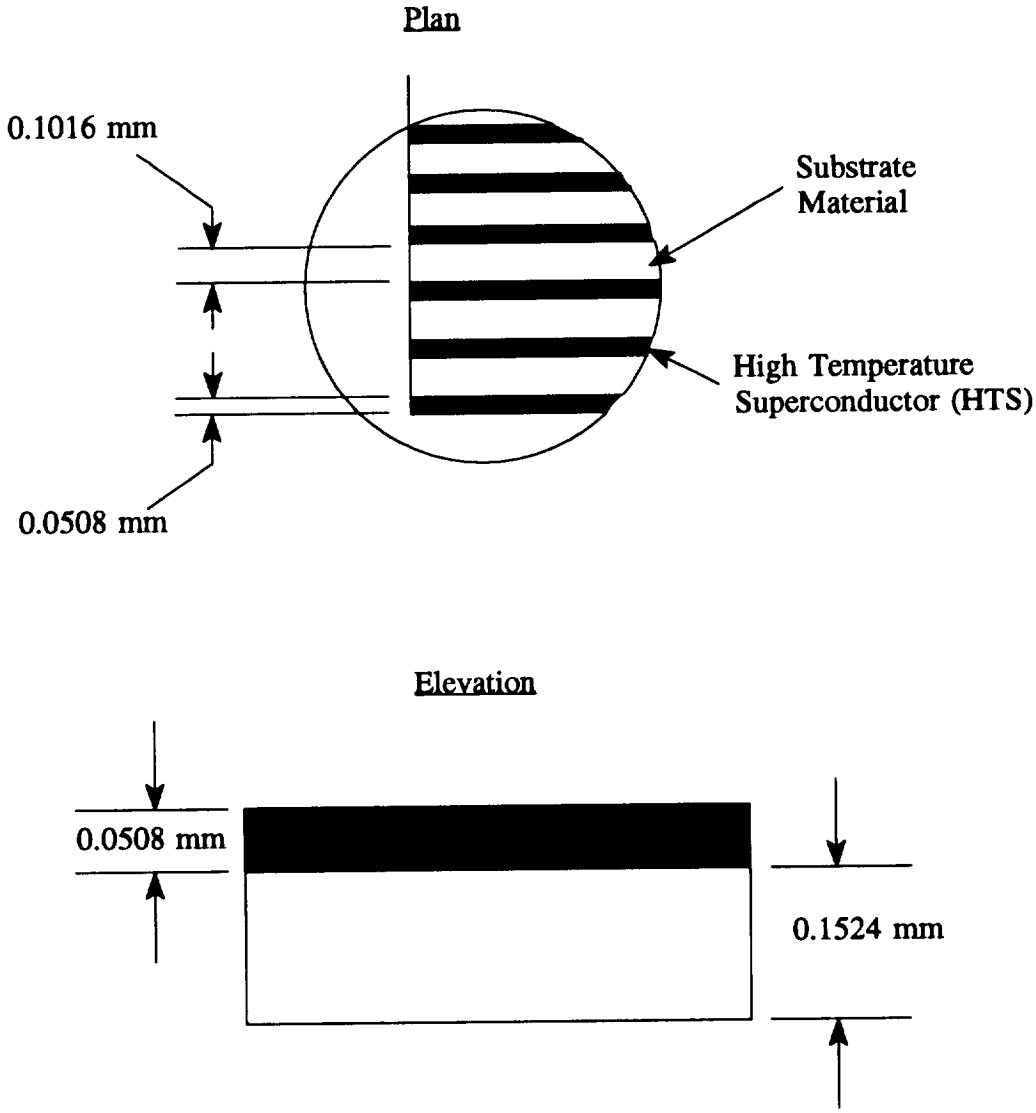


Figure 3.1.5. Detail A of HTS Leads on Substrate (Thermal Bridge).

3.2 Determination of the Conductive Heat Load on the Cryogen

The goal of the following analysis is to determine the conductive heat load supplied to the cryogen by each thermal bridge. This goal enables the quantitative comparison of the performance of HTS materials and manganin wires for use as electronic leads to cryogenic sensors in the space environment. The control-volume-based finite difference program ORTHO3D (Creel and Nelson, 1994) was utilized to describe the conduction in the thermal bridges. Each thermal bridge was considered separately in the conductive mathematical models. The models were formulated assuming that all other heat transfer (radiation, convection) are negligible.

The first subsection discusses the main characteristics of the control-volume-based finite difference method and introduces the finite difference program ORTHO3D. The second subsection details the geometric modeling for both the HTS and manganin thermal bridges. In the next subsection, the theoretical development for the incorporation of the electrical current in the manganin model is presented. The final subsection focuses on the convergence of the conductive mathematical models and provides the results, which are the temperature distributions along the thermal bridges and the heat load on the cryogen for the HTS and manganin models.

3.2.1 Control-Volume-Based Finite Difference Program ORTHO3D

A practical alternative procedure for solving multidimensional situations is to use a numerical method. The control-volume-based finite difference method is one of the more

popular numerical methods used in heat transfer problems. This method involves the discretization of the domain of interest into small volumes. Each control volume is represented by a grid point in its center. In the control-volume method, the properties attached to a grid point are assumed to prevail over the entire corresponding control volume. This assumption allows for the approximation of the differential equation governing the heat transfer problem to a set of discretized algebraic equations. This set of equations can then be solved, providing an estimate of the exact solution. In the limit of spatial refinement, the numerical solution becomes exact.

The program ORTHO3D, used in this research to analyze the conduction in the thermal bridges, is a control-volume, finite difference heat transfer program. It has been developed by Dr. D.J. Nelson, professor of Mechanical Engineering at Virginia Tech. ORTHO3D is based on an extension of the program CONDUCT created by Patankar (1991). The modifications performed on the original program allow for the study of three-dimensional materials with orthotropic properties. ORTHO3D also has the capability to add a contact resistance between layers in the z direction. Creel and Nelson (1994) used ORTHO3D to approximate a layer by a contact resistance in the thermal model of a three dimensional microelectronic package. It should be noted that the pre-packaged finite difference modeling programs PATRAN (1990) and SINDA (1985) were initially used in this research to study the conductive heat transfer in the thermal bridges. However, the limitation of the computer program SINDA, specifically its inability to converge to the correct temperature distribution, created the need to use ORTHO3D. This finite difference program performed efficiently in this investigation.

The structure of the original program CONDUCT, which is comprised of two main parts, has been maintained in ORTHO3D. While the *invariant* part, which solves the set of algebraic discretization equations, is common to all applications, the *adapt* part is a problem-dependent part and has to be modified by the user for each application. This original structure makes the program very easy to use. The program ORTHO3D and its capabilities have been described in detail by Creel (1994).

The program ORTHO3D has the ability to solve a three-dimensional conductive heat transfer occurring within a domain which can be defined by rectangular coordinates. The governing equation of such problem is

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}[k_x \frac{\partial T}{\partial x}] + \frac{\partial}{\partial y}[k_y \frac{\partial T}{\partial y}] + \frac{\partial}{\partial z}[k_z \frac{\partial T}{\partial z}] + S_{gen} , \qquad (3.2.1)$$

where ρ is the density, c is the specific heat, k_x, k_y and k_z are the conductivities in the x, y and z directions, respectively, T is the temperature and S_{gen} is the volumetric heat source in the domain. In this study, ORTHO3D is employed to analyze the heat conduction through the thermal bridges, assuming that all other heat transfer (radiation, conduction) are negligible. This conduction problem is steady state and the materials constituting the thermal bridges are assumed to be isotropic, that is $k_x=k_y=k_z=k_i$ for each material i. Therefore, the differential equation that needs to be solved is

$$\frac{\partial}{\partial x}[k_i \frac{\partial T}{\partial x}] + \frac{\partial}{\partial y}[k_i \frac{\partial T}{\partial y}] + \frac{\partial}{\partial z}[k_i \frac{\partial T}{\partial z}] + S_{gen} = 0 . \qquad (3.2.2)$$

This is a nonlinear problem because k depends on temperature. The volumetric heat source, S_{gen}, is detailed later in Section 3.2.3.

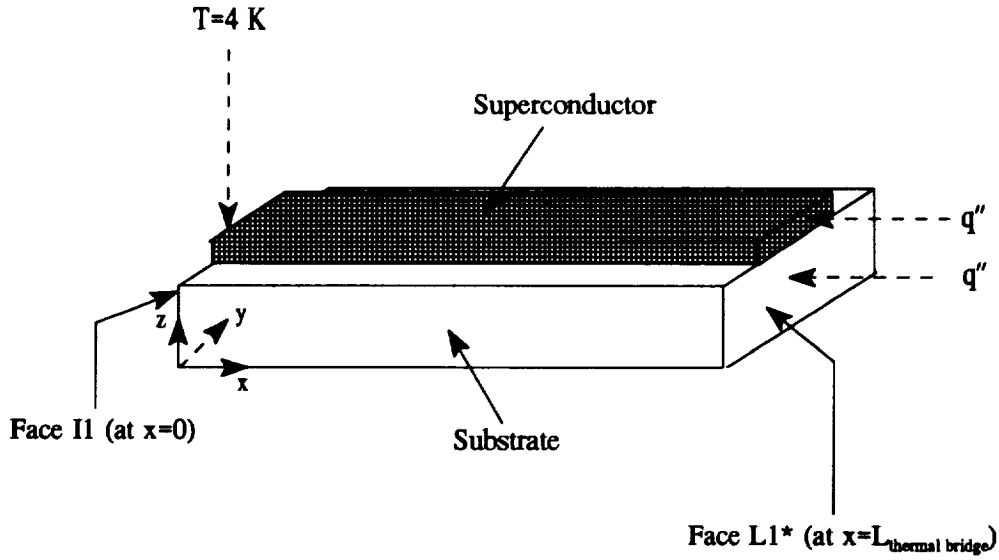
Before developing the boundary conditions that are applied in Eq. (3.2.2), the geometric models of the thermal bridges need to be defined.

3.2.2 Thermal Bridge Geometric Modeling

The control volume design was performed using the scheme called *Practice B* (Patankar, 1991). First, the domain was divided into control volumes, and then grid points were placed at the center of each control volume. This practice ensures the coincidence of the location of discontinuity in conductivity or heat generation with a control volume face. The grid locations in the x, y and z directions are denoted by i, j and k, respectively. The node numbering scheme ranges from 1 to L1, M1 and N1 for the x, y and z directions, respectively.

3.2.2.1 HTS Thermal Bridges

The geometric modeling of the HTS thermal bridges with ORTHO3D is similar to the one realized by Lee (1994) with the software PATRAN. The individual leads of the HTS material are lumped as a single lead on the substrate (Figure 3.2.1). Due to the symmetry in the y direction, only one-half of the HTS thermal bridges need to be modeled. Because of the non-regularity of the geometric model, a nominal rectangular domain is first drawn around it, as shown by Figure 3.2.1. Then this nominal domain is discretized into active and inactive zones, which are divided into control volumes. The inactive zones lie outside the real domain and no solution is sought there. The thermal conductivity is hence set to zero in the inactive region.



* Detail of Face L1
Half of the thermal bridge is discretized in zones

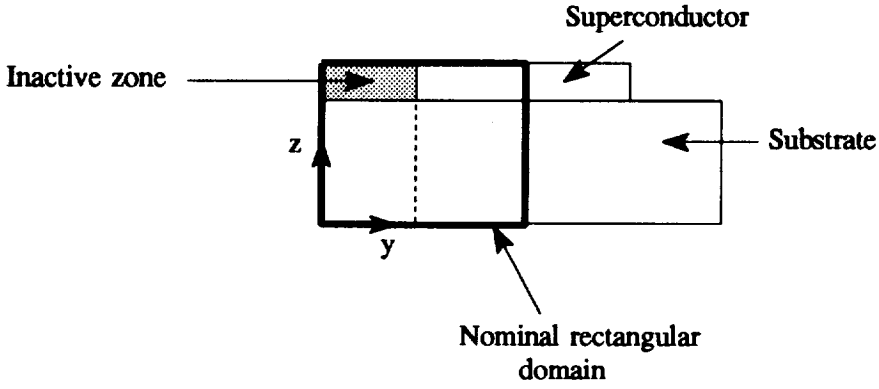
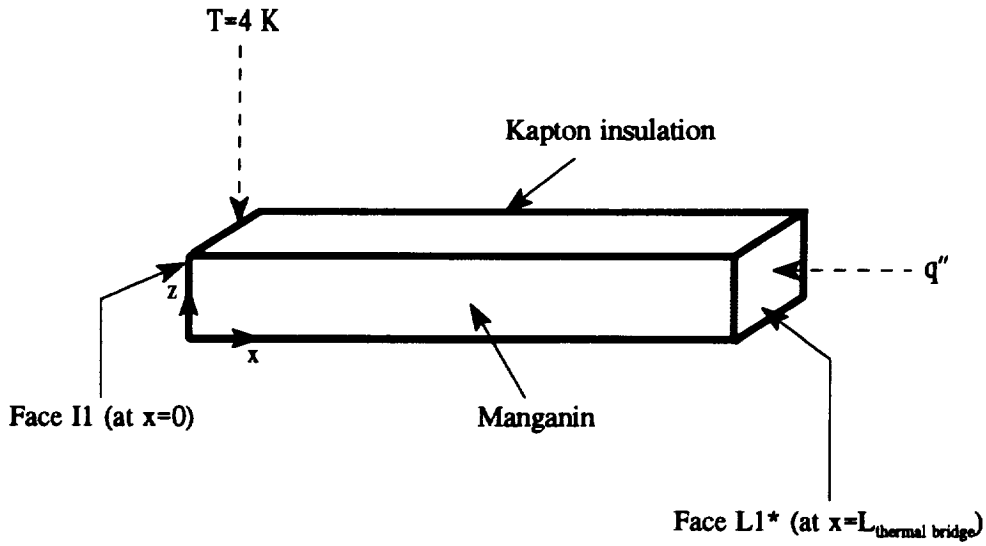


Figure 3.2.1. HTS Thermal Bridge Geometric Modeling Using ORTHO3D.

The nominal domain is discretized into one zone in the x direction, two zones in the y direction and two zones, or three, depending on the presence of a buffer layer between the substrate and the HTS leads, in the z direction. Figure 3.2.1 provides the details of face L1 (i=L1, $x=L_{thermal\ bridge}$) and displays the zone discretization in the y and z directions. This figure also supplies a schematic of the boundary conditions. Face I1 (i=1, x=0) has a fixed temperature of 4 K and face L1 is submitted to a constant heat flux so that the temperature on this face reaches 80 K. This constant heat flux was calculated for each HTS-substrate combination using a one-dimensional parallel flow assumption (Lee, 1994). The four other surfaces of the nominal rectangular domain are insulated.

3.2.2.2 Manganin Thermal Bridge

The manganin wires were also modeled as a single wire (Figure 3.2.2). However, in order to use the program ORTHO3D developed in rectangular coordinates, this single wire was modeled to be rectangular, with same cross-sectional area. Taking advantage of the symmetry, only a fourth of the manganin thermal bridge was studied. As for the HTS thermal bridges, the nominal domain is discretized into zones. One zone is set in the x direction and two zones are specified in the y and z directions. The details of face L1 are shown by Figure 3.2.2. The boundary conditions are analogous to those for the HTS thermal bridges.



* Detail of Face L1

A fourth of the thermal bridge is discretized in control volumes

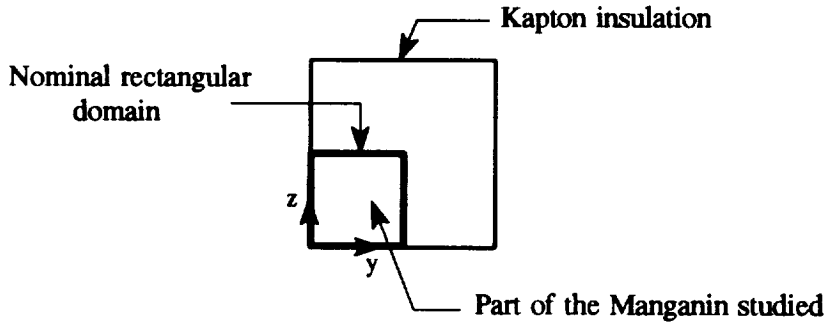


Figure 3.2.2. Manganin Thermal Bridge Geometric Modeling Using ORTHO3D.

3.2.3 Incorporation of the Electrical Current in the Manganin Wires

The IR detectors transmit typical signals of 1 μA or less. The electrical power ($P_{elect} = R_{elect}I^2$) of the electrical current, I, flowing through the thermal bridge which has an electrical resistance R_{elect}, generates heat along its length. This electrical heat source could be a significant source of heat load on the cryogen. Because the HTS materials do not have electrical resistance at cryogenic temperatures, only the manganin wires can be affected by the electronic signals. Therefore, the electrical conduction was implemented into the conductive mathematical model for the manganin wire in order to obtain the effective conductive heat transfer through the wire. The incorporation of the electrical conduction in the governing heat conduction equation, Eq. (3.2.2), is realized via the term S_{gen}, which represents the volumetric heat generated in the domain.

The electronic signals create in the manganin wires a volumetric heat source, S_{gen}, defined as the electrical power dissipated (W) over the volume (m^3), and is expressed as

$$S_{gen} = \frac{R_{elect}I^2}{A_c L} , \tag{3.2.3}$$

where R_{elect} is the electrical resistance of manganin, I is the electrical current (1 μA) and A_c and L are the cross-sectional area and length of the manganin link, respectively. Introducing the current density J and the electrical resistivity ρ_{elect} reduces the Joule heating term to

$$S_{gen} = J^2 \rho_{elect} . \tag{3.2.4}$$

The current density through the manganin wires, J, can be calculated because the

electrical current, I, and the cross-sectional area, A_c, are both known. The electrical resistivity of manganin, ρ_{elect}, is a function of temperature. Its variation needs to be determined for the temperature range of the thermal bridge.

The literature (Standard Handbook for Electrical Engineers, 13th ed., 1989) provides the electrical resistivity, $\rho_{elect,20}$, and the temperature coefficient per °C, $\alpha_{elect,20}$, for the manganin material at 20°C. Over moderate ranges of temperatures, such as 100°C, the change of resistivity is usually proportional to the change of temperature, that is

$$\rho_{elect,T_2} = \rho_{elect,T_1}[1 + \alpha_{elect,T_1}(T_2 - T_1)] , \qquad (3.2.5)$$

where $\rho_{elect,T2}$ and $\rho_{elect,T1}$ are the electrical resistivities at temperatures T_2 and T_1 (usually T_1=20°C), respectively, and $\alpha_{elect,T1}$ is the temperature coefficient at temperature T_1. However, over wide ranges of temperatures, the linear relationship of this formula is not applicable. Consequently, knowing $\rho_{elect,20}$ cannot be used to determine the electrical resistivity for temperatures less than -80°C (193 K).

Let us look now at the general behavior of the electrical resistivity of manganin alloys. This electrical behavior has been shown to decrease with temperature for temperatures lower than 20°C (Metal Handbook, 9th ed., 1980). Therefore a conservative value for the resistivity at a temperature range of [4-80 K] could be the resistivity at -80°C (193 K). This resistivity of manganin at -80°C, $\rho_{elect,-80}$, is calculated using $\rho_{elect,20}$ and $\alpha_{elect,20}$ in Eq. (3.2.5), with T_1=20°C and T_2=-80°C. We obtain

$$\rho_{elect,-80} = 4.814 \times 10^{-7} \ \Omega\text{-m} .$$

The volumetric electrical source for the manganin thermal bridge, S_{gen}, can then be determined using Eq. (3.2.4),

$$S_{gen} = 5.29\text{x}10^{-6}\ \text{W/m}^3\ .$$

This Joule heating is assumed constant within the manganin wires, that is S_{gen} is a constant at each grid point. Recall that for the HTS thermal bridges, S_{gen} equals zero.

It is important to note that, a priori, the resistance self-heating through the manganin wires should affect neither the heat load on the cryogen, nor the temperature distribution along the thermal bridge. Indeed, with a volume of $4.6\text{x}10^{-8}$ m^3 for the 152.4 mm long manganin thermal bridge, the Joule heating term is about 10^{-13} W for an applied current of 1 μA. The comparison of this generated heat source with the heat input of 10^{-3} W (Lee, 1994) at the warm end of this 152.4 mm long manganin thermal bridge makes the Joule heating term insignificant. This result was verified numerically using ORTHO3D.

3.2.4 Results and Discussion

The Fortran subroutines HTS.FOR (Appendix A) and MANG.FOR (Appendix B) were written as the *adapt* subroutines of ORTHO3D to solve this steady-state conductive heat transfer problem for the five combinations of HTS thermal bridges and for the manganin thermal bridge. In these subroutines, the physical parameters of the models such as the geometry, the boundary conditions, the conductivities and the volumetric heat source term for the manganin wires, are defined. The results are the temperature distributions within the thermal bridges and the conductive heat load on the cryogen by each thermal bridge.

The convergence of the conductive mathematical models is discussed in the first subsection. The complete results, obtained using ORTHO3D without the consideration of the Joule heating in the manganin wires, have actually already been given and largely discussed by Scott and Lee (1994). Therefore, only the results which have been shown to minimize the cryogen evaporation, are provided with brief comments in the final subsection.

3.2.4.1 Convergence of the Conductive Mathematical Models

Several parameters were studied in the convergence analysis of the models. These are the grid size, the initial temperature distribution along the thermal bridges, the energy balance of the domain and the number of iterations used by the solver.

The grid refinement of the domain examined has to be performed cautiously. Indeed, the aim is to obtain a good numerical accuracy with the fewest grid points possible. This allows for the saving of computational resources. The grid size should be homogeneous in the three directions; that is the length, width and thickness of each control volume should be in the same range of dimensions. In this research, the thermal bridges are characterized by a large aspect ratio, which is defined by the ratio of the length of the thermal bridge over its thickness. For instance, the value of the aspect ratio of the 152.4 mm long HTS thermal bridges is one thousand when calculated with respect to the thickness of the substrate. This large value shows the importance of the resolution in the x direction in order to obtain a reasonable aspect ratio for each control volume and hence an homogeneous grid size. The grid was tested by running the HTS models with a

different resolution in the x direction and with the same resolution in the y and z directions (7×5 or 6 when the models contain a buffer layer). It was found that one hundred control volumes in the x direction provided trade-off between the numerical accuracy obtained and the computing time required for this grid size. The numerical accuracy was considered reasonable when both the temperature at the warm end of the thermal bridges reached 80 K ($\pm$ 0.2 K) and the temperature gradient in the z direction was less than 0.2 K. The aspect ratio of the control volumes in the substrate becomes 30 for one hundred control volumes in the x direction.

The grid was refined to $100 \times 7 \times 5$ (or 6 in the case of a buffer layer between the substrate and the superconductor) for the HTS thermal bridges; for the manganin thermal bridge, the grid size used is $100 \times 4 \times 4$.

Another prominent parameter in the convergence of the models is the selection of accurate initial guessed temperatures. In this research, this selection is important not only because of the nonlinearity of the conduction problem but also because of the extremely low thermal conductivity values of the materials at cryogenic temperatures (Lee, 1994). To provide the most accurate initial temperatures along the thermal bridges, the temperature distribution of the combination YBCO/FSI, previously obtained using the softwares PATRAN and SINDA (Lee, 1994), was implemented in each model as a function of the length of the thermal bridge. This initial temperature distribution provided a good point of departure for the calculation by the solver of the material thermal conductivities at each grid point. This also had the advantage of improving the

convergence speed of the models.

It is strongly advised by the user in the *adapt* subroutine (Nelson, 1994) to implement a convergence criterion specific to the problem studied. The convergence criterion chosen in this work is based on the change in the conductive heat load transferred out of the domain. When the change in this heat output becomes negligible (less than 10^{-5}), the solution is considered to be converged. At this point, the computation of a second criterion based on an overall energy balance could be appreciated as a double-check of convergence. An overall energy balance, E_{bal}, is defined by

$$E_{bal} = Q_{in} - Q_{out} + Q_{gen} \ , \tag{3.2.6}$$

where Q_{in} and Q_{out} are the heat transferred in and out of the domain and Q_{gen} is the heat generated in the domain, as shown by Figure 3.2.3. A useful double-check is to compute the absolute value of the ratio E_{bal} over Q_{in}. When convergence is reached with respect to the first convergence criterion, the value of $|E_{bal}/Q_{in}|$ should be very small. Numerically, $|E_{bal}/Q_{in}|$ was about 10^{-5} for the HTS thermal bridges and about 10^{-10} for the manganin thermal bridge after an average of twelve iterations for each model. These values show that convergence has been reached for each model.

The last parameter of interest in the convergence of the models is the number of inner iterations, NTC, used by the solver. The inner iterations in the program have to be distinguished from the outer iterations. The inner iterations are applied to solve the algebraic equations for a specific temperature distribution, whereas the outer iterations are applied to update the temperature distribution. For the first outer iterations, NTC reached generally NTIMES which is the maximum number of inner iterations allowed in the

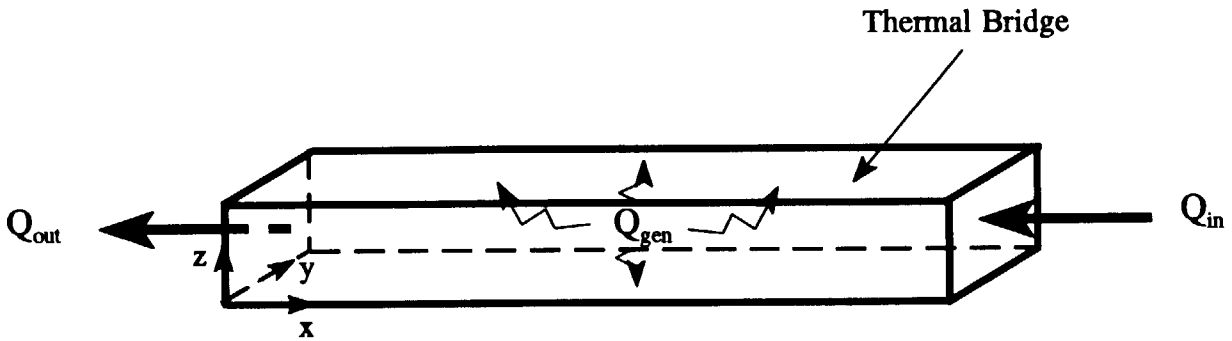


Figure 3.2.3. Overall Energy Balance Scheme in a Thermal Bridge.

solver. This behavior is normal for nonlinear problems. However, as more outer iterations were performed and convergence was approached, the number of inner iterations became less than NTIMES. This decrease in the NTC values indicates that the models are not only converged but are also properly formulated.

3.2.4.2 Temperature Distributions and Conductive Heat Load on the Cryogen

As mentioned earlier, the results for the temperature distributions and conductive heat load on the cryogen, without the consideration of the Joule heating in the manganin wires, have been provided and discussed in detail by Scott and Lee (1994). Although the manganin mathematical model, presented in Section 3.2.3, includes a Joule heating term, the results for manganin were not affected by this volumetric heat source, as expected. Recall that the current for the envisioned application is quite low (1 μA) and develops an insignificant heat source. This negligible Joule heating source has been previously observed by Caton and Selim (1992) for the same experimental conditions as those of this research. The results presented by Scott and Lee for the manganin wires can therefore be considered to represent the effective conductive heat transfer through the wires.

Lee (1994) studied the effect of both the length of the thermal bridges (101.6 mm and 152.4 mm) and the input heat flux at the warm end on the temperature distributions and on the conductive heat load on the cryogen. Lee showed that longer thermal bridges and lower input heat fluxes generate lower heat loads, as one would expect. A general result of Lee's study is that the substrate material is the largest contributor to the heat transfer through the HTS-substrate combination.

The heat load on the cryogen was shown to be minimum for the 152.4 mm long thermal bridges. The results obtained in this case (Scott and Lee, 1994) are presented next. Figure 3.2.4 shows the temperature distribution for each conductive model. The temperature distribution is a function of the length of the thermal bridge (x direction) but is independent of the y and z directions of the model. This result validates the one-dimensional parallel flow assumption made in the calculation of the input heat flux at the warm end (Lee, 1994). The temperature distribution is required for the estimation of the thermal conductivities; this is developed in Chapter 4.

Table 3.2.1 summarizes for each model the heat load on the cryogen, the percentage of heat load displayed and the extended life expected for the satellite on a five-year mission. One can see that the heat load on the cryogen by four of the five HTS models represents less than fifteen percent of the total heat load on the cryogen, while the manganin model produces a heat load of about twenty percent. These particular HTS

Table 3.2.1. Heat Load on the Cryogen, Percentage of Heat Load Displayed and Extended life on a Five-year Mission for the Different Types of 152.4 mm Long Thermal Bridges (Scott and Lee, 1994).

	Manganin	YBCO/ YSZ	BSCCO/ YSZ	YBCO/ GREEN	YBCO/ FSI	BSCCO/ FSI
Heat Load on the Cryogen (W)	1.00E-3	7.48E-4	5.89E-4	3.74E-3	4.03E-4	2.44E-4
Percentage of Heat load (%)	20.0	14.9	11.7	74.7	8.1	4.9
Extended Life on a 5-year Mission (months)	0.0	+3.0	+4.9	-32.8	+7.2	+9.1

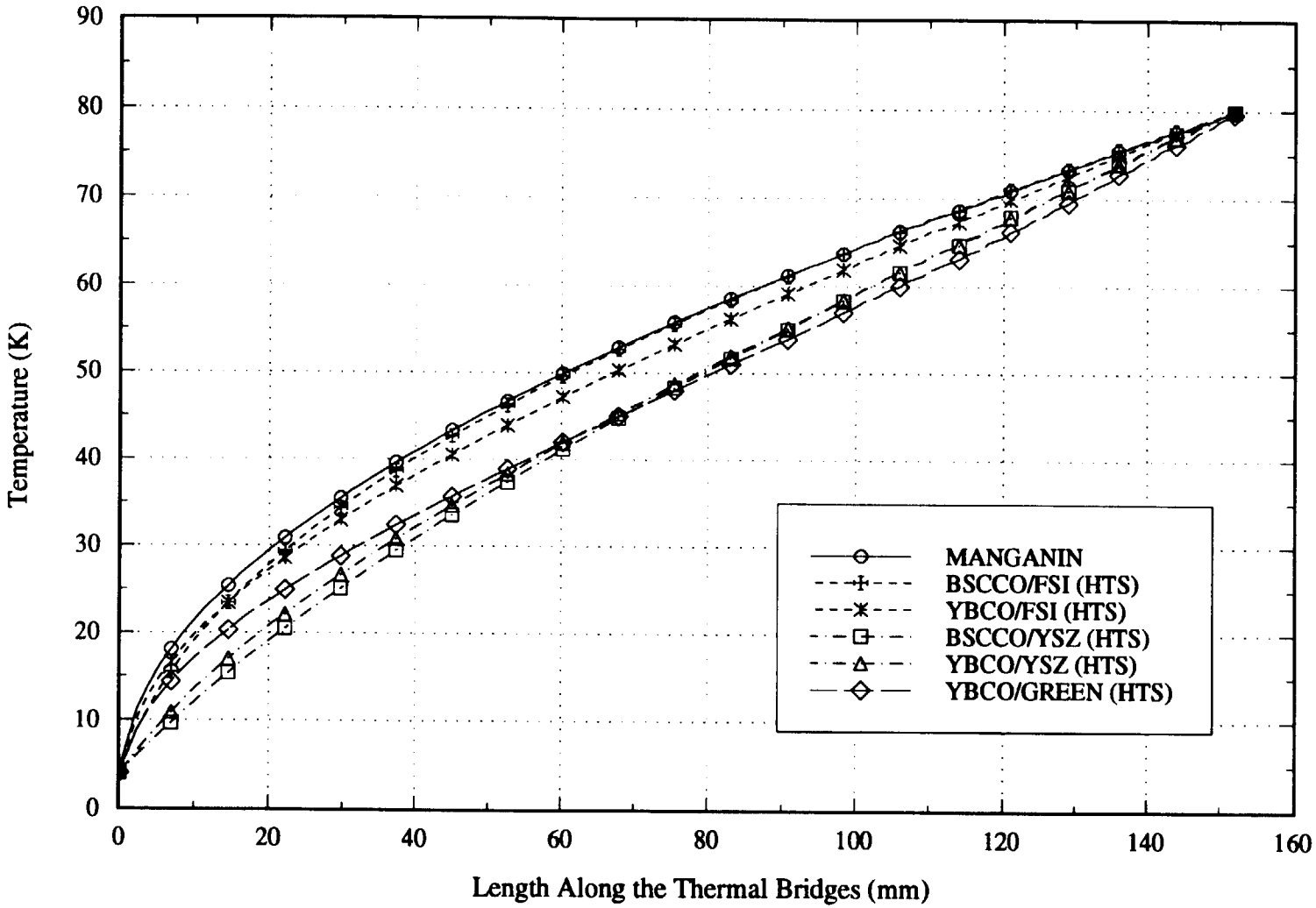


Figure 3.2.4. Temperature Distributions of 152.4 mm Long Thermal Bridges (Scott and Lee, 1994).

thermal bridges are therefore potential replacements for the manganin wires as electronic leads in cryogenic infrared sensor satellites.

3.3 Evaluation of the Radiative Heat Source on the HTS Thermal Bridges

Let us now examine the heat transfer assumptions made in the conduction analysis. In Lee's experimental design, described in Section 3.1, convection and solar radiation can both be neglected because of the vacuum environment of 6.8 atm and the protective shielding on the spacecraft, respectively. But no reliable assumption can be made about the radiative exchange within the thermal bridge housing area. In reality, radiation could affect the conductive heat load on the cryogen calculated for the high-temperature superconductor thermal bridges. The goal of the following investigation is to conduct a radiant interchange analysis within a high-temperature superconductor housing area and to determine if neglecting radiation in the conduction analysis is a valid assumption.

The first subsection focuses on describing the radiation problem. The next subsection is devoted to the theoretical developments used in a Monte-Carlo analysis to determine the distribution factors in the enclosure, and the radiative heat fluxes. Recall that the distribution factors reveal the radiative exchange between the surfaces within the enclosure. The results are provided and discussed in the final subsection. The distribution factors and the radiative heat fluxes were computed using the programs MC.FOR (Appendix C) and TQ.FOR (Appendix D), respectively.

3.3.1 Problem Description

To study the radiant interchange within an HTS isolator housing area, a specific enclosure was defined and divided into *n* surfaces. Either the temperature distribution or the heat flux distribution was specified on each surface. The enclosure is described in detail in the first part of this subsection. Discussed next is the estimation of the radiative properties, the emissivity, absorptivity and reflectivity ratio, of the materials in the enclosure. The surfaces were assumed to be gray and diffuse emitters which allowed the emissivity and absorptivity to be equal. The emissivities were estimated using the electromagnetic theory applied to radiative-property estimation. Because no procedure was found to predict the reflectivity ratios, these were evaluated based on reasonable estimated values.

3.3.1.1 Description of the Enclosure

Since the thermal housing area (see Figure 3.1.2) is symmetric with respect to the three separate chambers, only a single housing chamber needed to be analyzed. It has the shape of the third of a cylinder, as shown by Figure 3.3.1. The housing wall material is pure copper (copper OFHC). The thermal bridge was approximated as the substrate material alone. This simplification is acceptable geometrically because the HTS material printed on the substrate is very thin. The three different substrates, Fused Silica (FSI), Yttrium Stabilized Zirconia (YSZ) and Green Phase (GREEN), studied in the conduction analysis, were used. The surface numbering scheme is presented in Figure 3.3.2. The single housing chamber was divided into five surfaces: surfaces 1 and 2 are the surfaces

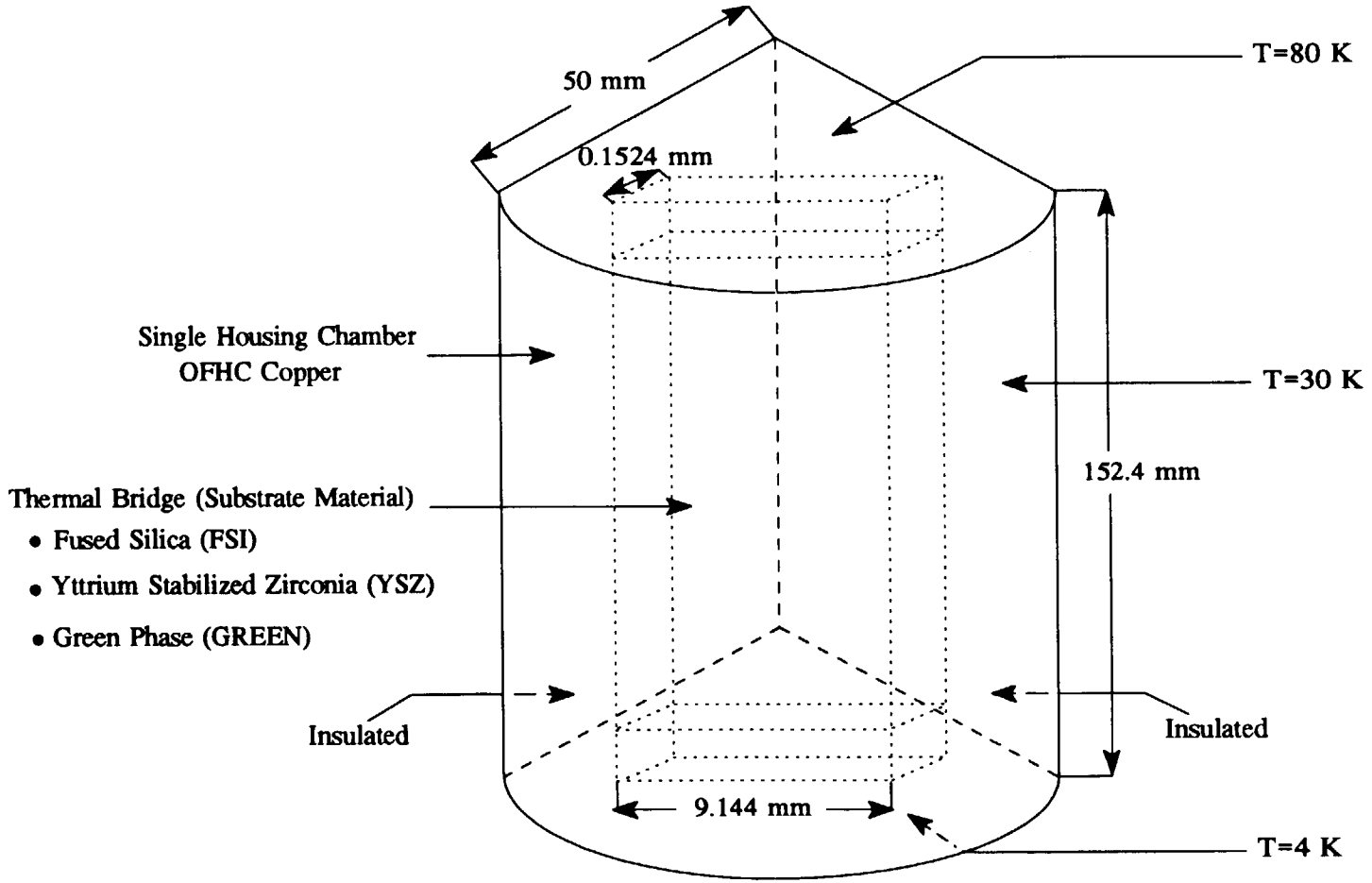
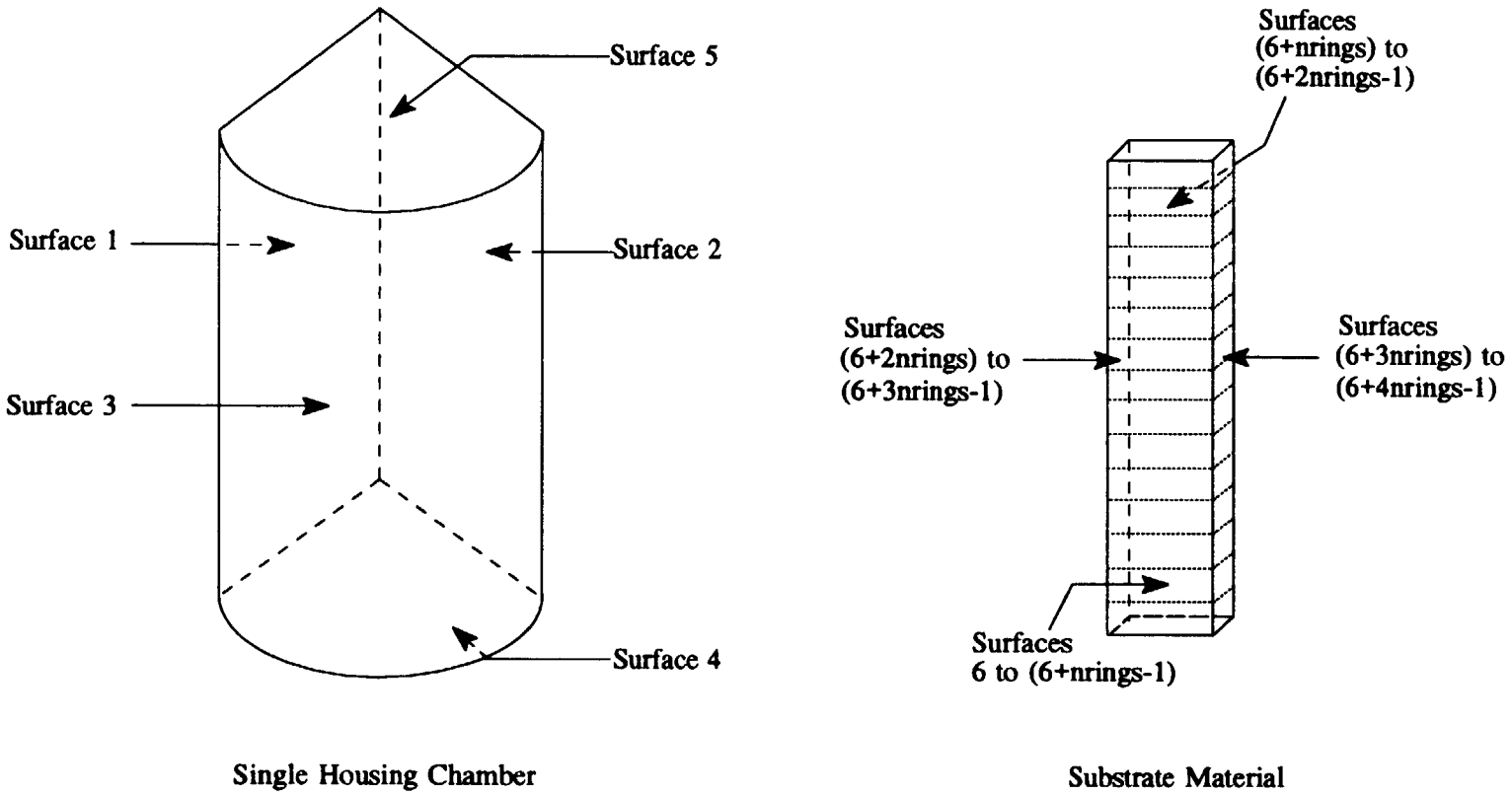


Figure 3.3.1. Description of the Enclosure.



Note: The enclosure is composed of n surfaces.
n = 5+4nrings.

Figure 3.3.2. Surface Numbering Scheme.

isolating the single housing chamber from the other two; surface 3 is the cylindrical wall and surfaces 4 and 5 are the bottom and top surfaces of the chamber, respectively. Because the temperatures or heat fluxes are uniform on the surfaces of the single housing chamber, these surfaces were not subdivided into smaller areas. The substrate material, however, was subdivided into several horizontal rings, denoted by *nrings*. Each ring has four surfaces. The enclosure contains n surfaces, where n is given by

$$n = 5 + 4(nrings) \ . \tag{3.3.1}$$

Note that the total area of the surfaces of the single housing chamber is much larger than the total area of the surfaces of the substrate material. This factor will be important in the interpretation of the radiant interchange results since the area parameter is taken into account in the calculation of the distribution factors.

The boundary conditions were specified on each surface. Because surfaces 1 and 2 play the role of isolator between the three single vacuum chambers, the flux is zero on these surfaces. For the other surfaces, the temperatures are known. On surface 3, the cylindrical wall, the temperature is assumed fixed at 30 K. The temperature at the end walls of the chamber are 4 K and 80 K, respectively. Since the substrate material is assumed to be bonded to the end walls of the cylinder, its temperature distribution varies from 4 K to 80 K. The temperature distribution obtained for the combination superconductor/substrate, using the finite difference program ORTHO3D, was adopted for the substrate temperature. This is appropriate because in the conduction analysis, the substrate has been shown to have the most effect on the temperature distribution of the combination. Using a polynomial fit, the temperature distribution of each substrate was

approximated as a function of the length of the thermal bridge. Polynomial fits were performed on the combinations YBCO/GREEN, BSCCO/FSI, and BSCCO/YSZ. The combinations BSCCO/FSI and BSCCO/YSZ were chosen for the substrates FSI and YSZ, respectively, because, from the conduction analysis, these generate less heat loads on the cryogen than the combinations YBCO/FSI and YBCO/YSZ. Consequently, when comparing the radiative heat source with the conductive heat source generated on the cryogen, the results will be conservative.

To complete the description of the enclosure, the surface radiative properties must be discussed. The basic radiative properties used in a radiant interchange analysis are the emissivity and absorptivity. Since the radiant interchange is modeled to account for mixed specular and diffuse reflections, the reflectivity ratio of each surface must also be known. The reflectivity ratio R is defined by

$$R = \frac{\rho^s}{\rho^s + \rho^d}, \tag{3.3.2}$$

where ρ^s is the specular component of the reflectivity and ρ^d is the diffuse component of the reflectivity. Unfortunately, a literature review was not able to yield the radiative properties of the specific substrate materials used in this research. This points out the lack of information in this area. Actually, we see that on one hand there has been a considerable increase of work in the radiation analysis field, but on the other hand insufficient work has been done in determining radiative properties. Hopefully within the next few years, more laboratories will be established for conducting such experiments. Until then, the properties can only be estimated.

3.3.1.2 Estimation of the Radiative Properties

The temperature in the enclosure varies from 4 K to 80 K. Referring to the hemispherical spectral emissive power of a blackbody (Siegel and Howell, 1992), in this temperature range the spectrum of the electromagnetic radiation within the enclosure should be dominated by long wavelengths (IR). Emission, absorption and reflection occur then in the same range of wavelengths for all surfaces. Therefore this problem is assumed to be independent of wavelength; or in other words, it is assumed that the surfaces are gray. This assumption could also have been concluded by simply considering the maximum temperature difference in the enclosure. Indeed, with a maximum temperature difference of "only" 76 K (compared to the temperature difference existing between the sun and the earth), the range of wavelengths at which each surface emits is expected to be roughly the same for all surfaces.

By applying the gray assumption, directional spectral radiative properties are simplified to directional properties. We can then write for the emissivity,

$$\epsilon'_{\lambda}(\lambda,T,\theta,\phi) = \epsilon'(T,\theta,\phi) \ , \tag{3.3.3}$$

and for the absorptivity,

$$\alpha'_{\lambda}(\lambda,T,\theta,\phi) = \alpha'(T,\theta,\phi) \ . \tag{3.3.4}$$

Now from Kirchhoff's law,

$$\epsilon'(T,\theta,\phi) = \alpha'(T,\theta,\phi) \ , \tag{3.3.5}$$

or more specifically

$$\epsilon'_{n}(T,\theta=0,\phi) = \alpha'_{n}(T,\theta=0,\phi) \ . \tag{3.3.6}$$

Equation (3.3.6) is a useful result because usually only the normal directional properties

are given in the literature. In addition, all surfaces in the enclosure are assumed to be diffuse emitters, which is equivalent to saying that the total directional incident intensity of radiation is independent of direction,

$$i'_i(T,\theta,\phi) = i'_i(T) \ . \tag{3.3.7}$$

It can be shown that

$$\alpha(T) = \frac{\int_{2\pi} \alpha'_n(T,\theta_i=0,\phi_i) i'_i(T) \cos\theta_i d\omega_i}{\int_{2\pi} i'_i(T) \cos\theta_i d\omega_i} = \epsilon(T) \ . \tag{3.3.8}$$

Therefore, only the emissivity, for which more information is given in the literature, needs to be estimated.

The electromagnetic theory applied to radiative-property prediction was used to estimate the emissivity of both the substrate material and the copper. This theory is based on the wave-surface interaction analysis, assuming an ideal interaction between the incident electromagnetic wave and the surface (Siegel and Howell, 1992). It has two limitations for practical calculations, which are the restriction to wavelengths greater than the visible spectrum and the application only for pure substances with ideally smooth surfaces. The first limitation is completely respected because the electromagnetic radiation within the enclosure is in the long wavelength region as previously explained. However, with respect to the second limitation, the effects of surface conditions of the copper and the substrate material on their radiative properties have to be neglected. This assumption is reasonable for the pure copper used in this research, which can be considered to be a highly polished copper. In the case of the substrate material, however,

the assumption of neglecting the effects of surface conditions could be responsible for large variations between the actual emissivity and the theoretical predicted emissivity.

The electromagnetic theory predicts the relations between the hemispherical and the normal emissivities. These relations are provided by Siegel and Howell (1992) for dielectric materials (Figure 4.7.b p. 116) and for metals (Figure 4.10.b p. 122). To estimate the unknown emissivities of the materials used in the enclosure, these relations were applied considering the substrate as a dielectric material and the copper as a metal. The normal emissivity (ϵ'_n) of another ceramic material, the magnesia (MgO) found in the literature (Siegel and Howell, 1992; Incropera and De Witt, 1990), was adopted for the substrate ($\epsilon'_n \in [0.7\text{-}0.9]$). In the case of the copper, the normal emissivity of a highly polished copper was used ($\epsilon'_n \in [0.01\text{-}0.02]$). The emissivities of the substrate and the copper were predicted to be in the ranges [0.7-0.85] and [0.013-0.025], respectively.

Because no method was found to provide an estimation of the reflectivity ratio R ($=\rho^s/(\rho^s+\rho^d)$), reasonable estimates from a professional in radiation heat transfer, Dr. J.R. Mahan (1994), were used.

Table 3.3.1 summarizes the predicted radiative properties. Nominal, minimum and maximum values are given for both materials (copper and substrate). These different values of the radiative properties will permit the study of the influence of the radiative properties in the problem. The percentage of deviation for the computed radiative heat flux on the substrate material, between the minimum and the maximum values of the radiative properties, is of interest. Table 3.3.1 shows the considerable difference between the radiative properties of the copper and those of the substrate material. The emissivity

Table 3.3.1. Predicted Radiative Properties of the Single Housing Chamber Material (Copper) and of the Substrate Material.

	Emissivity ϵ			Absorptivity α			Reflectivity ratio R		
	ϵ_{min}	ϵ_{nom}	ϵ_{max}	α_{min}	α_{nom}	α_{max}	R_{min}	R_{nom}	R_{max}
Substrate	0.70	0.80	0.85	0.70	0.80	0.85	0.10	0.15	0.20
Copper	0.013	0.020	0.025	0.013	0.020	0.025	0.90	0.95	1.00

(and absorptivity) of the copper is shown to be much smaller than the emissivity (and absorptivity) of the substrate material, whereas the reflectivity ρ of the copper is much higher (ρ=1-α). From this, large distribution factors for the surfaces of the substrate material will be expected in the Monte-Carlo analysis of the enclosure. Recall that the distribution factor D_{ij} represents the fraction of diffusely emitted radiation from surface i that is absorbed by surface j due to direct radiation and to all directional diffuse and specular reflections. However, the considerable difference between the total areas of the surfaces made of copper and those made of substrate material is also directly proportional in the distribution factor solution. Therefore no accurate conclusion can be anticipated from the distribution factor results.

Note that since the three substrate materials studied in this research have the same estimated radiative properties, the variation in their respective radiative heat loads will be a result of the variation in their temperature distributions.

3.3.2 Monte-Carlo Formulation

Now that the enclosure has been defined along with its physical properties, the next step is to analyze the radiation exchange between the surface elements. The radiant interchange model has to account for mixed specular and diffuse reflection. Furthermore it must have the ability to treat the specific geometry of the enclosure. With these required conditions, the Monte-Carlo method was chosen to model the present radiative problem.

The analysis of the radiation exchange between the surface elements of the enclosure was performed in two phases. First the distribution factors were computed using the Monte-Carlo method and second, the distribution factor results were used in calculating the radiative heat flux supplied to each surface.

3.3.2.1 Distribution Factors

The Monte-Carlo method is a statistical numerical method which models radiation by following the life of discrete energy bundles from emission to absorption using the probabilistic interpretation of the surface properties. The Monte-Carlo approach is straightforward and consists of six basic steps that have been explained in detail by, for example, Bongiovi (1993). Figure 3.3.3 provides a flow chart for the procedure. Distribution factors D_{ij} are calculated by counting the number of energy bundles emitted from each surface i (counter N_i in Figure 3.3.3) and the number of those absorbed by each surface j (counter N_{ij} in Figure 3.3.3). The solution converges if a sufficiently large number (depending on the problem) of energy bundles has been emitted. Consequently,

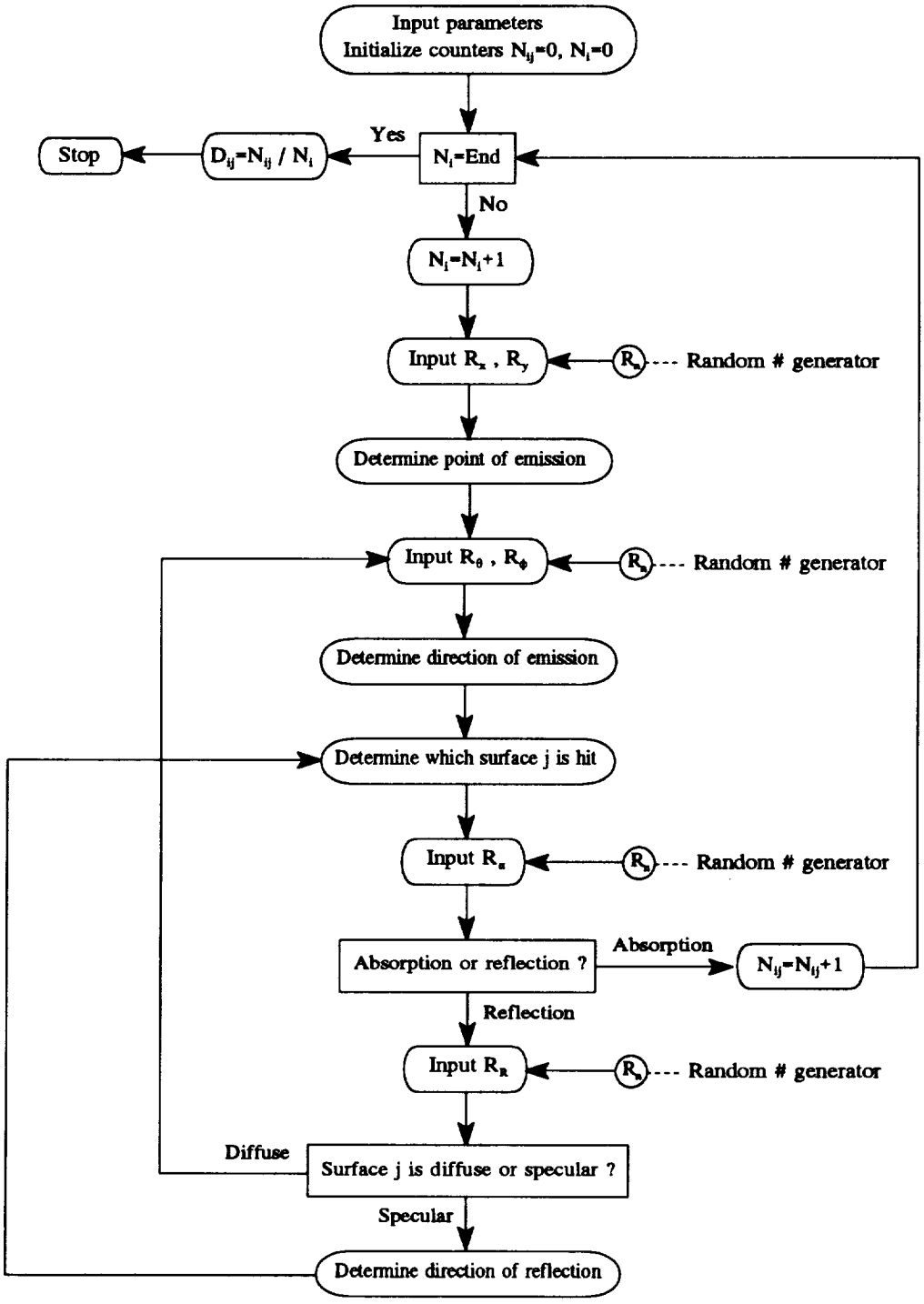


Figure 3.3.3. Flow Chart for the Monte-Carlo Procedure.

this procedure has the disadvantage of sometimes requiring an excessive computer CPU time. Thus, engineering judgment is necessary to settle compromises between the size and number of surface elements and the number of energy bundles emitted.

The Monte-Carlo analysis of the enclosure was performed by the Fortran code MC.FOR (see Appendix C). The output provided the distribution factors D_{ij} characteristic of the radiative exchange within the enclosure. The purpose of computing the distribution factors in the enclosure was to determine the radiative heat flux supplied to each surface, especially to the substrate material and to surface 4 which is in contact with the cold tip disk at a temperature of 4 K in the experimental design (see Figures 3.1.3 and 3.3.2).

A method to check convergence of the distribution factors is to calculate the weighted error E. From reciprocity,

$$\epsilon_i A_i D_{ij} = \epsilon_j A_j D_{ji} \ . \tag{3.3.9}$$

Summing this result over j ,

$$\sum_{j=1}^{n} \epsilon_i A_i D_{ij} = \sum_{j=1}^{n} \epsilon_j A_j D_{ji} \ , \quad i=1,2,...,n \ . \tag{3.3.10}$$

From the conservation of energy,

$$\sum_{j=1}^{n} D_{ij} = 1 \ , \quad i=1,2,...,n \ . \tag{3.3.11}$$

Therefore,

$$\epsilon_i A_i = \sum_{j=1}^{n} \epsilon_j A_j D_{ji} \ , \quad i=1,2,...,n \ , \tag{3.3.12}$$

and the error e_i on each surface i is

$$e_i = \sum_{j=1}^{n} \frac{\epsilon_f A_f D_{ji}}{\epsilon_f A_i} - 1 \ , \quad i=1,2,...,n \ . \tag{3.3.13}$$

The weighted error E is then

$$E = \frac{\sum_{i=1}^{n} e_f A_i}{\sum_{i=1}^{n} A_i} \ . \tag{3.3.14}$$

The error E should decrease as convergence is obtained or, in other words, as the number of energy bundle increases.

3.3.2.2 Radiative Heat Flux

The radiative heat flux Q_i (in watts) on surface i can be defined as the difference between the emitted and absorbed radiation by surface i. When Q_i is negative, surface i globally absorbs energy and when Q_i is positive, surface i globally emits energy. In the case of the enclosure described in this study, which consists of n diffusely emitting and absorbing, diffuse-specularly reflecting, gray, opaque surfaces, the radiation emitted by surface i at temperature T_i with area A_i and emissivity ϵ_i is

$$Q_{i,emit} = A_i \epsilon_i \sigma T_i^4 \ , \tag{3.3.15}$$

where σ is the Stefan-Boltzmann constant (σ=5.6696x10^{-8} W/m^2-K^4).

Using the distribution factors, the radiation emitted by surface j at temperature T_j with area A_j and emissivity ϵ_j and absorbed by surface i is

$$Q_{ji,abs} = A_j \epsilon_j \sigma T_j^4 D_{ji} \ . \tag{3.3.16}$$

Then summing over j gives the total radiation absorbed by surface i,

$$Q_{i,abs} = \sum_{j=1}^{n} A_j \epsilon_j \sigma T_j^4 D_{ji} \ . \tag{3.3.17}$$

Applying reciprocity (see Eq. (3.3.9)),

$$Q_{i,abs} = \sum_{j=1}^{n} A_i \epsilon_i \sigma T_j^4 D_{ij} \ . \tag{3.3.18}$$

The radiative heat flux Q_i on surface i can therefore be expressed as

$$Q_i = Q_{i,emit} - Q_{i,abs} \ . \tag{3.3.19}$$

Substituting $Q_{i,emit}$ and $Q_{i,abs}$,

$$Q_i = A_i \epsilon_i \sigma T_i^4 - \sum_{j=1}^{n} A_i \epsilon_i \sigma T_j^4 D_{ij} \ , \quad i=1,2,...,n \ . \tag{3.3.20}$$

Introducing the Kronecker delta function δ_{ij} simplifies the flux to

$$Q_i = \sum_{j=1}^{n} A_i \epsilon_i \sigma T_j^4 (\delta_{ij} - D_{ij}) \ , \quad i=1,2,...,n \ . \tag{3.3.21}$$

When all surfaces in an enclosure have a specified temperature, the radiative heat fluxes are then easily solved by using Eq. (3.3.21). However, in this study, the enclosure contains two surfaces (surfaces 1 and 2, see Section 3.3.1) with a specified heat flux. In this case, the unknown temperatures must first be determined before solving for the unknown heat fluxes. A general solution was written assuming that in the enclosure of interest, N surfaces have specified heat flux and $(n\text{-}N)$ surfaces have specified temperatures. Therefore, the radiative heat flux Q_i is known for $1 \le i \le N$, and the temperature T_i is known for $N+1 \le i \le n$.

Using Eq. (3.3.21), for $1 \le i \le N$, we can write

$$\frac{Q_i}{A_i \epsilon_i \sigma} = \sum_{k=1}^{N} (\delta_{ik} - D_{ik}) T_k^4 + \sum_{j=N+1}^{n} (-D_{ij}) T_j^4 \ . \tag{3.3.22}$$

Rearranging,

$$\sum_{k=1}^{N} (\delta_{ik} - D_{ik}) T_k^4 = \frac{Q_i}{A_i \epsilon_i \sigma} + \sum_{j=N+1}^{n} D_{ij} T_j^4 \ , \quad i=1,2,...,N \ . \tag{3.3.23}$$

If we define

$$U_{ik} = \sum_{k=1}^{N} (\delta_{ik} - D_{ik}) \ , \quad i=1,2,...,N \ , \tag{3.3.24}$$

and

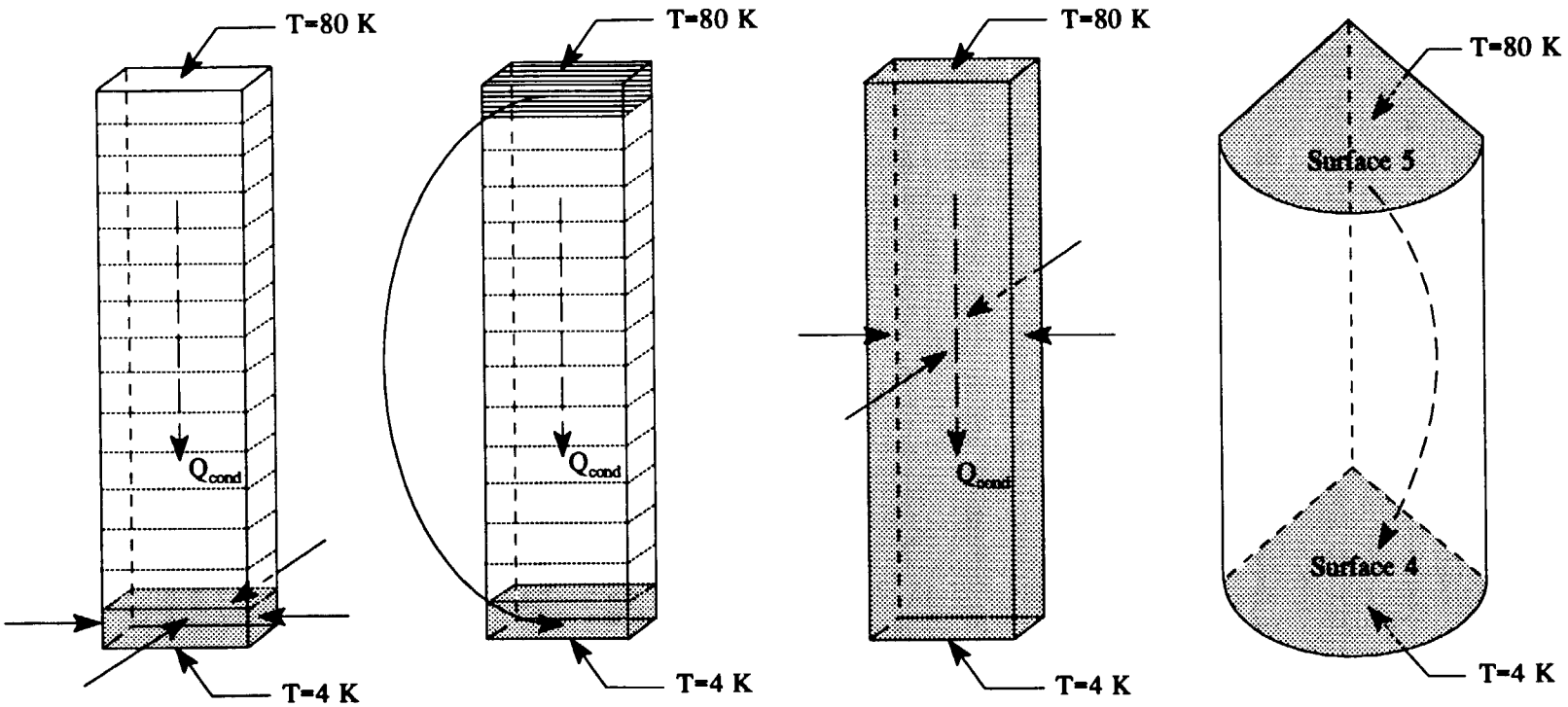
$$V_{ij} = \frac{Q_i}{A_i \epsilon_i \sigma} + \sum_{j=N+1}^{n} D_{ij} T_j^4 \ , \quad i=1,2,...,N \ , \tag{3.3.25}$$

Equation (3.3.23) yields

$$U_{ik} \ [T_k^4] = V_{ij} \ , \quad i=1,2,...,N \ , \tag{3.3.26}$$

where the summations over k (from 1 to N) and over j (from N+1 to n) are implied by the repeated subscripts. This matrix form allows us to solve for the unknown temperatures T_k by inverting the U_{ik} matrix. Going back to our specific case where T_1 and T_2 are unknown, Eq. (3.3.26) is applied for N=2. The unknown radiative heat fluxes Q_i can then be determined using Eq. (3.3.21).

The final objective is to determine whether or not the radiative heat source is a significant source of heat load on the cryogen. In doing so, it is interesting to consider four different radiative heat loads, as seen in Figure 3.3.4, and to compare them with the conductive heat load supplied to the cryogen. In the first case, the radiative heat load



Case 1: $Q_{rad\text{-}bot}$

Radiative Heat Load
On the Bottom of
the Substrate From
the Entire Enclosure

Case 2: $Q_{rad\text{-}topbot}$

Radiative Heat Load
On the Bottom of
the Substrate From
the Top of the Substrate

Case 3: $Q_{rad\text{-}tot}$

Radiative Heat Load
On the Entire Substrate
From the Entire
Enclosure

Case 4: $Q_{rad\text{-}54}$

Radiative Heat Load
On Surface 4
From Surface 5

Figure 3.3.4. Radiative Heat Loads Analyzed in the Enclosure.

($Q_{rad\text{-}bot}$) on the bottom of the substrate (T=4 K) from the entire enclosure is analyzed. Then in the second case the radiative heat load ($Q_{rad\text{-}topbot}$) on the bottom of the substrate (T=4 K) from the top of the substrate (T=80 K) is considered. The radiative heat load ($Q_{rad\text{-}tot}$) on the entire substrate from the entire enclosure is studied in the third case. Eventually, in the final case the radiative heat load ($Q_{rad\text{-}54}$) on surface 4 (bottom wall of the housing chamber at 4 K) from surface 5 (top wall of the housing chamber at 80 K) is calculated. The computation of these four radiative heat loads is described below. Recall that the substrate is divided into several rings (see Figure 3.3.2); and a ring is constituted by 4 surfaces. The bottom of the substrate is the first ring (surfaces 6, 6+*nrings*, 6+2*nrings*, 6+3*nrings*) and the top of the substrate is the last ring (surfaces 6+*nrings*-1, 6+2*nrings*-1, 6+3*nrings*-1, 6+4*nrings*-1). In this notation, "2nrings", for example, implies a multiplication of "nrings" by "2". It is important to point out that the radiative heat loads represented by cases 1 and 2 are arbitrary as these heat loads depend on the number and size of the rings in the substrate material, whereas the radiative heat loads represented by cases 3 and 4 are absolute values. Therefore, cases 3 and 4 will be more meaningful than the two first cases in the interpretation of the importance of radiation on the cryogenic heat load.

For the first case we define $Q_{rad\text{-}bot}$ as

$$Q_{rad\text{-}bot} = Q(6)+Q(6+nrings)+Q(6+2nrings)+Q(6+3nrings) \ . \qquad (3.3.27)$$

Then for the second case $Q_{rad\text{-}topbot}$ is described by

$$Q_{rad\text{-}topbot} = Q_{bot,emit} - Q_{topbot,abs} \ , \qquad (3.3.28)$$

where $Q_{bot,emit}$ is the radiation emitted by the bottom of the substrate (see Eq. (3.3.15)),

and $Q_{topbot,abs}$ is the radiation emitted by the top of the substrate and absorbed by the bottom (see Eq. (3.3.16)). Substituting,

$$Q_{rad\text{-}topbot} = \sum_i [A_i \epsilon_i \sigma T_i^4 - \sum_j [A_i \epsilon_i \sigma T_j^4 D_{ij}] , \tag{3.3.29}$$

where i=6, 6+*nrings*, 6+2*nrings*, 6+3*nrings*, and j=6+*nrings*-1, 6+2*nrings*-1, 6+3*nrings*-1, 6+4*nrings*-1. Rearranging,

$$Q_{rad\text{-}topbot} = \sum_i A_i \epsilon_i \sigma \, [T_i^4 - \sum_j T_j^4 D_{ij}] , \tag{3.3.30}$$

where the summations over i and j are specified above.

For the third case $Q_{rad\text{-}tot}$ is stated as

$$Q_{rad\text{-}tot} = \sum_{i=6}^{n} Q(i) , \tag{3.3.31}$$

where the surfaces 6 to n are those which constitute the substrate material.

Finally, for the fourth case $Q_{rad\text{-}54}$ is expressed as

$$Q_{rad\text{-}54} = Q_{4,emit} - Q_{54,abs} , \tag{3.3.32}$$

where $Q_{4,emit}$ is the radiation emitted by surface 4 (see Eq. (3.3.15)), and $Q_{54,abs}$ is the radiation emitted by surface 5 and absorbed by surface 4 (see Eq. (3.3.16)). Substituting,

$$Q_{rad\text{-}54} = A_4 \epsilon_4 \sigma [T_4^4 - T_5^4 D_{45}] . \tag{3.3.33}$$

3.3.3 Results and Discussion

This subsection is devoted to the results obtained from the radiant interchange analysis within the enclosure described in Section 3.3.1. The results for the distribution factors, determined using the program MC.FOR (Appendix C), are discussed first. Then the results for the four radiative heat loads described in the previous section and calculated using the program TQ.FOR(Appendix D), are analyzed.

3.3.3.1 Results of the Distribution Factors

In the next subsections, the convergence and symmetry of the solution of the distribution factors and the influence of the estimated radiative properties are carefully examined. Note that to analyze the distribution factors, the substrate was not divided into several rings. Therefore *nrings* equals 1 and then the total number of surfaces n in the enclosure is nine. This case was chosen because it requires less computing time.

3.3.3.1.1 Convergence of the Distribution Factors

First let us be sure that the solution for the distribution factors D_{ij} converges before analyzing the heat flux. The convergence is checked by calculating the weighted error E using Eq. (3.3.14). Since the solution converges if a sufficiently large number of energy bundles has been emitted, the error E should decrease with the increase of the number of bundles. This behavior is outlined by Figure 3.3.5. As expected, the slope of the decrease of E is very large for small numbers of bundles emitted. Then the slope decreases and becomes very slight after 100,000 energy bundles have been emitted

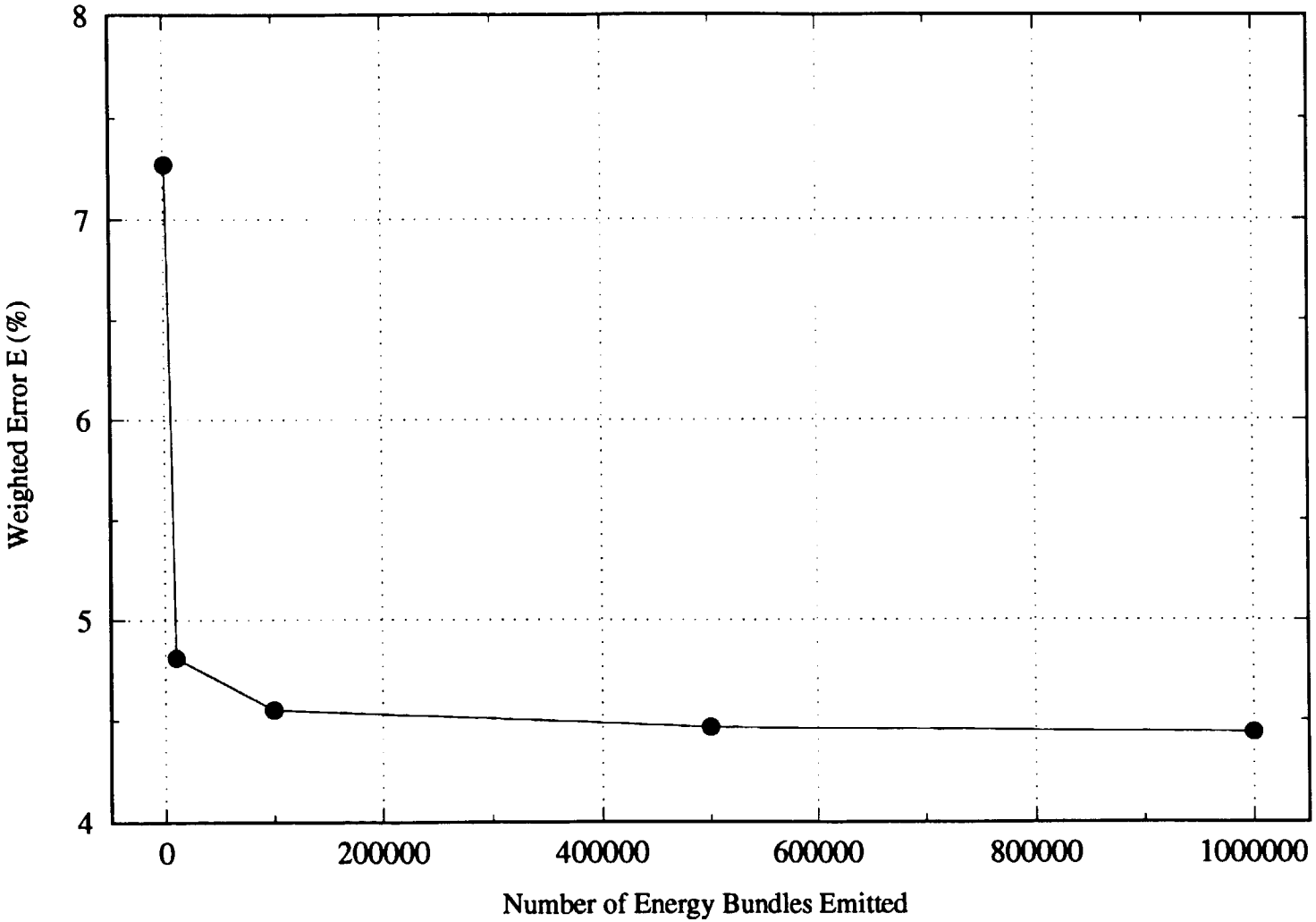


Figure 3.3.5. Weighted Error E vs. Number of Energy Bundles Emitted.

(E=4.55 percent). For the emission of a million bundles, the weighted error has decreased to 4.44 percent. These results were obtained using the nominal radiative properties. The weighted error was found to be slightly smaller for the maximum values of the properties and slightly larger for the minimum values of the properties. This phenomenon is logical because the minimum absorptivity of the copper corresponds to its maximum reflectivity, and as more energy bundles are reflected, the numerical error becomes larger.

For the following runs, the number of energy bundles emitted was chosen to be 100,000 bundles. This choice allows for a reasonable compromise between the accuracy of the solution and the long computing time required to run with a maximum absorptivity of 0.025 for the larger surfaces (single housing chamber surfaces).

3.3.3.1.2 Respect of the Symmetry

Now let us check that the geometric symmetry in the enclosure is respected in the distribution factors. The examples below show that, for two identical surfaces in the enclosure, the same (± the error E) distribution factors are obtained:

$$D(1,1) = 0.0578 \sim D(2,2) = 0.0576$$

$$D(1,2) = 0.0667 \sim D(2,1) = 0.0651$$

$$D(3,1) = 0.0721 \sim D(3,2) = 0.0740$$

$$D(9,4) = 0.0191 \sim D(8,5) = 0.0189$$

$$D(9,1) = 0.0520 \sim D(8,2) = 0.0518$$

These results were obtained for 100,000 energy bundles emitted, using the nominal

radiative properties. For 1,000,000 bundles emitted, the results are more accurate, as one would expect. It can thus be concluded that the solution is symmetric.

3.3.3.1.3 Influence of the Radiative Properties on the Distribution Factors

Figure 3.3.6 shows the effects of the surface radiative property estimated values on the distribution factors *D(1,j)*, for *nrings*=1 and hence $1 \leq j \leq 9$. The variation for *D(1,j)* depends strongly on the properties. For the minimum values of the absorptivity (α_{copper}=0.013 in the single housing chamber), surface 1 sees itself less than for the nominal values (α_{copper}=0.02) and almost half as much as for the maximum values (α_{copper}=0.025). This behavior is expected since the value of the minimum absorptivity is half the value of the maximum absorptivity. For each surface, the distribution factors obtained using the nominal properties are approximately the average between the results obtained using the minimum properties and those obtained using the maximum properties.

An interesting result is the indirect effect of the housing chamber material properties. The actual housing chamber material is the OFHC copper with a very low emissivity and absorptivity, and a very high reflectivity. For the minimum values of the absorptivity or, in other words, for the maximum values of the reflectivity (recall that ρ=1-α), surface 1 sees the substrate better than for larger absorptivities (smaller reflectivities) due to all the reflections on the copper. This demonstrates the fact that a material with lower reflectivity for the housing chamber would contribute to lower distribution factors to the substrate and hence lower radiative heat flux on the substrate. Indeed, the lower the distribution factors to the substrate, the lower the radiative heat flux on the substrate.

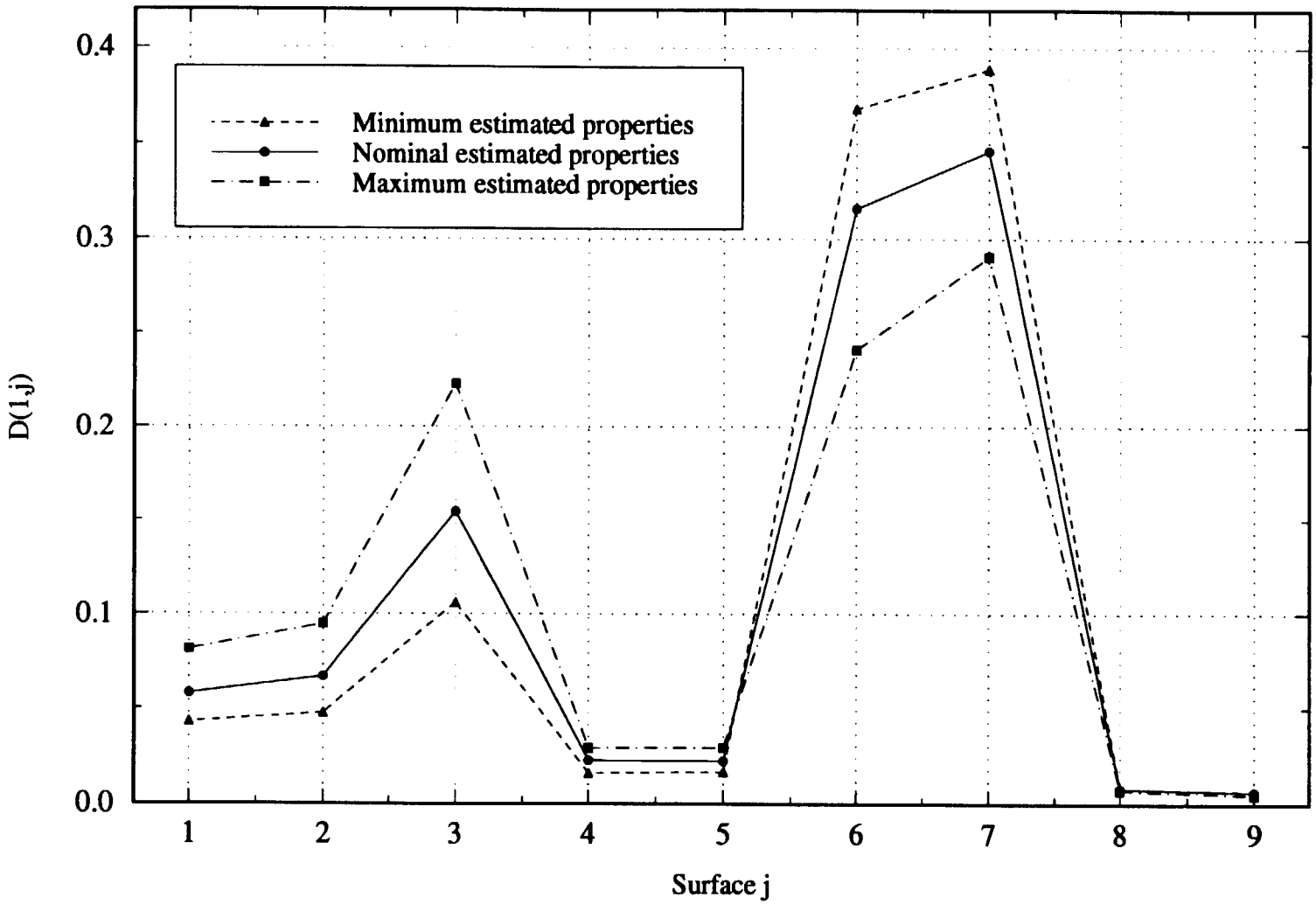


Figure 3.3.6. Distribution Factors D(1,j) for Nrings = 1 and for the Minimum, Nominal and Maximum Values of the Estimated Radiative Properties.

Note, however, that the high reflectivity (low absorptivity) of the single housing chamber material will help to obtain a low radiative heat load on surface 4 from surface 5.

3.3.3.2 Results of the Radiative Heat Flux

The interpretation of the results of the radiative heat flux on the substrate and on surface 4 is the focal point in this investigation. The influence on the radiative heat flux of both the substrate material division into several rings and the estimated radiative properties, is studied. The four different radiative heat loads, developed in Section 3.3.2.2, are then analyzed for the three different substrate materials studied, and compared with the conductive heat loads generated on the cryogen by these materials. The conductive heat loads were predicted in the conduction analysis using the finite difference program ORTHO3D. The comparisons allow us to conclude whether or not radiation can be neglected in the conduction analysis of the HTS thermal bridges.

3.3.3.2.1 Influence of the Number of Rings

The subdivision of the substrate material into as many rings as possible is advised since it provides an accurate temperature distribution in the substrate. Indeed, for each ring, the temperature is assumed uniform and its value is taken in the middle of the ring. The results obtained for one ring in the substrate thus have no physical meaning.

The maximum number of rings studied was sixteen which, in the optimization of the computer CPU time, gave good trade-off between the number of surfaces in the enclosure (n=69 for *nrings*=16) and the number of energy bundles emitted (100,000).

3.3.3.2.2 Influence of the Radiative Properties on the Radiative Heat Flux

A study was done on the influence of the estimated radiative properties on the radiative heat load $Q_{rad\text{-}topbot}$ for the fused silica substrate divided into sixteen rings. $Q_{rad\text{-}topbot}$ was described in Section 3.3.2.2 as the radiative heat load on the bottom of the substrate (T=4 K) from the top of the substrate (T=80 K). The fused silica (FSI) substrate was preferred to the two other substrates because it generates less of a conductive heat load.

Table 3.3.2 shows that the amount of radiation received by the bottom of the FSI substrate from its top increases as the absorptivity of both materials, and especially the substrate, increases. This relationship indicates that the single housing chamber material properties (specifically the reflectivity) have less effect on the radiative heat load on the substrate than the properties of the substrate itself. Indeed, there is a direct relationship between the absorptivity (or the reflectivity) of the substrate and its radiative heat flux, which is: the larger the absorptivity (the lower the reflectivity), the larger the radiative heat flux (see Eq. (3.3.18) which gives the total radiation absorbed by surface i, and recall

Table 3.3.2. Radiative Heat Load on the Bottom of the FSI Substrate (T=4 K) from its Top (T=80 K) for the Minimum, Nominal, and Maximum Radiative Property Values and for *nrings*=16.

Estimated Values	Minimum	Nominal	Maximum	% difference (max-nom)/ nom	% difference (min-nom)/ nom
$Q_{rad\text{-}topbot}$ (W)	-9.19 E-6	-9.45 E-6	-9.96 E-6	5.4	-2.9

* Note: the negative sign means that the substrate material receives radiation.

that $\epsilon_i=\alpha_i$ in the enclosure).

To be conservative with the results, the percentage difference between the maximum and the nominal radiative properties needs to be taken into account. This percentage displays an increase of 5.4 percent in the radiative heat load, which is relatively small compared to the difference between the maximum and the nominal radiative properties.

3.3.3.2.3 Importance of the Radiative Heat Source

The four different radiative heat loads, $Q_{rad\text{-}bot}$, $Q_{rad\text{-}topbot}$, $Q_{rad\text{-}tot}$ and $Q_{rad\text{-}54t}$ developed in Section 3.3.2.2, were evaluated for the three substrate materials (FSI, YSZ and GREEN) divided into sixteen rings, using the nominal radiative property values. The radiative heat loads were compared to the conductive heat load generated on the cryogen by each substrate. The conductive heat loads were obtained in the conduction analysis using the finite difference program ORTHO3D. In performing these comparisons, ratios *Rad/Cond*, which shows the importance of radiation over conduction, were computed.

- The first case studied, the radiative heat load ($Q_{rad\text{-}bot}$) on the bottom of the substrate (T=4 K) from the entire enclosure, was the worst case regarding the radiative heat load. Table 3.3.3 shows that $Q_{rad\text{-}bot}$ represents almost 30 percent of the conductive heat load if the substrate is FSI and almost 10 percent if the substrate is YSZ. The small percentage (1.46 percent) obtained for the GREEN substrate comes from the fact that this substrate generates a much larger conductive heat load on the cryogen than the two other substrates.

- In the second case, the radiative heat load ($Q_{rad\text{-}topbot}$) from the 80 K end wall of the substrate (last ring) to the 4 K end wall (first ring) was analyzed. Table 3.3.4 shows that the bottom of the FSI substrate receives from its top a radiative load smaller than 3.9 percent of the conductive load flowing between the two end walls of the substrate. This percentage decreases to 1.59 percent and 0.24 percent when the YSZ and GREEN substrate are employed, respectively. These percentages are very small and indicate that, compared to conduction, radiation from the top of the substrate has a very slight effect on the bottom of the substrate.

- The third case displays the radiative heat load ($Q_{rad\text{-}tot}$) on the entire substrate from the entire enclosure. From Table 3.3.5, it is obvious that radiation on each substrate is negligible compared to the conduction through each substrate.

- Finally, in the fourth case the radiative heat load ($Q_{rad\text{-}54}$) on surface 4 from surface 5 is presented. Table 3.3.6 shows that this radiative source represents less than 1.7 percent of the conductive source if the substrate is FSI and is negligible if the substrate is YSZ or GREEN.

Table 3.3.3. Radiative Heat Load on the Bottom of the Substrate (T=4 K) from the Entire Enclosure.

Substrate	FSI	YSZ	GREEN
$Q_{rad\text{-}bot}$ (W)	-7.21 E-5	-5.69 E-5	-5.45 E-5
Q_{cond} (W)	2.44 E-4	5.89 E-4	3.74 E-3
Ratio (%) *Rad/Cond*	29.59	9.66	1.46

Table 3.3.4. Radiative Heat Load on the Bottom of the Substrate (T=4 K) from the Top of the Substrate (T=80 K).

Substrate	FSI	YSZ	GREEN
$Q_{rad\text{-}topbot}$ (W)	-9.45 E-6	-9.40 E-6	-9.04 E-6
Q_{cond} (W)	2.44 E-4	5.89 E-4	3.74 E-3
Ratio (%) *Rad/Cond*	3.87	1.59	0.24

Table 3.3.5. Radiative Heat Load on the Entire Substrate from the Entire Enclosure.

Substrate	FSI	YSZ	GREEN
$Q_{rad\text{-}tot}$ (W)	-1.46 E-6	-1.24 E-6	-1.13 E-6
Q_{cond} (W)	2.44 E-4	5.89 E-4	3.74 E-3
Ratio (%) *Rad/Cond*	0.59	0.21	0.03

Table 3.3.6. Radiative Heat Load on Surface 4 from Surface 5.

Substrate	FSI	YSZ	GREEN
$Q_{rad\text{-}54}$ (W)	-4.07 E-6	-4.07 E-6	-4.07 E-6
Q_{cond} (W)	2.44 E-4	5.89 E-4	3.74 E-3
Ratio (%) *Rad/Cond*	1.67	0.69	0.11

The previous results show that in the first case, the radiative heat load on the bottom of the substrate from the entire enclosure can be negligible for the Green Phase but not for the Fused Silica and the Yttrium-Stabilized Zirconia. In the second case, the radiative heat load applied from the top end of the substrate to the bottom end represents only 3.87 percent of the conductive heat load generated on the cryogen for the FSI substrate and can be negligible (< 2 percent) for the YSZ and GREEN substrates. The results for the third and fourth case are of interest because, as previously mentioned, the radiative heat loads represented by these cases are absolute values and do not depend on the number and size of the rings in the substrate material. The third case indicated that the radiative heat load on the entire substrate from the entire enclosure was found to be negligible for each substrate. Finally, the radiative heat load on surface 4 from surface 5 can be negligible (< 2 percent) for each substrate.

Considering that the comparisons of the conductive heat load with the radiative heat loads of the third and fourth case ($Q_{rad\text{-}tot}$ and $Q_{rad\text{-}54}$, respectively) provide the best information for the significance of radiation on the cryogenic heat load, it is reasonable to conclude that neglecting radiation on the HTS thermal bridges is a valid assumption in the conduction analysis.

An important feature also displayed by this study is the use, for the housing chamber, of a material with lower reflectivity than the reflectivity of pure copper. A lower reflectivity would give fewer reflections on the housing walls, and then for instance, two rings of the substrate would not be able to "see" each other any more. This would lower radiation on the substrate. A lower reflectivity would also make the housing walls absorb

more radiant energy since the absorptivity would be higher (ρ=1-α), but they would also emit more radiant energy (recall that α=ϵ). The overall repercussion of using a housing chamber material with lower reflectivity has been stressed in Section 3.3.3.1.3 when using the maximum absorptivity values (minimum reflectivity values) to calculate the distribution factors. The minimum reflectivity for the housing chamber material helped to lower the distribution factors to the substrate. Thus, it is logical to assume that it would help to lower radiation on the substrate. Consequently, it could then be possible to obtain negligible radiative heat loads also for the first case analyzed previously. Note, however, that in order to maintain the radiative heat load on surface 4 from surface 5 negligible, it is advised to keep the pure copper material with high reflectivity for surfaces 4 and 5 of the housing chamber.

It should be noted that since the HTS thermal bridges were approximated as the substrate materials alone, the radiative properties of the superconductors were not taken into account. However, the reflectivity of superconductors has been shown to be very high (Siegel and Howell, 1992). In addition the study of the influence of the radiative properties on the radiative heat flux showed that the larger the reflectivity of the substrate material, the lower the radiative heat load on the substrate. Therefore, the difference between the high reflectivity of superconductors and the predicted low reflectivity of substrate materials is expected to affect the results on the superconductor/substrate combinations in a positive manner, that is, it should help to obtain lower radiative heat loads on the combinations. However, this difference is anticipated to increase the radiative heat load on surface 4 from surface 5.

CHAPTER 4

Thermal Conductivity Estimation of the HTS Thermal Bridges

This chapter focuses on the estimation of the thermal conductivities of the HTS thermal bridges in the space environment. The capability to develop a methodology for the determination of these thermal properties is a key strategy to assess the feasibility of HTS-substrate combinations as electronic leads in infrared sensor satellite systems. Two temperature-dependent thermal conductivity models for the HTS thermal bridges were sought in this investigation. These allowed for the analysis of both the thermal conductivities of the individual HTS thermal bridge materials and the effective thermal conductivities of the HTS thermal bridges. Detailed sensitivity studies were conducted on both thermal conductivity models. These studies resulted in the impossibility to estimate the thermal conductivities as functions of the temperature along the thermal bridges. Therefore, constant effective thermal conductivities were eventually estimated for the HTS thermal bridges using the modified Box-Kanemasu estimation procedure.

The first section provides the theoretical development of the analysis used in estimating the thermal conductivity model parameters. The results of the investigation

for the estimation of thermal conductivities of the HTS thermal bridges are presented and discussed in the next section.

4.1 Theoretical Considerations

In this section, the theoretical development used to analyze both the thermal conductivities of the individual HTS thermal bridge materials and the effective thermal conductivities of the composite HTS thermal bridges are presented. The material thermal conductivities and the effective thermal conductivities were formulated as functions of temperature. The estimation procedure for the thermal conductivities, a minimization method called the modified Box-Kanemasu method, requires both calculated and experimental temperatures. To estimate the individual thermal conductivities, that is for the HTS and the substrate materials, or "material" thermal conductivities, the calculated temperatures were obtained using the program ORTHO3D. The analysis of an "effective" thermal conductivity, or the combined analysis of the material properties, allowed for the simplification of the conductive heat transfer analysis within the thermal bridges. Therefore, to estimate the effective thermal conductivities, the calculated temperatures were obtained using a one-dimensional numerical scheme. In implementing the estimation procedure (Section 4.2), simulated temperature data were generated by adding random errors with a known variance to the temperature solution produced in the conductive analysis (Section 3.2.4.2). Prior to the actual implementation of this procedure, a detailed sensitivity study was performed to ensure reliable parameter

estimates.

The first subsection focuses on the material and effective thermal conductivity models. The second subsection provides the mathematical details of the parameter estimation technique. The simulation of measured temperature data is discussed in the following subsection. The final subsection describes the calculation of the sensitivity coefficients and then details the methodology for the sensitivity analysis.

4.1.1 Thermal Conductivity Models

The capability to determine the thermal conductivities of the HTS thermal bridges in the space environment will enable the assessment of the performance of HTS materials as electronic leads in sensor satellites. In this investigation, mathematical models were needed to estimate the material (eg., HTS, substrate and buffer layer if there is one) and the effective thermal conductivities as functions of the temperature along the length of the thermal bridges.

4.1.1.1 Thermal Conductivity Model of the HTS Thermal Bridge Materials

The HTS thermal bridges studied in this research are comprised of HTS-substrate combinations with and without a buffer layer, depending on the substrate material used, as described in Section 3.1. The equations and the plots of the material thermal conductivities have been provided by Lee (1994), and can be found in Appendix E. The thermal conductivities, k_i, were modeled as sixth-degree polynomials in temperature, that is

$$k_i = \beta_{i1}+\beta_{i2}T+\beta_{i3}T^2+\beta_{i4}T^3+\beta_{i5}T^4+\beta_{i6}T^5+\beta_{i7}T^6 \ , \qquad (4.1.1)$$

where i denotes a specific material (eg., for the BSCCO/FSI thermal bridge, i=FSI for the FSI substrate, i=BSCCO for the BSCCO superconductor and i=buffer layer for the buffer layer). The estimation of k_i will be performed through the study of the thermal conductivity coefficients β_{ij} (j=1,7).

4.1.1.2 Effective Thermal Conductivity Model of the HTS Thermal Bridges

The HTS thermal bridges can be described by parallel thermal circuit configurations, as shown in Figure 4.1.1. Note that because the temperature distribution is one-dimensional along the thermal bridges, there is no temperature gradient, and hence no thermal resistance, between the parallel elements in the configurations. The equivalent thermal circuit is characterized by an effective thermal resistance, $R_{th_{eff}}$. Using the thermal resistances, R_{th_i}, of each element i in the network, $R_{th_{eff}}$ is then expressed as

$$\frac{1}{R_{th_{eff}}} = \sum_i \frac{1}{R_{th_i}} \ . \qquad (4.1.2)$$

For a circuit of length L and cross-sectional area A_c, the thermal conductivity, k, can be defined from the thermal resistance, R_{th}, by the relationship

$$k = \frac{L}{A_c}\frac{1}{R_{th}} \ . \qquad (4.1.3)$$

Combining Eqs. (4.1.2) and (4.1.3), we obtain an effective thermal conductivity, k_{eff}, for the HTS thermal bridges,

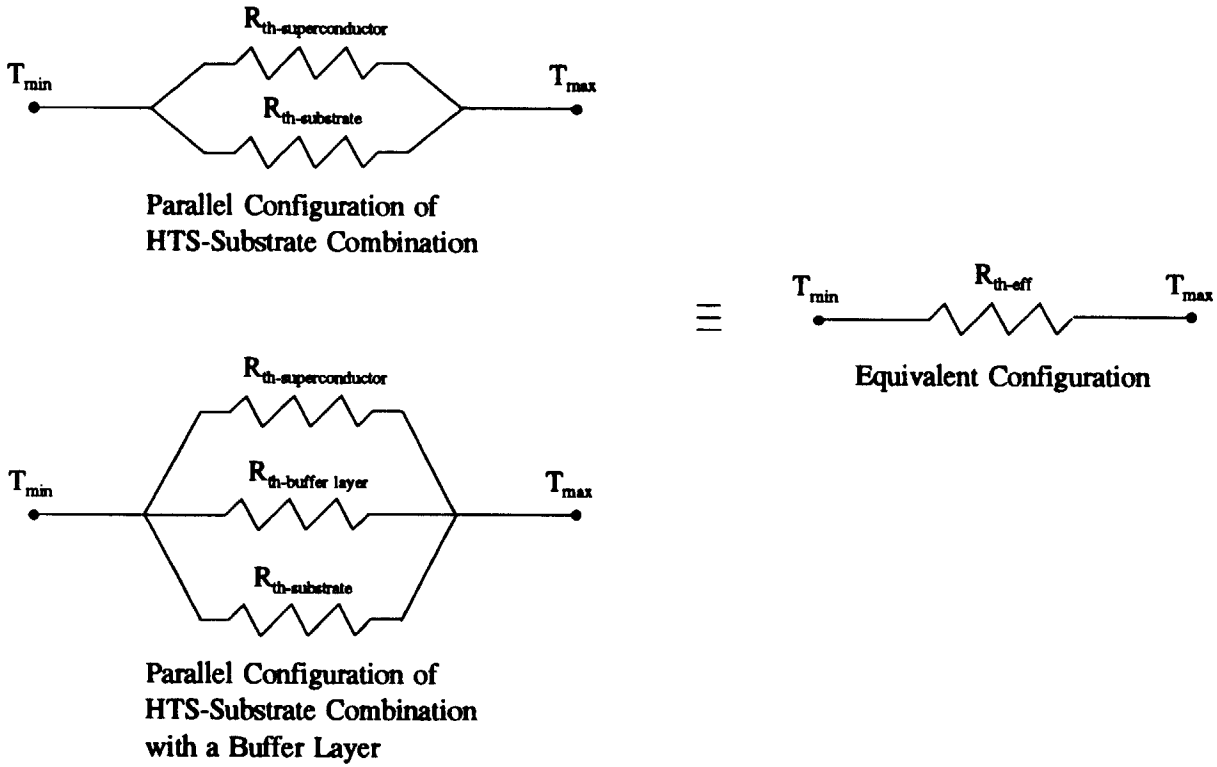


Figure 4.1.1. Parallel and Equivalent Thermal Circuit Configurations of HTS Thermal Bridges.

$$k_{eff} = \sum_i \frac{A_{c_i}}{A_{c_{tot}}} k_i \ , \tag{4.1.4}$$

where the subscript i includes the superconductor, the substrate material, and the buffer layer if there is one. $A_{c_{tot}}$ represents the total cross-sectional area of the specific thermal bridge and is calculated from

$$A_{c_{tot}} = \sum_i A_{c_i} \ . \tag{4.1.5}$$

As mentioned previously, each k_i is defined by a sixth-degree polynomial in temperature. Therefore, k_{eff} is first determined as a sixth-degree polynomial and, using a curve fit on the temperature range investigated (4-80 K), it is then restricted to a third-degree polynomial to account for nonnegligible coefficients in the polynomial equation only. The effective thermal conductivities of the HTS thermal bridges are eventually expressed as

$$k_{eff_i} = \beta_{i1} + \beta_{i2} T + \beta_{i3} T^2 + \beta_{i4} T^3 \ , \tag{4.1.6}$$

where i denotes a specific thermal bridge. Again, the estimation of k_{eff_i} will be performed through the study of the parameters β_{ij} (j=1,4).

Figure 4.1.2 shows the effective thermal conductivity, $k_{eff_{BF}}$, of the thermal bridge BSCCO/FSI with the thermal conductivities of the superconductor BSCCO, k_{BSCCO}, the substrate FSI, k_{FSI}, and the buffer layer (zirconia), $k_{buffer\ layer}$. One should notice that the effective thermal conductivity distribution is similar to the substrate thermal conductivity distribution. This behavior validates the general result presented by Lee (1994) that the substrate material is the dominant factor in the HTS thermal bridge. The plots and the coefficients of the effective thermal conductivities of the five different HTS thermal

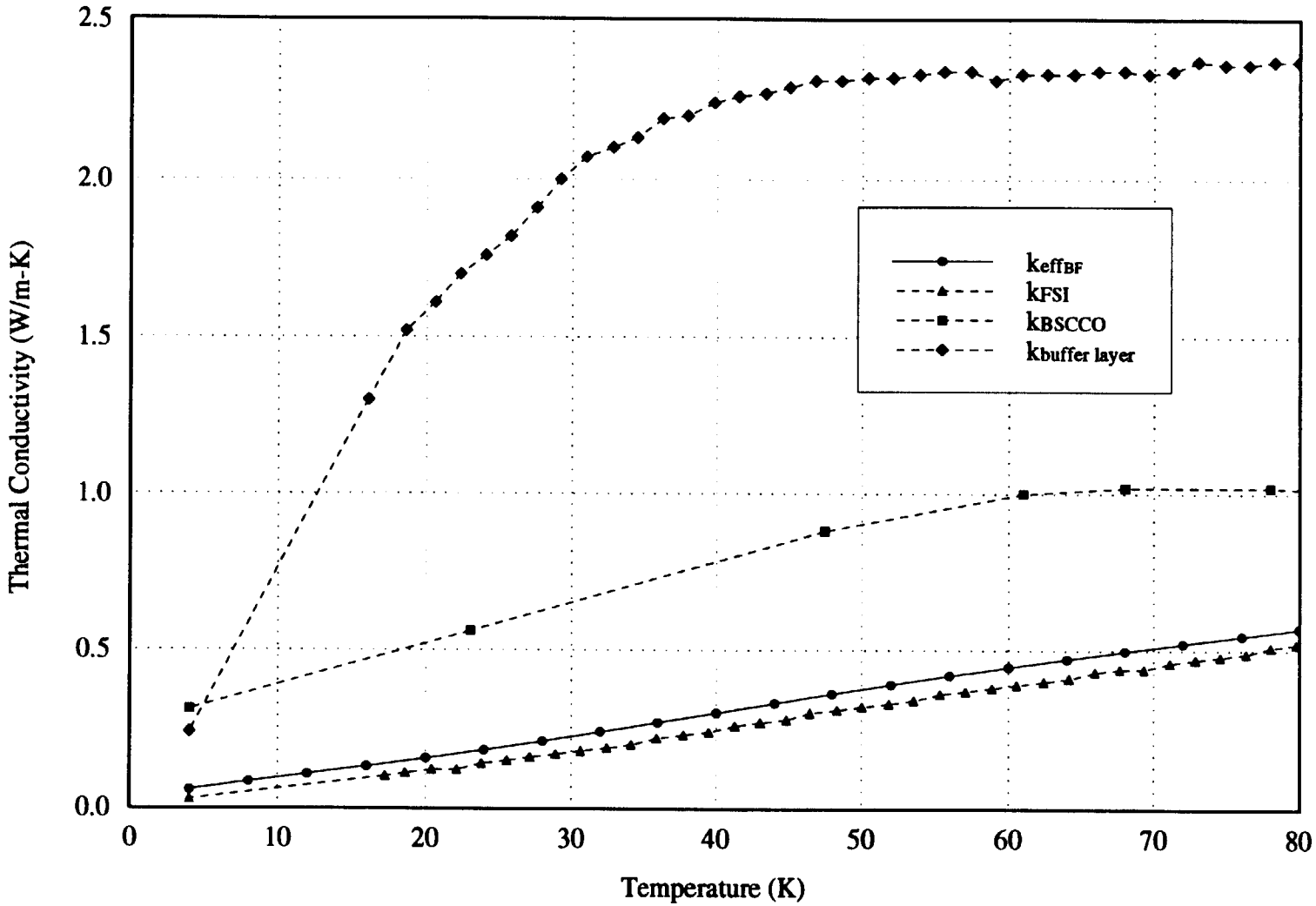


Figure 4.1.2. Material and Effective Thermal Conductivities for the Thermal Bridge BSCCO/FSI.

bridges are provided in Appendix E.

The effective thermal conductivity of the HTS thermal bridges is a useful tool to quantify the conductive heat transfer through the thermal bridges. Indeed, the determination of the effective thermal conductivity allows us to apply the one-dimensional conduction equation within the entire thermal bridge and to write

$$q'' = k_{eff}(T)\frac{dT}{dx} , \tag{4.1.7}$$

where q'' is the input heat flux at the warm end of the thermal bridge given by Lee (1994), and T is the temperature distribution along the thermal bridge (recall that the temperature has been shown to be one-dimensional). Integrating Eq. (4.1.7),

$$q''\int_0^x d\xi = \int_{4K}^{T} k_{eff}(\xi)d\xi , \tag{4.1.8}$$

and then integrating Eq. (4.1.8), gives

$$q''x = \beta_1(T-4)+\frac{\beta_2}{2}(T^2-4^2)+\frac{\beta_3}{3}(T^3-4^3)+\frac{\beta_4}{4}(T^4-4^4) . \tag{4.1.9}$$

Finally, rearranging Eq. (4.1.9),

$$\frac{\beta_4}{4}T^4+\frac{\beta_3}{3}T^3+\frac{\beta_2}{2}T^2+\beta_1 T = q''x+4\beta_1+8\beta_2+\frac{4^3}{3}\beta_3+64\beta_1 . \tag{4.1.10}$$

Equation (4.1.10), which governs the temperature distribution along the thermal bridges, is in the form $f(T) = q''x+C$, where f is a polynomial function of degree three, and C is a constant. Because of the physics of this problem, only one root is possible for the temperature in the range [4-80 K]. The temperature distribution can be solved for easily

applying the bisection method. This method uses the intermediate-value theorem of continuous functions and finds the root of a continuous function, f, in an interval $[a,b]$. A one-dimensional numerical scheme based on the bisection method was used to obtain calculated temperatures in the effective thermal conductivity estimation.

It is important to point out that the temperature distributions obtained using both the material and the effective thermal conductivity models are in good agreement.

4.1.2 Estimation of the Thermal Conductivities

The method used to estimate the thermal conductivities described in the previous section is the modified Box-Kanemasu estimation method. This method is a direct modification of the Gauss Linearization method (Beck and Arnold, 1977), and allows for nonlinearities in the model. It is based on the minimization of an objective function, the least squares function S, which can be expressed mathematically as

$$S = [\boldsymbol{Y}-\boldsymbol{T}(\underline{\beta})]^T[\boldsymbol{Y}-\boldsymbol{T}(\underline{\beta})] \ , \qquad (4.1.11)$$

where $\boldsymbol{Y}$ is the measured temperature vector, $\boldsymbol{T}(\underline{\beta})$ is the calculated temperature vector, and $\underline{\beta}$ is the exact parameter vector that contains the unknown thermal conductivity coefficients defined in Section 4.1.1. For the estimation of the thermal conductivities of the HTS thermal bridge materials, the calculated temperatures were obtained using the finite difference program ORTHO3D. Note that since the temperature distribution along the thermal bridges has been found in Section 3.2.4.2 to be one dimensional, either ORTHO3D or the original program CONDUCT (see Section 3.2.1) could have been used to calculate temperatures. For the case of the estimation of the effective thermal

conductivities, the calculated temperatures were determined from Eq. (4.1.10) using a one-dimensional numerical scheme based on the bisection method. The procedure to simulate measured temperatures is discussed later in Section 4.1.3.

In the estimation procedure, the least squares function, S, is minimized with respect to the unknown parameters, $\underline{\beta}$, resulting in

$$\nabla_{\underline{\beta}} S = 2[-\boldsymbol{X}^T(\underline{\beta})][\boldsymbol{Y}-\boldsymbol{T}(\underline{\beta})] = 0 , \tag{4.1.12}$$

where $\boldsymbol{X}(\underline{\beta})$ is the sensitivity coefficient matrix (Beck and Arnold, 1977), and is defined as

$$\boldsymbol{X}(\underline{\beta}) = [\nabla_{\underline{\beta}} \boldsymbol{T}^T(\underline{\beta})]^T . \tag{4.1.13}$$

The sensitivity coefficients are the derivatives of temperature with respect to the thermal conductivity coefficients being estimated. They represent the sensitivity of the temperature response to changes in the unknown parameters.

Because the conductive heat transfer in the HTS thermal bridges is a nonlinear problem, Eq. (4.1.12) cannot be explicitly solved for the parameter vector $\underline{\beta}$. Therefore, two approximations are used to linearize this equation. First, the sensitivity coefficient matrix, $\boldsymbol{X}(\underline{\beta})$, is replaced with $\boldsymbol{X}(\boldsymbol{b})$, where $\boldsymbol{b}$ is an estimate of $\underline{\beta}$; then the vector of calculated temperatures, $\boldsymbol{T}(\underline{\beta})$, is approximated by using the first two terms of a Taylor series of $\boldsymbol{T}(\underline{\beta})$ about $\boldsymbol{b}$, which gives

$$\boldsymbol{T}(\underline{\beta}) \approx \boldsymbol{T}(\boldsymbol{b})+[\nabla_{\underline{\beta}} \boldsymbol{T}^T(\boldsymbol{b})]^T(\underline{\beta}-\boldsymbol{b}) . \tag{4.1.14}$$

Equation (4.1.12) then reduces to

$$\underline{\beta} = \boldsymbol{b}+\boldsymbol{P}(\boldsymbol{b})[\boldsymbol{X}^T(\boldsymbol{b}) \ (\boldsymbol{Y}-\boldsymbol{T}(\boldsymbol{b}))] , \tag{4.1.15}$$

where the vector $\boldsymbol{P}(\boldsymbol{b})$ is defined as

$$P(b) = [X^T(b)X(b)]^{-1} . \tag{4.1.16}$$

Implementing an iterative scheme as described by Beck and Arnold (1977), the estimated parameter vector b can be eventually derived, yielding

$$b^{(k+1)} = b^{(k)} + P^{(k)}[X^{T(k)}(Y - T^{(k)})] , \tag{4.1.17}$$

where the subscript k is the iteration number. In this iterative process, an initial estimate $b^{(0)}$ is required. Equation (4.1.17) is known as the Gauss linearization equation.

To eliminate oscillations and nonconvergence which can sometimes occur for nonlinear problems, the Box-Kanemasu method incorporates a scalar interpolation factor, h, in the direction of the parameter variation. Equation (4.1.17) becomes

$$b^{(k+1)} = b^{(k)} + h^{(k+1)}\Delta_g b^{(k)} , \tag{4.1.18}$$

where the vector $\Delta_g b^{(k)}$ is expressed as

$$\Delta_g b^{(k)} = P^{(k)}[X^{T(k)}(Y - T^{(k)})] . \tag{4.1.19}$$

At each iteration, the sum of squares, S, is approximated by a quadratic function in h, the scalar interpolation factor. The value for h is then calculated by minimizing this approximated form of S, giving

$$h^{(k+1)} = G^{(k)}\alpha^2[S_\alpha^{(k)} - S_o^{(k)} + 2G^{(k)}\alpha]^{-1} , \tag{4.1.20}$$

where the scalar G is defined by

$$G^{(k)} = [\Delta_g b^{(k)}]^T(X^{T(k)}X^{(k)})[\Delta_g b^{(k)}] . \tag{4.1.21}$$

The value of the parameter α is initially set equal to one; $S_\alpha^{(k)}$ and $S_o^{(k)}$ are the values of S at α and zero, respectively. Note that the modified Box-Kanemasu method includes a check to ensure the continuous decrease of S from one iteration to another. This is done by reducing α by one-half if $S_\alpha^{(k)}$ is not less than $S_o^{(k)}$. Figure 4.1.3 presents a flowchart

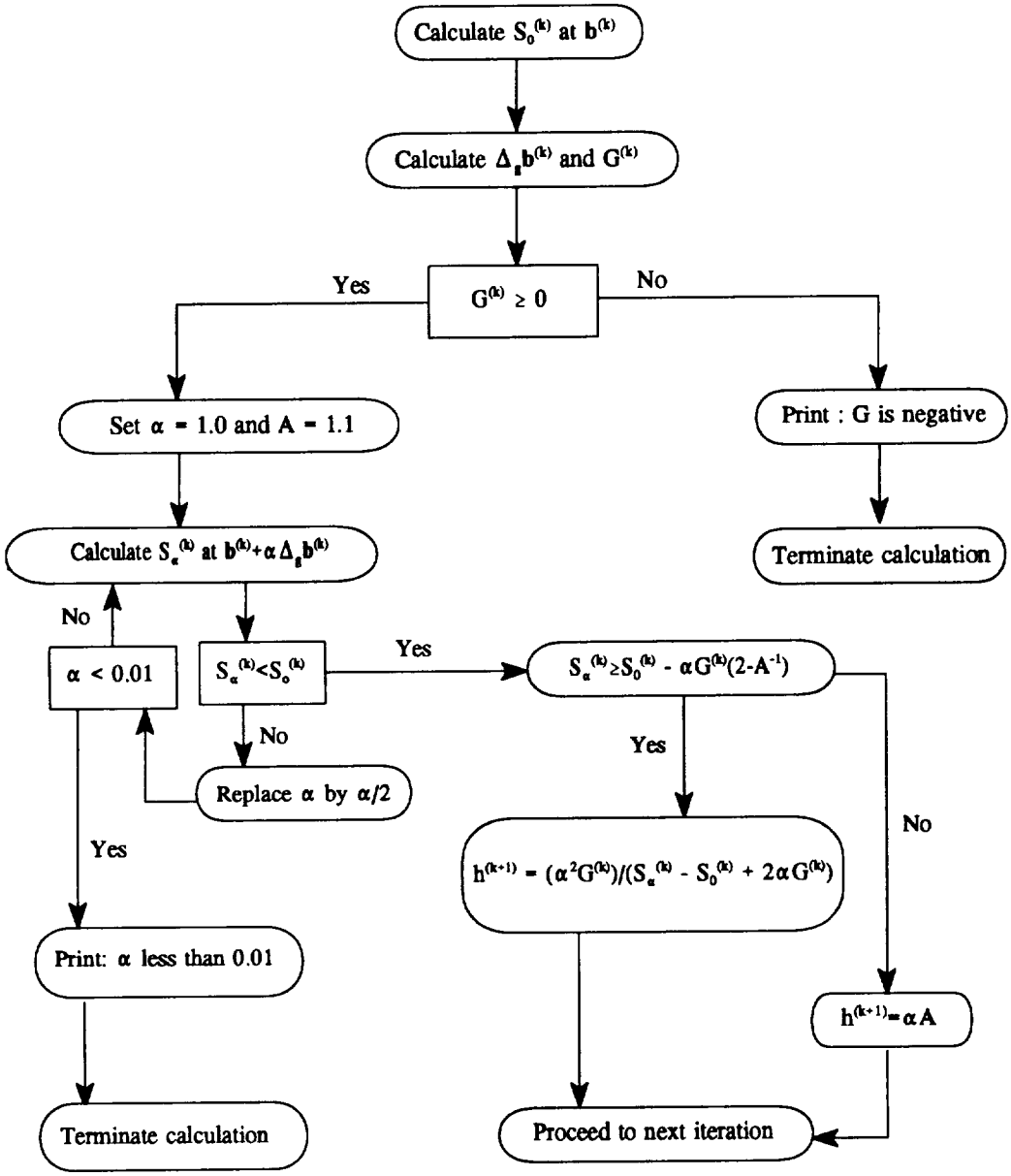


Figure 4.1.3. Flow Chart for the Modified Box-Kanemasu Estimation Procedure (Moncman, 1994).

illustrating the modified Box-Kanemasu method.

The estimation procedure can sometimes encounter an unstable behavior resulting in α less than 0.01 which terminates the calculations, or simply nonconvergence of the estimates. This could be the effect of near-linear dependence of the parameter sensitivity coefficients and/or very poor initial parameter estimates. The importance of a sensitivity analysis prior to the estimation procedure is detailed in Section 4.1.4.

To estimate the thermal conductivity parameters of the HTS thermal bridge materials, the subroutine KBOX3D.FOR (Appendix F) was written as the *adapt* subroutine of the program ORTHO3D. For the estimation of the effective thermal conductivity parameters of the HTS thermal bridges, a parameter estimation program called KBOXEFF.FOR (Appendix G) was written using the bisection method to solve Eq. (4.1.10) which governs the temperature distribution along the thermal bridges. Both KBOX3D.FOR and KBOXEFF.FOR use the modified Box-Kanemasu method.

It should be noted that, when the temperature solution is governed by a transient mathematical model, the concept of sequential estimation (Beck and Arnold, 1977), in which the parameters are evaluated at each time step, is generally utilized in the parameter estimation procedure. The advantage of applying this sequential estimation technique to transient models is that it allows the user to observe the effects of additional data on the sequential estimates and to evaluate the adequacy of the experimental design. Ideally, at the conclusion of an experiment, any additional data should not affect the parameter estimates. This concept was not of interest in this research since the heat conduction process within the thermal bridges is steady state.

4.1.3 Simulation of Measured Temperatures

The utilization of the modified Box-Kanemasu method for the estimation of the thermal conductivities requires experimental temperatures. As no experimental temperature measurements are available for this research project, simulated measurements must then be generated. This was performed by adding random errors with a known variance to the numerical temperatures obtained in the conduction analysis using the finite difference program ORTHO3D (Section 3.2.4.2). In doing this, temperature sensors are assumed to be placed along the HTS thermal bridges. However, in the proposed spaceflight experiment, only one temperature sensor is considered at each end of the thermal bridges (Section 3.1). Furthermore, since the temperature at the cold end is fixed at 4 K in the conductive mathematical models, temperature information for the parameter estimation can only be obtained from the sensor at the warm end of the thermal bridges. To meet this experimental design requirement, the parameter estimation procedure was also conducted using calculated and measured temperatures only at the warm end of the thermal bridges.

The program YI.FOR (Appendix H) was written to simulate temperature data both along the HTS thermal bridges and at the warm end of the thermal bridges. In this program, normally distributed random errors with standard deviations of 0.1 K, 0.5 K and 1.0 K were added to the temperature values obtained using ORTHO3D. Recall that these temperatures were obtained using the thermal conductivities of the materials in the conductive mathematical models (Section 3.2). Ten different sets of random errors were used for each standard deviation, which generated a total of thirty data sets.

4.1.4 Sensitivity Analysis

The parameter estimation procedure used in this research allows for the simultaneous estimation of the thermal conductivity parameters. However, if the sensitivity coefficients of these parameters are small, sufficient information might not be available to estimate the parameters. Furthermore, if the parameters are found to be correlated, they cannot be simultaneously estimated as independent values (Beck and Arnold, 1977). Indeed, correlation between the parameters induces the least squares function, S, to have no unique minimum, and therefore results in the nonconvergence of the minimization process. From these considerations, one understands the relevance in determining not only the magnitude of the parameters sensitivity coefficients, but also the degree of correlation between the various parameters prior to the estimation procedure. This is done through the analysis of the parameter sensitivity coefficients.

In the next subsections, the mathematical details inherent in the determination of the sensitivity coefficients of the thermal conductivity parameters are first presented. Then the methodology to carefully examine the sensitivity coefficients is discussed.

4.1.4.1 Determination of the Sensitivity Coefficients

As mentioned previously, the sensitivity coefficients represent the sensitivity of the temperature response to changes in the unknown parameters, namely the thermal conductivity parameters β. In the sensitivity study, it is meaningful to examine (Beck and Arnold, 1977)

$$(X_{\beta,T^+})^+ = \beta\left[\frac{\partial T^+}{\partial \beta}\right]_{\xi_i \neq \beta, const}, \tag{4.1.22}$$

where $(X_{\beta,T^+})^+$ is the dimensionless sensitivity coefficient of the parameter β, T^+ is the dimensionless temperature and $\xi_i \neq \beta$ are all parameters other than β that remain constant. Considering the one-dimensionality of the temperature distribution within the HTS thermal bridges, a convenient dimensionless temperature to be used is

$$T^+ = \frac{T - T_0}{T_L - T_0}, \tag{4.1.23}$$

where T_0 and T_L are the temperatures at the cold (x=0) and warm end (x=L) of the thermal bridges, respectively. Recall that T_0 is fixed at 4 K but T_L depends on the value of the parameter β. Because the sensitivity coefficients cannot be solved analytically, the term $\left[\frac{\partial T^+}{\partial \beta}\right]$ is approximated as

$$\frac{\partial T^+}{\partial \beta} \approx \frac{\Delta T^+}{\Delta \beta} = \frac{T^+(\beta+\Delta\beta) - T^+(\beta)}{\Delta \beta}, \tag{4.1.24}$$

where $T^+(\beta)$ and $T^+(\beta+\Delta\beta)$ are the dimensionless temperatures without and with the change $\Delta\beta$ in the parameter β, respectively. Note that $T^+(\beta)$ and $T^+(\beta+\Delta\beta)$ are nondimensionalized with respect to $T(\beta)$, that is

$$T^+(\beta) = \frac{T(\beta) - T_0(\beta)}{T_L(\beta) - T_0(\beta)}, \quad T^+(\beta+\Delta\beta) = \frac{T(\beta+\Delta\beta) - T_0(\beta)}{T_L(\beta) - T_0(\beta)}. \tag{4.1.25.a,b}$$

The value for each thermal conductivity coefficient, β, was obtained from the curve fits of the thermal conductivity plots, which are provided in Appendix E. The change in the parameter, $\Delta\beta$, was chosen to represent one percent of the value of β, which gives

$\Delta\beta=0.01\beta$. Equation (4.1.22) reduces then to

$$(X_{\beta,T^*})^* = \frac{1}{0.01}\left[\frac{T(\beta+\Delta\beta)-T(\beta)}{T_L(\beta)-T_0(\beta)}\right]. \tag{4.1.26}$$

The subroutine XI3D.FOR (Appendix I) was written as the *adapt* subroutine of the program ORTHO3D to compute the dimensionless sensitivity coefficients of the thermal conductivity parameters for the HTS thermal bridge materials. The sensitivity coefficients of the effective thermal conductivity parameters for the HTS thermal bridges were determined using the program XIEFF.FOR (Appendix J).

4.1.4.2 Methodology for the Sensitivity Analysis

The methodology used to examine the dimensionless sensitivity coefficients includes the analysis of both the magnitude of the sensitivity coefficients and the linear dependence between these coefficients. In case of near-linear dependence between the sensitivity coefficients, the correlation matrix should be computed to check for any correlation between the parameters to be estimated.

Small magnitudes ($<10^{-3}$) for dimensionless sensitivity coefficients indicate that the dimensionless temperature profile is insensitive to changes in a specific parameter, while large magnitudes (>1) represent extreme sensitivity to changes in a parameter (Scott, 1994). It should be noted that the limit 10^{-3} is generally representative of the limiting sensitivity in a given variable due to a change in a parameter. However, this limit is flexible, especially in cases where near-linear dependence between the sensitivity coefficients exists.

The magnitude of the sensitivity coefficients can also be interpreted as the amount of information about the value of the parameter available from the temperature measurement data. Indeed, parameters estimated from data with large sensitivity coefficients are generally more accurate than parameters estimated from data with small sensitivity coefficients (Scott, 1994).

The second step in the careful examination of the sensitivity coefficients is to consider the possibility of linear dependence between the sensitivity coefficients. The initial step to investigate linear dependence is to simply plot the sensitivity coefficients against each other. If the sensitivity coefficients appear to be linearly dependent, the corresponding parameters are correlated and cannot be estimated simultaneously (Beck and Arnold, 1977). Again a comment is required here to point out that even if the sensitivity coefficients are not linearly dependent over the entire range of temperatures investigated (if the temperature is the given variable of interest), near-linear dependency can sometimes occur in the temperature range of interest and result in inaccurate parameter estimates.

The next step in the present methodology is to compute the correlation matrix according to Beck and Arnold (1977) for the uncorrelated parameters determined by the plots of the sensitivity coefficients. This is especially important when a plot of the sensitivity coefficients is inconclusive. The diagonal terms of the correlation matrix are all unity and the off-diagonal terms must be in the interval [-1,1]. Whenever all the off-diagonal terms exceed 0.9 in magnitude, the estimates are highly correlated and tend to be inaccurate. One reason this could occur is that near-linear dependence between the

sensitivity coefficients exits, causing the parameters to be correlated. Note that the calculation of the correlation matrix should always be the final step in the investigation of linear dependence between the sensitivity coefficients. Indeed, viewing the sensitivity coefficients against each other in an initial step gives insight on the proportionality of these coefficients over the entire range of temperatures investigated, whereas the off-diagonal of the correlation matrix provides only an overall number. The subroutine KBOX3D.FOR and the program KBOXEFF.FOR, both described in Section 4.1.2, were used to compute the correlation matrix for the estimation of the thermal conductivity coefficients of the HTS thermal bridge materials and the estimation of the effective thermal conductivity coefficients of the HTS thermal bridges, respectively.

The analysis of the sensitivity coefficient magnitude and linear dependence should be concluded with the determination of which parameters are to be estimated. On the basis of the previous developments, uncorrelated parameters with the highest sensitivity coefficients should be chosen to be estimated.

4.2 Results and Discussion

The results of the analysis for the estimation of both the HTS thermal bridge material thermal conductivities and the HTS thermal bridge effective thermal conductivities are presented and discussed in this section. Recall that these thermal conductivities are modeled as polynomials in temperature, as described in Section 4.1.1, to account for temperature dependence. In both models, correlations between the polynomial parameters

are carefully examined prior to the estimation procedure. This is done through the sensitivity analysis using the methodology detailed in Section 4.1.4.2. As both thermal conductivity models display correlations between the parameters, simultaneous estimation of the thermal conductivity parameters was concluded to be impossible. Therefore, the estimation of constant effective thermal conductivities was performed for the HTS thermal bridges.

The first subsection provides the results for the sensitivity analysis of the material thermal conductivity model. The results for the estimation of the effective thermal conductivities of the HTS thermal bridges are given in the next subsection. These results include the sensitivity analysis of the effective thermal conductivity model developed in Section 4.1.1.2, and the estimates obtained when studying the effective thermal conductivities as constants.

4.2.1 Estimation of the Thermal Conductivities of the HTS Thermal Bridge Materials

The material thermal conductivities are modeled as six degree polynomials in temperature, as described by Eq. (4.1.1). The estimation of the thermal conductivity, k_i, of a specific material i in a HTS thermal bridge involves the simultaneous estimation of the polynomial parameters β_{ij} (j=1,7). This parameter estimation problem becomes complex as the HTS thermal bridges comprised of two, or three with the presence of a buffer layer, materials. Before implementing the parameter estimation, the sensitivity analysis of these multiple parameters is an imperative step to determine which parameters

can be estimated. This analysis is performed following the methodology described Section 4.1.4.2.

4.2.1.1 Sensitivity Analysis of the Polynomial Parameters of the Material Thermal Conductivities

Let us first study the HTS thermal bridge BSCCO/FSI which was shown in Section 3.2.4.2 to minimize the heat load on the cryogen. Using Eq. (4.1.1), the thermal conductivities of the substrate, k_{FSI}, the superconductor, k_{BSCCO}, and the buffer layer (zirconia), $k_{buffer\ layer}$, are respectively expressed as,

$$k_{FSI} = \beta_{f1}+\beta_{f2}T+\beta_{f3}T^2+\beta_{f4}T^3+\beta_{f5}T^4+\beta_{f6}T^5+\beta_{f7}T^6 \ , \tag{4.2.1}$$

$$k_{BSCCO} = \beta_{bs1}+\beta_{bs2}T+\beta_{bs3}T^2+\beta_{bs4}T^3+\beta_{bs5}T^4+\beta_{bs6}T^5+\beta_{bs7}T^6 \ , \tag{4.2.2}$$

$$k_{bufferlayer} = \beta_{z1}+\beta_{z2}T+\beta_{z3}T^2+\beta_{z4}T^3+\beta_{z5}T^4+\beta_{z6}T^5+\beta_{z7}T^6 \ . \tag{4.2.3}$$

The dimensionless sensitivity coefficients of the 21 parameters β_{ij}, where the subscript i denotes either FSI, BSCCO, or buffer layer and the subscript j denotes the place of the parameter in the polynomial equation (j=1,7), are computed using Eq. (4.1.26) in the *adapt* subroutine XI3D.FOR of the program ORTHO3D. The nominal values of the parameters are provided in Appendix E. Following the methodology for the sensitivity analysis, the magnitude of the dimensionless sensitivity coefficients is first examined. Recall that in the magnitude investigation, the number 10^{-3} is generally representative of the limit between small and large magnitudes. Table 4.2.1 shows that the magnitude of the sensitivity coefficients of the parameters β_{z1} and β_{z7} is smaller than 10^{-3}, which indicates that the temperature response is insensitive to changes in these parameters.

Table 4.2.1. Orders of Magnitude of the Dimensionless Sensitivity Coefficients of the Material Thermal Conductivity Parameters for the Thermal Bridge BSCCO/FSI.

FSI	β_{f1}	β_{f2}	β_{f3}	β_{f4}	β_{f5}	β_{f6}	β_{f7}
$(X_{\beta f})^+$	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
Zirconia	β_{z1}	β_{z2}	β_{z3}	β_{z4}	β_{z5}	β_{z6}	β_{z7}
$(X_{\beta z})^+$	10^{-4}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-4}
BSCCO	β_{bs1}	β_{bs2}	β_{bs3}	β_{bs4}	β_{bs5}	β_{bs6}	β_{bs7}
$(X_{\beta bs})^+$	10^{-2}	10^{-1}	1	1	1	1	10^{-1}

Therefore, these parameters cannot be estimated. The smallest magnitude (10^{-3}) is obtained for the sensitivity coefficients $(X_{\beta zj})^+$, j=2,6. This result verifies the little influence of the buffer layer on the temperature solution of the thermal bridge BSCCO/FSI. Figure 4.2.1 displays the sensitivity coefficients with magnitudes equal to or larger than 10^{-3} along the thermal bridge. As one can see, the largest magnitude of the sensitivity coefficients is obtained at the end of the thermal bridge, where the temperature is the highest. Recall that the magnitude of the sensitivity coefficients can be interpreted as the amount of information about the value of the parameter available from the temperature measurement data. This behavior therefore stresses the importance of placing in the experimental design a temperature sensor at the warm end of the thermal bridge to acquire the most temperature information for the parameter estimation.

Figure 4.2.1 shows that the dimensionless sensitivity coefficients with magnitudes smaller than 10^{-1} (sensitivity coefficients of the parameters β_{f1}, β_{f7}, β_{zj} (j=2,6) and β_{bs1}) have a similar distribution compared to the other sensitivity coefficients with larger magnitude.

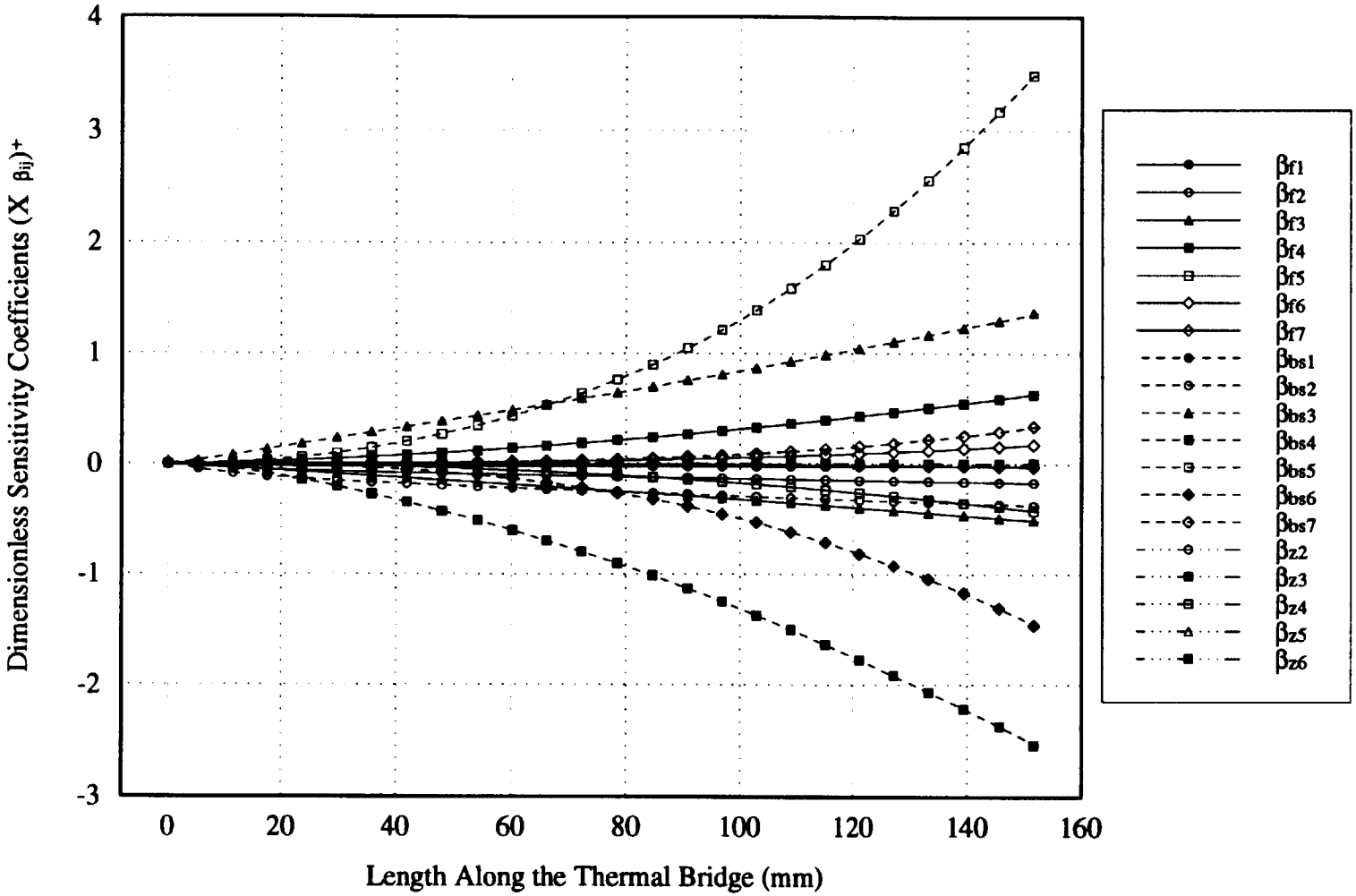


Figure 4.2.1. Dimensionless Sensitivity Coefficients (> 10^{-3}) of the Material Thermal Conductivity Parameters for BSCCO/FSI.

This indicates linear dependence (Beck and Arnold, 1977), and because of this, coupled with the rather low magnitudes, the limit 10^{-3} was reconsidered in this investigation, as mentioned in Section 4.1.2.2, and chosen as 10^{-1} instead. The dimensionless sensitivity coefficients with magnitude higher than 10^{-1} are plotted along the thermal bridge in Figure 4.2.2. There are four groups of sensitivity coefficients with similar behavior evident in this figure. The first group includes the sensitivity coefficients $(X_{\beta\text{-}f4,f6,bs5,bs7})^+$. This notation refers to the sensitivity coefficients of the parameters β_{f4}, β_{f6}, β_{bs5}, and β_{bs7}, respectively. The second group is comprised of the sensitivity coefficients $(X_{\beta\text{-}f5,bs4,bs6})^+$. The sensitivity coefficients $(X_{\beta\text{-}f2,bs2})^+$ and $(X_{\beta\text{-}f3,bs3})^+$ constitute the third and fourth group, respectively. The correlations between the sensitivity coefficients of these four groups are shown Figures 4.2.3.a-d. The determination of correlation between sensitivity coefficients from these plots is based on previous work on linear dependence between sensitivity coefficients (Beck and Arnold, 1977). For each group of correlated parameters, the parameter with the highest sensitivity coefficient is chosen to be analyzed, which gives the four parameters β_{bs2}, β_{bs3}, β_{bs4}, and β_{bs5}.

In the continuation of this sensitivity analysis, the dimensionless sensitivity coefficients of the four parameters mentioned above need to be plotted against each other to further investigate any linear dependence. This is done in Figure 4.2.4. Note that the use of the same scale when plotting the sensitivity coefficients against each other makes any linear dependence between the sensitivity coefficients clearer. The results are summarized in Table 4.2.2. A (+) sign indicates that linear dependence, or near-linear dependence, is found and a (-) sign indicates that no linear dependence is evident. It

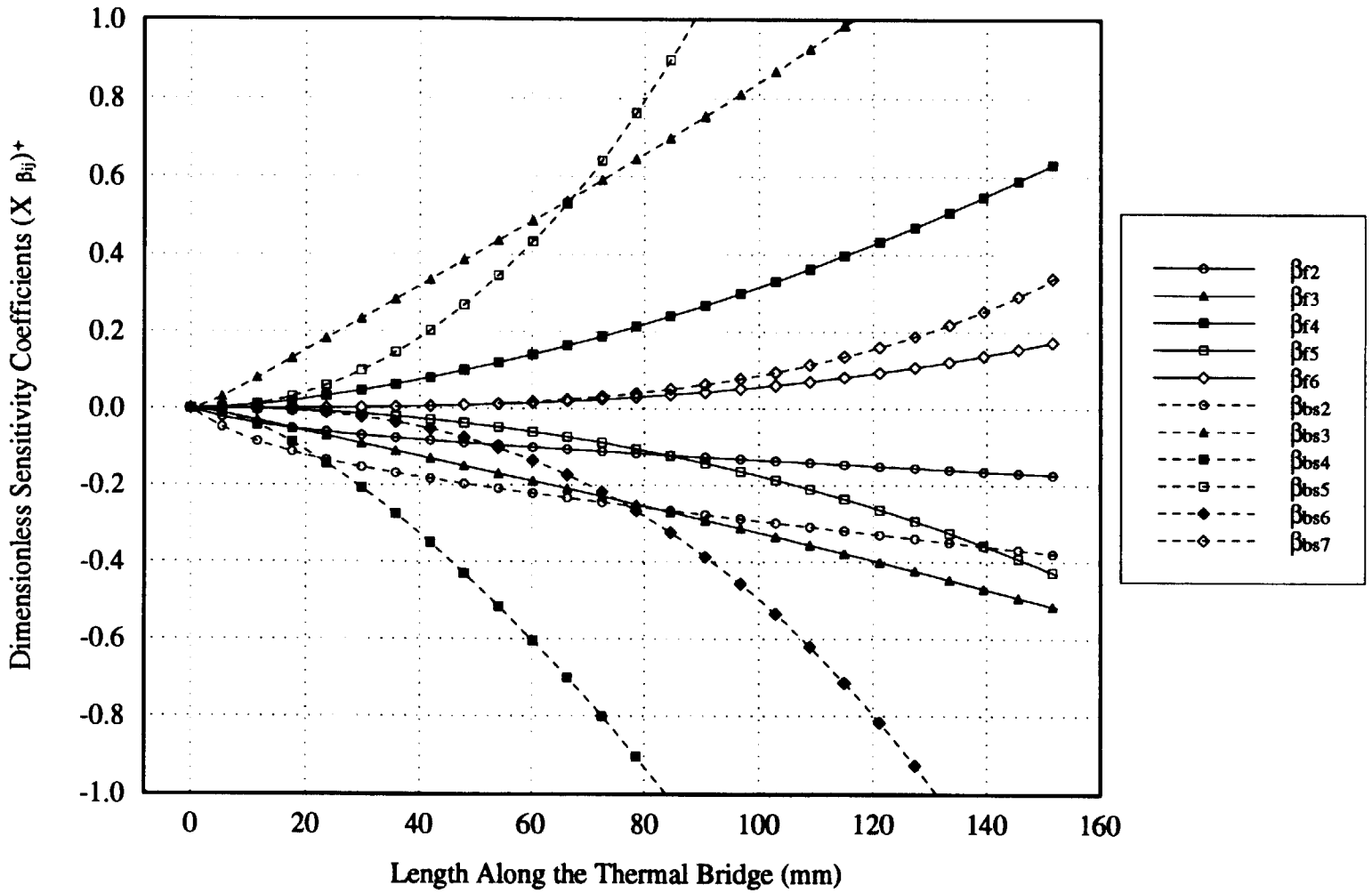


Figure 4.2.2. Dimensionless Sensitivity Coefficients (> 10^{-1}) of the Material Thermal Conductivity Parameters for BSCCO/FSI.

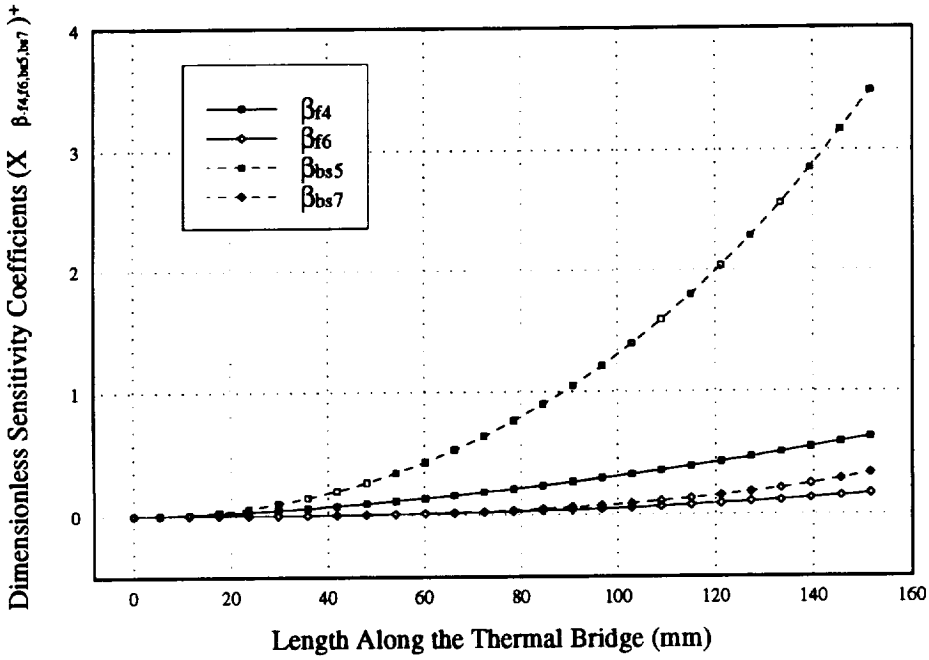


Figure 4.2.3.a. Similar Behavior of the Dimensionless Sensitivity Coefficients $(X_{\beta\text{-}f4,f6,bs5,bs7})^{+}$.

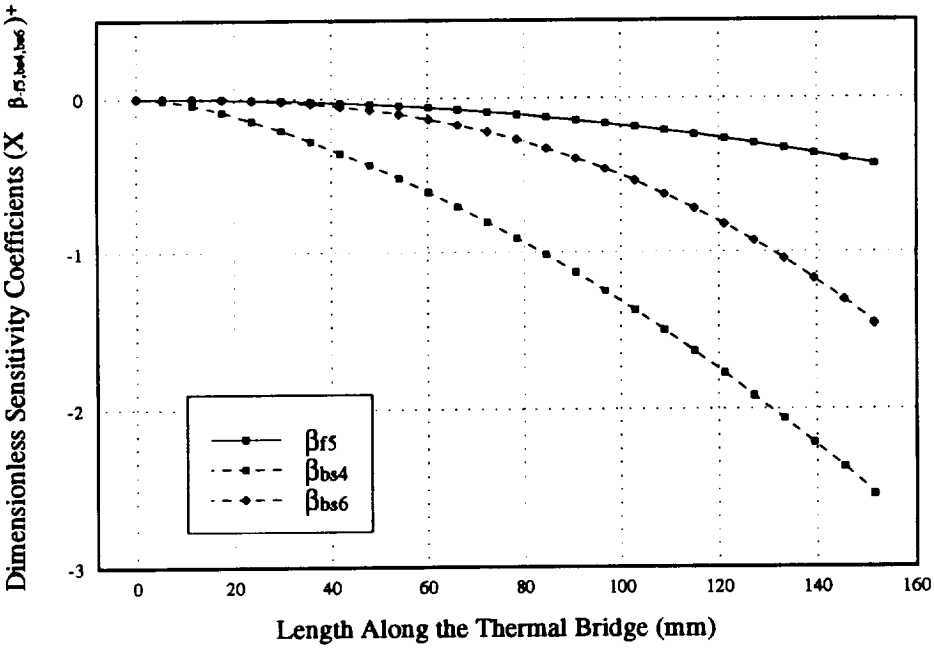


Figure 4.2.3.b. Similar Behavior of the Dimensionless Sensitivity Coefficients $(X_{\beta\text{-}f5,bs4,bs6})^{+}$.

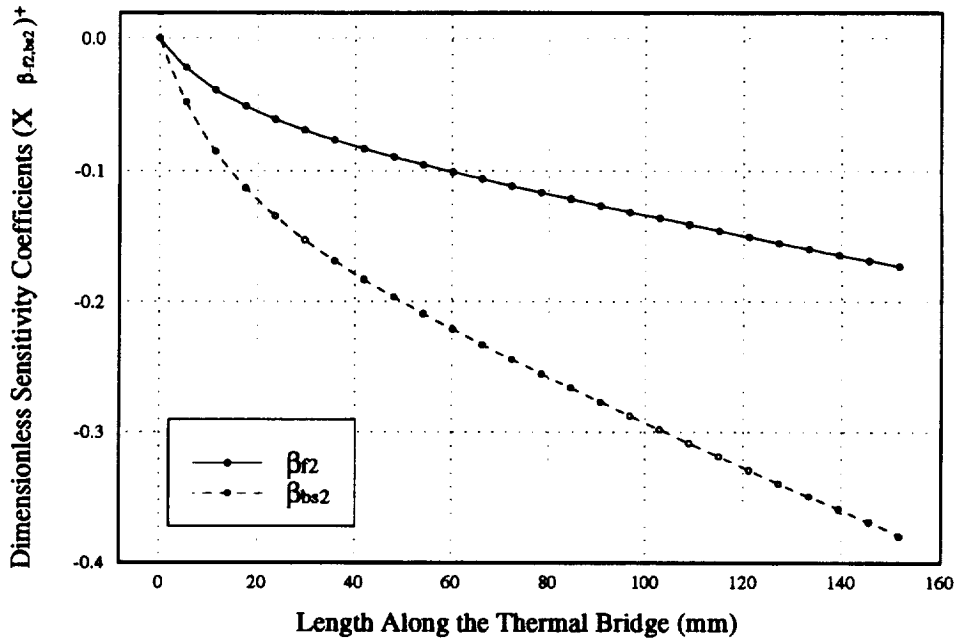


Figure 4.2.3.c. Similar Behavior of the Dimensionless Sensitivity Coefficients $(X_{\beta\text{-}f2,bs2})^{+}$.

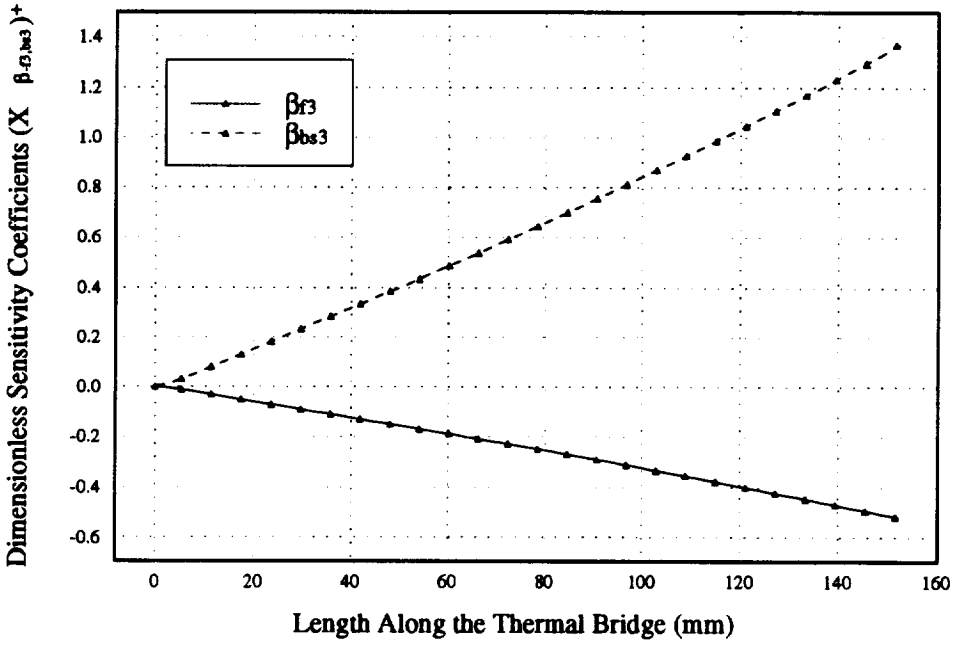


Figure 4.2.3.d. Similar Behavior of the Dimensionless Sensitivity Coefficients $(X_{\beta\text{-}f3,bs3})^{+}$.

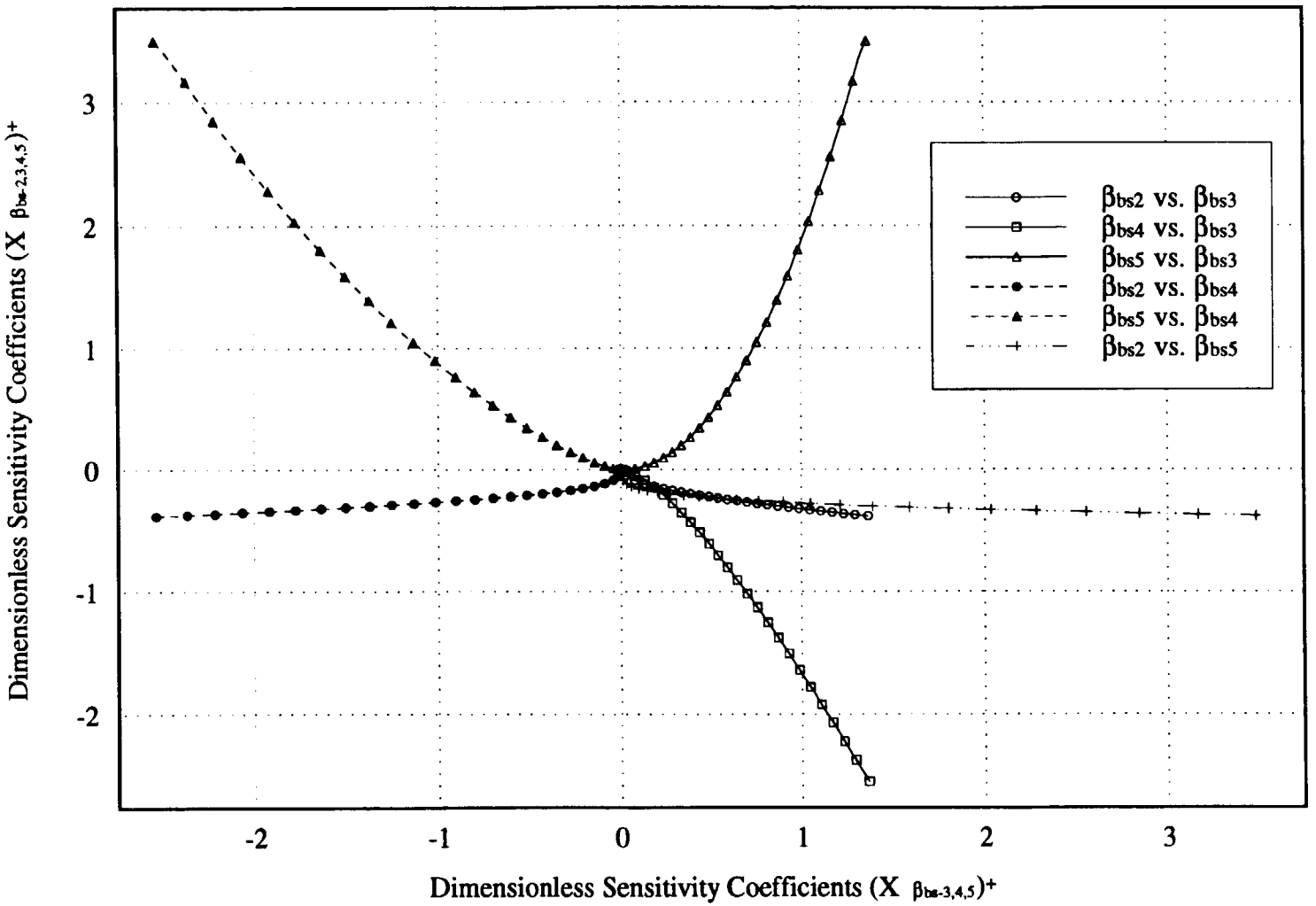


Figure 4.2.4. Linear Dependence Between the Dimensionless Sensitivity Coefficients ($X\beta_{bs-2,3,4,5}$)+.

Table 4.2.2. Linear Dependence of the Dimensionless Sensitivity Coefficients of the Material Thermal Conductivity Parameters β_{bs2}, β_{bs3}, β_{bs4}, and β_{bs5} for the Thermal Bridge BSCCO/FSI. ((+) indicates linear dependence; (-) indicates no linear dependence).

$(X_\beta)^+$	β_{bs2}	β_{bs3}	β_{bs4}	β_{bs5}
β_{bs2}	(+)	(+)	(+)	(+)
β_{bs3}	(+)	(+)	(+)	(-)
β_{bs4}	(+)	(+)	(+)	(-)
β_{bs5}	(+)	(-)	(-)	(+)

should be noted that if the sensitivity coefficients are linearly (or near-linearly) dependent over almost the entire range investigated except at the very beginning of the range (smallest value for the sensitivity coefficients), linear dependence was considered. Table 4.2.2 indicates that both simultaneous estimations of the parameters β_{bs3} with β_{bs5}, and β_{bs4} with β_{bs5} can be studied. As explained in the methodology for the sensitivity coefficient analysis, uncorrelated parameters with the highest sensitivity coefficients should be chosen. This results in the choice of the parameters β_{bs4} and β_{bs5}.

To conclude the present sensitivity analysis, the correlation matrix according to Beck and Arnold (1977) should be computed prior to the simultaneous estimation of the parameters β_{bs4} and β_{bs5}. This is of importance in this investigation because near-linear dependence could exist between the sensitivity coefficients $(X_{\beta bs4})^+$ and $(X_{\beta bs5})^+$, as seen in Figure 4.2.4. The computation of the correlation matrix was performed using the *adapt* subroutine KBOX3D.FOR of the program ORTHO3D. Ten sets of calculated and measured temperatures simulated with a standard deviation of 1.0 K along the thermal

bridge, were utilized. In the first simulated experiment, the off-diagonal term of the correlation matrix was found to be 0.987 which exceeded the limit 0.9 given by Beck and Arnold (1977). This result indicates that the estimates are highly correlated and tend to be inaccurate. Note, however, that the value of the estimates for the parameters β_{bs4} and β_{bs5} obtained for this first estimation were 1.241×10^{-4} and -2.096×10^{-6}, respectively, and were very close to the nominal values of these parameters (1.243×10^{-4} and -2.100×10^{-6}, for β_{bs4} and β_{bs5} respectively, provided in Appendix E).

In the other simulated experiments, the estimation procedure was terminated because the value of the variable α was less than 0.01 in the modified Box-Kanemasu method (see Figure 4.1.3). This stresses the instability of the simultaneous estimation of β_{bs4} and β_{bs5}. The correlation matrix of these experiments exhibited an off-diagonal term of about 0.988 which is slightly higher than in the first experiment. From these results, the simultaneous estimation of the parameters β_{bs4} and β_{bs5} was concluded to be impossible. Therefore, among the polynomial parameters described at the beginning of this analysis for the thermal bridge BSCCO/FSI, only one parameter of a specific material could be estimated.

The correlation between the parameters describing the thermal conductivities of the individual materials analyzed was expected. Indeed, it was improbable that the material thermal conductivities could have been distinguished because of the similarities of the temperature profiles when the substrate, the superconductor and the buffer layer thermal paths are considered separately. However, the correlation between the parameters with regards to the temperature was not evident. Therefore, the sensitivity analysis proved to be a useful tool in assessing the possibility of the parameters' simultaneous estimation.

Based on this analysis and on preliminary calculations, the same conclusion was drawn for the four other HTS thermal bridges. As the estimation of one thermal conductivity parameter for a specific material in a HTS thermal bridge was not of interest in this research, no further investigations were conducted in the study of the individual thermal conductivities of the HTS thermal bridge materials. It should be noted, however, that this study would have been pursued if the simultaneous estimation of at least two parameters, one for each material in a HTS thermal bridge, could have been performed (these uncorrelated parameters could have then been estimated as constants for the thermal conductivities of the corresponding materials).

4.2.2 Estimation of the Effective Thermal Conductivities of the HTS Thermal Bridges

The effective thermal conductivities of the HTS thermal bridges were modeled as third-degree polynomials in temperature, as described by Eq. (4.1.6). These models were defined from the material thermal conductivity models using the equivalent thermal circuit configurations in Section 4.1.1.2. Recall that the effective thermal conductivities were restricted to third-degree polynomials to account for only nonnegligible coefficients in the polynomial equations. This results in the analysis of four parameters β_{ij} (j=1,4) in the estimation of each HTS thermal bridge effective thermal conductivities k_{eff_i}. Proceeding similarly as in Section 4.2.1, the sensitivity analysis of the parameters β_{ij} is performed for each HTS thermal bridge prior to the implementation of the estimation procedure. Again, the four parameters describing the effective thermal conductivities as functions of

temperature were found to be correlated for each HTS thermal bridge. Therefore, the last option available to perform an estimation of the HTS thermal bridge thermal conductivities is to consider the effective thermal conductivities as constants.

In the first subsection, the results of the sensitivity coefficient analysis of the effective thermal conductivity polynomial model are provided for each HTS thermal bridge. The estimation of constant effective thermal conductivities for the HTS thermal bridges is presented and discussed in the next subsection.

4.2.2.1 Sensitivity Analysis of the Polynomial Parameters of the Effective Thermal Conductivities

The sensitivity analysis of the polynomial parameters of the effective thermal conductivities presented here is similar to that of the polynomial parameters of the material thermal conductivities conducted in the previous subsection. Once again, using the BSCCO/FSI thermal bridge as an example, the effective thermal conductivity $k_{eff_{BF}}$, is described by

$$k_{eff_{BF}} = \beta_{BF1}+\beta_{BF2}T+\beta_{BF3}T^2+\beta_{BF4}T^3 \ . \tag{4.2.4}$$

The dimensionless sensitivity coefficients of the four parameters β_{BFj} (j=1,4) are computed using Eq. (4.1.26) in the program XIEFF.FOR. The nominal values of the parameters are provided in Appendix E. Table 4.2.3 displays the magnitudes of the sensitivity coefficients, which are higher than the general limit 10^{-3}.

The sensitivity coefficients $(X_{\beta BFj})^+$ are plotted along the length of the thermal bridge in Figure 4.2.5. As one can see, the four sensitivity coefficients have a similar linear

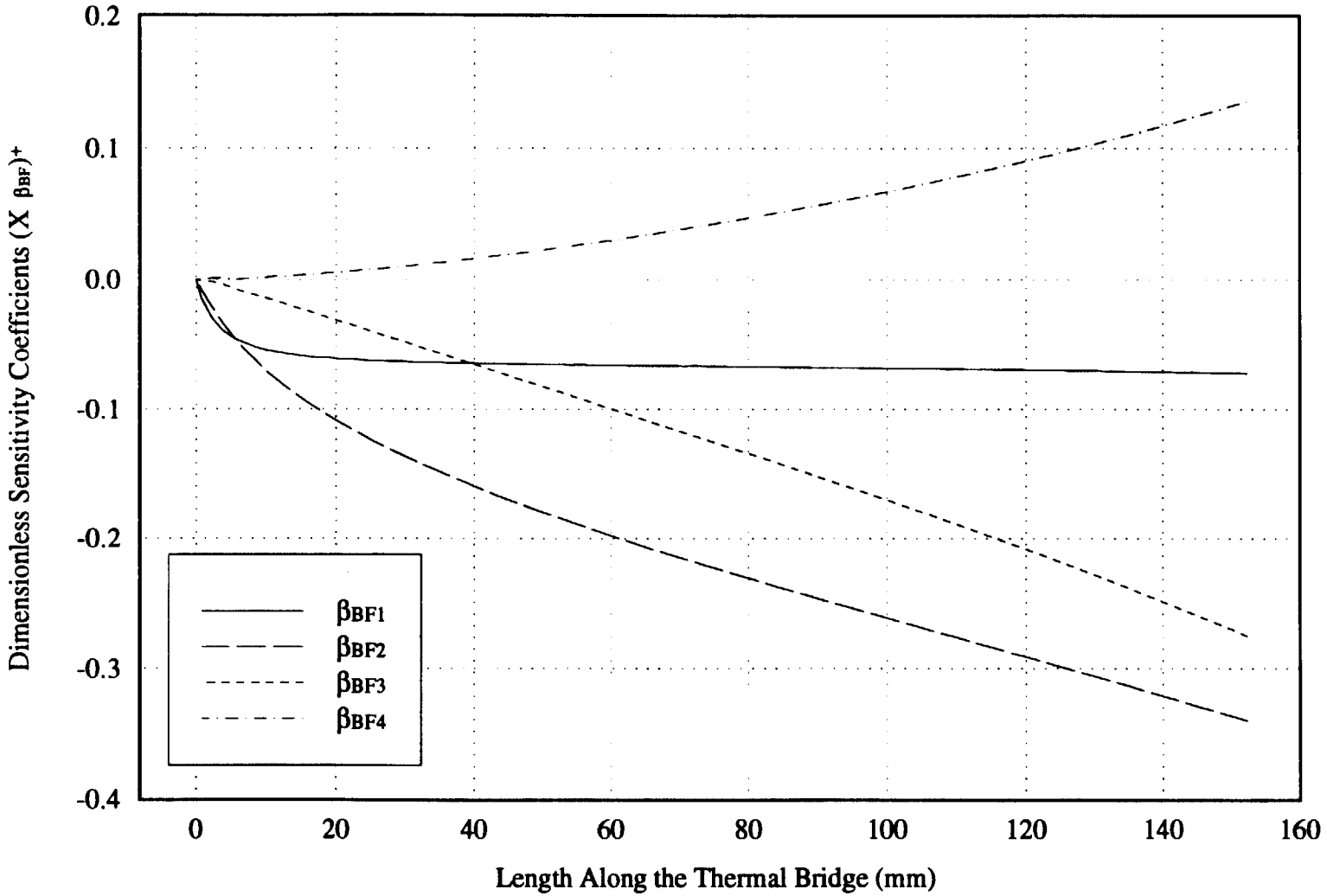


Figure 4.2.5. Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for BSCCO/FSI.

Table 4.2.3. Orders of Magnitude of the Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for the Thermal Bridge BSCCO/FSI.

BSCCO/FSI	β_{BF1}	β_{BF2}	β_{BF3}	β_{BF4}
$(X_{\beta BF})^+$	10^{-2}	10^{-1}	10^{-1}	10^{-1}

behavior over the last one-third of the thermal bridge where the sensitivity coefficients are the highest or, in other words, where the temperature provides the most information for the simultaneous estimation of the parameters β_{BFj}. This observation is of interest as previous work has shown that even though the parameters are not linearly dependent over the entire range investigated, near-linear dependency occurring in a fraction of the range can result in inaccurate parameter estimates and in instability of the estimation procedure (Saad, 1991).

In order to investigate linear dependence between the sensitivity coefficients, these are plotted against each other in Figure 4.2.6. Table 4.2.4 summarizes the results. The same notation as in Section 4.2.1 is used here. Also recall that if the sensitivity coefficients are linearly (or near-linearly) dependent over the entire range investigated except at the very beginning of the range, where the temperature provides the least information, linear dependence was considered. The similar linear behavior of the sensitivity coefficients over the last one-third of the thermal bridge, observed in Figure 4.2.5, can also be seen in Figure 4.2.6. As mentioned previously, this linear dependence between the sensitivity coefficients for the highest temperatures could result in the instability of the estimation procedure for the simultaneous estimation of the parameters,

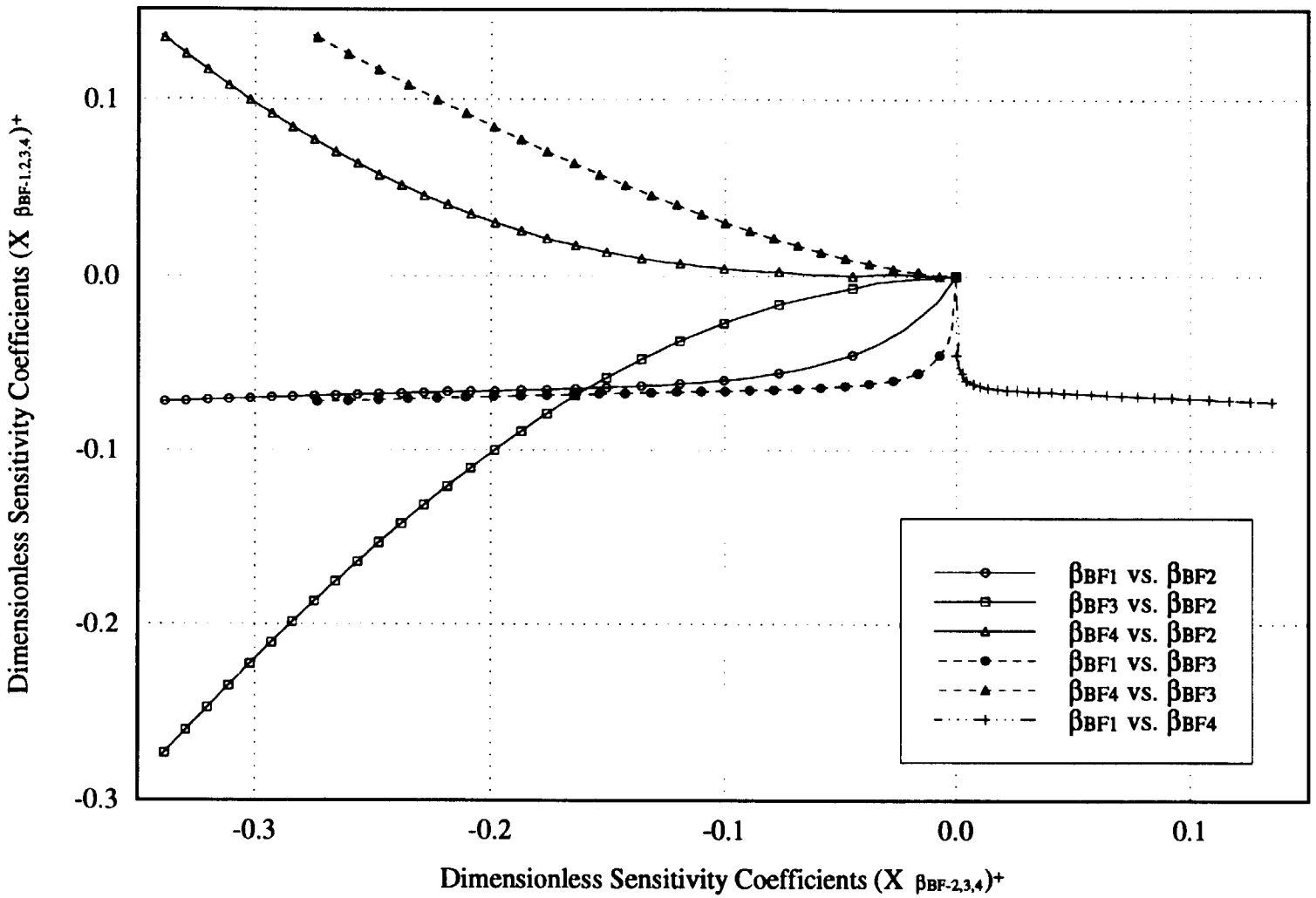


Figure 4.2.6. Linear Dependence Between the Dimensionless Sensitivity Coefficients ($X\,\beta_{BF\text{-}1,2,3,4}$)$^{+}$.

Table 4.2.4. Linear Dependence of the Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters β_{BF1}, β_{BF2}, β_{BF3}, and β_{BF4} for the Thermal Bridge BSCCO/FSI.

$(X_{\beta BF})^+$	β_{BF1}	β_{BF2}	β_{BF3}	β_{BF4}
β_{BF1}	(+)	(+)	(+)	(+)
β_{BF2}	(+)	(+)	(-)	(-)
β_{BF3}	(+)	(-)	(+)	(+)
β_{BF4}	(+)	(-)	(+)	(+)

even though they are not linearly dependent over the entire range investigated.

Table 4.2.4 indicates that both simultaneous estimations of the parameters β_{BF2} with β_{BF3}, and β_{BF2} with β_{BF4} can be studied as both pairs of parameters were found to be globally uncorrelated over the range investigated. The uncorrelated parameters with the highest sensitivity coefficients, β_{BF2} and β_{BF3}, were chosen for analysis.

The last phase to perform is to compute the correlation matrix according to Beck and Arnold (1977). This computation was realized using the program KBOXEFF.FOR. Ten sets of calculated and measured temperatures simulated with a standard deviation of 1.0 K along the thermal bridge, were utilized. The off-diagonal term of the correlation matrix was found to be about 0.981 for all experiments conducted. This value, larger than the limit 0.9 given by Beck and Arnold (1977), indicates that the estimates are highly correlated and tend to be inaccurate, as anticipated. Few experiments provided final estimates; most of the experiments performed were actually terminated because the value of the variable α became too small (<0.01) in the modified Box-Kanemasu method.

This analysis concluded with the impossibility of simultaneously estimating the polynomial parameters β_{BFj} (j=1,4), which describe the temperature dependence of the effective thermal conductivity of the thermal bridge BSCCO/FSI. Similar analysis concluding to the same result, were proceeded for the four other HTS thermal bridges. The results are recapitulated below.

The polynomial equations of the effective thermal conductivities $k_{eff_{BY}}$, $k_{eff_{YF}}$, $k_{eff_{YY}}$ and $k_{eff_{YG}}$ and the nominal values of the polynomial parameters β_{BYj}, β_{YFj}, β_{YYj} and β_{YGj} (j=1,4), are provided in Appendix E for the thermal bridges BSCCO/YSZ, YBCO/FSI, YBCO/YSZ AND YBCO/GREEN, respectively. Table 4.2.5 displays the magnitude of the dimensionless sensitivity coefficients of the effective thermal conductivity parameters for the four HTS thermal bridges. Figures 4.2.7.a to 4.2.10.a show the sensitivity coefficients larger than 10^{-3} in magnitude along the length of the thermal bridges. The sensitivity coefficients are plotted against each other in Figures 4.2.7.b to 4.2.10.b. Table 4.2.6 provides the results of the linear dependence investigation between the sensitivity coefficients of the effective thermal conductivity parameters for the four HTS thermal bridges. Eventually, the off-diagonal terms of the correlation matrix of the parameters chosen to be simultaneously estimated are recapitulated in Table 4.2.7. In the estimation procedure, instability was logically observed to increase as the off-diagonal term of the correlation matrix approaches one.

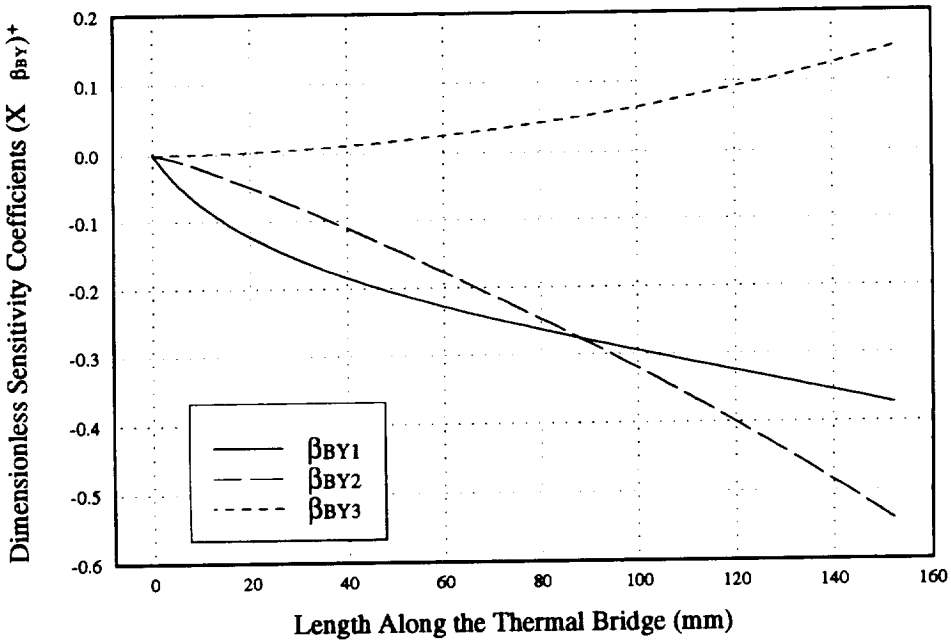


Figure 4.2.7.a. Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for BSCCO/YSZ.

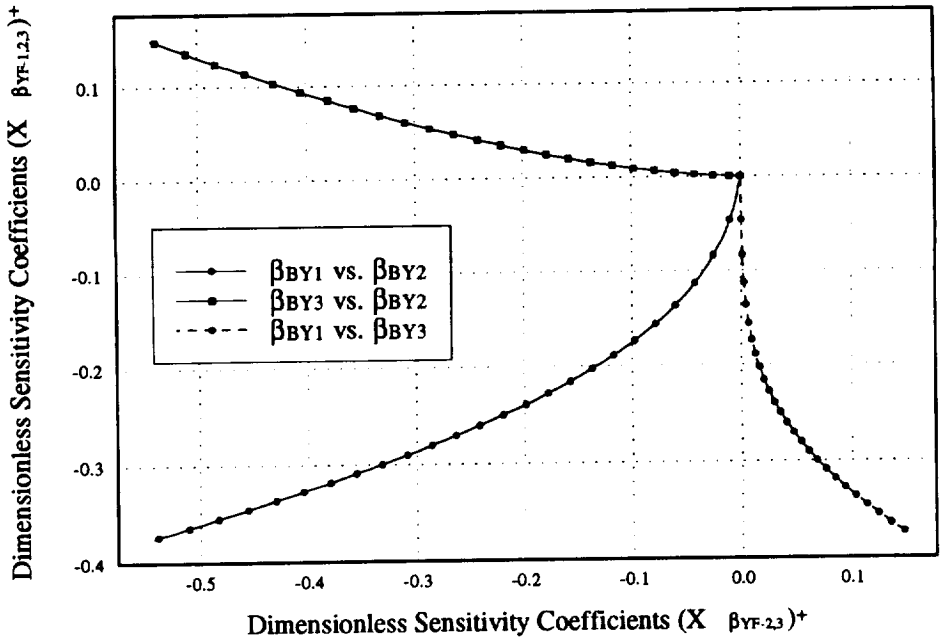


Figure 4.2.7.b. Linear Dependence Between the Dimensionless Sensitivity Coefficients $(X\,\beta_{BY\text{-}1,2,3})^{+}$.

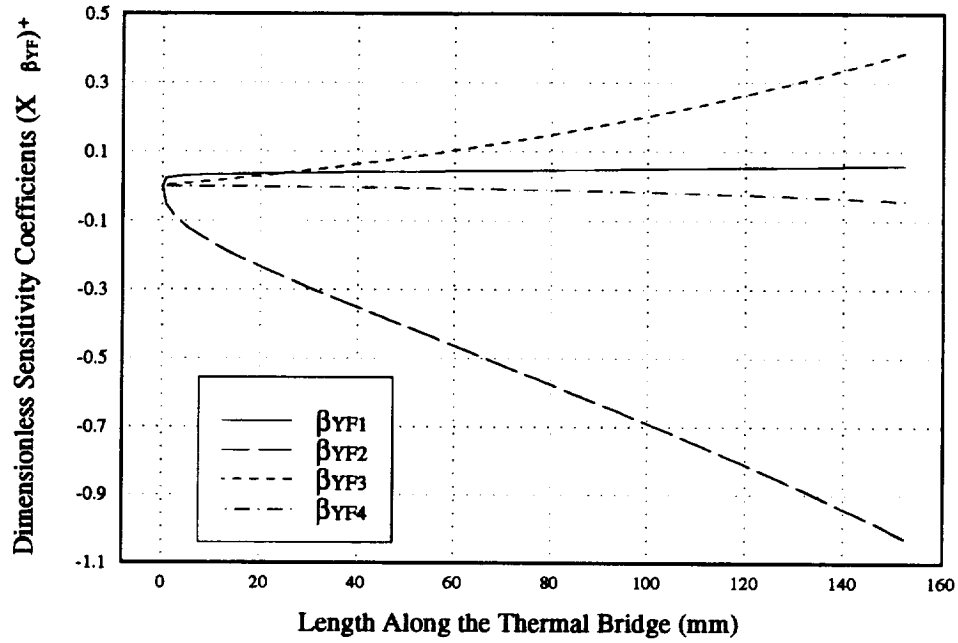


Figure 4.2.8.a. Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for YBCO/FSI.

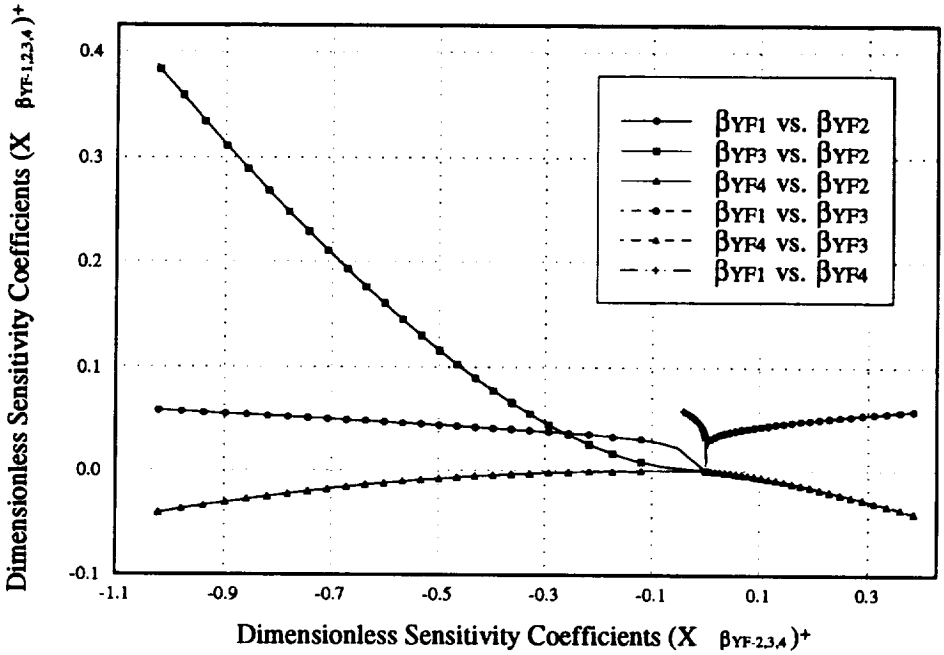


Figure 4.2.8.b. Linear Dependence Between the Dimensionless Sensitivity Coefficients $(X\ \beta_{YF\text{-}1,2,3,4})^{+}$.

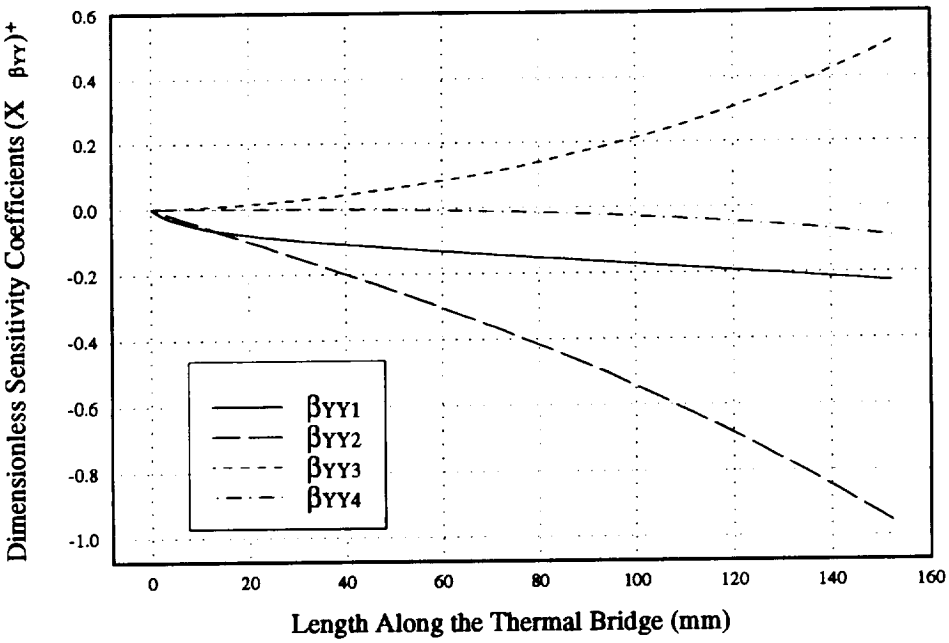


Figure 4.2.9.a. Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for YBCO/YSZ.

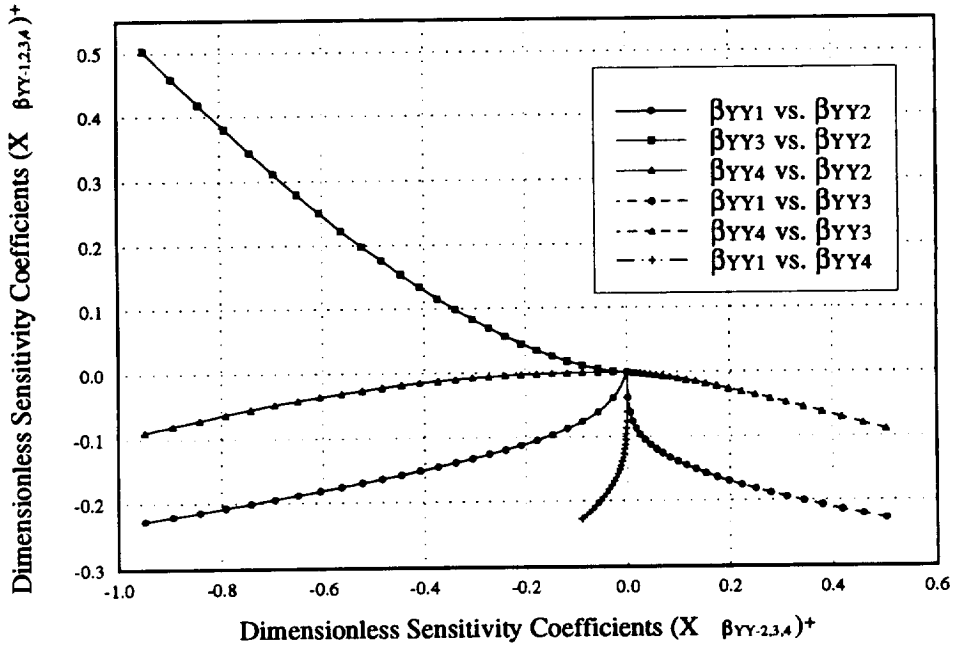


Figure 4.2.9.b. Linear Dependence Between the Dimensionless Sensitivity Coefficients ($X\,\beta_{YY\text{-}1,2,3,4}$)$^{+}$.

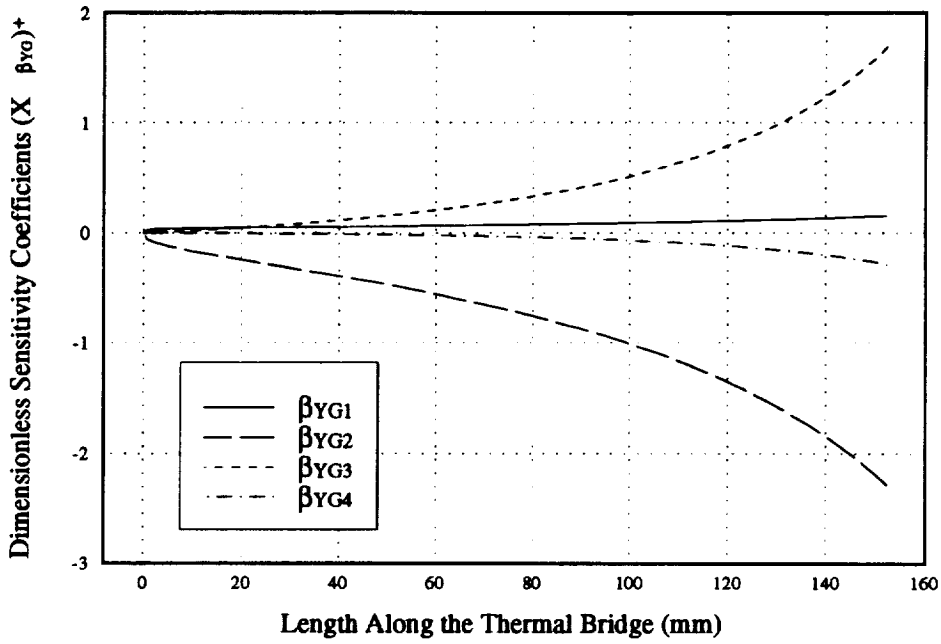


Figure 4.2.10.a. Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for YBCO/GREEN.

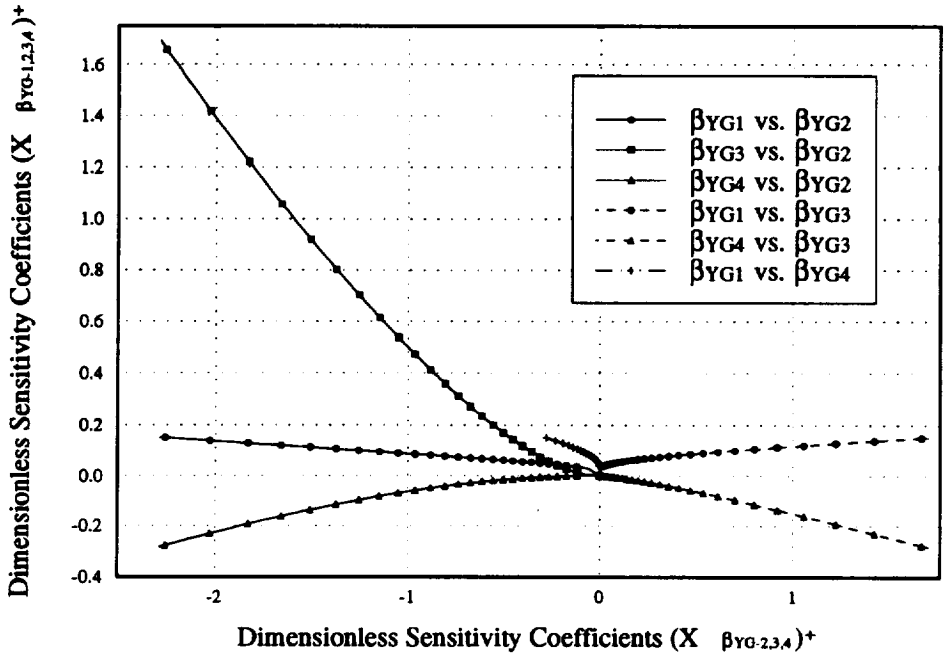


Figure 4.2.10.b. Linear Dependence Between the Dimensionless Sensitivity Coefficients ($X\beta_{YG-1,2,3,4}$)$^{+}$.

Table 4.2.5. Orders of Magnitude of the Dimensionless Sensitivity Coefficients of the Effective Thermal Conductivity Parameters for the Thermal Bridges BSCCO/YSZ, YBCO/FSI, YBCO/YSZ and YBCO/GREEN.

BSCCO/YSZ	β_{BY1}	β_{BY2}	β_{BY3}	β_{BY4}
$(X_{\beta BY})^+$	10^{-1}	10^{-1}	10^{-1}	10^{-4}
YBCO/FSI	β_{YF1}	β_{YF2}	β_{YF3}	β_{YF4}
$(X_{\beta YF})^+$	10^{-2}	1	10^{-1}	10^{-2}
YBCO/YSZ	β_{YY1}	β_{YY2}	β_{YY3}	β_{YY4}
$(X_{\beta YY})^+$	10^{-1}	10^{-1}	10^{-1}	10^{-2}
YBCO/GREEN	β_{YG1}	β_{YG2}	β_{YG3}	β_{YG4}
$(X_{\beta YG})^+$	10^{-1}	1	1	10^{-1}

Table 4.2.6. Linear Dependence of the Sensitivity Coefficients of the Effective Thermal Conductivity Parameters β_{BYj} (j=1,3), β_{YFj}, β_{YYj}, and β_{YGj} (j=1,4) for the Thermal Bridges BSCCO/YSZ, YBCO/FSI, YBCO/YSZ and YBCO/GREEN, respectively. ((+) indicates linear dependence; (-) indicates no linear dependence).

$(X_{\beta BY})^+$	$\underline{\beta_{BY1}}$	$\underline{\beta_{BY2}}$	$\underline{\beta_{BY3}}$	
β_{BY1}	(+)	(-)	(-)	
β_{BY2}	(-)	(+)	(-)	
β_{BY3}	(-)	(-)	(+)	
$(X_{\beta YF})^+$	β_{YF1}	$\underline{\beta_{YF2}}$	$\underline{\beta_{YF3}}$	β_{YF4}
β_{YF1}	(+)	(+)	(+)	(+)
β_{YF2}	(+)	(+)	(-)	(-)
β_{YF3}	(+)	(-)	(+)	(+)
β_{YF4}	(+)	(-)	(+)	(+)
$(X_{\beta YY})^+$	β_{YY1}	$\underline{\beta_{YY2}}$	$\underline{\beta_{YY3}}$	β_{YY4}
β_{YY1}	(+)	(+)	(+)	(-)
β_{YY2}	(+)	(+)	(-)	(-)
β_{YY3}	(+)	(-)	(+)	(+)
β_{YY4}	(-)	(-)	(+)	(+)
$(X_{\beta YG})^+$	β_{YG1}	$\underline{\beta_{YG2}}$	$\underline{\beta_{YG3}}$	β_{YG4}
β_{YG1}	(+)	(+)	(+)	(+)
β_{YG2}	(+)	(+)	(-)	(-)
β_{YG3}	(+)	(-)	(+)	(+)
β_{YG4}	(+)	(-)	(+)	(+)

*Note: the parameters underlined for each thermal bridge are the uncorrelated parameters chosen to be simultaneously estimated.

Table 4.2.7. Off-diagonal Terms of the Correlation Matrix of the Effective Thermal Conductivity Parameters Chosen to be Simultaneously Estimated for the Thermal Bridges BSCCO/YSZ, YBCO/FSI, YBCO/YSZ and YBCO/GREEN.

BSCCO/YSZ	β_{BY1}	β_{BY3}
β_{BY2}	0.977	0.987
β_{BY3}	0.932	
YBCO/FSI	β_{YF2}	
β_{YF3}	0.982	
YBCO/YSZ	β_{YY2}	
β_{YY3}	0.984	
YBCO/GREEN	β_{YG2}	
β_{YG3}	0.982	

4.2.2.2 Estimation of Constant Effective Thermal Conductivities for the HTS Thermal Bridges

As the effective thermal conductivities of the HTS thermal bridges could not be estimated as functions of temperature, the estimation of the effective thermal conductivities as constants was investigated. The constant effective thermal conductivities are estimated using the modified Box-Kanemasu method. Recall that in the parameter estimation procedure (Section 4.1.2), calculated and measured temperatures both along the thermal bridges and at only the warm end of the thermal bridges were to be utilized. The measured temperatures were simulated using the program YI.FOR described in Section 4.1.3. Two different estimates were therefore obtained depending on whether

temperatures along the thermal bridges or at the warm end of the thermal bridges were employed.

The first subsection is devoted to the mathematical description of the two different estimates obtained for the constant thermal conductivities of the HTS thermal bridges. The next subsection discusses the results obtained for these two estimates.

4.2.2.2.1 Mathematical Description of the Two Constant Effective Thermal Conductivities Estimated

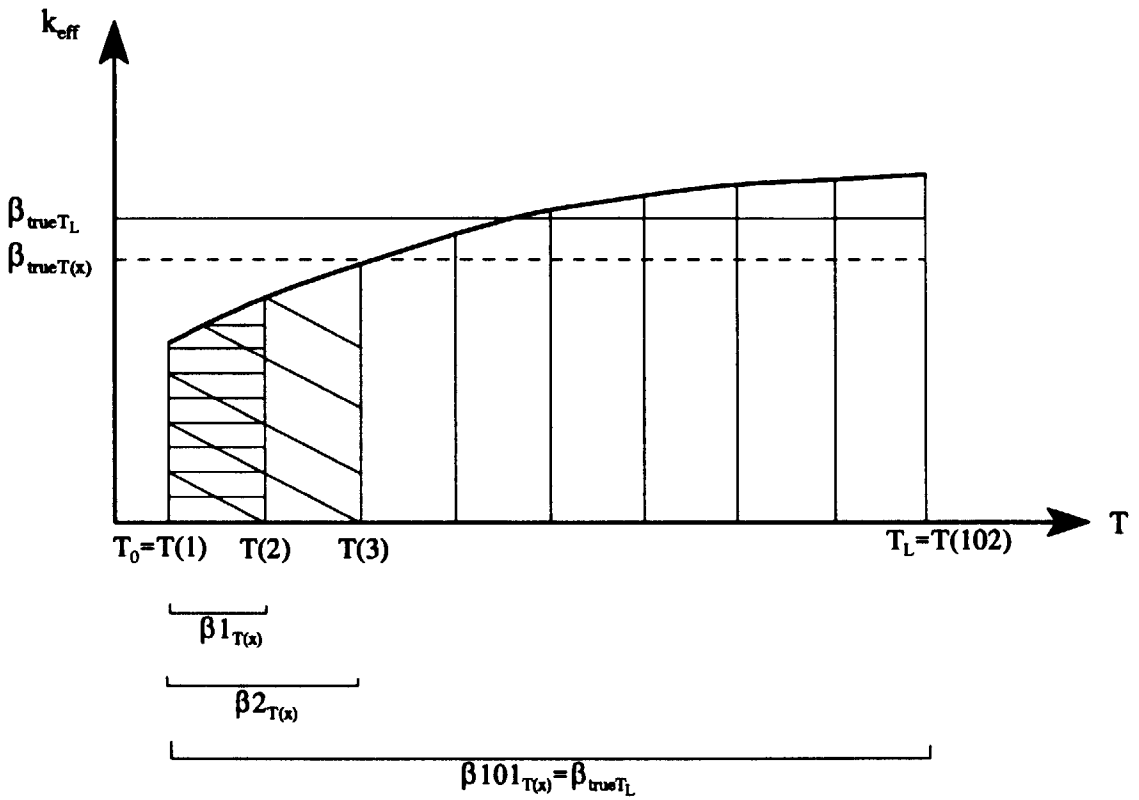
The procedure to estimate the effective thermal conductivities of the HTS thermal bridges as constants uses calculated and simulated measured temperatures both along the thermal bridges and at only the warm end of the thermal bridges, denoted by $T(x)$ and T_L, respectively. The utilization of temperatures at the warm end of the thermal bridges was specified to meet the experimental design requirement which includes two temperatures sensors at both end of the thermal bridges only. Two different estimates are therefore sought for the constant effective thermal conductivities, depending on whether $T(x)$ or T_L are used.

Let us describe mathematically what these two estimates represent. The estimate, β_{T_L}, calculated using T_L, represents the average value over the temperature range along the thermal bridge of the temperature-dependent effective thermal conductivity, $k_{eff}(T)$. Indeed, the final temperature T_L is obtained using $k_{eff}(T)$ over the entire temperature range along the thermal bridge. The true value, $\beta_{true T_L}$, for the estimate of a constant effective thermal conductivity calculated using T_L, can then be expressed as,

$$\beta_{true T_L} = \frac{\int_{T_0}^{T_L} k_{eff}(T)dT}{T_L - T_0}, \tag{4.2.5}$$

where T_0 is the temperature fixed at 4 K at the cold end of the thermal bridge. Figure 4.2.11 shows the true value $\beta_{true T_L}$ for a distribution $k_{eff}(T)$ which was chosen to generally characterize the profiles of the effective thermal conductivities of the HTS thermal bridges provided in Appendix E.

The mathematical description of the constant effective thermal conductivity estimated using $T(x)$, is more complex. In reality, this estimate, $\beta_{T(x)}$, represents a weighted average of $k_{eff}(T)$ in the temperature range $[T_0\text{-}T_L]$. In order to define the true value for $\beta_{T(x)}$, $\beta_{true T(x)}$, consider the discretization into several intervals of the distribution $k_{eff}(T)$ over the temperature range $[T_0\text{-}T_L]$ as shown in Figure 4.2.11. It is important to point out that the limits of these intervals must coincide with the numerical temperatures obtained using ORTHO3D in Section 3.2. This coincidence is required because in the estimation of $\beta_{T(x)}$, the measured temperatures are simulated by adding random errors to the temperature solution produced using ORTHO3D. In doing this curve discretization, recall that the geometric model of the HTS thermal bridges was discretized in a hundred control volumes along the length which provided 102 numerical values for the temperature distribution. Therefore the curve describing $k_{eff}(T)$ should be divided in 101 (102-1) intervals. The average value of $k_{eff}(T)$ over the first interval $[T_0\text{-}T(2)]$, is denoted by $\beta 1_{T(x)}$ as seen in Figure 4.2.11; the average value of $k_{eff}(T)$ over both the first and second intervals, that is on the range $[T_0\text{-}T(3)]$, is denoted by $\beta 2_{T(x)}$; eventually, $\beta 101_{T(x)}$ denotes



Note: T(i), $1 \le i \le 102$, are the numerical temperatures obtained using ORTHO3D.
Draw not to scale.

Figure 4.2.11. Description of the True Constant Effective Thermal Conductivity Estimates.

the average value of $k_{eff}(T)$ over the entire range investigated $[T_0\text{-}T_L]$. One can notice that $\beta 101_{T(x)}=\beta_{trueT_L}$. These different means $\beta i_{T(x)}$ (i=1,101), are expressed as,

$$\beta i_{T(x)} = \frac{\int_{T_0}^{T(i)} k_{eff}(T)dT}{T(i)-T_0} . \tag{4.2.6}$$

The true value, $\beta_{trueT(x)}$, for the estimate of a constant effective thermal conductivity calculated using $T(x)$, can now be defined as the weighted average of the different means $\beta i_{T(x)}$, and described as,

$$\beta_{trueT(x)} = \frac{\sum_{i=1}^{101} \beta i_{T(x)} Area(i)}{\sum_{i=1}^{101} Area(i)} , \tag{4.2.7}$$

where $Area(i)$ is the area under the curve $k_{eff}(T)$ over the temperature range $[T_0\text{-}T(i)]$.

As described in Figure 4.2.11, the value for $\beta_{trueT(x)}$ is expected to be less than the value for β_{trueT_L}, because all the means $\beta i_{T(x)}$ are less (or equal for $\beta 101_{T(x)}$) than β_{trueT_L}. This behavior should be obtained for the true estimates of the constant effective thermal conductivities of the HTS thermal bridges because the distribution $k_{eff}(T)$ in Figure 4.2.11 generally characterizes the profiles of the effective thermal conductivities of the HTS thermal bridges. The true estimates, $\beta_{trueT(x)}$, and β_{trueT_L}, were computed for each HTS thermal bridge using the program KEFF.FOR (Appendix K).

4.2.2.2.2 Constant Effective Thermal Conductivity Estimates

The results obtained for the estimated constant thermal conductivities of the HTS thermal bridges are presented and discussed in this subsection. These results include the constant thermal conductivities, $\beta_{T(x)}$ and β_{T_L}, estimated using temperatures along the thermal bridges and at only the warm end of the thermal bridges, respectively. Both thermal properties were estimated using the parameter estimation program KBOXEFF.FOR. The calculated temperatures, $T(x)$ and T_L, are governed by a simple one-dimensional equation resulting in,

$$T(x) = \frac{q''}{\beta_{T(x)}}x+4 \ . \tag{4.2.8}$$

The measured temperatures were simulated by adding normally distributed random errors with three different standard deviations (0.1, 0.5 and 1.0 K) to the temperature values obtained using ORTHO3D. Recall that 102 numerical values for the temperature distribution were computed along the thermal bridges. Therefore, in the case of temperature measurements along the thermal bridges, 102 data points could be simulated. Since no restriction exists for the temperature measurement number at the warm end of the thermal bridges in this case, 500 data were simulated for each experiment. Ten simulated experiments were performed for each standard deviation, which generated a total of thirty experiments for each case. This was performed using the program YI.FOR.

The estimates, $\beta_{T(x)}$ and β_{T_L}, obtained for each experiment are given in Tables 4.2.8 to 4.2.12 for the thermal bridges BSCCO/FSI, YBCO/FSI, BSCCO/YSZ, YBCO/YSZ and YBCO/GREEN, respectively. For each standard deviation of measurement errors, the

mean value of the estimates is provided, along with its 95-percent confidence interval. The 95-percent confidence intervals were calculated from

$$\bar{b}_i \pm \frac{t_{\alpha/2} s}{\sqrt{N_p}} , \tag{4.2.9}$$

where $\bar{b}_i$ and s are the mean and standard deviation of the estimate, respectively, N_p is the number of data points used, and $t_{\alpha/2}$ is the value of the t distribution with $(N_p\text{-}1)$ degrees of freedom and $\alpha/2$ confidence region (Walpole and Myers, 1978). Constant effective thermal conductivities were also estimated using the temperatures directly obtained from ORTHO3D for the measured temperatures, resulting in the estimates $\beta_{0T(x)}$ and β_{0T_L}. The temperatures obtained from ORTHO3D are called "exact" data because these temperatures are taken as reference in this research. The true estimates, $\beta_{trueT(x)}$ and β_{trueT_L}, described in the previous subsection, are supplied for each thermal bridge. Eventually, the percentage difference between both the mean value of each standard deviation and the estimate obtained using exact data, and between the estimate using exact data and the true estimate, is given.

In the analysis of Tables 4.2.8 to 4.2.12, general results can be commented for the five HTS thermal bridges. First, as anticipated in Section 4.2.2.2.1, larger values are obtained for the estimation of the constant effective thermal conductivities at the warm end of the thermal bridges; that is $\beta_{T_L} > \beta_{T(x)}$.

Second, for all experiments conducted, the estimates using exact data, $\beta_{0T(x)}$ and β_{0T_L}, fall within the 95-percent confidence intervals of the respective mean values. This result ensures that reasonable estimates have been obtained and allows for the validation of the

Table 4.2.8. Estimated Constant Effective Thermal Conductivities for the Thermal Bridge BSCCO/FSI.

Simulated Experiment	Standard Deviation of Measurement Errors (s)					
	0.1		0.5		1.0	
	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}
1	0.2696	0.3153	0.270	0.3154	0.2702	0.3155
2	0.2696	0.3153	0.2697	0.3152	0.2699	0.3151
3	0.2696	0.3153	0.2697	0.3153	0.270	0.3154
4	0.2695	0.3153	0.2696	0.3154	0.2697	0.3154
5	0.2696	0.3153	0.2697	0.3153	0.2698	0.3154
6	0.2695	0.3153	0.2693	0.3152	0.2690	0.3151
7	0.2695	0.3153	0.2694	0.3154	0.2693	0.3155
8	0.2696	0.3152	0.2698	0.3151	0.2700	0.3149
9	0.2695	0.3153	0.2696	0.3154	0.2697	0.3155
10	0.2695	0.3153	0.2695	0.3153	0.2695	0.3154
Mean	0.2695 ±2.6E-5	0.3153 ±1.5E-5	0.2696 ±1.3E-4	0.3153 ±7.3E-5	0.2697 ±2.6E-4	0.3153 ±1.5E-4
Difference (%) mean/exact data	0.007	0.001	0.033	0.007	0.067	0.013
Exact Data (s=0)	$\beta_{0T(x)}$	β_{0T_L}				
	0.2695	0.3153				
Difference (%) exact data/ true estimate	2.61	0.04				
True Estimate	$\beta_{trueT(x)}$	β_{trueT_L}				
	0.2627	0.3152				

Table 4.2.9. Estimated Constant Effective Thermal Conductivities for the Thermal Bridge YBCO/FSI.

Simulated Experiment	Standard Deviation of Measurement Errors (s)					
	0.1		0.5		1.0	
	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}
1	0.4564	0.5214	0.4568	0.5215	0.4574	0.5217
2	0.4563	0.5213	0.4566	0.5213	0.4570	0.5211
3	0.4563	0.5214	0.4566	0.5215	0.4569	0.5216
4	0.4563	0.5214	0.4564	0.5215	0.4566	0.5217
5	0.4563	0.5214	0.4565	0.5215	0.4568	0.5216
6	0.4562	0.5213	0.4558	0.5212	0.4553	0.5211
7	0.4562	0.5214	0.4561	0.5215	0.4559	0.5217
8	0.4564	0.5213	0.4567	0.5211	0.4572	0.5208
9	0.4563	0.5214	0.4564	0.5216	0.4565	0.5218
10	0.4563	0.5214	0.4563	0.5215	0.4563	0.5216
Mean	0.4563 ±4.5E-5	0.5214 ±2.4E-5	0.4564 ±2.3E-4	0.5214 ±1.2E-4	0.4566 ±4.5E-4	0.5215 ±2.4E-4
Difference (%) mean/exact data	0.007	0.001	0.033	0.007	0.067	0.013
Exact Data (s=0)	$\beta_{0T(x)}$	β_{0T_L}				
	0.4563	0.5214				
Difference (%) exact data/ true estimate	2.26	0.11				
True Estimate	$\beta_{trueT(x)}$	β_{trueT_L}				
	0.4462	0.5277				

Table 4.2.10. Estimated Constant Effective Thermal Conductivities for the Thermal Bridge BSCCO/YSZ.

Simulated Experiment	Standard Deviation of Measurement Errors (s)					
	0.1		0.5		1.0	
	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}
1	0.7046	0.7602	0.7053	0.7603	0.7062	0.7605
2	0.7045	0.7601	0.7050	0.7599	0.7055	0.7597
3	0.7045	0.7601	0.7049	0.7603	0.7055	0.7604
4	0.7044	0.7602	0.7047	0.7603	0.7049	0.7605
5	0.7045	0.7601	0.7048	0.7602	0.7052	0.7604
6	0.7042	0.7601	0.7036	0.7598	0.7028	0.7596
7	0.7043	0.7602	0.7040	0.7603	0.7037	0.7606
8	0.7045	0.7600	0.7051	0.7597	0.7058	0.7593
9	0.7044	0.7602	0.7046	0.7604	0.7047	0.7607
10	0.7044	0.7601	0.7044	0.7603	0.7044	0.7604
Mean	0.7044 ±7.3E-5	0.7601 ±3.5E-5	0.7046 ±3.7E-4	0.7602 ±1.8E-4	0.7049 ±7.3E-4	0.7602 ±3.5E-4
Difference (%) mean/exact data	0.007	0.001	0.033	0.007	0.067	0.013

Exact Data (s=0)	$\beta_{0T(x)}$	β_{0T_L}
	0.7044	0.7601
Difference (%) exact data/ true estimate	2.23	0.06
True Estimate	$\beta_{trueT(x)}$	β_{trueT_L}
	0.6890	0.7606

Table 4.2.11. Estimated Constant Effective Thermal Conductivities for the Thermal Bridge YBCO/YSZ.

Simulated Experiment	Standard Deviation of Measurement Errors (s)					
	0.1		0.5		1.0	
	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}
1	0.8919	0.9666	0.8928	0.9669	0.8940	0.9671
2	0.8918	0.9665	0.8924	0.9664	0.8931	0.9661
3	0.8918	0.9666	0.8923	0.9668	0.8930	0.9670
4	0.8917	0.9666	0.8920	0.9668	0.8923	0.9671
5	0.8918	0.9666	0.8922	0.9668	0.8927	0.9669
6	0.8914	0.9665	0.8907	0.9663	0.8897	0.9659
7	0.8916	0.9666	0.8912	0.9669	0.8908	0.9671
8	0.8918	0.9665	0.8926	0.9661	0.8935	0.9655
9	0.8917	0.9667	0.8919	0.9670	0.8921	0.9673
10	0.8916	0.9666	0.8917	0.9668	0.8917	0.9670
Mean	0.8917 ±9.3E-5	0.9666 ±4.5E-5	0.8920 ±4.6E-4	0.9667 ±2.2E-4	0.8923 ±9.3E-4	0.9667 ±4.5E-4
Difference (%) mean/exact data	0.007	0.001	0.033	0.007	0.067	0.013

Exact Data (s=0)	$\beta_{0T(x)}$	β_{0T_L}
	0.8917	0.9666
Difference (%) exact data/ true estimate	2.36	0.10
True Estimate	$\beta_{trueT(x)}$	β_{trueT_L}
	0.8710	0.9656

Table 4.2.12. Estimated Constant Effective Thermal Conductivities for the Thermal Bridge YBCO/GREEN.

Simulated Experiment	Standard Deviation of Measurement Errors (s)					
	0.1		0.5		1.0	
	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}	$\beta_{T(x)}$	β_{T_L}
1	4.5154	4.8459	4.5201	4.8470	4.5261	4.8484
2	4.5149	4.8454	4.5180	4.8445	4.5217	4.8433
3	4.5149	4.8459	4.5177	4.8466	4.5212	4.8475
4	4.5145	4.8460	4.5160	4.8470	4.5118	4.8483
5	4.5147	4.8458	4.5169	4.8465	4.5196	4.8474
6	4.5132	4.8453	4.5092	4.8440	4.5041	4.8424
7	4.5138	4.8459	4.5120	4.8470	4.5099	4.8484
8	4.5151	4.8451	4.5189	4.8430	4.5236	4.8403
9	4.5144	4.8461	4.5153	4.8476	4.5165	4.8496
10	4.5142	4.8459	4.5142	4.8466	4.5143	4.8476
Mean	4.5145 ±4.8E-4	4.8457 ±2.2E-4	4.5158 ±2.4E-3	4.8460 ±1.1E-3	4.5175 ±4.7E-3	4.8463 ±2.2E-3
Difference (%) mean/exact data	0.007	0.001	0.033	0.007	0.067	0.013

Exact Data (s=0)	$\beta_{0T(x)}$	β_{0T_L}
	4.5142	4.8457
Difference (%) exact data/ true estimate	2.41	0.03
True Estimate	$\beta_{trueT(x)}$	β_{trueT_L}
	4.4080	4.8470

estimation procedure.

Larger percentage differences with respect to the estimates from exact data are obtained for the mean values of the estimates $\beta_{T(x)}$, which indicates that the thermal parameter $\beta_{T(x)}$ is more difficult to estimate than the thermal parameter β_{T_L}. This occurs because, based on the sensitivity analysis, the most temperature information for the parameter estimation is provided at the warm end of the thermal bridges. Indeed, recall that the sensitivity coefficient magnitudes have been shown in Sections 4.2.1 and 4.2.2.1 to be maximum at the warm end of the thermal bridges; furthermore, note that parameters estimated from data with large sensitivity coefficients are generally more accurate than parameters estimated from data with small sensitivity coefficients (Scott, 1994). Therefore, the estimation of $\beta_{T(x)}$ is more sensitive to experimental errors. This result is of importance in this research as it confirms the placement of a temperature sensor at the warm end of the thermal bridges in the preliminary experimental design.

As shown in Tables 4.2.8 to 4.2.12, the addition of random measurement errors with standard deviation of 0.1, 0.5 and 1.0 K, induces an overall decrease in the accuracy of the estimates, with an associated increase in the corresponding 95-percent confidence intervals. The maximum percentage difference with respect to exact data is contained in the mean values of the estimates obtained using measurement errors with a standard deviation of 1.0 K; this results for the five HTS thermal bridges in percentage differences of 0.013 percent for the estimates β_{T_L} and in percentage differences ranging from 0.067 to 0.073 percent for the estimates $\beta_{T(x)}$. The small values for these percentage differences confirm that reasonable estimates have been obtained.

A comment should be added about the influence of the initial estimate in the parameter estimation method, as it bears on the performances of the procedure. Indeed, when the initial estimate was chosen as the estimate using exact data, convergence was reached in no more than two iterations; whereas the choice of a poor initial estimate resulted in the increase of the number of iterations to reach convergence.

Looking now at the percentage differences between the estimates using exact data and the true estimates, one can see that the estimates β_{0T_L} closely match the true estimates β_{trueT_L} with a maximum percentage difference of 0.11 percent occurring for the thermal bridge YBCO/FSI. This result points out the reliability of the estimates obtained at the warm end of the thermal bridges. The percentage differences between $\beta_{0T(x)}$ and $\beta_{trueT(x)}$ are however larger, with a maximum difference of 2.61 percent occurring for the thermal bridge BSCCO/FSI. Percentage differences between $\beta_{0T(x)}$ and $\beta_{trueT(x)}$ were actually expected to be higher than those obtained between β_{0T_L} and β_{trueT_L}. In an attempt to explain these higher percentage differences, recall that in the computation of the value $\beta_{trueT(x)}$ in Section 4.2.2.2.1, the discretization of the distribution $k_{eff}(T)$ was restricted to 101 intervals. This restriction was required so that the limits of the intervals coincide with the numerical temperature values produced using ORTHO3D. From this, one can expect that the use of more data points along the thermal bridges should help obtaining closer values for $\beta_{0T(x)}$ and $\beta_{trueT(x)}$.

Finally, the values of the constant effective thermal conductivities estimated for each thermal bridge were compared. This comparison shows that the smallest values for $\beta_{T(x)}$ and β_{T_L} are logically obtained for the thermal bridge BSCCO/FSI which displays the less

heat load on the cryogen, or in other words, which conducts heat the less. In addition, the proportionality between the constant effective thermal conductivities β_{0T_L}, estimated using exact data, is found to correspond to the proportionality between the heat loads on the cryogen, Q_{cond}, provided in Table 3.2.1, by the HTS thermal bridges. Using the thermal bridge BSCCO/FSI (BF) as a reference, Table 4.2.13 displays the ratios $\frac{\beta_{0T_Lj}}{\beta_{0T_LBF}}$ and $\frac{Q_{condj}}{Q_{condBF}}$, where j denotes the four other HTS thermal bridges. As one can see, for each thermal bridge, the same value is obtained for both ratios. This result was expected because β_{0T_L} represents the estimate using exact data of the average value over the temperature range $[T_0\text{-}T_L]$ of the HTS thermal bridge effective thermal conductivity. The excellent agreement between both ratios not only demonstrates the accuracy of the estimation of the average values over the temperature range $[T_0\text{-}T_L]$ of the HTS thermal bridge effective thermal conductivities, but also shows that this estimation is an effective way to demonstrate the respective heat loads on the cryogen.

Table 4.2.13. Proportionality Between Both the Constant Effective Thermal Conductivities Estimated at the Warm End of the HTS Thermal Bridges Using Exact Data and the Respective Heat Loads on the Cryogen.

	BSCCO/ FSI	YBCO/ FSI	BSCCO/ YSZ	YBCO/ YSZ	YBCO/ GREEN
β_{0T_L} (W/m-K)	0.3153	0.5214	0.7601	0.9666	4.8457
Q_{cond} (W)	2.44E-4	4.03E-4	5.89E-4	7.48E-4	3.74E-3
$\beta_{0T_L}/\beta_{0T_LBF}$	1	1.65	2.41	3.07	15.37
Q_{cond}/Q_{condBF}	1	1.65	2.41	3.07	15.33

CHAPTER 5

Conclusions and Summary

The focus of this study was on the analysis of a space experimental design for high-T_c superconductive thermal bridges (Lee, 1994). The primary objectives were to verify that the sources of heat transfer (electrical and radiative heat sources) neglected in the preliminary conductive analysis of the thermal bridges by Lee were indeed negligible, and to develop a methodology for the estimation as temperature dependent of the thermal conductivities of the HTS thermal bridges. The following conclusions were drawn based on the results obtained.

5.1 Electrical and Radiative Heat Sources

In this investigation, the electrical and radiative heat sources on the thermal bridges were evaluated in order to determine whether or not these sources contribute significantly on the heat load on the cryogen. The evaluation of the radiative heat source was performed only for the HTS thermal bridges; therefore, the cryogenic heat load generated

by the manganin thermal bridge is a conservative result. The electrical heat source created in the manganin wires by the electronic signals from the IR detectors was directly implemented into the conductive model of the manganin thermal bridge. The radiative heat source on the HTS thermal bridges was determined by performing a separate radiant interchange analysis within a high-T_c superconductor housing chamber in the experimental design.

5.1.1 Conclusions for the Electrical Heat Source

The finite difference program ORTHO3D used to construct the conductive mathematical models of the thermal bridges allowed for the analysis of a volumetric heat source generated in the geometric domain. The Joule heating term created by the electrical current was therefore incorporated as a volumetric heat source into the manganin conductive model. The following conclusions can be drawn from the results:

1) The electrical heat source in the manganin wires does not contribute significantly on the cryogenic heat load.
2) The temperature distribution along the manganin wires is not affected by the electrical heat source.

5.1.2 Conclusions for the Radiative Heat Source

The analysis of the radiant interchange within a HTS housing chamber was performed in two phases. First the distribution factors were computed using the Monte-Carlo method and then the distribution factor results were used in calculating the radiative

heat load on the HTS thermal bridges. Due to the geometric complexity, the HTS thermal bridges were approximated as only the substrate materials. Because the radiative properties of the specific substrates used in this research (fused silica, yttrium stabilized zirconia and green phase) could not be found in the literature, these properties had to be predicted. This prediction could be, however, responsible for variations between the actual and the calculated radiative heat load on the HTS thermal bridges. Three different radiative heat loads, based on geometric considerations, were compared to the conductive heat load on the cryogen. From the results obtained, the following conclusions can be made:

1) The solution for the distribution factors is converged and symmetric.
2) The larger the reflectivity of the substrate material, the lower the distribution factors to the substrate and the lower the radiative heat load on the substrate.
3) The lower the reflectivity of the housing chamber material, the lower the distribution factors to the substrate and the lower the radiative heat load on the substrate.
4) The radiative heat load on the bottom of the substrate from the entire enclosure is negligible for the GREEN substrate but not for the FSI and YSZ substrates.
5) The radiative heat load on the bottom end of the substrate from the top end represents less than 4 percent of the conductive heat load on the cryogen for the three substrates (FSI, YSZ and GREEN).
6) The radiative heat load on the entire substrate from the entire enclosure is negligible for all three substrates studied.
7) The radiative heat load on surface 4 from surface 5 represents less than 2 percent of

the conductive heat load on the cryogen for the three substrates.

Considering that conclusions 6 and 7 provide the best information for the relevance of radiation on the cryogenic heat load, the radiative heat source on the HTS thermal bridges can then be reasonably neglected in the conductive analysis.

The following conclusion can also be made when examining the combined results of the evaluation of the electrical and radiative heat sources:

1) The conductive heat loads on the cryogen and the temperature distributions along the thermal bridges obtained in the conductive analysis (Scott and Lee, 1994) are valid.

5.2 Thermal Conductivity Estimation Methodology

A methodology was presented for the estimation of the thermal conductivities of the individual HTS thermal bridge materials and the effective thermal conductivities of the composite HTS thermal bridges, as functions of temperature. This methodology included a sensitivity analysis and the demonstration of the estimation procedure using simulated data with added random errors. The estimation procedure used was the modified Box-Kanemasu method. The following conclusions can be drawn from the results obtained:

1) The parameters describing the material thermal conductivities as functions of temperature in a HTS thermal bridge are correlated and cannot be estimated simultaneously.
2) The parameters describing the HTS thermal bridge effective thermal conductivities as functions of temperature are correlated and cannot be estimated simultaneously.

3) The placement of a temperature sensor at the warm end of the thermal bridges provides the most information for the parameter estimation.

Based on the two first conclusions, the effective thermal conductivities of the HTS thermal bridges were analyzed to be estimated as constants. The estimation procedure was demonstrated using simulated and exact data both along the thermal bridges and at the warm end of the thermal bridges to account for the location of the temperature sensor in the experimental design. The results obtained allow for the following conclusions:

1) The estimation procedure using simulated data resulted in good agreement between the estimated and predicted constant effective thermal conductivities.
2) The estimation of the constant effective thermal conductivities is more sensitive to measurement errors using simulated data along the thermal bridges than at the warm end of the thermal bridges.
3) The estimates for the constant effective thermal conductivities obtained using exact data at the warm end represent with accuracy the average values over the temperature range along the thermal bridges of the temperature dependent effective thermal conductivities.
4) The proportionality between the estimates for the constant effective thermal conductivities obtained using exact data at the warm end exactly corresponds to the proportionality between the heat loads on the cryogen by the respective HTS thermal bridges. The estimation at the warm end of the thermal bridges of the HTS thermal bridge effective thermal conductivities as constants is then an effective way to demonstrate the respective cryogenic heat loads.

CHAPTER 6

Recommendations

From the analysis of the conclusions drawn in chapter five, recommendations can be deduced for the two majors areas of this research study, namely the heat transfer analysis of the experimental design and the thermal conductivity estimation of the HTS thermal bridges.

In order to minimize the radiative heat source in the HTS housing area the use, for only surfaces 1, 2 and 3 of the housing chamber (see Figure 3.3.2), of a material with lower reflectivity than the reflectivity of the currently used pure copper is suggested. This would allow to minimize the radiative heat load on the HTS thermal bridges and to maintain the radiative heat load on surface 4 from surface 5 negligible. The choice for this material would also have to meet the specific requirements set for the housing chamber with respect to the experimental design.

To account for the radiative properties of the superconductors, the HTS leads should be incorporated on the substrate materials in the radiation analysis.

Finally, with the primary goal to compare the performance between the HTS and the

manganin thermal bridges, a radiation analysis needs to be performed on the manganin thermal bridge.

The heat transfer through the HTS thermal bridges does not account for any heat contribution from the mechanical supports as these are currently being designed to have no contact with the bridges in space. However, if the structural analysis of the support mechanisms results in an effective contact in space between the supports and the bridges, then the heat transfer model of the HTS thermal bridges would have to be reconsidered.

The estimates for the constant effective thermal conductivities of the HTS thermal bridges obtained using exact data at the warm end represented with accuracy the average values over the temperature range along the thermal bridges of the temperature dependent effective thermal conductivities. The investigation of different temperatures, lower than 80 K, at the warm end (with the cold end kept fixed at 4 K) would provide an efficient way to determine different average values of the temperature dependent effective thermal conductivities for different temperature gradients between the ends of the thermal bridges. The objective would be to build a model describing the average values of the effective thermal conductivities as functions of the temperature gradient between the ends of the thermal bridges. Note that these average values would be expected to be more accurate as the temperature would get closer to 80 K and hence provide more information for the parameter estimation. An interpolation procedure would need to be implemented to obtain the distribution of the effective thermal conductivities with respect to the temperature along the thermal bridges (ranging from 4 to 80 K) from the distribution of the average values with respect to the temperature gradient between the ends of the

thermal bridges (ranging from 0 to 76 K). Such experiment to vary the temperature at the warm end from 4 to 80 K could be performed by programming the heater (see Section 3.1) with a step function, and estimating the thermal conductivities at steady-state conditions throughout the interval from 4 to 80 K.

BIBLIOGRAPHY

Allen, N. P., ed. by D. Fishlock, 1969, *Guide to Superconductivity*, MacDonald & Co., London, pp. 1-16.

Bard, Y., 1970, "Comparison of Gradient Methods for the Solution of Nonlinear Parameter Estimation Problems," *SIAM Journal of Numerical Analysis*, Vol. 7, pp. 157-186.

Bardeen, J., ed. by M. H. Cohen, 1968, *Superconductivity in Science and Technology*, University of Chicago Press, Chicago, pp. 1-17.

Bardeen, J., L. N. Cooper, and J. R. Schrieffer, 1957, "Theory of Superconductivity," *Phys. Rev.*, Vol. 108, No. 5, December 1, pp.1175-1204.

Beck, J. V. and K. J. Arnold, 1977, *Parameter Estimation in Engineering and Science*, John Wiley & Sons, New York.

Berdnorz, J. G., and K. A. Müller, 1986, "Possible High T_c Superconductivity in the Ba-La-Cu-O System," *Z. Phys. B—Condensed Matter 64*, pp.189-193.

Bongiovi, R. P., 1993, *A Parametric Study of the Radiative and Optical Characteristics of a Scanning Radiometer for Earth Radiation Budget Applications Using the Monte-Carlo Method*, M.S. Thesis, Department of Mechanical Engineering, Virginia Polytechnic and State University, Blacksburg, VA.

Box, G. E. P. and H. Kanemasu, 1972, "Topics in Model Building, Part II, on Non-linear Least Squares," Tech. Report No. 321, University of Wisconsin, Dept. of Statistics, Madison, Wisconsin.

Caton, R., R. Selim, and A. M. Buoncristiani, 1992, "A Comparison of Superconductor and Manganin Technology for Electronic Links Used in Space Mission Applications," NASA Contractor Report 4477.

Creel, K. E., 1994, *A Thermal Analysis Tool for Three-Dimensional Models of Multilayer Microelectronics*, M.S. Thesis, Department of Mechanical Engineering, Virginia

Polytechnic and State University, Blacksburg, VA.

Creel, K. E., and D. J. Nelson, 1994, "A Thermal Analysis Tool for Three-Dimensional Thermal Models of Multilayer Microelectronics," *Proceedings of the Virginia Power Electronics Center Seminar*, Virginia Tech, Blacksburg, VA, pp. 113-127.

Doss, J. D., 1989, *Engineer's Guide to High-Temperature Superconductivity*, John Wiley & Sons, New York.

Eckert, E. R. G., and E. M. Sparrow, 1961, "Radiative Heat Exchange Between Surfaces with Specular Reflections," *International Journal of Heat Transfer*, Vol. 3, pp. 42-54.

Geballe, T. H., and J. K. Hulm, 1988, "Superconductivity—The State That Came in from the Cold," *Science*, Vol. 239, pp. 367-375.

Hunt, V. D., 1989, *Superconductivity Sourcebook*, John Wiley & Sons, Toronto.

Incropera, F. P., and D. P. De Witt, 1990, *Fundamentals of Heat and Mass Transfer*, 3rd Ed., John Wiley & Sons, New York.

Jurkowski, T., Y. Jarny and D. Delaunay, 1992, "Simultaneous Identification of Thermal Conductivity and Thermal Contact Resistances without Internal Temperature Measurements," *Institution of Chemical Engineers Symposium Series*, Vol. 2, No. 129, pp. 1205-1211.

Krishen, K., and A. Ignatiev, 1988, "Future Superconductivity Applications in Space - A Review," *World Congress on Superconductivity*, Houston, Texas, Vol. 8, pp. 141-154.

Lagües, M., X. Ming Xie, H. Tebbji, X. Z. Xu, V. Mairet, C. Hatterer, C. F. Beuran, C. Deville-Cavellin, 1993, "Evidence suggesting Superconductivity at 250K in a Sequentially Deposited Cuprate Film," *Science*, Vol. 262, pp. 1850-1852.

Lee, K. M., 1994, *Experimental Design for the Evaluation of High-T_c Superconductive Thermal Bridges in a Sensor Satellite*, M.S. Thesis, Department of Mechanical Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Mahan, 1994, Personal Communication.

Mahan, J. R., and L. D. Eskin, 1984, "The radiation Distribution Factor - Its Calculation Using Monte-Carlo and an Example of its Application," *First UK National Heat Transfer Conference*, Leeds, Yorkshire, England, pp. 1001-1012.

Metal Handbook, 1980, 9th Ed., American Society for Metals, Vol. 3, pp. 640-644.

Moncman D. A., *Optimal Experimental Designs for the Estimation of Thermal Properties of Composite Materials*, M.S. Thesis, Department of Mechanical Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Nelson, 1994, Personal Communication.

Patankar, S. V., 1991, *Computation of Conduction and Duct Flow Heat Transfer*, Innovative Research, Maple Grove, MN.

PATRAN, 1990, PDA Engineering, Scranton, PA.

Rose-Innes, A. C., and E. H. Rhoderick, 1969, *Introduction to Superconductivity*, Pergamon Press, London.

Saad, Z., 1991, Estimation of the Kinetic Parameters of an Amine-Epoxy Resin During Cure, M.S. Thesis, Department of Mechanical Engineering, Michigan State University, East Lansing, MI.

Scott, E. P., 1994, "An Analytical Solution and Sensitivity Study of Sublimation-Dehydration Within a Porous Medium With Volumetric Heating, "*Journal of Heat Transfer*, Transactions of the ASME, Vol. 116, pp. 686-693.

Scott, E.P., and K. M. Lee, 1994, "Experimental Design for the Evaluation of High-T_c Superconductive Thermal Bridges in a Sensor Satellite," NASA Annual Report for Contract No. NAG-1-1500.

Scott E. P., and Z. Saad, 1993, "Estimation of Kinetic Parameters Associated with the Curing of Thermoset Resins. Part I: Theoretical Investigation," *Polymer Engineering and Science*, Vol. 33, No. 18, pp. 1157-1164.

Scott E. P., and Z. Saad, 1993, "Estimation of Kinetic Parameters Associated with the Curing of Thermoset Resins. Part II: Experimental Results," *Polymer Engineering and Science*, Vol. 33, No. 18, pp. 1165-1169.

Scott, L. A., and E. P. Scott, 1993, "An Inverse Approach to the Determination of the Optimal Treatment Time in Cryosurgical Applications," *Advances in Bioheat and Mass Transfer: Microscale Analysis of Thermal Injury Processes, Instrumentation, Modeling, and Clinical Applications*, HTD-Vol. 268, pp. 1-8.

Siegel, R., and J. R. Howel, 1992, *Thermal Radiation Heat Transfer*, 3rd Ed., Hemisphere Publishing Corporation, Washington.

SINDA '85/FLUINT, 1985, NASA Contract NAS9-1744B, Costa Mesa, CA.

Spencer, B., 1994, *Design of a Support System for a Dynamically Loaded Thin Plate*, Senior Project, Department of Engineering Science and Mechanics, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Standard Handbook for Electrical Engineers, ed. by D. G. Fink and H. W. Beaty, 1989, 13th Ed., pp. 4-75.

Taktak, R., E. P. Scott and J. V. Beck, 1991, "Optimal Experimental Designs for Estimating the Thermal Properties of Composite Materials," Proceedings of the 3rd ASME - JSME Thermal Engineering Joint Conference, ASME, Vol. 3, pp. 481-488.

Vidali G., 1993, *Superconductivity: the Next Revolution?*, Cambridge University Press, Cambridge, Great Britain.

Walpole, R. E. and R. H. Myers, 1978, *Probability and Statistics for Engineers and Scientists*, 2nd Edition, Macmillan Publishing Co., NY.

Wise, S. A., J. D. Buckley, I. Nolt, M. W. Hooker, G. H. Haertling, R. Selim, R. Caton and A. M. Buoncristiani, 1992, "High-T_c Thermal Bridges for Space-Borne Cryogenic Infrared Detectors," FED-Sensor Technology Branch, NASA-Langley Research Center, Hampton, VA.

Appendix A

The Fortran Subroutine HTS.FOR

This subroutine, HTS.FOR, was written as the *adapt* part of the program ORTHO3D (provided at the end of the subroutine). HTS.FOR is used to determine the temperature distribution and the cryogenic heat load for the five HTS thermal bridges.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE ADAPT
c
c  HP f77 version,  3-D dp Iso version - DJN
c
c$noextensions
C-----------------------------------------------------------------------
C----- STEADY CONDUCTION IN A 3D HALF HTS THERMAL BRIDGE -----
C                  with isotropic gamma
C
C----- Temperature Determination
C      Subroutine HTS.FOR, written by Sandrine Garcia, 1994.
C-----------------------------------------------------------------------
      INCLUDE 'common3d.f'
C**********************************************************************
      DIMENSION              T(NI,NJ,NK)
      EQUIVALENCE (F(1,1,1,1),T(1,1,1))
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY GRID
C
      HEADER='HALF YBCO/YSZ - L=152.4mm - Q'
      PRINTF='T'
C      PLOTF ='output.pl'  (not set up for 3D plots yet)
C
c  set geometric dimensions
      rL  = 0.1524d0
```

```
c  if there is a buffer layer (the substrate is FSI)
ccc      thk2=0.0000003d0
c  if there is no buffer layer
      thk2 = 0.0d0
      rlsuperc = 0.003048d0
      e1 = 0.0001524d0
      e2 = 0.0000508d0
      a  = 0.003048d0
C
c  zoned grid method
c  set x, y and z zones
      NZX = 1
      XZONE(1) = rL
      NCVX(1) = 100
C
      NZY = 2
      YZONE(1) = a
      NCVY(1) = 4
      YZONE(2) = rlsuperc/2.d0
      NCVY(2) = 3
C
      NZZ = 2
c  if the substrate is FSI, add 1 CV for the buffer layer
ccc      NZZ = 3
      ZZONE(1) = e1
      NCVZ(1) = 3
c  if there is no buffer layer
      ZZONE(2) = e2
      NCVZ(2) = 2
c  if there is a buffer layer
ccc      ZZONE(2) = thk2
ccc      NCVZ(2) = 1
ccc      ZZONE(3) = e2
ccc      NCVZ(3) = 2
C
      CALL ZGRID
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY BEGIN
C
      TITLE(1) ='  TEMPERATURE '
      KSOLVE(1)=1
      KPRINT(1)=0
      KPLOT(1) =0
      KSTOP    =0
C
c  set maximum number of outer iterations
      LAST   = 50
c  set minimum number of outer iterations
      ITRMIN = 8
c  set convergence parameter
      epsi = 1.d-5
c  set initial temperature (K)
      DO 100 K=1,N1
      DO 100 J=1,M1
```

```
      T(1,J,K) = 4.d0
      DO 100 I=2,L1
        T(I,J,K) = 4.7336407d0 + 2376.7446289d0*X(I)
     +           - 85404.94531d0*X(I)**2
     +           + 2106631.25d0*X(I)**3
     +           - 29870884.d0*X(I)**4
     +           + 238106752.d0*X(I)**5
     +           - 991233280.d0*X(I)**6
     +           + 1674745088.d0*X(I)**7
100   CONTINUE
C
c  set thermal conductivity coefficients
      Bg1 = 0.3558d0
      Bg2 = 0.07173d0
      Bg3 = 0.01066d0
      Bg4 = -3.706d-4
      Bg5 = 4.814d-6
      Bg6 = -2.839d-8
      Bg7 = 6.37d-11
C
      By1 = 0.4464d0
      By2 = -0.002426d0
      By3 = 9.229d-4
      By4 = -2.793d-5
      By5 = 3.772d-7
      By6 = -2.395d-9
      By7 = 5.839d-12
C
      Bf1 = 0.01565d0
      Bf2 = 0.002761d0
      Bf3 = 1.561d-4
      Bf4 = -3.076d-6
      Bf5 = 3.403d-8
      Bf6 = -2.009d-10
      Bf7 = 4.826d-13
C
      Bz1 = -0.2045d0
      Bz2 = 0.1159d0
      Bz3 = -0.001041d0
      Bz4 = -2.761d-5
      Bz5 = 6.671d-7
      Bz6 = -5.127d-9
      Bz7 = 1.367d-11
C
      Bybco1 = 0.1567d0
      Bybco2 = 0.01403d0
      Bybco3 = 0.007463d0
      Bybco4 = -2.51d-4
      Bybco5 = 3.437d-6
      Bybco6 = -2.201d-8
      Bybco7 = 5.45d-11
C
      Bbscco1 = 0.143d0
      Bbscco2 = 0.05445d0
      Bbscco3 = -0.003517d0
```

```
      Bbscco4 = 1.243d-4
      Bbscco5 = -2.1d-6
      Bbscco6 = 1.665d-8
      Bbscco7 = -5.035d-11
C
c  set input heat flux (W/m2)
c   - if the thermal bridge is YBCO/GREEN
      Qyg = 2415.429d0
c   - if the thermal bridge is YBCO/YSZ
      Qyy = 483.237d0
c   - if the thermal bridge is BSCCO/YSZ
      Qby = 380.095d0
c   - if the thermal bridge is YBCO/FSI
      Qyf = 260.383d0
c   - if the thermal bridge is BSCCO/FSI
      Qbf = 157.302d0
C
      QOUT0 = 0.D0
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY OUTPUT
C
      QIN  = 0.D0
      QOUT = 0.D0
      DO 20 J=2,M2
      DO 20 K=2,N2
        QIN  = QIN  + YCV(J)*ZCV(K)*FLUXL1(J,K,1)
        QOUT = QOUT + YCV(J)*ZCV(K)*FLUXI1(J,K,1)
20    CONTINUE
C
      DO 200 IUNIT=IU1,IU2
        IF(ITER.EQ.0) WRITE(IUNIT,210)
210       FORMAT(2X,'ITER',3X,'T(L1,M2,11)',5X,'T(L1,M2,12)',9X,'QIN',
     +          13X,'QOUT',9X,'NTC(1)')
        WRITE(IUNIT,220)ITER,T(L1,M2,11),T(L1,M2,12),QIN,QOUT,NTC(1)
220       FORMAT(2X,I3,3X,1PE10.2,5X,1PE10.2,7X,1PE10.3,7X,1PE10.3,
     +          7X,I2)
200   CONTINUE
C
c  create a convergence criterion
      IF (ITER.LT.ITRMIN) RETURN
      DIFF = ABS((QOUT-QOUT0)/(QOUT+SMALL))
      QOUT0=QOUT
      IF (DIFF.LE.epsi.OR.ITER.EQ.LAST) THEN
C
c  calculate overall energy balance
      HTBAL = QIN + QOUT
      EBAL  = ABS(HTBAL/QIN)
      DO 40 IUNIT=IU1,IU2
        WRITE(IUNIT,50)EBAL
50        FORMAT(/,2X,'EBAL',1PE11.3)
40    CONTINUE
C
c  Record temperature solution on specific file
      open(unit=3,file='yy.100')
```

```
      do i=1,L1
        write(3,'(E18.5)')T(i,M2,11)
      enddo
      close(3)
C
      CALL PRINT
      close (iu2)
      KSTOP=1
      ENDIF
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY PHI
C
c  set conductivities
      DO 300 K=2,N2
      DO 300 J=2,M2
      DO 300 I=2,L2
C
      IF (Z(K).LE.e1) THEN
c  if the substrate is the GREEN PHASE :
ccc          GAM(I,J,K) = Bg1 + Bg2*T(I,J,K)
ccc    +                 + Bg3*T(I,J,K)**2
ccc    +                 + Bg4*T(I,J,K)**3
ccc    +                 + Bg5*T(I,J,K)**4
ccc    +                 + Bg6*T(I,J,K)**5
ccc    +                 + Bg7*T(I,J,K)**6
c  if the substrate is YSZ :
          GAM(I,J,K) = By1 + By2*T(I,J,K)
     +                 + By3*T(I,J,K)**2
     +                 + By4*T(I,J,K)**3
     +                 + By5*T(I,J,K)**4
     +                 + By6*T(I,J,K)**5
     +                 + By7*T(I,J,K)**6
c  if the substrate is FSI :
ccc          GAM(I,J,K) = Bf1 + Bf2*T(I,J,K)
ccc    +                 + Bf3*T(I,J,K)**2
ccc    +                 + Bf4*T(I,J,K)**3
ccc    +                 + Bf5*T(I,J,K)**4
ccc    +                 + Bf6*T(I,J,K)**5
ccc    +                 + Bf7*T(I,J,K)**6
        ELSE
C
        IF (Z(K).GT.e1.AND.Y(J).LT.a) GAM(I,J,K) = 0.d0
C
c  if there is a buffer layer (the substrate is FSI)
ccc        IF (Z(K).GT.e1.AND.Z(K).LE.(e1+thk2).AND.
ccc    +       Y(J).GE.a) THEN
ccc        GAM(I,J,K) = Bz1 + Bz2*T(I,J,K)
ccc    +                 + Bz3*T(I,J,K)**2
ccc    +                 + Bz4*T(I,J,K)**3
ccc    +                 + Bz5*T(I,J,K)**4
ccc    +                 + Bz6*T(I,J,K)**5
ccc    +                 + Bz7*T(I,J,K)**6
ccc        ENDIF
C
```

```
        IF (Z(K).GT.(e1+thk2).AND.Z(K).LE.(e1+thk2+e2).AND.
     +      Y(J).GE.a) THEN
c  if the superconductor is YBCO :
        GAM(I,J,K) = Bybco1 + Bybco2*T(I,J,K)
     +             + Bybco3*T(I,J,K)**2
     +             + Bybco4*T(I,J,K)**3
     +             + Bybco5*T(I,J,K)**4
     +             + Bybco6*T(I,J,K)**5
     +             + Bybco7*T(I,J,K)**6
c  if the superconductor is BSCCO :
ccc        GAM(I,J,K) = Bbscco1 + Bbscco2*T(I,J,K)
ccc     +             + Bbscco3*T(I,J,K)**2
ccc     +             + Bbscco4*T(I,J,K)**3
ccc     +             + Bbscco5*T(I,J,K)**4
ccc     +             + Bbscco6*T(I,J,K)**5
ccc     +             + Bbscco7*T(I,J,K)**6
        ENDIF
C
       ENDIF
300   CONTINUE
C
c  set boundary conditions
      DO 310 K=2,N2
      DO 310 J=2,M2
        KBCL1(J,K) = 2
c      - for the substrate
        IF (Z(K).LE.e1) THEN
         FLXCL1(J,K) = Qyy
        ELSE
C
c      - if there is a buffer layer
ccc         IF (Y(J).GE.a.AND.Z(K).GT.e1.AND.Z(K).LE.
ccc     +       (e1+thk2)) THEN
ccc          FLXCL1(J,K) = Qyf
ccc         ENDIF
C
c      - for the superconductor
         IF (Y(J).GE.a.AND.Z(K).GT.(e1+thk2).AND.Z(K).LE.
     +       (e1+thk2+e2)) THEN
          FLXCL1(J,K) = Qyy
C
         ENDIF
        ENDIF
310   CONTINUE
C
      DO 320 K=2,N2
      DO 320 I=2,L2
        KBCJ1(I,K)=2
        KBCM1(I,K)=2
320   CONTINUE
C
      DO 330 J=2,M2
      DO 330 I=2,L2
        KBCK1(I,J)=2
        KBCN1(I,J)=2
```

```
330   CONTINUE
C
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
************************************************************************
      PROGRAM ORTHO3D
*
* Control Volume Method solution of three-dimensional, isotropic
* heat conduction
*
* User portion of code contained in layer*.f
* layer3d.f - standard user subroutine
*
* RS/6000 version - double precision
*
* Program structure similar to that detailed in
* "Computation of Conduction and Duct Flow Heat Transfer",
*  S.V. Patankar, Maple Grove, MN:  Innovative Research, Inc.
*
************************************************************************
      INCLUDE 'common3d.f'
C
      CALL DEFLT
      CALL GRID
      CALL READY
      CALL BEGIN
C
10    CONTINUE
C
c start iteration of outer loop
      CALL OUTPUT
c check to see if convergence has occured
      IF(KSTOP.NE.0) STOP 'HTSISOX.F done'
      CALL HEART
      GO TO 10
C
      STOP
      END
************************************************************************
c include other subroutines in invariant part
      INCLUDE 'defrd3d.f'
      INCLUDE 'heart3d.f'
      INCLUDE 'solve3d.f'
      INCLUDE 'tools3d.f'
```

Appendix B

The Fortran Subroutine MANG.FOR

This subroutine, MANG.FOR, was written as the *adapt* part of the program ORTHO3D (provided at the end of the subroutine HTS.FOR in appendix A). MANG.FOR is used to determine the temperature distribution and the cryogenic heat load for the manganin thermal bridge.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE ADAPT
c
c  HP f77 version,  3-D dp Ortho version - DJN
c
c$noextensions
C-----------------------------------------------------------------------
C----- STEADY CONDUCTION IN A 3D FOURTH MANGANIN THERMAL BRIDGE -----
c                 with isotropic gamma
C
C----- Temperature Determination
C       Subroutine MANG.FOR, written by Sandrine Garcia, 1994.
C-----------------------------------------------------------------------
      INCLUDE 'common3d.f'
C***********************************************************************
      DIMENSION              T(NI,NJ,NK)
      EQUIVALENCE (F(1,1,1,1),T(1,1,1))
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY GRID
C
      HEADER=' FOURTH MANGANIN - L=101.6mm - Q+g'
      PRINTF='mgg2.pr'
C       PLOTF ='output.pl'  (not set up for 3D plots yet)
```

```
C
c  set geometric dimensions
     rL  = 0.1524d0
     rlmang   = 0.0005491747d0
     rlkapt = 0.0000017436298d0
C
c  zoned grid method
c  set x, y and z zones
     NZX = 1
     XZONE(1) = rL
     NCVX(1) = 100
C
     NZY = 2
     YZONE(1) = rlkapt
     NCVY(1) = 1
     YZONE(2) = rlmang/2.d0
     NCVY(2) = 4
C
     NZZ = 2
     ZZONE(1) = rlkapt
     NCVZ(1) = 1
     ZZONE(2) = rlmang/2.d0
     NCVZ(2) = 4
C
     CALL ZGRID
     RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
     ENTRY BEGIN
C
     TITLE(1) ='   TEMPERATURE '
     KSOLVE(1)=1
     KPRINT(1)=1
     KPLOT(1) =0
C
c  set maximum number of outer iterations
     LAST   = 100
c  set minimum number of outer iterations
     ITRMIN = 30
c  set convergence parameter
     epsi = 1.d-5
C
c  set initial temperature (K)
     DO 100 K=1,N1
     DO 100 J=1,M1
       T(1,J,K) = 4
     DO 100 I=2,L1
       T(I,J,K) = 4.7336407d0 + 2376.7446289d0*X(I)
    +            - 85404.94531d0*X(I)**2
    +            + 2106631.25d0*X(I)**3
    +            - 29870884.d0*X(I)**4
    +            + 238106752.d0*X(I)**5
    +            - 991233280.d0*X(I)**6
    +            + 1674745088.d0*X(I)**7
100  CONTINUE
C
```

```
c  set input heat flux calculated by Lee (W/m2)
      Q = 3321.5d0
c  set electrical source (W/m3)
      g = 5.29d-6
c  define constant conductivity of kapton (W/m*K)
      CDKAP = 0.16d0
C
      QOUT0 = 0.D0
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY OUTPUT
C
      QIN  = 0.D0
      QOUT = 0.D0
      DO 20 J=2,M2
      DO 20 K=2,N2
        QIN  = QIN  + YCV(J)*ZCV(K)*FLUXL1(J,K,1)
        QOUT = QOUT + YCV(J)*ZCV(K)*FLUXI1(J,K,1)
20    CONTINUE
C
      DO 200 IUNIT=IU1,IU2
        IF(ITER.EQ.0) WRITE(IUNIT,210)
210      FORMAT(2X,'ITER',3X,'T(L1,M2,N2)',5x,'T(L1,M2,2)',
     +          8X,'QIN',13X,'QOUT',9X,'NTC(1)')
        WRITE(IUNIT,220)ITER,T(L1,M2,N2),T(L1,M2,2),QIN,QOUT,NTC(1)
220      FORMAT(2X,I3,3X,1PE10.2,5x,1PE10.2,5X,1PE10.3,7X,1PE10.3,
     +          7X,I2)
200   CONTINUE
C
c  create a convergence criterion
      IF (ITER.LT.ITRMIN) RETURN
      DIFF = ABS((QOUT-QOUT0)/(QOUT+SMALL))
      QOUT0=QOUT
      IF (DIFF.LE.epsi.OR.ITER.EQ.LAST) THEN
C
c  calculate overall energy balance
      HTBAL = QIN + QOUT
      EBAL  = ABS(HTBAL/QIN)
      DO 40 IUNIT=IU1,IU2
        WRITE(IUNIT,50)EBAL
50       FORMAT(/,2X,'EBAL',1PE11.3)
40    CONTINUE
C
      CALL PRINT
      close (iu2)
      KSTOP=1
      ENDIF
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY PHI
C
c  set conductivities and electrical source
      DO 300 K=2,N2
      DO 300 J=2,M2
      DO 300 I=2,L2
```

```
C
        IF (Y(J).LT.rlkapt.OR.Z(K).LT.rlkapt)
     +    GAM(I,J,K) = CDKAP
C
        IF (Y(J).GE.rlkapt.AND.Z(K).GE.rlkapt) THEN
         GAM(I,J,K) = 0.01449D0 + 0.1005D0*T(I,J,K)
     +              + 0.005584D0*T(I,J,K)**2
     +              - 1.911D-4*T(I,J,K)**3
     +              + 3.283D-6*T(I,J,K)**4
     +              - 2.88D-8*T(I,J,K)**5
     +              + 9.859D-11*T(I,J,K)**6
C
c set volumetric heat source
         SC(I,J,K) = g
        ENDIF
300   CONTINUE
C
c  set boundary conditions
      DO 310 K=2,N2
      DO 310 J=2,M2
        KBCL1(J,K) = 2
c      - for the manganin
        IF (Y(J).GE.rlkapt.AND.Z(K).GE.rlkapt)
     +    FLXCL1(J,K) = Q
310   CONTINUE
C
      DO 320 K=2,N2
      DO 320 I=2,L2
        KBCJ1(I,K)=2
        KBCM1(I,K)=2
320   CONTINUE
C
      DO 330 J=2,M2
      DO 330 I=2,L2
        KBCK1(I,J)=2
        KBCN1(I,J)=2
330   CONTINUE
C
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Appendix C

The Fortran Program MC.FOR

This program, MC.FOR, performs a Monte-Carlo analysis of the HTS thermal bridges designed by Kasey M. Lee. The distribution factors D_{ij} are computed in order to then determine the radiative heat transfer in the enclosure and more specifically on the substrate material. Therefore MC.FOR has the ability to increase the number of rings in the substrate material. This program was developed by Sandrine Garcia, 1994.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Program MCHTS
C
C
C     Name of the main variables used in this code
C      integer
C      * nelemt : number of surfaces in the enclosure
C      * nrings : number of rings in the substrate material
C      * nbundles : number of energy bundles emitted by each surface
C      * countelt : counter of energy bundles emitted by each surface
C      * i1 : emitting surface
C      * i2 : bundle number
C      * i : source surface
C      * j : surface striked by an energy bundle
C      * k(nelemt) : determines the z coordinate of each rings in the substrate
C                   material
C      * seed : implemented to the random number generator urand()
C
C      real
C      * countDij(nelemt,nelemt) : number of energy bundle emitted by surface i
C                                  and absorbed by surface j
```

```
C     * Dij(nelemt,nelemt) : distribution factor
C     * Dtot : summation of all Dij; should be equal to 1
C     * alpha : angle between surfaces 1 and 2 in the single housing chamber;
C              since it is a third of a cylinder, alpha=2*pi/3
C     * H : Height of the thermal bridge
C     * R : radius of the single housing chamber
C     * l : width of the substrate material
C     * e : thickness of the substrate material
C     * (xa,ya) : coordinates of point A in the plane (x-y)
C     * (xb,yb) : coordinates of point B      "
C     * (xc,yc) : coordinates of point C      "
C     * (xd,yd) : coordinates of point D      "
C     * (xe,ye) : coordinates of point E      "
C     * (x1,y1,z1) : coordinates of the point of emission
C     * (x2,y2,z2) : coordinates of the point of intersection between the energy
C                bundle emitted from point 1 on surface i and surface j
C     * (l1,m1,n1) : direction cosines of the energy bundle unit vector emitted
C                from surface i
C     * (l2,m2,n2) : direction cosines of the energy bundle unit vector reflected
C                on surface j
C     * emiss(nelemt) : emissivity of each surface
C     * absorpt(nelemt) : absorptivity of each surface
C     * ratio(nelemt) : reflectivity ratio of each surface
C
      integer nelemt,nrings,nbundles,countelt(69),max,
     +        i,i1,i2,j,k(69),seed1,seed2
      double precision Dij(69,69),Dtot(69),countDij(69,69),
     +   pi,alpha,H,R,l,e,xa,ya,xb,yb,xc,yc,xd,yd,xe,ye,
     +   l1,m1,n1,l2,m2,n2,x1,y1,z1,x2,y2,z2,Lmin,
     +   A(69),emiss(69),absorpt(69),ratio(69),urand
C
      COMMON/RANDOM/seed1,seed2
      COMMON/GEOM1/nrings,alpha,H,R,pi
      COMMON/GEOM2/l,e
      COMMON/POINTABC/xa,ya,xb,yb,xc,yc
      COMMON/POINTDE/xd,yd,xe,ye
      COMMON/PROP1/A,emiss,nelemt
      COMMON/PROP2/absorpt,ratio,nbundles
      COMMON/COUNTER/countDij,Dij,Dtot,countelt
      COMMON/SUBSTRATE/k
      COMMON/POINT1/x1,y1,z1
      COMMON/POINT2/x2,y2,z2
      COMMON/DIRECTION1/l1,m1,n1
      COMMON/DIRECTION2/l2,m2,n2
C
C     set constants
      call const
C
      open(10,file='100m16n.out')
      open(20,file='100m16n.dat')
C     output title :
      write(*,*)'MCproject running ...'
      write(10,*)'output : 100m16n.out'
      write(10,*)'emiss1(housing)=0.020'
      write(10,*)'absorpt1(housing)=0.020'
```

```
      write(10,*)'ratio1(housing)=0.95'
      write(10,*)'emiss2(substrate)=0.80'
      write(10,*)'absorpt2(substrate)=0.80'
      write(10,*)'ratio2(substrate)=0.15'
      write(10,*)'nrings = 16'
      write(10,*)'nbundles = 100000'
      write(10,*)'seed1 = 12056'
      write(10,*)'seed2 = 08013'
C
CCC   DO-LOOP TO STUDY EMISSION FROM EACH SURFACE
C       in the single housing chamber :
C          * surfaces 1 and 2 are the flat vertical surfaces
C          * surface  3 is the cylindrical wall
C       * surfaces 4 and 5 are the bottom and the top surfaces
C     in the substrate material :
C       * surfaces 6 to 6+nrings-1 are the front surfaces
C       * surfaces 6+nrings to 6+2*nrings-1 are the back surfaces
C       * surfaces 6+2*nrings to 6+3*nrings-1 are the left surfaces
C       * surfaces 6+3*nrings to 6+4*nrings-1 are the right surfaces
C
      DO 999 i1=1,nelemt
C     to follow the progress of the program on the  screen
      write(*,*)'i1= ',i1
      max=nbundles
C     initialize countelt(i) and countDij(i,j)
      countelt(i1)=0
      do j=1,nelemt
       countDij(i1,j)=0.
      enddo
C
CC    do-loop on the number of energy bundles
      do 1 i2=1,max
       countelt(i1)=countelt(i1)+1
C
C     step 1 : locate point of emission on surface i
      i=i1
      call ptofem(i)
C
C     step 2 : find the direction of emission
2     call fdirofem(i)
      if (dabs(n1).lt.1d-10) go to 2
C
C     step 3 : find where the emitted energy bundle strikes the enclosure wall
C           and identify which surface j the energy bundle striked
3     call finterse(i1,i2,i,j,Lmin)
      if (Lmin.eq.10d10) then
       countelt(i1)=countelt(i1)-1
       max=max+1
       go to 1
      endif
C
C     step 4 : is the energy bundle absorbed or reflected on surface j ?
4     if (absorpt(j).ge.urand()) then
C     the energy bundle is absorbed
       countDij(i1,j)=countDij(i1,j)+1
```

```
       go to 1
      else
C      the energy bundle is reflected
       go to 5
      endif
C
C      step 5 : is the reflection diffuse or specular ?
5     if (ratio(j).lt.urand()) then
C      the reflection is diffuse
C      consider emission from the point (x2,y2,z2) on surface j
       x1=x2
       y1=y2
       z1=z2
       i=j
       go to 2
      else
C      the reflection is specular
      call specular(j)
C      consider now emission from the point(x2,y2,z2) on surface j with
C      the direction cosines (l2,m2,n2)
       x1=x2
       y1=y2
       z1=z2
       l1=l2
       m1=m2
       n1=n2
       i=j
       go to 3
      endif
1     continue
999   CONTINUE
C
C     OUTPUT SOLUTION
      call output
      close(10)
      close(20)
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine const
C     This subroutine set the constants of the problem
C
      integer nbundles,nrings,nelemt,i,k(69),seed1,seed2
      double precision pi,alpha,H,R,l,e,xa,ya,xb,yb,xc,yc,xd,yd,xe,ye,
     + emiss1,absorpt1,ratio1,emiss2,absorpt2,ratio2,
     + A(69),emiss(69),absorpt(69),ratio(69),
     + t,beta
C
      COMMON/RANDOM/seed1,seed2
      COMMON/GEOM1/nrings,alpha,H,R,pi
      COMMON/GEOM2/l,e
      COMMON/POINTABC/xa,ya,xb,yb,xc,yc
      COMMON/POINTDE/xd,yd,xe,ye
      COMMON/PROP1/A,emiss,nelemt
      COMMON/PROP2/absorpt,ratio,nbundles
```

```
      COMMON/SUBSTRATE/k
C
C     geometric constants :
      pi=dacos(-1.d0)
      alpha=2.d0*pi/3.d0
      H=15.24d-2
      l=9.144d-3
      e=0.1524d-3
      R=5.d-2
      t=dsqrt((l/2.d0)**2+((R-e)/2.d0)**2)
      beta=datan(l/(R-e))
      xa=t*cos(alpha/2.d0-beta)
      ya=t*sin(alpha/2.d0-beta)
      xb=xa+e*cos(alpha/2.d0)
      yb=ya+e*sin(alpha/2.d0)
      xc=t*cos(alpha/2.d0+beta)
      yc=t*sin(alpha/2.d0+beta)
      xd=xc+e*cos(alpha/2.d0)
      yd=yc+e*sin(alpha/2.d0)
      xe=-R*sin(alpha-pi/2.d0)
      ye=R*cos(alpha-pi/2.d0)
      nbundles=100000
      nrings=16
      nelemt=5+4*nrings
C
C     seed for the random number generator :
      seed1=12056
      seed2=08013
      call rmarin(seed1, seed2)
C
C     radiative properties :
C     in the single housing chamber
      emiss1=0.020d0
      absorpt1=0.020d0
      ratio1=0.95d0
C     in the substrate
      emiss2=0.80d0
      absorpt2=0.80d0
      ratio2=0.15d0
      do i=1,nelemt
       if (i.le.5) then
          emiss(i)=emiss1
          absorpt(i)=absorpt1
          ratio(i)=ratio1
       else
          emiss(i)=emiss2
          absorpt(i)=absorpt2
          ratio(i)=ratio2
       endif
      enddo
C
C     define function k used for the surfaces in the substrate material
      do i=1,6
       k(i)=0
      enddo
```

```
      do i=7,6+nrings-1
       k(i)=k(i-1)+1
      enddo
      k(6+nrings)=0
      do i=7+nrings,6+2*nrings-1
       k(i)=k(i-1)+1
      enddo
      k(6+2*nrings)=0
      do i=7+2*nrings,6+3*nrings-1
       k(i)=k(i-1)+1
      enddo
      k(6+3*nrings)=0
      do i=7+3*nrings,6+4*nrings-1
       k(i)=k(i-1)+1
      enddo
C
C     area of each surface
      A(1)=H*R
      A(2)=H*R
      A(3)=alpha*R*H
      A(4)=pi*R**2/3.d0
      A(5)=pi*R**2/3.d0
      do i=6,(6+4*nrings-1)
        if (6.le.i.and.i.le.(6+2*nrings-1)) A(i)=l*H/nrings
        if ((6+2*nrings).le.i.and.i.le.(6+4*nrings-1)) A(i)=e*H/nrings
      enddo
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine ptofem(i)
C     This subroutine determines the point of emission on surface i
C
      integer i,k(69),seed1,seed2,nrings
      double precision R,H,l,e,alpha,pi,xa,ya,xb,yb,xc,yc,x1,y1,z1,
     + urand,rand1,rand2,const1,const2
C
      COMMON/RANDOM/seed1,seed2
      COMMON/GEOM1/nrings,alpha,H,R,pi
      COMMON/GEOM2/l,e
      COMMON/POINTABC/xa,ya,xb,yb,xc,yc
      COMMON /SUBSTRATE/k
      COMMON/POINT1/x1,y1,z1
C
C     if surface 1 is emitting :
      if (i.eq.1) then
       x1=R*urand()
       y1=0.d0
       z1=H*urand()
      else
C
C     if surface 2 is emitting :
       if (i.eq.2) then
      rand1=urand()
      x1=-cos(alpha/2.d0)*R*rand1
```

```
      y1=sin(alpha/2.d0)*R*rand1
      z1=H*urand()
      else
C
C     if surface 3 is emitting :
      if (i.eq.3) then
       rand1=urand()
       x1=R*dcos(alpha*rand1)
       y1=R*dsin(alpha*rand1)
       z1=H*urand()
      else
C
C     if surface 4 is emitting :
       if (i.eq.4) then
10      rand1=urand()
        rand2=urand()
        x1=R*dsqrt(rand1)*dcos(alpha*rand2)
        y1=R*dsqrt(rand1)*dsin(alpha*rand2)
        z1=0.0
C       neglect the case where the point of emission is in the 'hole'
C       corresponding to the substrate material
          const1=((x1-xa)*(yb-ya)-(y1-ya)*(xb-xa))/
     +           ((xc-xa)*(yb-ya)-(yc-ya)*(xb-xa))
          const2=((x1-xa)-const1*(xc-xa))/(xb-xa)
          if (0.d0.lt.const1.and.const1.lt.1.and.
     +        0.d0.lt.const2.and.const2.lt.1) go to 10
        else
C
C       if surface 5 is emitting :
        if (i.eq.5) then
11        rand1=urand()
          rand2=urand()
          x1=R*dsqrt(rand1)*dcos(alpha*rand2)
          y1=R*dsqrt(rand1)*dsin(alpha*rand2)
          z1=H
C       neglect the case where the point of emission is in the 'hole'
C       corresponding to the substrate material
          const1=((x1-xa)*(yb-ya)-(y1-ya)*(xb-xa))/
     +           ((xc-xa)*(yb-ya)-(yc-ya)*(xb-xa))
          const2=((x1-xa)-const1*(xc-xa))/(xb-xa)
          if (0.d0.lt.const1.and.const1.lt.1.and.
     +        0.d0.lt.const2.and.const2.lt.1) go to 11
        else
C
C     if a surface on the front side of the substrate material is emitting :
        if (6.le.i.and.i.le.(6+nrings-1)) then
         rand1=urand()
         x1=-sin(alpha/2.d0)*l*rand1+xb
         y1=cos(alpha/2.d0)*l*rand1+yb
         z1=H/nrings*urand()+k(i)*H/nrings
        else
C
C     if a surface on the back side of the substrate material is emitting :
        if ((6+nrings).le.i.and.i.le.(6+2*nrings-1)) then
         rand1=urand()
```

```
         x1=-sin(alpha/2.d0)*l*rand1+xa
         y1=cos(alpha/2.d0)*l*rand1+ya
         z1=H/nrings*urand()+k(i)*H/nrings
        else
C
C    if a surface on the left side of the substrate material is emitting :
          if ((6+2*nrings).le.i.and.i.le.(6+3*nrings-1)) then
           rand1=urand()
           x1=cos(alpha/2.d0)*e*rand1+xa
           y1=sin(alpha/2.d0)*e*rand1+ya
           z1=H/nrings*urand()+k(i)*H/nrings
          else
C
C    if a surface on the right side of the substrate material is emitting :
           if ((6+3*nrings).le.i.and.i.le.(6+4*nrings-1)) then
            rand1=urand()
            x1=cos(alpha/2.d0)*e*rand1+xc
            y1=sin(alpha/2.d0)*e*rand1+yc
            z1=H/nrings*urand()+k(i)*H/nrings
              endif
             endif
            endif
           endif
          endif
         endif
        endif
       endif
      endif
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine adjust(l,m,n)
C     This subroutine adjusts the direction cosines to avoid accumulated errors
C
      double precision l,m,n,o
C
      o=dsqrt(l**2+m**2+n**2)
      l=l/o
      m=m/o
      n=n/o
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine fdirofem(i)
C     This subroutine determines the direction cosines (l1,m1,n1) of emission
C     from surface i
C
      integer i,i2,k(69),nrings,seed1,seed2
      double precision H,alpha,R,pi,x1,y1,z1,xa,ya,xb,yb,xc,yc,urand,
     + l1,m1,n1,theta,phi
C
      COMMON/RANDOM/seed1,seed2
      COMMON/GEOM1/nrings,alpha,H,R,pi
```

```
      COMMON/POINTABC/xa,ya,xb,yb,xc,yc
      COMMON /SUBSTRATE/k
      COMMON/POINT1/x1,y1,z1
      COMMON/DIRECTION1/l1,m1,n1
C
C     for a diffuse emitter, the angles theta and phi giving the direction of
C     emission in the enclosure are known :
      theta=dasin(dsqrt(urand()))
      phi=2.d0*pi*urand()
C
C     if surface 1 is emitting :
      if (i.eq.1) then
       l1=sin(theta)*sin(phi)
       m1=cos(theta)
       n1=sin(theta)*cos(phi)
C       check n1.s1>0
       if (m1.lt.0.d0) write(10,*)'surf1 : n1.s1<0 !'
      else
C
C     if surface 2 is emitting :
       if (i.eq.2) then
        l1=cos(alpha/2.d0)*sin(theta)*sin(phi)+
     +     sin(alpha/2.d0)*cos(theta)
        m1=cos(alpha/2.d0)*cos(theta)-
     +     sin(alpha/2.d0)*sin(theta)*sin(phi)
        n1=sin(theta)*cos(phi)
C        check n1.s1>0
        if ((sin(alpha/2.d0)*l1+cos(alpha/2.d0)*m1).lt.0.d0)
     +    write(10,*)'surf2 : n1.s1<0 !'
       else
C
C     if surface 4 is emitting :
        if (i.eq.4) then
         l1=-sin(theta)*sin(phi)
         m1=sin(theta)*cos(phi)
         n1=cos(theta)
C         check n1.s1>0
         if (n1.lt.0.d0) write(10,*)'surf4 : n1.s1<0 !'
        else
C
C     if surface 5 is emitting :
         if (i.eq.5) then
          l1=sin(theta)*sin(phi)
          m1=sin(theta)*cos(phi)
          n1=-cos(theta)
C          check n1.s1>0
          if (-n1.lt.0.d0) write(10,*)'surf5 : n1.s1<0 !'
         else
C
C    if a surface on the front side of the substrate material is emitting :
          if (6.le.i.and.i.le.(6+nrings-1)) then
           l1=sin(alpha/2.d0)*sin(theta)*sin(phi)+
     +        cos(alpha/2.d0)*cos(theta)
           m1=sin(alpha/2.d0)*cos(theta)-
     +        cos(alpha/2.d0)*sin(theta)*sin(phi)
```

```
        n1=sin(theta)*cos(phi)
C        check n1.s1>0
        if ((cos(alpha/2.d0)*l1+sin(alpha/2.d0)*m1).lt.0.d0)
     +     write(10,*)'case6 : n1.s1<0 !'
       else
C
C   if a surface on the back side of the substrate material is emitting :
        if ((6+nrings).le.i.and.i.le.(6+2*nrings-1)) then
         l1=-sin(alpha/2.d0)*sin(theta)*sin(phi)-
     +       cos(alpha/2.d0)*cos(theta)
         m1=-sin(alpha/2.d0)*cos(theta)+
     +       cos(alpha/2.d0)*sin(theta)*sin(phi)
         n1=sin(theta)*cos(phi)
C        check n1.s1>0
        if ((-cos(alpha/2.d0)*l1-sin(alpha/2.d0)*m1).lt.0.d0)
     +     write(10,*)'case7 : n1.s1<0 !'
       else
C
C   if a surface on the left side of the substrate material is emitting :
         if ((6+2*nrings).le.i.and.i.le.(6+3*nrings-1)) then
          l1=sin(alpha/2.d0)*cos(theta)-
     +       cos(alpha/2.d0)*sin(theta)*sin(phi)
          m1=-cos(alpha/2.d0)*cos(theta)-
     +       sin(alpha/2.d0)*sin(theta)*sin(phi)
          n1=sin(theta)*cos(phi)
C         check n1.s1>0
        if ((sin(alpha/2.d0)*l1-cos(alpha/2.d0)*m1).lt.0.d0)
     +     write(10,*)'case8 : n1.s1<0 !'
         else
C
C   if a surface on the right side of the substrate material is emitting :
          if ((6+3*nrings).le.i.and.i.le.(6+4*nrings-1)) then
           l1=-sin(alpha/2.d0)*cos(theta)+
     +       cos(alpha/2.d0)*sin(theta)*sin(phi)
           m1=+cos(alpha/2.d0)*cos(theta)+
     +       sin(alpha/2.d0)*sin(theta)*sin(phi)
           n1=sin(theta)*cos(phi)
C         check n1.s1>0
        if ((-sin(alpha/2.d0)*l1+cos(alpha/2.d0)*m1).lt.0.d0)
     +     write(10,*)'case9 : n1.s1<0 !'
         else
C
C    if surface 3 is emitting :
          if (i.eq.3) then
            n1=sin(theta)*cos(phi)
            m1=sin(theta)*sin(phi)*x1/R-cos(theta)*y1/R
            l1=-sin(theta)*sin(phi)*y1/R-cos(theta)*x1/R
C        check n1.s1>0
       if ((-x1/R*l1-y1/R*m1).lt.0.d0) write(10,*)'surf3 : n1.s1<0 !'
           endif
          endif
         endif
        endif
       endif
      endif
```

```
      endif
     endif
    endif
C
C    adjust the direction cosines to avoid accumulated errors
    call adjust(l1,m1,n1)
C
    return
    end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    subroutine chedim45(x2,y2,z2,alpha,pi,xe,R,L2)
C    This subroutine checks if point 2 (x2,y2,z2) is on surface 4 or 5.
C    If the coordinates satisfy the surface limits then L2 is computed.
C
    integer rep
    double precision x1,y1,z1,x2,y2,z2,alpha,pi,xe,R,L2,
   +  const1,const2,xa,ya,xb,yb,xc,yc,length,beta
C
    COMMON/POINTABC/xa,ya,xb,yb,xc,yc
    COMMON/POINT1/x1,y1,z1
C
    length=dsqrt(x2**2+y2**2)
    if (length.le.R.and.(0.d0).le.y2.and.y2.le.R.and.
   +    xe.le.x2.and.x2.le.R) then
C     neglect the case where point 2 is in the 'hole'corresponding to the
C     substrate material
      const1=((x1-xa)*(yb-ya)-(y1-ya)*(xb-xa))/
   +          ((xc-xa)*(yb-ya)-(yc-ya)*(xb-xa))
      const2=((x1-xa)-const1*(xc-xa))/(xb-xa)
      if (0.d0.lt.const1.and.const1.lt.1.and.
   +       0.d0.lt.const2.and.const2.lt.1) go to 45
     if (x2.lt.(0.d0)) then
       if (y2.lt.(1.d-10)) y2=1d-10
       beta=datan(dabs(x2/y2))
       if (beta.lt.(alpha-pi/2.d0)) then
         rep=0
       else
         rep=1
       endif
     endif
     if (x2.ge.(0.d0)) rep=0
    else
     rep=1
    endif
    if (rep.eq.0) L2=dsqrt((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)
C
45   return
    end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    subroutine chedim289(x2,y2,z2,H,a,b,c,d,L2)
C    This subroutine checks if point 2 (x2,y2,z2) is on surface 2, or
C    on the left or right side of the substrate material.
C    If the coordinates satisfy the surface limits then L2 is computed.
C
    double precision x2,y2,z2,x1,y1,z1,H,a,b,c,d,L2
```

```
C
      COMMON/POINT1/x1,y1,z1
C
      if ((0.d0).lt.z2.and.z2.lt.H.and.a.le.x2.and.x2.le.b.
     +    and.c.le.y2.and.y2.le.d) then
       L2=dsqrt((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)
      endif
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine chedim67(x2,y2,z2,H,a,b,c,d,L2)
C     This subroutine checks if point 2 (x2,y2,z2) is on the front or back side
C     of the substrate material.
C     If the coordinates satisfy the surface limits then L2 is computed.
C
      double precision x2,y2,z2,x1,y1,z1,H,a,b,c,d,L2
C
      COMMON/POINT1/x1,y1,z1
C
      if ((0.d0).lt.z2.and.z2.lt.H.and.a.lt.x2.and.x2.lt.b.
     +    and.c.lt.y2.and.y2.lt.d) then
       L2=dsqrt((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)
      endif
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine chedim1(x2,y2,z2,H,a,b,L2)
C     This subrourine cheks if point 2 (x2,y2.z2) is on surface 1.
C     If the coordinates satisfy the surface limits then L2 is computed.
C
      double precision x2,y2,z2,x1,y1,z1,H,a,b,L2
C
      COMMON/POINT1/x1,y1,z1
C
      if ((0.d0).lt.z2.and.z2.lt.H.and.a.lt.x2.and.x2.le.b) then
       L2=dsqrt((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)
      endif
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine chedim3(x2,y2,z2,xe,H,R,L2)
C     This subrourine cheks if point 2 (x2,y2.z2) is on surface 3.
C     If the coordinates satisfy the surface limits then L2 is computed.
C
      double precision x2,y2,z2,x1,y1,z1,xe,H,R,L2
C
      COMMON/POINT1/x1,y1,z1
C
      if ((0.d0).lt.z2.and.z2.lt.H.and.(0.d0).le.y2.and.y2.le.R.and.
     +    xe.lt.x2.and.x2.le.R) then
      L2=dsqrt((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)
      endif
```

```
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine quadsolve(i1,i2,i,R,root)
C      this subroutine gives the roots of the equation ax**2+b*x+c=0
C
      integer i1,i2,i
      double precision a,b,c,d,root(2),x1,y1,z1,l1,m1,n1,R
C
      COMMON/POINT1/x1,y1,z1
      COMMON/DIRECTION1/l1,m1,n1
C
      a=(l1/n1)**2+(m1/n1)**2
      b=2.d0/n1*(l1*(x1-l1*z1/n1)+m1*(y1-m1*z1/n1))
      c=(x1-l1*z1/n1)**2+(y1-m1*z1/n1)**2-R**2
      d=(b**2-4.d0*a*c)
      if (d.lt.0) then
        write(10,*)'quadsolve : complex roots !'
        write(10,*)'i1= ',i1
        write(10,*)'i2= ',i2
        write(10,*)'i= ',i
        write(10,*)'d= ',d
        stop
      else
        root(1)=(-b-dsqrt(d))/(2.d0*a)
        root(2)=(-b+dsqrt(d))/(2.d0*a)
      endif
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine finterse(i1,i2,i,j,Lmin)
C      This subroutine finds point 2 (x2,y2,z2) where the emitted energy bundle
C      strikes the enclosure wall.
C      It also identifies which surface j the energy bundle striked
C
      integer i,k,k2,nrings,j,i2,i1
      double precision pi,x1,y1,z1,l1,m1,n1,H,R,alpha,x2,y2,z2,Lmin,
     +      x(9),y(9),z(9),L(9),
     +  root(2),xa,ya,xb,yb,xc,yc,xd,yd,xe,ye
C
      COMMON/GEOM1/nrings,alpha,H,R,pi
      COMMON/POINTABC/xa,ya,xb,yb,xc,yc
      COMMON/POINTDE/xd,yd,xe,ye
      COMMON/POINT1/x1,y1,z1
      COMMON/POINT2/x2,y2,z2
      COMMON/DIRECTION1/l1,m1,n1
C
C      initialize all distances to a big value 10d10
      do k=1,9
        L(k)=10.d10
      enddo
C
C      case 1 : the energy bundle strikes surface 1
```

```
C     if the emitting surface is 1 or a surface on the right side of
C     the substrate material, then reject case  1
      if ((i.eq.1).or.((6+3*nrings).le.i.and.i.le.(6+4*nrings-1))) then
       go to 102
      endif
101   y(1)=0.d0
      z(1)=z1-n1*y1/m1
      x(1)=l1*z(1)/n1+x1-l1*z1/n1
C     check if (x2,y2,z2) is on surface 1
      call chedim1(x(1),y(1),z(1),H,(0.d0),R,L(1))
C
C     case 2 : the energy bundle strikes surface 2
C     if the emitting surface is 2 or a surface on the left side of
C     the substrate material, then reject case 2
      if ((i.eq.2).or.((6+2*nrings).le.i.and.i.le.(6+3*nrings-1))) then
       go to 103
      endif
102   z(2)=n1/(sin(alpha/2.d0)*l1+cos(alpha/2.d0)*m1)*(sin(alpha/2.d0)*
     +      (l1*z1/n1-x1)+cos(alpha/2.d0)*(m1*z1/n1-y1))
      x(2)=l1*z(2)/n1+x1-l1*z1/n1
      y(2)=m1*z(2)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on surface 2
      call chedim289(x(2),y(2),z(2),H,xe,(0.d0),(0.d0),ye,L(2))
C
C     case 3 : the energy bundle strikes surface 3
C     if the emitting surface is on the back side of the substrate,
C     then reject case 3
      if ((6+nrings).le.i.and.i.le.(6+2*nrings-1)) then
       go to 104
      endif
103   call quadsolve(i1,i2,i,R,root)
      z(3)=root(1)
      x(3)=l1*root(1)/n1+x1-l1*z1/n1
      y(3)=m1*root(1)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on surface 3
      call chedim3(x(3),y(3),z(3),xe,H,R,L(3))
C     if (x2,y2,z2) is not on surface 3 or if (x2,y2,z2)=(x1,y1,z1),
C     then reject root(1) and study root(2)
      if ((L(3).lt.1d-10).or.(L(3).eq.10d10)) then
      z(3)=root(2)
      x(3)=l1*root(2)/n1+x1-l1*z1/n1
      y(3)=m1*root(2)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on surface 3
      call chedim3(x(3),y(3),z(3),xe,H,R,L(3))
C     if (x2,y2,z2)=(x1,y1,z1), then reject the solution
      if (L(3).lt.1d-10) L(3)=10d10
      endif
C
C     case 4 : the energy bundle strikes surface 4
C     if the emitting surface is 4, then reject case 4
      if (i.eq.4) then
       go to 105
      endif
104   z(4)=0.d0
      x(4)=x1-l1*z1/n1
```

```
      y(4)=y1-m1*z1/n1
C     check if (x2,y2,z2) is on surface 4
      call chedim45(x(4),y(4),z(4),alpha,pi,xe,R,L(4))
C
C     case 5 : the energy bundle strikes surface 5
C     if the emitting surface is 5, then reject case 5
      if (i.eq.5) then
        go to 106
      endif
105   z(5)=H
      x(5)=l1*z(5)/n1+x1-l1*z1/n1
      y(5)=m1*z(5)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on surface 5
      call chedim45(x(5),y(5),z(5),alpha,pi,xe,R,L(5))
C
C     if the emitting surface is in the substrate material,
C     then reject cases 6,7,8,9
      if (6.le.i.and.i.le.(6+4*nrings-1)) then
      go to 110
      endif
C
C     case 6 : the energy bundle strikes the front surfaces of the substrate
C           material (surfaces 6 to 6+nrings-1)
106   z(6)=n1/(cos(alpha/2.d0)*l1+sin(alpha/2.d0)*m1)*(cos(alpha/2.d0)*
     +      (xb+l1*z1/n1-x1)+sin(alpha/2.d0)*(yb+m1*z1/n1-y1))
      x(6)=l1*z(6)/n1+x1-l1*z1/n1
      y(6)=m1*z(6)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on the front side of the substrate material
      call chedim67(x(6),y(6),z(6),H,xd,xb,yb,yd,L(6))
C
C     case 7 : the energy bundle strikes the back surfaces of the substrate
C           material (surfaces 6+nrings to 6+2nrings-1)
C     if the emitting surface is 3, then reject case 7
      if (i.eq.3) then
        go to 108
      endif
107   z(7)=n1/(cos(alpha/2.d0)*l1+sin(alpha/2.d0)*m1)*(cos(alpha/2.d0)*
     +      (xa+l1*z1/n1-x1)+sin(alpha/2.d0)*(ya+m1*z1/n1-y1))
      x(7)=l1*z(7)/n1+x1-l1*z1/n1
      y(7)=m1*z(7)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on the back side of the substrate material
      call chedim67(x(7),y(7),z(7),H,xc,xa,ya,yc,L(7))
C
C     case 8 : the energy bundle strikes the left surfaces of the substrate
C           material (surfaces 6+2nrings to 6+3nrings-1)
C     if the emitting surface is 2, then reject case 8
      if (i.eq.2) then
        go to 109
      endif
108   z(8)=n1/(-sin(alpha/2.d0)*l1+cos(alpha/2.d0)*m1)*
     +      (-sin(alpha/2.d0)*(xa+l1*z1/n1-x1)+
     +      cos(alpha/2.d0)*(ya+m1*z1/n1-y1))
      x(8)=l1*z(8)/n1+x1-l1*z1/n1
      y(8)=m1*z(8)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on the left side of the substrate material
```

```
      call chedim289(x(8),y(8),z(8),H,xa,xb,ya,yb,L(8))
C
C     case 9 : the energy bundle strikes the right surfaces of the substrate
C              material (surfaces 6+3nrings to 6+4nrings-1)
C     if the emitting surface is 1, then reject case 9
      if (i.eq.1) then
        go to 110
      endif
109   z(9)=n1/(-sin(alpha/2.d0)*l1+cos(alpha/2.d0)*m1)*
     +      (-sin(alpha/2.d0)*(xc+l1*z1/n1-x1)+
     +      cos(alpha/2.d0)*(yc+m1*z1/n1-y1))
      x(9)=l1*z(9)/n1+x1-l1*z1/n1
      y(9)=m1*z(9)/n1+y1-m1*z1/n1
C     check if (x2,y2,z2) is on the right side of the substrate material
      call chedim289(x(9),y(9),z(9),H,xc,xd,yc,yd,L(9))
C
C     find the shortest length of all possible solutions :
110   Lmin=MIN(L(1),L(2),L(3),L(4),L(5),L(6),L(7),L(8),L(9))
      If (Lmin.eq.L(1)) k2=1
      If (Lmin.eq.L(2)) k2=2
      If (Lmin.eq.L(3)) k2=3
      If (Lmin.eq.L(4)) k2=4
      If (Lmin.eq.L(5)) k2=5
      If (Lmin.eq.L(6)) k2=6
      If (Lmin.eq.L(7)) k2=7
      If (Lmin.eq.L(8)) k2=8
      If (Lmin.eq.L(9)) k2=9
C     the correct point of intersection corresponds to the shortest lenght
      x2=x(k2)
      y2=y(k2)
      z2=z(k2)
C
C     now identify the surface for (x2,y2,z2)
      if (k2.eq.1) j=1
      if (k2.eq.2) j=2
      if (k2.eq.3) j=3
      if (k2.eq.4) j=4
      if (k2.eq.5) j=5
      if (k2.eq.6) then
      do k=0,(nrings-1)
        if ((k*H/nrings).le.z2.and.z2.lt.((k+1)*H/nrings)) j=k+6
      enddo
      else
       if (k2.eq.7) then
       do k=0,(nrings-1)
         if ((k*H/nrings).le.z2.and.z2.lt.((k+1)*H/nrings))
     +       j=(k+nrings)+6
       enddo
       else
       if (k2.eq.8) then
         do k=0,(nrings-1)
           if ((k*H/nrings).le.z2.and.z2.lt.((k+1)*H/nrings))
     +       j=(k+2*nrings)+6
         enddo
       else
```

```
      if (k2.eq.9) then
        do k=0,(nrings-1)
          if ((k*H/nrings).le.z2.and.z2.lt.((k+1)*H/nrings))
     +         j=(k+3*nrings)+6
        enddo
      endif
     endif
     endif
    endif
C
    return
    end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    subroutine fcos2(l,m,n,l2,m2,n2)
C    This subroutine determines the direction (l2,m2,n2) of reflection on a surface.
C    The direction (l1,m1,n1) of emission  and the normal unit vector to the
C    surface are known.
C
    double precision l1,m1,n1,l,m,n,l2,m2,n2
C
    COMMON/DIRECTION1/l1,m1,n1
C
    l2=l1-2.d0*l*(l1*l+m1*m+n1*n)
    m2=m1-2.d0*m*(l1*l+m1*m+n1*n)
    n2=n1-2.d0*n*(l1*l+m1*m+n1*n)
C
    return
    end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    subroutine specular(j)
C    For a given direction (l1,m1,n1) of emission, this subroutine computes
C    the direction (l2,m2,n2) of reflection on each surface of the enclosure.
C    (l,m,n) are the coordinates of the normal unit vector to the surface.
C
    integer j,nrings
    double precision l1,m1,n1,l2,m2,n2,x2,y2,z2,l,m,n,alpha,H,R,pi
C
    COMMON/DIRECTION1/l1,m1,n1
    COMMON/DIRECTION2/l2,m2,n2
    COMMON/POINT2/x2,y2,z2
    COMMON/GEOM1/nrings,alpha,H,R,pi
C
C    if reflection occurs on surface 1 :
    if (j.eq.1) then
    l=0.d0
    m=1.d0
    n=0.d0
     call fcos2(l,m,n,l2,m2,n2)
    else
C
C    if reflection occurs on surface 2 :
     if (j.eq.2) then
     l=sin(alpha/2.d0)
     m=cos(alpha/2.d0)
     n=0.d0
```

```
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on surface 3 :
     if (j.eq.3) then
      l=-x2/R
      m=-y2/R
      n=0.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on surface 4 :
     if (j.eq.4) then
      l=0.d0
      m=0.d0
      n=1.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on surface 5 :
     if (j.eq.5) then
      l=0.d0
      m=0.d0
      n=-1.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on the front side of the substrate material :
     if (6.le.j.and.j.le.(6+nrings-1)) then
      l=cos(alpha/2.d0)
      m=sin(alpha/2.d0)
      n=0.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on the back side of the substrate material :
     if ((6+nrings).le.j.and.j.le.(6+2*nrings-1)) then
      l=-cos(alpha/2.d0)
      m=-sin(alpha/2.d0)
      n=0.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on the left side of the substrate material :
     if ((6+2*nrings).le.j.and.j.le.(6+3*nrings-1)) then
      l=sin(alpha/2.d0)
      m=-cos(alpha/2.d0)
      n=0.d0
      call fcos2(l,m,n,l2,m2,n2)
     else
C
C    if reflection occurs on the right side of the substrate material :
     if ((6+3*nrings).le.j.and.j.le.(6+4*nrings-1)) then
     l=-sin(alpha/2.d0)
     m=cos(alpha/2.d0)
```

```
          n=0.d0
          call fcos2(l,m,n,l2,m2,n2)
           endif
          endif
         endif
        endif
       endif
      endif
     endif
      endif
     endif
C
C     adjust the direction cosines to avoid accumulated errors
      call adjust(l2,m2,n2)
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine output
C     This subroutine computes the distribution factors Dij and Dtot.
C     It also checks the convergence of the solution.
C
      integer i,j,countelt(69),nelemt
      double precision emiss(69),A(69),countDij(69,69),Dij(69,69),
     +                 Dtot(69),error(69),E
C
      COMMON/COUNTER/countDij,Dij,Dtot,countelt
      COMMON/PROP1/A,emiss,nelemt
C
C     computation of the distribution factors Dij and Dtot
      do i=1,nelemt
        Dtot(i)=0.d0
        do j=1,nelemt
          Dij(i,j)=countDij(i,j)/countelt(i)
          Dtot(i)=Dtot(i)+Dij(i,j)
          write(10,*)'D(',i,',',j,')= ',Dij(i,j)
          write(20,*)Dij(i,j)
        enddo
        write(10,*)'Dtot(',i,')= ',Dtot(i)
      enddo
C
C     check convergence of the solution
C     the 'error' for each surface and the weighted 'Error' are computed :
      do i=1,nelemt
       error(i)=0.d0
       do j=1,nelemt
         error(i)=error(i)+emiss(j)*A(j)*Dij(j,i)
       enddo
       error(i)=error(i)/emiss(i)/A(i)-1.d0
      enddo
      E=0.d0
      Asum=0.d0
      do i=1,nelemt
        E=E+A(i)*error(i)
        Asum=Asum+A(i)
```

```
      enddo
      E=dabs(E/Asum)
      write(10,*)'E = ',E
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine rmarin(ij, kl)
C This is the initialization routine for the random number generator RANMAR()
C NOTE: The seed variables can have values between:    0 <= IJ <= 31328
C                                                      0 <= KL <= 30081
C The random number sequences created by these two seeds are of sufficient
C length to complete an entire calculation with. For example, if sveral
C different groups are working on different parts of the same calculation,
C each group could be assigned its own IJ seed. This would leave each group
C with 30000 choices for the second seed. That is to say, this random
C number generator can create 900 million different subsequences -- with
C each subsequence having a length of approximately 10^30.
C
C Use IJ = 1802 & KL = 9373 to test the random number generator. The
C subroutine RANMAR should be used to generate 20000 random numbers.
C Then display the next six random numbers generated multiplied by 4096*4096
C If the random number generator is working properly, the random numbers
C should be:
C          6533892.0  14220222.0  7275067.0
C          6172232.0  8354498.0   10633180.0
C
      implicit real*8 (a-h, o-z)
      real*8 u(97), c, cd, cm
      integer i97, j97
      logical test
      common /raset1/ u, c, cd, cm, i97, j97, test
      test = .false.
C
      if( IJ .lt. 0  .or.  IJ .gt. 31328  .or.
     1    KL .lt. 0  .or.  KL .gt. 30081 ) then
         write (*, *) ' The first random number seed must have a'
         write (*, *) ' value between 0 and 31328.'
         write (*, *)
         write (*, *) ' The second seed must have a value between 0'
         write (*, *) ' and 30081.'
         stop
      endif
      i = mod(IJ/177, 177) + 2
      j = mod(IJ    , 177) + 2
      k = mod(KL/169, 178) + 1
      l = mod(kl,     169)
      do 2 ii = 1, 97
         s = 0.0
         t = 0.5
         do 3 jj = 1, 24
            m = mod(mod(i*j, 179)*k, 179)
            i = j
            j = k
            k = m
```

```
          l = mod(53*l+1, 169)
          if (mod(l*m, 64) .ge. 32) then
            s = s + t
          endif
          t = 0.5 * t
3       continue
        u(ii) = s
2     continue
      c = 362436.0 / 16777216.0
      cd = 7654321.0 / 16777216.0
      cm = 16777213.0 /16777216.0
      i97 = 97
      j97 = 33
      test = .true.
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      real*8 function urand()
C This is the random number generator proposed by George Marsaglia in
C Florida State University Report: FSU-SCRI-87-50
C
      implicit real*8 (a-h, o-z)
      real*8 u(97), c, cd, cm
      integer i97, j97
      logical test
      common /raset1/ u, c, cd, cm, i97, j97, test
C
      if(.not.test) then
        write (*, *) 'urand error #1: must call the initialization
     +               routine rmarin before calling urand.'
        stop
      endif
      uni = u(i97) - u(j97)
      if( uni .lt. 0.0 ) uni = uni + 1.0
      u(i97) = uni
      i97 = i97 - 1
      if(i97 .eq. 0) i97 = 97
      j97 = j97 - 1
      if(j97 .eq. 0) j97 = 97
      c = c - cd
      if( c .lt. 0.0 ) c = c + cm
      uni = uni - c
      if( uni .lt. 0.0 ) uni = uni + 1.0
      urand = uni
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Appendix D

The Fortran Program TQ.FOR

This program, TQ.FOR, computes the unknown temperatures and radiative net heat fluxes of the surfaces defined for the HTS thermal bridge housing area designed by Kasey M. Lee. The distribution factors were computed using the program MC.FOR (Appendix C). This program was developed by Sandrine Garcia, 1994.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PROGRAM TQMC
C
      integer nrings,nelemt,i
      double precision T(69),q(69),Qtot,Qbot,Qtopbot,Q54,area(69),
     +                 A(2,2),C(2),delta(69,69),Dij(69,69),
     +                 emiss(69),sigma,H,R,l,e,pi,alpha
C
      COMMON/PROP/sigma,emiss
      COMMON/ELEMT/nelemt
      COMMON/SUBSTRATE1/nrings,H
      COMMON/SUBSTRATE2/R,alpha,pi,l,e
      COMMON/TEMPERAT/T
      COMMON/FLUX/q
      COMMON/Q/Qbot,Qtopbot,Qtot,Q54
      COMMON/AREA/area
      COMMON/FACTOR/Dij
      COMMON/KRONEKER/delta
      COMMON/MATA/A
      COMMON/MATC/C
C
      open(unit=10,file='Q16ngre.out')
      write(10,*)'output : Q16ngre.out'
```

```
      write(10,*)'nrings=16'
      write(10,*)'nominal estimated values used for the radiative
     +properties'
      write(10,*)'temperature profile used for the substrate :'
cc    write(10,*)'     thermal bridge BSCCO/FSILICA'
cc    write(10,*)'     thermal bridge BSCCO/YSZ'
      write(10,*)'     thermal bridge YBCO/GREEN'
C
C     set constants
      call const
C
C     definition of the matrices Dij,delta,A,C
      call MDij
      call Mdelta
      call MatrixA
      call MatrixC
C
C     Computation of T1 and T2
C     solve A*T12=C by the Gaussian elimination method
      call solve
C
C     Computation of the Fluxes
      call compQ
C
      write(10,100)
      do i=1,nelemt
        write(10,110)i,T(i),Q(i),area(i)
      enddo
      write(10,120)
      write(10,121)
      write(10,122)Qbot
      write(10,130)
      write(10,131)
      write(10,132)Qtopbot
      write(10,140)
      write(10,141)
      write(10,142)Qtot
      write(10,150)
      write(10,151)
      write(10,152)Q54
100   format(/,1x,'surface',4x,'T(K)',10x,'Q(W) ',7x,'area(m2)',/)
110   format(2x,I3,5x,f7.3,4x,E11.5,4x,E11.5)
120   format(//,'1. Radiative heat load on the bottom of the
     +substrate from the entire enclosure :')
121   format(2x,'Qrad-bot (W) = ')
122   format(1x,E14.4)
130   format(/,'2. Radiative heat load on the bottom of the
     +substrate from the top of the substrate :')
131   format(2x,'Qrad-topbot (W) = ')
132   format(1x,E14.4)
140   format(/,'3. Radiative heat load on the entire substrate
     +from the entire enclosure :')
141   format(2x,'Qrad-tot (W) = ')
142   format(1x,E14.4)
150   format(/,'4. Radiative heat load from surface 5
```

```
     +to surface 4 :')
151  format(2x,'Qrad-54 (W) = ')
152  format(1x,E14.4)
      close(10)
C
      stop
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      double precision function Tfsi(x)
C     This function gives the temperature distribution as a function of x
C     in the thermal bridge BSCCO/FSILICA.
C     Tfsi was determined using a polynomial fitting of degree 6.
C
      double precision x,C1,C2,C3,C4,C5,C6,C7
C
C     set the constants
      C1=4.8490772d0
      C2=1725.5456543d0
      C3=-39113.32421d0
      C4=651411.5d0
      C5=-6100919.5d0
      C6=29254182.d0
      C7=-55807916.d0
      Tfsi=C1+C2*x+C3*x**2+C4*x**3+C5*x**4+C6*x**5+C7*x**6
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      double precision function Tysz(x)
C     This function gives the temperature distribution as a function of x
C     in the thermal bridge BSCCO/YSZ.
C     Tysz was determined using a polynomial fitting of degree 6.
C
      double precision x,C1,C2,C3,C4,C5,C6,C7
C
C     set the constants
      C1=3.9761219d0
      C2=887.8359985d0
      C3=-8196.110351d0
      C4=95402.742187d0
      C5=-729424.d0
      C6=3161748.25d0
      C7=-5818412.5d0
      Tysz=C1+C2*x+C3*x**2+C4*x**3+C5*x**4+C6*x**5+C7*x**6
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      double precision function Tgreen(x)
C     This function gives the temperature distribution as a function of x
C     in the thermal bridge YBCO/GREEN.
C     Tgreen was determined using a polynomial fitting of degree 6.
C
      double precision x,C1,C2,C3,C4,C5,C6,C7
```

```
C
C     set the constants
      C1=5.3292093d0
      C2=1454.3270264d0
      C3=-37262.3789d0
      C4=681398.3125d0
      C5=-6675988.5d0
      C6=33007696.d0
      C7=-64334344.d0
      Tgreen=C1+C2*x+C3*x**2+C4*x**3+C5*x**4+C6*x**5+C7*x**6
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine const
C     This subroutine set the constants of the problem
C
      integer nrings,nelemt,i,j
      double precision emiss1,emiss2,emiss(69),sigma,T(69),q(69),
     +        area(69),H,R,l,e,pi,alpha,Tfsi,Tysz,Tgreen
C
      COMMON/PROP/sigma,emiss
      COMMON/ELEMT/nelemt
      COMMON/SUBSTRATE1/nrings,H
      COMMON/SUBSTRATE2/R,alpha,pi,l,e
      COMMON/TEMPERAT/T
      COMMON/FLUX/q
      COMMON/AREA/area
C
C     geometric constants :
      pi=dacos(-1.d0)
      alpha=2.d0*pi/3.d0
      H=15.24d-2
      R=5.d-2
      l=9.144d-3
      e=0.1524d-3
      nrings=16
      nelemt=5+4*nrings
C
C     radiative properties :
      sigma=5.6696d-8
      emiss1=.020d0
      emiss2=.80d0
      do i=1,nelemt
       if (i.le.5) then
         emiss(i)=emiss1
       else
         emiss(i)=emiss2
       endif
      enddo
C
C     define the known Temperatures and Fluxes
      q(1)=0.d0
      q(2)=0.d0
      T(3)=30.d0
```

```
      T(4)=4.d0
      T(5)=80.d0
      do i=0,nrings-1
      j=i+6
cc    T(j)=Tfsi((i+.5d0)*H/nrings)
cc    T(j)=Tysz((i+.5d0)*H/nrings)
      T(j)=Tgreen((i+.5d0)*H/nrings)
      T(j+nrings)=T(j)
      T(j+2*nrings)=T(j)
      T(j+3*nrings)=T(j)
      enddo
C
C     define the area of each surface
      area(1)=H*R
      area(2)=H*R
      area(3)=alpha*R*H
      area(4)=pi*R**2/3.d0
      area(5)=pi*R**2/3.d0
      do i=6,(6+4*nrings-1)
        if (6.le.i.and.i.le.(6+2*nrings-1)) area(i)=l*H/nrings
        if ((6+2*nrings).le.i.and.i.le.(6+4*nrings-1))
     +      area(i)=e*H/nrings
      enddo
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       subroutine MDij
C      This subroutine reads the distribution factors in the enclosure analyzed
C      in the program MCproject.for.
C
      integer nelemt,i,j
      double precision Dij(69,69)
C
      COMMON/FACTOR/Dij
      COMMON/ELEMT/nelemt
C
      open(1,file='100m16n.dat')
      do i=1,nelemt
       do j=1,nelemt
         read(1,*)Dij(i,j)
       enddo
      enddo
      close(1)
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       subroutine Mdelta
C      This subroutine defines the kronecker delta function delta(i,j)
C
      integer nelemt,i,j
      double precision delta(69,69)
C
      COMMON/ELEMT/nelemt
```

```
      COMMON/KRONEKER/delta
C
      do i=1,nelemt
        do j=1,nelemt
          if (i.eq.j) THEN
            delta(i,j)=1.d0
          else
            delta(i,j)=0.d0
          endif
        enddo
      enddo
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       subroutine MatrixA
C      This subroutine defines the matrix A(2,2) such that A*T12=C
C
      double precision Dij(69,69),A(2,2)
C
      COMMON/FACTOR/Dij
      COMMON/MATA/A
C
      A(1,1)=1.d0-Dij(1,1)
      A(1,2)=-Dij(1,2)
      A(2,1)=-Dij(2,1)
      A(2,2)=1.d0-Dij(2,2)
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       subroutine MatrixC
C      This subroutine defines the vector C(2) such that A*T12=C
C
      integer nelemt,j
      double precision C(2),T12(2),T(69),Dij(69,69)
C
      COMMON/FACTOR/Dij
      COMMON/ELEMT/nelemt
      COMMON/TEMPERAT/T
      COMMON/MATC/C
C
      C(1)=0.d0
      C(2)=0.d0
      do j=3,nelemt
        C(1)=C(1) + (T(j))**4*Dij(1,j)
        C(2)=C(2) + (T(j))**4*Dij(2,j)
      enddo
C
      return
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       subroutine solve
C      this subroutine solve A*T12=C for T1 and T2 using the gaussian
C      elimination method
```

```
C
      integer n,i,j,k
      double precision A(2,2),C(2),T12(2),T(69),
     +                 AUG(2,3),PIVOT,TEMP,MULT
C
      COMMON/TEMPERAT/T
      COMMON/MATA/A
      COMMON/MATC/C
C
      n=2
C     *form the n*(n+1) augmented matrix AUG by adjoining C to A
      DO i=1,n
       DO j=1,n
         AUG(i,j)=A(i,j)
       ENDDO
      ENDDO
      DO i=1,n
       AUG(i,n+1)=C(i)
      ENDDO
      DO 70 i=1,n
C     *     locate nonzero diagonal entry
       IF (AUG(i,i).eq.0.D0) THEN
          PIVOT=0.D0
          j=i+1
30        IF ((PIVOT.eq.0.D0).AND.(j.le.n)) THEN
            IF (AUG(j,i).ne.0.D0) PIVOT=j
            GO TO 30
          ENDIF
          IF (PIVOT.eq.0.D0) THEN
           STOP 'MATRIX IS SINGULAR'
          ELSE
C     *               interchange rows i and PIVOT
           DO 40 j=1,n+1
             TEMP=AUG(i,j)
             AUG(i,j)=AUG(PIVOT,j)
             AUG(PIVOT,j)=TEMP
40         CONTINUE
          ENDIF
       ENDIF
C     *     eliminate ith unknown from equations i+1,...,n
        DO 60 j=i+1,n
          MULT=-AUG(j,i)/AUG(i,i)
          DO 50 k=i,n+1
           AUG(j,k)=AUG(j,k)+MULT*AUG(i,k)
50        CONTINUE
60      CONTINUE
70     CONTINUE
C     *find the solutions
      T12(n)=AUG(n,n+1)/AUG(n,n)
      DO 90 j=n-1,1,-1
        T12(j)=AUG(j,n+1)
        DO 80 k=j+1,n
          T12(j)=T12(j)-AUG(j,k)*T12(k)
80      CONTINUE
        T12(j)=T12(j)/AUG(j,j)
```

```
90   CONTINUE
C
     T(1)=(T12(1))**0.25
     T(2)=(T12(2))**0.25
C
     RETURN
     END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      subroutine compQ
C     This subroutine computes the unknown Fluxes in the
C     enclosure analyzed in the program MCproject.for
C
     integer nelemt,nrings,i,j
     double precision emiss(69),delta(69,69),Dij(69,69),q(69),
    +                T(69),area(69),H,sigma,K,
    +                Qbot,Qtopbot,Qtot,Q54
C
     COMMON/PROP/sigma,emiss
     COMMON/ELEMT/nelemt
     COMMON/SUBSTRATE1/nrings,H
     COMMON/TEMPERAT/T
     COMMON/FLUX/q
     COMMON/Q/Qbot,Qtopbot,Qtot,Q54
     COMMON/AREA/area
     COMMON/FACTOR/Dij
     COMMON/KRONEKER/delta
C
     do i=3,nelemt
      K=0.d0
      do j=1,nelemt
         K=K+emiss(i)*sigma*(T(j))**4*(delta(i,j)-Dij(i,j))
      enddo
      q(i)=K
       Q(i)=q(i)*area(i)
     enddo
C
C     1. Compute the radiative heat load on the bottom of the substrate
C     from the entire enclosure = Qbot(W)
     Qbot=Q(6)+Q(22)+Q(38)+Q(54)
C
C     2. Compute the radiative heat load contribution from the top (T=80K)
C     of the substrate to the bottom (T=4K) = Qtopbot(W)
     Qtopbot=0.d0
     do i=6,(nelemt-nrings+1),nrings
       Qtopbot=Qtopbot +
    +         area(i)*emiss(i)*sigma
    +         * ( (T(i))**4 - (T(6+nrings-1))**4 *
    +                  ( Dij(i,6+nrings-1)
    +                   + Dij(i,6+2*nrings-1)
    +                   + Dij(i,6+3*nrings-1)
    +                   + Dij(i,6+4*nrings-1) ) )
      enddo
C
C     3. Compute the radiative heat load on the entire substrate
C     from the entire enclosure = Qtot(W)
```

```
      Qtot=0.d0
      do i=6,nelemt
        Qtot=Qtot+Q(i)
      enddo
C
C     4. Compute the radiative heat load on surface 4 (T=4K)
C     from surface 5 (T=80K) = Q54(W)
        Q54=area(4)*emiss(4)*sigma*(T(4)**4-T(5)**4*Dij(4,5))
C
      return
      end
```

Appendix E

Thermal Conductivity Models

This appendix provides the models of the HTS thermal bridge material thermal conductivities and the HTS thermal bridge effective thermal conductivities.

E.1 Material Thermal Conductivity Model

The general thermal conductivity equation is (Lee, 1994):

$$k(T) = a+bT+cT^2+dT^3+eT^4+fT^5+gT^6 \ . \tag{E.1}$$

The constants a, b, c, d, e, f, and g are provided in Table E.1 for the materials studied in this project. Figure E.1 shows the plots of the material thermal conductivities.

E.2 HTS Thermal Bridge Effcetive Thermal Conductivity Model

The general effective thermal conductivity equation is:

$$k_{eff}(T) = a+bT+cT^2+dT^3 \ . \tag{E.2}$$

The constants a, b, c, and d are provided in Table E.2 for the HTS thermal bridges studied in this project. Figure E.2 shows the plots of the effective thermal conductivities.

Table E.1. Coefficients of the Material Thermal Conductivity Model.

Material	BSCCO	YBCO	FSI	YSZ	GREEN	Zirconia
a	1.430E-1	1.567E-1	1.565E-2	4.464E-1	3.558E-1	-2.045E-1
b	5.445E-2	1.403E-2	2.761E-3	-2.426E-3	7.173E-2	1.159E-1
c	-3.517E-3	7.463E-3	1.561E-4	9.229E-4	1.066E-2	-1.041E-3
d	1.243E-4	-2.510E-4	-3.076E-6	-2.793E-5	-3.706E-4	-2.761E-5
e	-2.100E-6	3.437E-6	3.403E-8	3.772E-7	4.814E-6	6.671E-7
f	1.665E-8	-2.201E-8	-2.009E-10	-2.395E-9	-2.839E-8	-5.127E-9
g	-5.035E-11	5.450E-11	4.826E-13	5.839E-12	6.370E-11	1.367E-11

Table E.2. Coefficients of the HTS Thermal Bridge Effective Thermal Conductivity Model.

Thermal Bridge	BSCCO/ FSI	YBCO/ FSI	BSCCO/ YSZ	YBCO/ YSZ	YBCO/ GREEN
a	4.095E-2	-4.749E-2	3.678E-1	2.793E-1	-7.615E-1
b	4.611E-3	2.008E-2	1.274E-2	2.822E-2	2.916E-1
c	6.988E-5	-1.375E-4	6.400E-5	-2.715E-4	-3.604E-3
d	-5.676E-7	2.505E-7	3.723E-9	8.222E-7	1.083E-5

*Note: the values for the coefficients a, b, c and d, of the effective thermal conductivities are valid over the temperature range [4-80 K].

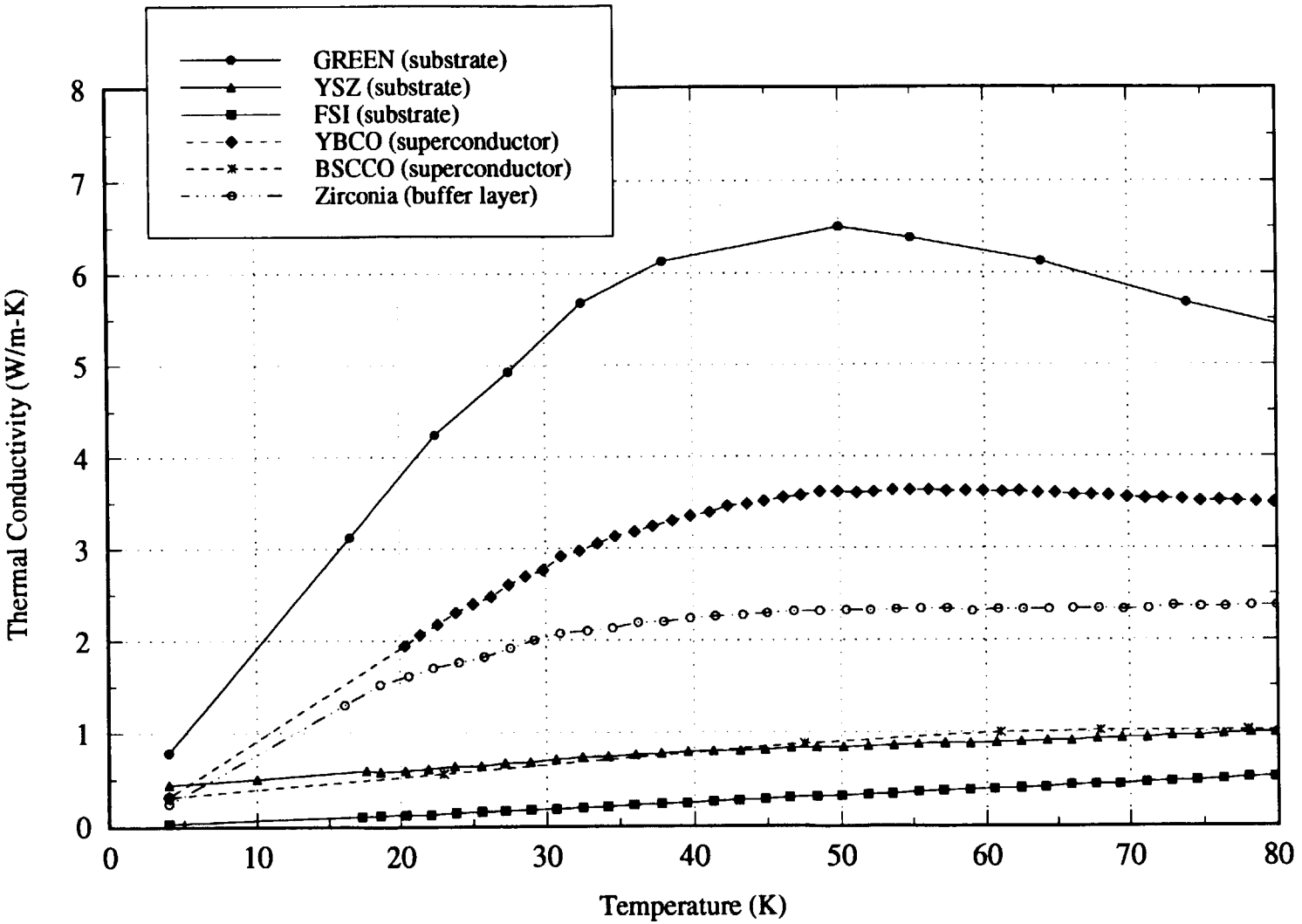


Figure E.1. Thermal Conductivities of the HTS Thermal Bridge Materials.

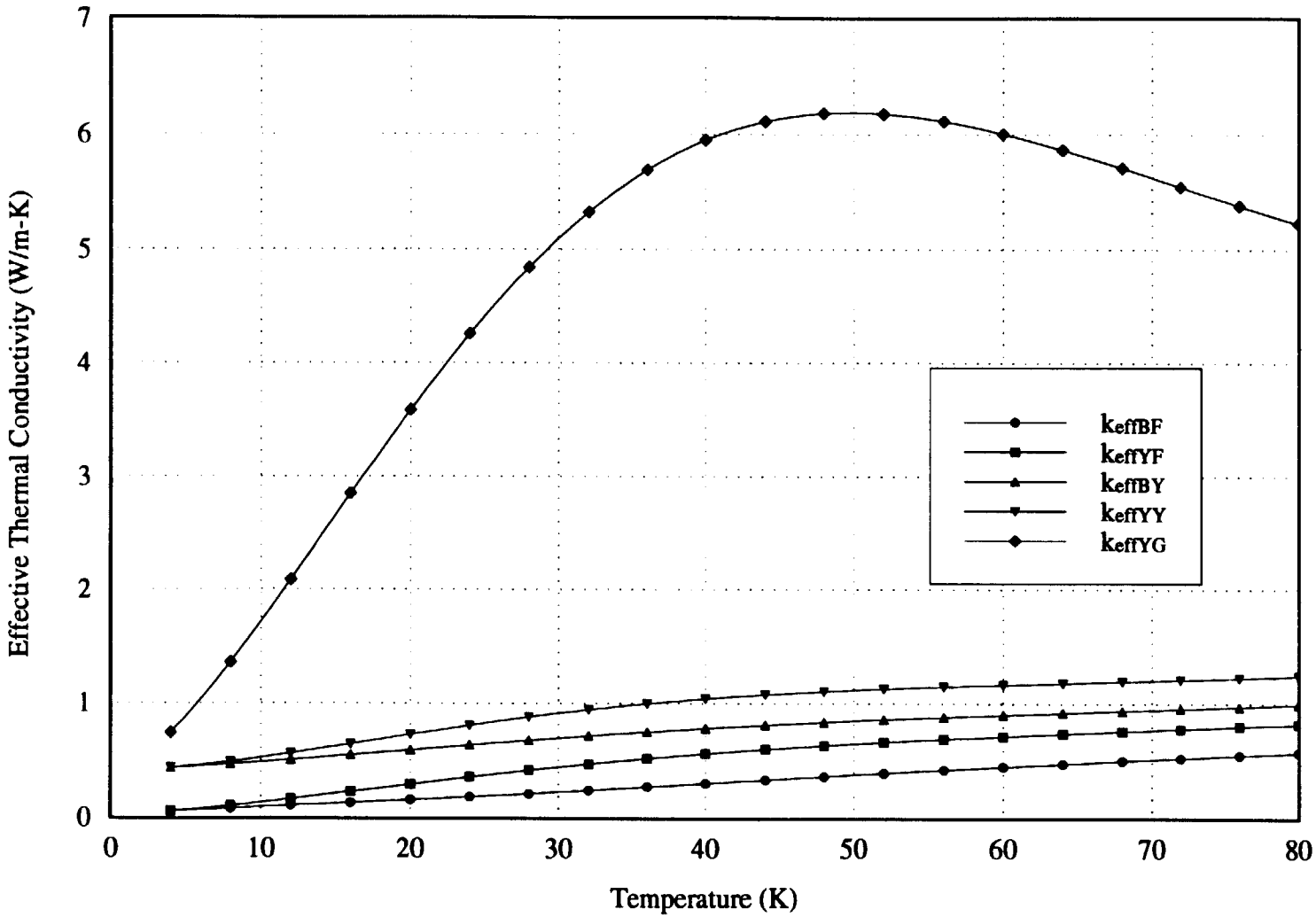


Figure E.2. Effective Thermal Conductivities of the HTS Thermal Bridges.

Appendix F

The Fortran Subroutine KBOX3D.FOR

This subroutine, KBOX3D.FOR, was written as the *adapt* part of the program ORTHO3D (provided at the end of the subroutine). KBOX3D.FOR uses the modified Box-Kanemasu method to estimate the thermal conductivity parameters of the material thermal conductivities in a HTS thermal bridge.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE ADAPT
c
c  HP f77 version,  3-D dp Iso version - DJN
c
c$noextensions
C-----------------------------------------------------------------------
C----- STEADY CONDUCTION IN A 3D HALF HTS THERMAL BRIDGE -----
C                  with isotropic gamma
C
C----- Material Thermal Conductivity Estimation
C      Subroutine KBOX3D.FOR, written by sandrine Garcia, 1994.
C-----------------------------------------------------------------------
      INCLUDE 'common3d.f'
C***********************************************************************
      DIMENSION T1(NI),T2(NI),YI(NI),XI(NI,2),XT(2,NI),
     +          B(2),b1(2),b2(2),
     +          RES(NI),XTX(2,2),XTY(2),dgb(2),P(2,2),RI(2,2),
     +          AUG(2,3),RR(2,2)
C
      DIMENSION              T(NI,NJ,NK)
      EQUIVALENCE (F(1,1,1,1),T(1,1,1))
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

```
      ENTRY GRID
C
      HEADER='HALF BSCCO/YSZ - L=152.4mm'
      PRINTF='K'
C
c  set geometric dimensions
      rL  = 0.1524d0
c  if there is a buffer layer (the substrate is FSI)
ccc       thk2=0.0000003d0
c  if there is no buffer layer
      thk2 = 0.0d0
      rlsuperc = 0.003048d0
      e1 = 0.0001524d0
      e2 = 0.0000508d0
      a  = 0.003048d0
C
c  zoned grid method
c  set x, y and z zones
      NZX = 1
      XZONE(1) = rL
      NCVX(1) = 100
C
      NZY = 2
      YZONE(1) = a
      NCVY(1) = 4
      YZONE(2) = rlsuperc/2.d0
      NCVY(2) = 3
C
      NZZ = 2
c  if the substrate is FSI, add 1 CV for the buffer layer
ccc       NZZ = 3
      ZZONE(1) = e1
      NCVZ(1) = 3
c  if there is no buffer layer
      ZZONE(2) = e2
      NCVZ(2) = 2
c  if there is a buffer layer
ccc       ZZONE(2) = thk2
ccc       NCVZ(2) = 1
ccc       ZZONE(3) = e2
ccc       NCVZ(3) = 2
C
      CALL ZGRID
C
      open(unit=99,file='1yby.d')
ccc       open(unit=99,file='by.100')
      do i=1,102
c  102 is the number of data point measurements (L1)
c  YI(i) is the simulated measured temperature value
        read(99,*)YI(i)
      enddo
      close(99)
      open(unit=98,file='1yby.pol')
C
c  Np is the number of parameters
```

```
      Np=2
c  dB is the parameter variation used in the determination of the
c  sensitivity coefficients
      dB=0.01d0
c  sigma is the standard deviation of YI(j)
      sigma=0.5d0
      sigma2=sigma**2
c  set convergence parameter
      epsi = 1.d-5
c  criter is the convergence parameter for the Box-Kanemasu method
      criter=1.d-3
c  delta1 is a constant used in the convergence criterion
      delta1=1.d-30
c  lastk is the maximum number of iterations in the Box-Kanemasu method
      lastk=15
c  last is the maximum number of iterations in the steady state conduction
c  problem
      last=50
c  itrmin is the minimum number of iterations in the steady state conduction
c  problem
      itrmin=8
C
      Kcount=1
      iterk=1
C
c  set thermal conductivity coefficients
      Bg1 = 0.3558d0
      Bg2 = 0.07173d0
      Bg3 = 0.01066d0
      Bg4 = -3.7060d-4
      Bg5 = 4.814d-6
      Bg6 = -2.839d-8
      Bg7 = 6.37d-11
C
      By1 = 0.4464d0
      By2 = -0.002426d0
      By3 = 9.229d-4
      By4 = -2.793d-5
      By5 = 3.772d-7
      By6 = -2.395d-9
      By7 = 5.839d-12
C
      Bf1 = 0.01565d0
      Bf2 = 0.002761d0
      Bf3 = 1.561d-4
      Bf4 = -3.076d-6
      Bf5 = 3.403d-8
      Bf6 = -2.009d-10
      Bf7 = 4.826d-13
C
      Bz1 = -0.2045d0
      Bz2 = 0.1159d0
      Bz3 = -0.001041d0
      Bz4 = -2.761d-5
      Bz5 = 6.671d-7
```

```
      Bz6 = -5.127d-9
      Bz7 = 1.367d-11
C
      Bybco1 = 0.1567d0
      Bybco2 = 0.01403d0
      Bybco3 = 0.007463d0
      Bybco4 = -2.51d-4
      Bybco5 = 3.437d-6
      Bybco6 = -2.201d-8
      Bybco7 = 5.45d-11
C
      Bbscco1 = 0.143d0
      Bbscco2 = 0.05445d0
      Bbscco3 = -0.003517d0
      Bbscco4 = 1.243d-4
      Bbscco5 = -2.1d-6
      Bbscco6 = 1.665d-8
      Bbscco7 = -5.035d-11
C
c  set input heat flux (W/m*K)
c   - if the thermal bridge is YBCO/GREEN
      Qyg = 2415.429d0
c   - if the thermal bridge is YBCO/YSZ
      Qyy = 483.237d0
c   - if the thermal bridge is BSCCO/YSZ
      Qby = 380.095d0
c   - if the thermal bridge is YBCO/FSI
      Qyf = 260.383d0
c   - if the thermal bridge is BSCCO/FSI
      Qbf = 157.302d0
C
c  Give first estimates for the parameters to be estimated
      b1(1)=Bbscco4
      b1(2)=Bbscco5
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY BEGIN
C
      TITLE(1) ='   TEMPERATURE '
      KSOLVE(1)=1
      KPRINT(1)=0
      KPLOT(1) =0
      ITER=0
      KSTOP=0
C
c  set initial temperatures
      do K=1,N1
        do J=1,M1
          open(unit=14,file='bf.100')
          do I=1,L1
            read(14,*)T(I,J,K)
          enddo
          close(14)
        enddo
      enddo
```

```
C
      if (Kcount.eq.1) then
c  set parameter values to b1(j)
        do j=1,Np
          B(j)=b1(j)
        enddo
      endif
C
      if (Kcount.eq.2) B(Kj)=B(Kj)*(1.d0+dB)
C
      if (Kcount.eq.3) then
        alpha=alpha/2.d0
        do j=1,Np
          B(j)=b1(j)+alpha*dgb(j)
        enddo
      endif
C
      QOUT0 = 0.D0
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY OUTPUT
C
      QIN  = 0.D0
      QOUT = 0.D0
      DO 203 J=2,M2
      DO 203 K=2,N2
        QIN  = QIN  + YCV(J)*ZCV(K)*FLUXL1(J,K,1)
        QOUT = QOUT + YCV(J)*ZCV(K)*FLUXI1(J,K,1)
203   CONTINUE
C
      DO 200 IUNIT=IU1,IU2
        IF(ITER.EQ.0) WRITE(IUNIT,210)
210       FORMAT(2X,'ITER',3X,'T(L1,M2,11)',6X,'T(L1,M2,12)',12X,
     +          'QIN',14X,'QOUT',10X,'NTC(1)')
        WRITE(IUNIT,220)ITER,T(L1,M2,11),T(L1,M2,12),QIN,QOUT,NTC(1)
220       FORMAT(2X,I4,3X,1PE11.3,5X,1PE11.3,7X,1PE12.3,7X,1PE12.3,
     +          7X,I2)
200   CONTINUE
C
c  create a convergence criterion
      IF (ITER.LT.ITRMIN) RETURN
      DIFF = ABS((QOUT-QOUT0)/(QOUT+SMALL))
      QOUT0=QOUT
      IF (DIFF.LE.epsi.OR.ITER.EQ.LAST) THEN
C
c  calculate overall energy balance
      HTBAL = QIN + QOUT
      EBAL  = ABS(HTBAL/QIN)
      DO 201 IUNIT=IU1,IU2
        WRITE(IUNIT,202)EBAL
202       FORMAT(/,2X,'EBAL',1PE11.3,/)
201   CONTINUE
C
      If (Kcount.eq.1) then
c  First part in the Box-Kanemasu Method
```

```
c  get numerical temperature solution for B(j)=b1(j)
c  So is the sum of squares for the B(j)=b1(j) parameter values
        So=0.d0
        do i=1,L1
          T1(i)=T(i,M2,11)
          RES(i)=YI(i)-T1(i)
          So=So+(RES(i))**2/sigma2
        enddo
        Kcount=Kcount+1
        Kj=1
        go to 13
      endif
C
      If (Kcount.eq.2) then
c  Second part in the Box-Kanemasu Method
c  get numerical temperature solutions for B(Kj)=B(Kj)*(1.d0+dB),Kj fixed
c  determine sensitivity coefficients
c  reinitialyze B(Kj)
        B(Kj)=b1(Kj)
        do i=1,L1
          T2(i)=T(i,M2,11)
          XI(i,Kj)=(T2(i)-T1(i))/(B(Kj)*dB)
          XT(Kj,i)=XI(i,Kj)
        enddo
C
        if (Kj.lt.2) then
          Kj=Kj+1
          go to 13
        else
c  determine matrix XTX(Np,Np)
          do j=1,Np
            do k=1,Np
              XTX(j,k)=0.d0
              do i=1,L1
                XTX(j,k)=XTX(j,k)+XT(j,i)*XI(i,k)/sigma2
              enddo
            enddo
          enddo
C
c  determine vector XTY(Np)
          do j=1,Np
            XTY(j)=0.d0
            do i=1,L1
              XTY(j)=XTY(j)+XT(j,i)*RES(i)/sigma2
            enddo
          enddo
C
c  Solve XTX*P=RI for P using the gaussian elimination method
c  first define RI(Np,Np), the matrix identity
          do j=1,Np
            do k=1,Np
              if (k.eq.j) then
                RI(j,k)=1.d0
              else
                RI(j,k)=0.d0
```

```
            endif
          enddo
        enddo
C
c  solve successively each column of P
        do l=1,Np
C    *form the Np*(Np+1) augmented matrix AUG by adjoining RI to XTX
          DO i=1,Np
           DO j=1,Np
             AUG(i,j)=XTX(i,j)
           ENDDO
          ENDDO
          DO i=1,Np
           AUG(i,Np+1)=RI(i,l)
          ENDDO
          DO 1070 i=1,Np
C    *locate nonzero diagonal entry
           IF (AUG(i,i).eq.0.D0) THEN
             PIVOT=0.D0
             j=i+1
1030              IF ((PIVOT.eq.0.D0).AND.(j.le.Np)) THEN
             IF (AUG(j,i).ne.0.D0) PIVOT=j
             GO TO 1030
             ENDIF
             IF (PIVOT.eq.0.D0) THEN
             STOP 'MATRIX IS SINGULAR'
             ELSE
C    *interchange rows i and PIVOT
              DO 1040 j=1,Np+1
                TEMP=AUG(i,j)
                AUG(i,j)=AUG(PIVOT,j)
                AUG(PIVOT,j)=TEMP
1040                  CONTINUE
             ENDIF
           ENDIF
C    *eliminate ith unknown from equations i+1,...,Np
           DO 1060 j=i+1,Np
             RMULT=-AUG(j,i)/AUG(i,i)
             DO 1050 k=i,Np+1
              AUG(j,k)=AUG(j,k)+RMULT*AUG(i,k)
1050             CONTINUE
1060           CONTINUE
1070         CONTINUE
C    *find the solutions
          P(Np,l)=AUG(Np,Np+1)/AUG(Np,Np)
          DO 1090 j=Np-1,1,-1
           P(j,l)=AUG(j,Np+1)
           DO 1080 k=j+1,Np
             P(j,l)=P(j,l)-AUG(j,k)*P(k,l)
1080           CONTINUE
           P(j,l)=P(j,l)/AUG(j,j)
1090         CONTINUE
        enddo
C
c  check the correlation matrix before getting to the parameter estimation
```

```
c  The diagonal terms of the correlation matrix are all unity and the off
c  -diagonal terms must be in the interval [-1,1]. Whenever all the off
c  -diagonal terms exceed 0.9 in magnitude, the estimates are highly
c  correlated and tend to be inaccurate
      write(98,'(/,"The correlation matrix is")')
      do j=1,Np
        do k=1,j
          ar=P(j,j)*P(k,k)
          RR(j,k)=P(j,k)/sqrt(ar)
        enddo
      enddo
      do j=1,Np
        write(98,'(3E15.7)') (RR(j,k),k=1,j)
      enddo
C
c  determine vector dgb(Np)
          do j=1,Np
            dgb(j)=0.d0
            do k=1,Np
              dgb(j)=dgb(j)+P(j,k)*XTY(k)
            enddo
          enddo
C
c  G is a measure of the slope; it should approach zero at convergence
          G=0.d0
          do j=1,Np
            sum=0.d0
            do k=1,Np
              sum=sum+XTX(j,k)*dgb(k)
            enddo
            G=G+dgb(j)*sum
          enddo
C
c  Third part in the Box-Kanemasu Method
c  By the definition of G, it should always be positive
          if (G.lt.0.d0) then
            write(98,*)'G is negative ! Terminate calculations'
            go to 12
          endif
C
c  Fourth part in the Box-Kanemasu Method
          alpha=2.000d0
          AA=1.1d0
          Kcount=Kcount+1
          go to 13
        endif
      endif
C
      If (Kcount.eq.3) then
c  Salpha is the sum of squares for the Bcoef(j)=b1(j)+alpha*dgb(j) parameter
c  values. Salpha decreases towards a positive constant and should be less
c  than So
        Salpha=0.d0
        do i=1,L1
          T1(i)=T(i,M2,11)
```

```
          RES(i)=YI(i)-T1(i)
          Salpha=Salpha+(RES(i))**2/sigma2
        enddo
C
        if (Salpha.gt.So) then
          if (alpha.le.0.01d0) then
            write(98,*)'alpha is too small !'
            write(98,'("alpha = ",F12.6,2x,"Salpha = ",E15.6,
     +              2x,"So = ",E15.6)')alpha,Salpha,So
            go to 12
          else
            go to 13
          endif
        endif
C
        sumch=So-alpha*G*(2.d0-1.d0/AA)
c  h is a scalar interpolation factor; its a fraction of the Gauss step
c  given by the Box-Kanemasu method
        if (Salpha.gt.sumch) then
          h=alpha**2*G/(Salpha-So+2.d0*alpha*G)
        else
          h=alpha*AA
        endif
C
c  Calculate the final parameter estimates using h
c  Also calculate ratio; if it is less than criter, then the change in
c  the estimated parameters is insignifiant and the iterative process is
c  terminated. change is used to determine when all parameters stop varying
        change=0
        do j=1,Np
          b2(j)=b1(j)+h*dgb(j)
          ratio=abs(b2(j)-b1(j))/(abs(b1(j))+delta1)
          if (ratio.le.criter) change=change+1
        enddo
      endif
C
c  Print out the calculate values for h, G, So and Salpha
       write(98,1300)
1300   format(5x,'iter',10x,'h',13x,'G',12x,'So',11x,'Salpha')
       write(98,1301)iterk,h,G,So,Salpha
1301   format(I8,4E14.6,/)
C
c  Print out the final parameter estimates
      write(98,*)'The final parameter estimate for this iteration is'
      write(98,1310) (b2(j),j=1,Np)
 1310   format(3E16.6)
C
c  Print out the P matrix
      write(98,'(/,"The P matrix is")')
       do j=1,Np
         write(98,1320) (P(j,k),k=1,Np)
           enddo
1320   format(3D15.7)
C
      if (Np.gt.change.and.iterk.le.lastk) then
```

```
      do j=1,Np
        b1(j)=b2(j)
      enddo
C
      Kcount=1
      iterk=iterk+1
      go to 13
     endif
C
12   close(iu2)
     close(98)
     STOP 'K ESTIMATION DONE'
C
13   KSTOP=1
     ENDIF
     RETURN
C*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*
     ENTRY PHI
C
c  set conductivities
     DO 300 K=2,N2
     DO 300 J=2,M2
     DO 300 I=2,L2
C
       IF (Z(K).LE.e1) THEN
c  if the substrate is the GREEN PHASE :
ccc         GAM(I,J,K) = Bg1 + Bg2*T(I,J,K)
ccc    +                + Bg3*T(I,J,K)**2
ccc    +                + Bg4*T(I,J,K)**3
ccc    +                + Bg5*T(I,J,K)**4
ccc    +                + Bg6*T(I,J,K)**5
ccc    +                + Bg7*T(I,J,K)**6
c  if the substrate is YSZ :
        GAM(I,J,K) = By1 + By2*T(I,J,K)
    +                + By3*T(I,J,K)**2
    +                + By4*T(I,J,K)**3
    +                + By5*T(I,J,K)**4
    +                + By6*T(I,J,K)**5
    +                + By7*T(I,J,K)**6
c  if the substrate is FSI :
ccc         GAM(I,J,K) = Bf1 + Bf2*T(I,J,K)
ccc    +                + Bf3*T(I,J,K)**2
ccc    +                + Bf4*T(I,J,K)**3
ccc    +                + Bf5*T(I,J,K)**4
ccc    +                + Bf6*T(I,J,K)**5
ccc    +                + Bf7*T(I,J,K)**6
       ELSE
C
        IF (Z(K).GT.e1.AND.Y(J).LT.a) GAM(I,J,K) = 0.d0
C
c  if there is a buffer layer (the substrate is FSI)
ccc        IF (Z(K).GT.e1.AND.Z(K).LE.(e1+thk2).AND.
ccc    +       Y(J).GE.a) THEN
ccc        GAM(I,J,K) = Bz1 + Bz2*T(I,J,K)
ccc    +                + Bz3*T(I,J,K)**2
```

```
ccc     +                + Bz4*T(I,J,K)**3
ccc     +                + Bz5*T(I,J,K)**4
ccc     +                + Bz6*T(I,J,K)**5
ccc     +                + Bz7*T(I,J,K)**6
ccc         ENDIF
C
          IF (Z(K).GT.(e1+thk2).AND.Z(K).LE.(e1+thk2+e2).AND.
     +        Y(J).GE.a) THEN
c  if the superconductor is YBCO :
ccc           GAM(I,J,K) = Bybco1 + Bybco2*T(I,J,K)
ccc     +                + Bybco3*T(I,J,K)**2
ccc     +                + Bybco4*T(I,J,K)**3
ccc     +                + Bybco5*T(I,J,K)**4
ccc     +                + Bybco6*T(I,J,K)**5
ccc     +                + Bybco7*T(I,J,K)**6
c  if the superconductor is BSCCO :
           GAM(I,J,K) = Bbscco1 + Bbscco2*T(I,J,K)
     +               + Bbscco3*T(I,J,K)**2
     +               + B(1)*T(I,J,K)**3
     +               + B(2)*T(I,J,K)**4
     +               + Bbscco6*T(I,J,K)**5
     +               + Bbscco7*T(I,J,K)**6
          ENDIF
         ENDIF
300   CONTINUE
C
c  set boundary conditions
      DO 310 K=2,N2
      DO 310 J=2,M2
        KBCL1(J,K) = 2
c     - for the substrate
         IF (Z(K).LE.e1) THEN
          FLXCL1(J,K) = Qyf
         ELSE
C
c     - if there is a buffer layer
ccc           IF (Y(J).GE.a.AND.Z(K).GT.e1.AND.Z(K).LE.
ccc     +         (e1+thk2)) THEN
ccc            FLXCL1(J,K) = Qbf
ccc           ENDIF
C
c     - for the superconductor
          IF (Y(J).GE.a.AND.Z(K).GT.(e1+thk2).AND.Z(K).LE.
     +         (e1+thk2+e2)) THEN
           FLXCL1(J,K) = Qyf
          ENDIF
         ENDIF
310   CONTINUE
C
      DO 320 K=2,N2
      DO 320 I=2,L2
        KBCJ1(I,K)=2
        KBCM1(I,K)=2
320   CONTINUE
C
```

```
      DO 330 J=2,M2
      DO 330 I=2,L2
        KBCK1(I,J)=2
        KBCN1(I,J)=2
330   CONTINUE
C
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
************************************************************************
       PROGRAM ORTHO3D
*
* Control Volume Method solution of three-dimensional, isotropic
* heat conduction
*
* User portion of code contained in layer*.f
* layer3d.f - standard user subroutine
*
* RS/6000 version - double precision
*
* Program structure similar to that detailed in
* "Computation of Conduction and Duct Flow Heat Transfer",
*  S.V. Patankar, Maple Grove, MN:  Innovative Research, Inc.
*
* Modified by Sandrine Garcia to implement the Box-Kanemasu estimation procedure
* (see subroutine KBOX3D.FOR) and to compute the dimensionless sensitivity
* coefficients. (see subroutine XI3D.FOR).
*
************************************************************************
      INCLUDE 'common3d.f'
C
      CALL DEFLT
      CALL GRID
      CALL READY
C
93    CALL BEGIN
C
10    CONTINUE
c start iteration of outer loop
     CALL OUTPUT
c check to see if convergence has occured
      IF (KSTOP.NE.0) go to 93
      CALL HEART
      GO TO 10
C
      STOP
      END
************************************************************************
c include other subroutines in invariant part
      INCLUDE 'defrd3d.f'
      INCLUDE 'heart3d.f'
      INCLUDE 'solve3d.f'
      INCLUDE 'tools3d.f'
```

Appendix G

The Fortran Program KBOXEFF.FOR

This program, KBOXEFF.FOR, uses the modified Box-Kanemasu method to estimate the effective thermal conductivities of the HTS thermal bridges. KBOXEFF.FOR has the ability to estimate these thermal properties either as functions of temperature or as constants. This program was written by Sandrine Garcia, 1994.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Program Kboxeff
C
      integer Np,max,iterk,lastk,set
      double precision T,z,sigma,criter,
     +                 So,Salpha,G,delta1
      double precision B(1),b1(1),b2(1),Texact(102),sol(10),
     +                 X(102),T1(102),T2(102),YY(1020),YI(102),
     +                 XI(102,1),XT(1,102),XTX(1,1),XTY(1),RES(102),
     +                 dgb(1),P(1,1),RI(1,1),AUG(1,2),RR(1,1)
ccc   +                  X(102),T1(500),T2(500),YY(5000),YI(500),
ccc   +                  XI(500,1),XT(1,500),XTX(1,1),XTY(1),RES(500),
ccc   +                  dgb(1),P(1,1),RI(1,1),AUG(1,2),RR(1,1)
C
      COMMON/BCOEF/B
      COMMON/b1COEF/b1
      COMMON/LENGTH/X
C
      open(unit=98,file='o1ybf.eff')
c  SET THE CONSTANTS
c  Np is the number of parameters to study
      Np=1
c  max is the number of points
```

```
c   if T(x) is used to estimate the parameters
      max=102
c   if TL is used to estimate the parameters
ccc       max=500
      maxL=102
c  sigma is the standard deviation of YI(j)
      sigma=0.1d0
      sigma2=sigma**2
c  criter is the convergence parameter for the Box-Kanemasu method
      criter=1.d-4
c  delta1 is a constant used in the convergence criterion
      delta1=1.d-30
c  lastk is the maximum number of iterations in the Box-Kanemasu method
      lastk=10
C
c  if the exact temperatures from ORTHO3D are used as measured data
ccc       open(unit=99,file='bf.100')
ccc       do i=1,maxL
ccc          read(99,*)Texact(i)
c   if T(x) is used to estimate the parameter
ccc          YI(i)=Texact(i)
ccc       enddo
c   if TL is used to estimate the parameter
ccc       do i=1,max
ccc          YI(i)=Texact(maxL)
ccc       enddo
C
c  if simulated temperature are used as measured data
c   read YYI(i) which contains 10 simulated data sets
c   if T(x) is used to estimate the parameter
      open(unit=99,file='o1ybf.d')
      do i=1,1020
        read(99,*)YY(i)
      enddo
c   if TL is used to estimate the parameter
ccc       open(unit=99,file='1yLbf.d')
ccc       do i=1,5000
ccc          read(99,*)YY(i)
ccc       enddo
      close(99)
C
c  LOOP ON THE NUMBER OF DATA SETS
      DO 999 SET=1,10
      write(98,*)
      write(98,*)'*****************************************************'
     write(98,*)'SET NUMBER: ',SET
C
      do i=1,max
c   if T(x) is used to estimate the parameter
        YI(i)=YY(i+102*(SET-1))
c   if TL is used to estimate the parameter
ccc          YI(i)=YY(i+500*(SET-1))
      enddo
C
c  define X(i),the position vector and initialize the b1 vector
```

```
      call init
C
c  start Box-Kanemasu method
      iterk = 0
1     iterk = iterk+1
C
c  set parameters values to b1(j)
      B(1) = b1(1)
C
c  First part in the Box-Kanemasu Method
c  solve for T(x) using B(j)=b1(j);
c   if keff is a polynomial use the bisection method
c    if T(x) is used to estimate the parameter
ccc       T1(1) = 4.d0
ccc       do i=2,max
ccc          z = X(i)
ccc          call dichot(z,T)
ccc          T1(i) = T
ccc       enddo
c    if TL is used to estimate the parameter
ccc          z = X(maxL)
ccc          call dichot(z,T)
ccc       do i=1,max
ccc          T1(i) = T
ccc       enddo
c   if keff is a constant
      do i=1,max
c     if T(x) is used to estimate the parameter
        T1(i)=157.302d0/B(1)*X(i)+4.d0
c     if TL is used to estimate the parameter
ccc          T1(i)=157.302d0/B(1)*X(maxL)+4.d0
      enddo
C
c  So is the sum of squares for the Bcoef(j)=b1(j) parameter values
      So=0.d0
      do i=1,max
        RES(i)=YI(i)-T1(i)
        So=So+(RES(i))**2/sigma2
      enddo
C
c  start sensitivity study
c  iterate on Np, the number of parameters to study
      do 20 Kj=1,Np
c
c   if keff is a polynomial, determine XI numerically
c    dB is the parameter variation used in the determination of the
c    sensitivity coefficients
ccc       dB = 0.01d0
c    modify B(Kj)
ccc          B(Kj) = B(Kj)*(1.d0+dB)
C
c    solve for T2 (including the influence of dB) using the bisection method
c     if T(x) is used to estimate the parameter
ccc          T2(1) = 4.d0
ccc          do i=2,max
```

```
ccc            z = X(i)
ccc            call dichot(z,T)
ccc            T2(i) = T
ccc          enddo
c     if TL is used to estimate the parameter
ccc          z = X(maxL)
ccc          call dichot(z,T)
ccc          do i=1,max
ccc            T2(i) = T
ccc          enddo
C
c    reinitialize B(Kj)
ccc          B(Kj) = b1(Kj)
C
c    study XB(Kj)
ccc          do i=1,max
ccc            XI(i,Kj) = (T2(i)-T1(i))/(B(Kj)*dB)
ccc            XT(Kj,i)=XI(i,Kj)
ccc          enddo
C
c   if keff is a constant, determine XI analytically
         do i=1,max
c    if T(x) is used to estimate the parameter
           XI(i,Kj)=-157.302d0*X(i)/B(Kj)**2
c    if TL is used to estimate the parameter
ccc            XI(i,Kj)=-157.302d0*X(maxL)/B(Kj)**2
           XT(Kj,i)=XI(i,Kj)
         enddo
20     continue
C
c  determine matrix XTX(Np,Np)
      do j=1,Np
        do k=1,Np
          XTX(j,k)=0.d0
          do i=1,max
            XTX(j,k)=XTX(j,k)+XT(j,i)*XI(i,k)/sigma2
          enddo
        enddo
      enddo
C
c  determine vector XTY(Np)
      do j=1,Np
        XTY(j)=0.d0
        do i=1,max
          XTY(j)=XTY(j)+XT(j,i)*RES(i)/sigma2
        enddo
      enddo
C
c  Solve XTX*P=RI for P using the gaussian elimination method
c  first define RI(Np,Np), the matrix identity
      do j=1,Np
        do k=1,Np
          if (k.eq.j) then
            RI(j,k)=1.d0
          else
```

```
          RI(j,k)=0.d0
        endif
      enddo
    enddo
C
c  solve successively each column of P
    do l=1,Np
C   *form the Np*(Np+1) augmented matrix AUG by adjoining RI to XTX
      DO i=1,Np
        DO j=1,Np
          AUG(i,j)=XTX(i,j)
        ENDDO
      ENDDO
      DO i=1,Np
        AUG(i,Np+1)=RI(i,l)
      ENDDO
      DO 1070 i=1,Np
C   *locate nonzero diagonal entry
        IF (AUG(i,i).eq.0.D0) THEN
          PIVOT=0.D0
          j=i+1
1030        IF ((PIVOT.eq.0.D0).AND.(j.le.Np)) THEN
            IF (AUG(j,i).ne.0.D0) PIVOT=j
            GO TO 1030
          ENDIF
          IF (PIVOT.eq.0.D0) THEN
            STOP 'MATRIX IS SINGULAR'
          ELSE
C   *interchange rows i and PIVOT
            DO 1040 j=1,Np+1
              TEMP=AUG(i,j)
              AUG(i,j)=AUG(PIVOT,j)
              AUG(PIVOT,j)=TEMP
1040            CONTINUE
          ENDIF
        ENDIF
C   *eliminate ith unknown from equations i+1,...,Np
        DO 1060 j=i+1,Np
          RMULT=-AUG(j,i)/AUG(i,i)
          DO 1050 k=i,Np+1
            AUG(j,k)=AUG(j,k)+RMULT*AUG(i,k)
1050        CONTINUE
1060      CONTINUE
1070    CONTINUE
C   *find the solutions
      P(Np,l)=AUG(Np,Np+1)/AUG(Np,Np)
      DO 1090 j=Np-1,1,-1
       P(j,l)=AUG(j,Np+1)
       DO 1080 k=j+1,Np
         P(j,l)=P(j,l)-AUG(j,k)*P(k,l)
1080     CONTINUE
        P(j,l)=P(j,l)/AUG(j,j)
1090    CONTINUE
    enddo
C
```

```
c  determine vector dgb(Np)
      do j=1,Np
        dgb(j)=0.d0
        do k=1,Np
          dgb(j)=dgb(j)+P(j,k)*XTY(k)
        enddo
      enddo
C
c  G is a measure of the slope; it should approach zero at convergence
      G=0.d0
      do j=1,Np
        sum=0.d0
        do k=1,Np
          sum=sum+XTX(j,k)*dgb(k)
        enddo
        G=G+dgb(j)*sum
      enddo
C
c  Third part in the Box-Kanemasu Method
c  By the definition of G, it should always be positive
      if (G.lt.0.d0) then
        write(98,*)'G is negative ! Terminate calculations'
        go to 5
      endif
C
c  Fourth part in the Box-Kanemasu Method
      alpha=2.000d0
      AA=1.1d0
2     alpha=alpha/2.d0
      do j=1,Np
        B(j)=b1(j)+alpha*dgb(j)
      enddo
C
c  solve for T(x) using B(j)=b1(j)+alpha*dgb(j);
c   if keff is a polynomial use the bisection method
c    if T(x) is used to estimate the parameter
ccc      T1(1) = 4.d0
ccc      do i=2,max
ccc        z = X(i)
ccc        call dichot(z,T)
ccc        T1(i) = T
ccc      enddo
c    if TL is used to estimate the parameter
ccc        z = X(maxL)
ccc        call dichot(z,T)
ccc      do i=1,max
ccc        T1(i) = T
ccc      enddo
c    if keff is a constant
      do i=1,max
c    if T(x) is used to estimate the parameter
        T1(i)=157.302d0/B(1)*X(i)+4.d0
c    if TL is used to estimate the parameter
ccc         T1(i)=157.302d0/B(1)*X(maxL)+4.d0
      enddo
```

```
C
c  Salpha is the sum of squares for the Bcoef(j)=b1(j)+alpha*dgb(j) parameter
c  values. Salpha decreases towards a positive constant and should be less
c  than So
      Salpha=0.d0
      do i=1,max
        RES(i)=YI(i)-T1(i)
        Salpha=Salpha+(RES(i))**2/sigma2
      enddo
C
      if (Salpha.gt.So) then
        if (alpha.le.0.01d0) then
          write(98,*)'alpha is too small !'
          write(98,'("alpha = ",F12.6,2x,"Salpha = ",E15.6,
     +              2x,"So = ",E15.6)')alpha,Salpha,So
          go to 5
        else
          go to 2
        endif
      endif
C
      sumch=So-alpha*G*(2.d0-1.d0/AA)
c  h is a scalar interpolation factor; its a fraction of the Gauss step
c  given by the Box-Kanemasu method
      if (Salpha.gt.sumch) then
        h=alpha**2*G/(Salpha-So+2.d0*alpha*G)
      else
        h=alpha*AA
      endif
C
c  Calculate the final parameter estimates using h
c  Also calculate ratio; if it is less than criter, then the change in
c  the estimated parameters is insignifiant and the iterative process is
c  terminated. change is used to determine when all parameters stop varying
      change=0
      do j=1,Np
        b2(j)=b1(j)+h*dgb(j)
        ratio=abs(b2(j)-b1(j))/(abs(b1(j))+delta1)
        if (ratio.le.criter) change=change+1
      enddo
C
c  Print out the calculate values for h, G, So and Salpha
      write(*,1300)
      write(98,1300)
1300  format(5x,'iterk',10x,'h',13x,'G',12x,'So',11x,'Salpha')
      write(*,1301)iterk,h,G,So,Salpha
      write(98,1301)iterk,h,G,So,Salpha
1301  format(I8,4E14.6,/)
C
c  Print out the final parameter estimates
      write(98,*)'The final parameter estimates for this iteration are'
      write(*,*)'The final parameter estimates for this iteration are'
      write(98,1310) (b2(j),j=1,Np)
      write(*,1310) (b2(j),j=1,Np)
1310  format(E16.6)
```

```
C
c  Print out the P matrix
      write(98,'(/,"The P matrix is")')
      do j=1,Np
        write(98,1320) (P(j,k),k=1,Np)
      enddo
1320  format(D15.7)
C
c  Determine and print out the correlation matrix
c  The diagonal terms of the correlation matrix are all unity and the off
c  -diagonal terms must be in the interval [-1,1]. Whenever all the off
c  -diagonal terms exceed 0.9 in magnitude, the estimates are highly
c  correlated and tend to be inaccurate
      write(98,'(/,"The correlation matrix is")')
      do j=1,Np
        do k=1,j
          ar=P(j,j)*P(k,k)
          RR(j,k)=P(j,k)/sqrt(ar)
        enddo
      enddo
      do j=1,Np
        write(98,'(E15.7)') (RR(j,k),k=1,j)
      enddo
C
      if (Np.gt.change.and.iterk.le.lastk) then
        do j=1,Np
          b1(j)=b2(j)
        enddo
        go to 1
      endif
C
c  for the estimation of a constant effective thermal conductivity
c  store final estimate
      sol(SET)=b2(1)
999   CONTINUE
C
c  come here to perform statistic calculations for the 10 data sets
c   compute the mean value of the estimates
      mean=0.d0
      do i=1,10
        mean=mean+sol(i)
      enddo
      mean=mean/10.d0
c   compute the standard deviation
      dev=0.d0
      do i=1,10
        dev=dev+(sol(i)-mean)**2
      enddo
      dev=sqrt(dev/9.d0)
c   compute the 95% confidence interval for the mean value
      confint=2.262d0*dev/sqrt(10.d0)
      write(98,*)
      write(98,1310)'mean= ',mean
      write(*,1310)'mean= ',mean
      write(98,1310)'95% confidence interval= ',confint
```

```
      write(*,1310)'95% confidence interval= ',confint
      write(98,1310)'standard deviation of the sample= ',dev
      write(*,1310)'standard deviation of the sample= ',dev
1310  format(E16.6)
C
5     close(98)
      STOP 'K ESTIMATION DONE'
      END
*************************************************************************
      subroutine init
C
      integer maxL,i
      double precision L
      double precision b1(1),X(102)
C
      COMMON/b1COEF/b1
      COMMON/LENGTH/X
C
c  define position vector
c  (so that it is equivalent to 100 CV in the x-direction in ORTHO3D)
      L = 0.1524d0
      maxL=102
      X(1)=0.d0
      X(2)=L/(100.d0*2.d0)
      do i=3,(maxL-1)
        X(i) = X(i-1)+L/100.d0
      enddo
      X(maxL) = L
c  define initial estimate for the parameter studied
      b1(1) = 3.0d-1
C
      return
      end
*************************************************************************
      subroutine dichot(x,T)
C
      integer Kcount
      double precision Byy2,Byy3,Byy4,
     +                 x,T,a1,a2,c,qin,K,Qa1,Qa2,Qc,epsi,Poly
      double precision B(1)
C
      COMMON/BCOEF/B
      COMMON/Byy/Byy2,Byy3,Byy4
C
c  define thermal conductivity coefficients not studied as parameters
      Byy2 = 2.78287d-2
      Byy3 = -3.02917d-4
      Byy4 = 1.26054d-6
C
c  define input heat flux
      qin = 483.237d0
C
c  define limits of the interval studied [a1,a2] for the T range
      a1 = 3.d0
      a2 = 100.d0
```

```
c  define constant K in the T solution
      K = 4.d0*B(1)+8.d0*Byy2+4.d0**3/3.d0*Byy3+64.d0*Byy4
C
c  solve for T(x) using the bisection method
      Kcount = 0
10    Kcount = Kcount+1
      Qa1 = Poly(a1)-(qin*x+K)
      Qa2 = Poly(a2)-(qin*x+K)
      c = (a1+a2)/2.d0
      Qc = Poly(c)-(qin*x+K)
C
      if ((Qa1*Qc).le.0.d0) then
        a2 = c
      else
        a1 = c
      endif
C
      epsi = (abs(Qa1)+abs(Qa2))/2.d0
      if (Kcount.gt.1000) STOP 'Kcount greater than 1000'
      if (epsi.gt.1d-4) then
        go to 10
      else
        T = c
      endif
C
      return
      end
*******************************************************************************
      double precision function Poly(T)
C
      double precision T,Byy2,Byy3,Byy4
      double precision B(1)
C
      COMMON/BCOEF/B
      COMMON/Byy/Byy2,Byy3,Byy4
C
C       Poly = B(1)*T+Byy2/2.d0*T**2+Byy3/3.d0*T**3+Byy4/4.d0*T**4
      return
      end
*******************************************************************************
```

Appendix H

The Fortran Program YI.FOR

This program, YI.FOR, reads a file of numerical temperatures obtained using ORTHO3D for a specific HTS thermal bridge and adds random errors to simulate measured temperatures. Sets of simulated temperatures are obtained both along the thermal bridge and at the warm end. The simulated measured temperatures are then used for the estimation of the thermal conductivity parameters. This program was written by Sandrine Garcia, 1994.

```
      PROGRAM YI
C
      COMMON/RAND/L1,STDDV
      COMMON NDAT
      DIMENSION DATA(20000)
      DIMENSION T(102)
C
      L1=102
      STDDV=1.0d0
C
c  read the numerical temperatures obtained using ORTHO3D
      open(unit=30,file='bf.100')
      do i=1,L1
        read(30,*)T(i)
      enddo
      close(30)
C
      open(unit=2,file='1yLbf.d')
```

```
ccc     open(unit=2,file='o1ybf.d')
      open(unit=50,file='Nseed')
C
ccc     do j=1,10
      do j=1,50
        call random(data)
c  Addition of random errors to calculated numerical Temperatures T(x)
ccc       do i=1,L1
ccc         Y=T(i) + data(i)
c  Addition of random errors to calculated numerical Temperatures T(L1)
        do i=1,100
          Y=T(L1) + data(i)
          write(2,*)Y
        enddo
      enddo
      close(50)
      close(2)
C
      stop
      end
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Subroutine random(data)
c  See Numerical Recipes by Press, Flannery, Teukolsky and Vetterling,
c  Cambridge Press, 1986 about page 192
c  Modified by J.V. Beck, Michigan State University
C
      COMMON/RAND/L1,STDDV
      COMMON NDAT
      parameter(PI=3.14159265,NBIN=1000)
      Dimension data(20000)
C
      read(50,*)idum
      NDAT=L1+NBIN
      rhon=0.0
      rhod=0.0
      do 500 idumi=1,1
        data(1)=gasdev(idum)*STDDV
        do 11 i=2,L1
          data(i)=gasdev(idum)*STDDV
          rhon=rhon+data(i-1)*data(i)
          rhod=rhod+data(i)*data(i)
11      continue
12      continue
        rho=rhon/rhod
        call moment(data,i-1,ave,adev,sdev,var,rho)
500   continue
      write(*,*)'                    Values of quantities'
      write(*,'(1x,T29,A,T42,A/)')' Sample  ',' Expected'
      write(*,'(1x,A,T25,2F12.4)')'Mean :',ave,0.0
      write(*,'(1x,A,T25,2F12.4)')'Average Deviation :',adev,STDDV
      write(*,'(1x,A,T25,2F12.4)')'Standard Deviation :',sdev,STDDV
      varth=stddv*stddv
      write(*,'(1x,A,T25,2F12.4)')'Variance :',var,varth
      write(*,'(1x,A,T25,F12.4)')'Est. Correlation Coeff.',rho
      write(*,*)'Average deviation comes from use of absolute values'
```

```
      return
      end
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      Subroutine moment(data,n,ave,adev,sdev,var,rho)
C
      Dimension data(20000)
C
      If (n.le.1) pause 'n must be at least 2'
      s=0.
      sd=0.
      sn=0.
      do 11 j=1,n
        s=s+data(j)
        if (j.eq.1) goto 11
        sn=sn+data(j)*data(j-1)
        sd=sd+data(j)+data(j)
11     continue
      ave=s/n
      adev=0.
      var=0.
      do 12 j=1,n
        s=data(j)-ave
        adev=adev+abs(s)
        p=s*s
        var=var+p
12     continue
      adev=adev/n
      var=var/(n-1)
      sdev=sqrt(var)
      rho=sn/sd
C
      return
      end
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      Function ran1(idum)
c  Returns uniformly distributed numbers between 0 and 1
C
      Dimension R(97)
      Parameter (M1=259200,IA1=7141,IC1=54773,RM1=3.8580247E-6)
      Parameter (M2=134456,IA2=8121,IC2=28411,RM2=7.4373773E-6)
      Parameter (M3=243000,IA3=4561,IC3=51349)
      Data IFF/0/
C
      if (idum.lt.0.or.IFF.eq.0) then
        IFF=1
        IX1=MOD(IC1-idum,M1)
        IX1=MOD(IA1*IX1+IC1,M1)
        IX2=MOD(IX1,M2)
        IX1=MOD(IA1*IX1+IC1,M1)
        IX3=MOD(IX1,M3)
        do 11 j=1,97
          IX1=MOD(IA1*IX1+IC1,M1)
          IX2=MOD(IA2*IX2+IC2,M2)
          R(j)=(FLOAT(IX1)+FLOAT(IX2)*RM2)*RM1
11        continue
```

```
        idum=1
      endif
      IX1=MOD(IA1*IX1+IC1,M1)
      IX2=MOD(IA2*IX2+IC2,M2)
      IX3=MOD(IA3*IX3+IC3,M3)
      j=1+(97*IX3)/M3
      if (j.gt.97.or.j.lt.1) pause
      ran1=R(j)
      R(j)=(FLOAT(IX1)+FLOAT(IX2)*RM2)*RM1
C
      return
      end
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      Function gasdev(idum)
c  Uses Box-Muller transformation fron uniform distribution to normal
c  distribution with unit standard deviation
C
      DATA ISET/0/
C
      if (ISET.eq.0) then
1       v1=2.*ran1(idum)-1.
        v2=2.*ran1(idum)-1.
        R=v1**2+v2**2
        if (R.ge.1..or.R.eq.0.) goto 1
        fac=sqrt(-2.*LOG(R)/R)
        gset=v1*fac
        gasdev=v2*fac
        ISET=1
      else
        gasdev=gset
        ISET=0
      endif
C
      return
      end
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Appendix I

The Fortran Subroutine XI3D.FOR

This subroutine, YI.FOR, was written as the *adapt* part of the program ORTHO3D (provided at the end of the subroutine KBOXEFF.FOR in Appendix F). XI3D.FOR is used to compute the dimensionless sensitivity coefficients of the thermal conductivity parameters for the HTS thermal bridge materials.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE ADAPT
c
c  HP f77 version,  3-D dp Iso version - DJN
c
c$noextensions
C-----------------------------------------------------------------------
C----- STEADY CONDUCTION IN A 3D HALF HTS THERMAL BRIDGE -----
C                  with isotropic gamma
C
C----- Dimensionless Sensitivity Coefficient Determination for the Material
C      Thermal conductivity Parameters.
C      Subroutine XI3D.FOR, written by Sandrine Garcia, 1994.
C-----------------------------------------------------------------------
      INCLUDE 'common3d.f'
C*********************************************************************
      DIMENSION Xsens(NI)
      DIMENSION T1(NI),T2(NI)
      DIMENSION              T(NI,NJ,NK)
      EQUIVALENCE (F(1,1,1,1),T(1,1,1))
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY GRID
C
      HEADER='HALF BSCCO/FSI - L=152.4mm - XF1 influence'
```

```
      PRINTF='X'
C
c  set geometric dimensions
      rL  = 0.1524d0
c  if there is a buffer layer (the substrate is FSI)
      thk2=0.0000003d0
c  if there is no buffer layer
ccc    thk2 = 0.0d0
      rlsuperc = 0.003048d0
      e1 = 0.0001524d0
      e2 = 0.0000508d0
      a  = 0.003048d0
C
c  zoned grid method
c  set x, y and z zones
      NZX = 1
      XZONE(1) = rL
      NCVX(1) = 100
C
      NZY = 2
      YZONE(1) = a
      NCVY(1) = 4
      YZONE(2) = rlsuperc/2.d0
      NCVY(2) = 3
C
ccc    NZZ = 2
c  if the substrate is FSI, add 1 CV for the buffer layer
      NZZ = 3
      ZZONE(1) = e1
      NCVZ(1) = 3
c  if there is no buffer layer
ccc    ZZONE(2) = e2
ccc    NCVZ(2) = 2
c  if there is a buffer layer
      ZZONE(2) = thk2
      NCVZ(2) = 1
      ZZONE(3) = e2
      NCVZ(3) = 2
C
      CALL ZGRID
C
c  set convergence parameter
      epsi = 1.d-5
c  set maximum number of outer iterations
      LAST  = 50
c  set minimum number of outer iterations
      ITRMIN = 8
C
c  set thermal conductivity coefficients
      Bg1 = 0.3558d0
      Bg2 = 0.07173d0
      Bg3 = 0.01066d0
      Bg4 = -3.706d-4
      Bg5 = 4.814d-6
      Bg6 = -2.839d-8
```

```
      Bg7 = 6.37d-11
C
      By1 = 0.4464d0
      By2 = -0.002426d0
      By3 = 9.229d-4
      By4 = -2.793d-5
      By5 = 3.772d-7
      By6 = -2.395d-9
      By7 = 5.839d-12
C
      Bf1 = 0.01565d0
      Bf2 = 0.002761d0
      Bf3 = 1.561d-4
      Bf4 = -3.076d-6
      Bf5 = 3.403d-8
      Bf6 = -2.009d-10
      Bf7 = 4.826d-13
C
      Bz1 = -0.2045d0
      Bz2 = 0.1159d0
      Bz3 = -0.001041d0
      Bz4 = -2.761d-5
      Bz5 = 6.671d-7
      Bz6 = -5.127d-9
      Bz7 = 1.367d-11
C
      Byb1 = 0.1567d0
      Byb2 = 0.01403d0
      Byb3 = 0.007463d0
      Byb4 = -2.51d-4
      Byb5 = 3.437d-6
      Byb6 = -2.201d-8
      Byb7 = 5.45d-11
C
      Bbs1 = 0.143d0
      Bbs2 = 0.05445d0
      Bbs3 = -0.003517d0
      Bbs4 = 1.243d-4
      Bbs5 = -2.1d-6
      Bbs6 = 1.665d-8
      Bbs7 = -5.035d-11
C
c  set input heat flux (W/m*K)
c   - if the thermal bridge is YBCO/GREEN
      Qyg = 2415.429d0
c   - if the thermal bridge is YBCO/YSZ
      Qyy = 483.237d0
c   - if the thermal bridge is BSCCO/YSZ
      Qby = 380.095d0
c   - if the thermal bridge is YBCO/FSI
      Qyf = 260.383d0
c   - if the thermal bridge is BSCCO/FSI
      Qbf = 157.302d0
C
      dB=0.d0
```

```
      KSTOPPP=0
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY BEGIN
C
      TITLE(1) ='   TEMPERATURE '
      KSOLVE(1)=1
      KPRINT(1)=0
      KPLOT(1) =0
      KSTOP=0
      ITER=0
C
c  modify one thermal conductivity coefficient
      Bf11 = Bf1*(1.d0+dB)
C
c  set initial temperature (K)
      DO K=1,N1
        DO J=1,M1
         open(unit=3,file='bf.100')
          DO I=1,L1
           read(3,*)T(I,J,K)
          ENDDO
          close(3)
        ENDDO
      ENDDO
C
      QOUT0 = 0.D0
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY OUTPUT
C
      QIN  = 0.D0
      QOUT = 0.D0
      DO 20 J=2,M2
      DO 20 K=2,N2
        QIN  = QIN  + YCV(J)*ZCV(K)*FLUXL1(J,K,1)
        QOUT = QOUT + YCV(J)*ZCV(K)*FLUXI1(J,K,1)
20    CONTINUE
C
      DO 200 IUNIT=IU1,IU2
        IF(ITER.EQ.0) WRITE(IUNIT,210)
210     FORMAT(2X,'ITER',3X,'T(L1,M2,11)',5X,'T(L1,M2,12)',9X,'QIN',
     +         13X,'QOUT',9X,'NTC(1)')
        WRITE(IUNIT,220)ITER,T(L1,M2,11),T(L1,M2,12),QIN,QOUT,NTC(1)
220     FORMAT(2X,I3,3X,1PE10.4,5X,1PE10.4,7X,1PE10.4,7X,1PE12.4,
     +         7X,I2)
200   CONTINUE
C
c  create a convergence criterion
      IF (ITER.LT.ITRMIN) RETURN
      DIFF = ABS((QOUT-QOUT0)/(QOUT+SMALL))
      QOUT0=QOUT
      IF (DIFF.LE.epsi.OR.ITER.EQ.LAST) THEN
C
c  calculate overall energy balance
```

```
      HTBAL = QIN + QOUT
      EBAL  = ABS(HTBAL/QIN)
      DO 40 IUNIT=IU1,IU2
        WRITE(IUNIT,50)EBAL
50      FORMAT(/,2X,'EBAL',1PE11.3)
40    CONTINUE
C
      If (Kstoppp.eq.0) then
       do i=1,L1
         T1(i)=T(i,M2,11)
       enddo
       dB=0.01d0
       Kstoppp=1
       go to 13
      endif
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Come here to start dimensionless sensitivity coefficient calculation
      open (unit=1,file='bfXf1.dat')
C
      do i=1,L1
        T2(i) = T(i,M2,11)
c  dimensionless temperatures
        T2(i) = (T2(i)-T1(1))/(T1(L1)-T1(1))
        T1(i) = (T1(i)-T1(1))/(T1(L1)-T1(1))
c  dimensionless coefficients
       Xsens(i) = (T2(i)-T1(i))/dB
        write(1,1000)X(i),Xsens(i)
1000  format(1F12.6,3x,1PE15.6)
      enddo
      close(1)
      STOP 'XI determination done'
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
13    KSTOP=1
      ENDIF
      RETURN
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
      ENTRY PHI
C
c  set conductivities
      DO 300 K=2,N2
      DO 300 J=2,M2
      DO 300 I=2,L2
C
       IF (Z(K).LE.e1) THEN
c  if the substrate is the GREEN PHASE :
ccc          GAM(I,J,K) = Bg1 + Bg2*T(I,J,K)
ccc     +               + Bg3*T(I,J,K)**2
ccc     +               + Bg4*T(I,J,K)**3
ccc     +               + Bg5*T(I,J,K)**4
ccc     +               + Bg6*T(I,J,K)**5
ccc     +               + Bg7*T(I,J,K)**6
c  if the substrate is YSZ :
ccc          GAM(I,J,K) = By1 + By2*T(I,J,K)
ccc     +               + By3*T(I,J,K)**2
ccc     +               + By4*T(I,J,K)**3
```

```
ccc     +                + By5*T(I,J,K)**4
ccc     +                + By6*T(I,J,K)**5
ccc     +                + By7*T(I,J,K)**6
c  if the substrate is FSI :
         GAM(I,J,K) = Bf11 + Bf2*T(I,J,K)
     +                     + Bf3*T(I,J,K)**2
     +                     + Bf4*T(I,J,K)**3
     +                     + Bf5*T(I,J,K)**4
     +                     + Bf6*T(I,J,K)**5
     +                     + Bf7*T(I,J,K)**6
       ELSE
C
       IF (Z(K).GT.e1.AND.Y(J).LT.a) GAM(I,J,K) = 0.d0
C
c  if there is a buffer layer (the substrate is FSI)
       IF (Z(K).GT.e1.AND.Z(K).LE.(e1+thk2).AND.
     +          Y(J).GE.a) THEN
         GAM(I,J,K) = Bz1 + Bz2*T(I,J,K)
     +                     + Bz3*T(I,J,K)**2
     +                     + Bz4*T(I,J,K)**3
     +                     + Bz5*T(I,J,K)**4
     +                     + Bz6*T(I,J,K)**5
     +                     + Bz7*T(I,J,K)**6
       ENDIF
C
       IF (Z(K).GT.(e1+thk2).AND.Z(K).LE.(e1+thk2+e2).AND.
     +          Y(J).GE.a) THEN
c  if the superconductor is YBCO :
ccc         GAM(I,J,K) = Byb1 + Byb2*T(I,J,K)
ccc  +                + Byb3*T(I,J,K)**2
ccc  +                + Byb4*T(I,J,K)**3
ccc  +                + Byb5*T(I,J,K)**4
ccc  +                + Byb6*T(I,J,K)**5
ccc  +                + Byb7*T(I,J,K)**6
c  if the superconductor is BSCCO :
         GAM(I,J,K) = Bbs1 + Bbs2*T(I,J,K)
     +                     + Bbs3*T(I,J,K)**2
     +                     + Bbs4*T(I,J,K)**3
     +                     + Bbs5*T(I,J,K)**4
     +                     + Bbs6*T(I,J,K)**5
     +                     + Bbs7*T(I,J,K)**6
       ENDIF
      ENDIF
300   CONTINUE
C
c  set boundary conditions
      DO 310 K=2,N2
      DO 310 J=2,M2
        KBCL1(J,K) = 2
c      - for the substrate
        IF (Z(K).LE.e1) THEN
        FLXCL1(J,K) = Qbf
        ELSE
C
c      - if there is a buffer layer
```

```
      IF (Y(J).GE.a.AND.Z(K).GT.e1.AND.Z(K).LE.
     +          (e1+thk2)) THEN
         FLXCL1(J,K) = Qbf
      ENDIF
C
c     - for the superconductor
       IF (Y(J).GE.a.AND.Z(K).GT.(e1+thk2).AND.Z(K).LE.
     +        (e1+thk2+e2)) THEN
       FLXCL1(J,K) = Qbf
       ENDIF
      ENDIF
310   CONTINUE
C
      DO 320 K=2,N2
      DO 320 I=2,L2
        KBCJ1(I,K)=2
        KBCM1(I,K)=2
320   CONTINUE
C
      DO 330 J=2,M2
      DO 330 I=2,L2
        KBCK1(I,J)=2
        KBCN1(I,J)=2
330   CONTINUE
C
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Appendix J

The Fortran Program XIEFF.FOR

This program, XIEFF.FOR, is used to compute the dimensionless sensitivity coefficients of the effective thermal conductivity parameters for the HTS thermal bridges. This program was written by Sandrine Garcia, 1994.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 Program XIEFF
C
      integer Np,max,i,Kj
      double precision a1,z
      double precision B(4),Bi(4),X(102),T1(102),T2(102),XI(102,4),
     +                 T22(102),T11(102)
C
      COMMON/BCOEF/B
      COMMON/BCOEFI/Bi
      COMMON/LENGTH/X
C
c  Np is the number of parameters to study
      Np = 4
c  max is the number of points
      max = 102
c  define the position and the Bcoef vectors
      call init(max)
C
c  solve for T(x) using the nominal values for the parameters
c  use the bisection method
      T1(1) = 4.d0
      a1 = 4.d0
      do i=2,max
        z = X(i)
        call dichot(z,a1)
```

```
        T1(i) = a1
        write(*,'(" T1: i,T1 ",i3,3x,F12.6)')i,T1(i)
      enddo
C
      open(unit=1,file='ygeff.T')
      do i=1,max
        write(1,*)T1(i)
      enddo
      close(1)
C
c  start sensitivity study
      dB = 0.01d0
      open(unit=2,file='ygXeff.dat')
c  iterate on Np, the number of parameters to study
      do 20 Kj=1,Np
C
c  modify B(Kj)
        B(Kj) = B(Kj)*(1.d0+dB)
        write(2,'("B: ",4E12.6)')B(1),B(2),B(3),B(4)
C
c  solve for T2 (including the influence of dB)
        T2(1) = 4.d0
        a1 = 4.d0
        do i=2,max
          z = X(i)
          call dichot(z,a1)
          T2(i) = a1
        enddo
C
c  reinitialize B(Kj)
          B(Kj) = Bi(Kj)
C
c  study XB(Kj)
        do i=1,max
c  dimensionless T
          T22(i) = (T2(i)-T1(1))/(T1(max)-T1(1))
          T11(i) = (T1(i)-T1(1))/(T1(max)-T1(1))
c  dimensionless sensitivity coefficient
          XI(i,Kj) = (T22(i)-T11(i))/dB
          write(2,1000)X(i),XI(i,Kj)
        enddo
1000  format(1F12.6,3x,1E12.6)
20    continue
      close(2)
      STOP 'Xi determination done'
      END
*************************************************************************
      subroutine init(max)
C
      integer max,i
      double precision L
      double precision B(4),Bi(4),X(102)
C
      COMMON/BCOEF/B
      COMMON/BCOEFI/Bi
```

```
      COMMON/LENGTH/X
C
c  define position vector
c  (so that it is equivalent to 100 CV in the x-direction in Conduct)
      L = 0.1524d0
      X(1)=0.d0
      X(2)=L/(100.d0*2.d0)
      do i=3,(max-1)
        X(i) = X(i-1)+L/100.d0
      enddo
      X(max) = L
C
c  define nominal values for the parameters studied
c    if the thermal bridge is BSCCO/FSI
ccc       Bi(1) = 4.094868d-2
ccc       Bi(2) = 4.611036d-3
ccc       Bi(3) = 6.98767d-5
ccc       Bi(4) = -5.675586d-7
c    if the thermal bridge is YBCO/FSI
ccc       Bi(1) = -4.74875d-2
ccc       Bi(2) = 2.008335d-2
ccc       Bi(3) = -1.37505310d-4
ccc       Bi(4) = 2.504543d-7
c    if the thermal bridge is BSCCO/YSZ
ccc       Bi(1) = 3.677809d-1
ccc       Bi(2) = 1.274091d-2
ccc       Bi(3) = -6.4d-5
ccc       Bi(4) = 3.722766d-9
c    if the thermal bridge is YBCO/YSZ
ccc       Bi(1) = 2.792925d-1
ccc       Bi(2) = 2.82224d-2
ccc       Bi(3) = -2.71507d-4
ccc       Bi(4) = 8.222185d-7
c    if the thermal bridge is YBCO/GREEN
      Bi(1) = -7.614828d-1
      Bi(2) = 2.915835d-1
      Bi(3) = -3.604426d-3
      Bi(4) = 1.083079d-5
C
c  initialize the parameters
      B(1) = Bi(1)
      B(2) = Bi(2)
      B(3) = Bi(3)
      B(4) = Bi(4)
C
      return
      end
*************************************************************************
      subroutine dichot(x,a1)
C
      integer Kcount
      double precision x,a1,a2,c,qin,K,Qa1,Qa2,Qc,epsi,P
      double precision B(4)
C
      COMMON/BCOEF/B
```

```
C
c  define input heat flux
ccc      qin = 157.302d0
ccc      qin = 260.383d0
ccc      qin = 380.095d0
ccc      qin = 483.237d0
      qin = 2415.429d0
C
c  define 2nd limit b of the interval studied [a,b] for the T range
      a2 = 100.d0
c  define constant K in the T solution
      K = 4.d0*B(1)+8.d0*B(2)+4.d0**3/3.d0*B(3)+64.d0*B(4)
C
c  solve for T(x) using the bisection method
      Kcount = 0
10    Kcount = Kcount+1
      Qa1 = P(a1)-(qin*x+K)
      Qa2 = P(a2)-(qin*x+K)
      c = (a1+a2)/2.d0
      Qc = P(c)-(qin*x+K)
      if ((Qa1*Qc).lt.0.d0) then
        a2 = c
      else
        a1 = c
      endif
C
      epsi = (abs(Qa1)+abs(Qa2))/2.d0
      if (Kcount.gt.1000) STOP 'Kcount greater than 1000'
      if (epsi.gt.1d-4) go to 10
C
      return
      end
*************************************************************************
      double precision function P(T)
C
      double precision T
      double precision B(4)
C
      COMMON/BCOEF/B
C
      P = B(1)*T+B(2)/2.d0*T**2+B(3)/3.d0*T**3+B(4)/4.d0*T**4
      return
      end
*************************************************************************
```

Appendix K

The Fortran Program KEFF.FOR

This program, KEFF.FOR, is used to compute the true estimates of the constant effective thermal conductivities of the HTS thermal bridges. This program was written by Sandrine Garcia, 1994.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Program KEFF
C
      integer i
      double precision ratiosup,ratiosub,ratiobl
      double precision B(7),Bsup(7),Bsub(7),Bbl(7),T,keff,
     +                 truekl,truekx,TL,dT,T1(102)
C
c  set the surface ratios
      Asup = 0.1548384d0
      Asub = 1.3935456d0
c  - if there is no buffer layer
ccc       Atot = 1.54838406d0
c  - if there is a buffer layer
      Abl  = 0.0009144d0
      Atot = 1.5492984d0
C
      ratiosup = Asup/Atot
      ratiosub = Asub/Atot
c  if there is a buffer layer
      ratiobl  = Abl/Atot
C
c  set the material thermal conductivity coefficients
c  if the substrate is the GREEN Phase
ccc       Bsub(1) = 0.3558d0
ccc       Bsub(2) = 0.07173d0
```

```
ccc      Bsub(3) = 0.01066d0
ccc      Bsub(4) = -3.706d-4
ccc      Bsub(5) = 4.814d-6
ccc      Bsub(6) = -2.839d-8
ccc      Bsub(7) = 6.37d-11
C
c  if the substrate is YSZ
ccc      Bsub(1) = 0.4464d0
ccc      Bsub(2) = -0.002426d0
ccc      Bsub(3) = 9.229d-4
ccc      Bsub(4) = -2.793d-5
ccc      Bsub(5) = 3.772d-7
ccc      Bsub(6) = -2.395d-9
ccc      Bsub(7) = 5.839d-12
C
c  if the substrate is FSI
      Bsub(1) = 0.01565d0
      Bsub(2) = 0.002761d0
      Bsub(3) = 1.561d-4
      Bsub(4) = -3.076d-6
      Bsub(5) = 3.403d-8
      Bsub(6) = -2.009d-10
      Bsub(7) = 4.826d-13
c  for FSI add the buffer layer
      Bbl(1) = -0.2045d0
      Bbl(2) = 0.1159d0
      Bbl(3) = -0.001041d0
      Bbl(4) = -2.761d-5
      Bbl(5) = 6.671d-7
      Bbl(6) = -5.127d-9
      Bbl(7) = 1.367d-11
C
c  if the superconductor is YBCO
ccc      Bsup(1) = 0.1567d0
ccc      Bsup(2) = 0.01403d0
ccc      Bsup(3) = 0.007463d0
ccc      Bsup(4) = -2.51d-4
ccc      Bsup(5) = 3.437d-6
ccc      Bsup(6) = -2.201d-8
ccc      Bsup(7) = 5.45d-11
C
if the superconductor is BSCCO
      Bsup(1) = 0.143d0
      Bsup(2) = 0.05445d0
      Bsup(3) = -0.003517d0
      Bsup(4) = 1.243d-4
      Bsup(5) = -2.1d-6
      Bsup(6) = 1.665d-8
      Bsup(7) = -5.035d-11
C
c  DETERMINATION OF THE COEFFICIENT OF Keffective
c  if there is no buffer layer
ccc      B(1) = ratiosup*Bsup(1) + ratiosub*Bsub(1)
ccc      B(2) = ratiosup*Bsup(2) + ratiosub*Bsub(2)
ccc      B(3) = ratiosup*Bsup(3) + ratiosub*Bsub(3)
```

```
ccc      B(4) = ratiosup*Bsup(4) + ratiosub*Bsub(4)
ccc      B(5) = ratiosup*Bsup(5) + ratiosub*Bsub(5)
ccc      B(6) = ratiosup*Bsup(6) + ratiosub*Bsub(6)
ccc      B(7) = ratiosup*Bsup(7) + ratiosub*Bsub(7)
C
c  if there is a buffer layer
      B(1) = ratiosup*Bsup(1) + ratiobl*Bbl(1) + ratiosub*Bsub(1)
      B(2) = ratiosup*Bsup(2) + ratiobl*Bbl(2) + ratiosub*Bsub(2)
      B(3) = ratiosup*Bsup(3) + ratiobl*Bbl(3) + ratiosub*Bsub(3)
      B(4) = ratiosup*Bsup(4) + ratiobl*Bbl(4) + ratiosub*Bsub(4)
      B(5) = ratiosup*Bsup(5) + ratiobl*Bbl(5) + ratiosub*Bsub(5)
      B(6) = ratiosup*Bsup(6) + ratiobl*Bbl(6) + ratiosub*Bsub(6)
      B(7) = ratiosup*Bsup(7) + ratiobl*Bbl(7) + ratiosub*Bsub(7)
C
c  record keff data
ccc      open(unit=1,file='kbf.eff')
ccc      T=0.d0
ccc      do i=1,101
ccc         keff = B(1)+B(2)*T+B(3)*T**2+B(4)*T**3
ccc     +          +B(5)*T**4+B(6)*T**5+B(7)*T**6
ccc         write(1,*)T,keff
ccc         T = T+1.d0
ccc      enddo
ccc      close(1)
C
c  reads exact temperatures obtained from ORTHO3D
      open(unit=99,file='bf.100')
      do i=1,102
        read(99,*)T1(i)
      enddo
      close(99)
C
c  determine the area under the keff curve for the range of temperature
c  investigated [4-80K]
c  set T interval to discretize the curve
      dT=0.00001d0
c  set initial T and initialize area to zero
      T=4.d0
      area=0.d0
      sumarea=0.d0
      sumbeta=0.d0
c  determine the true estimate truekl (average value of keff on [To,TL]
      do while (T.le.T1(102))
        i=i+1
        keff = B(1)+B(2)*T+B(3)*T**2+B(4)*T**3
     +          +B(5)*T**4+B(6)*T**5+B(7)*T**6
        area = area+keff*dT
        T = T+dT
      enddo
      truekl = area/(T1(102)-T1(1))
C
c  initialize area, sumbeta and sumarea to zero
      area=0.d0
      sumbeta=0.d0
      sumarea=0.d0
```

```
c  determine the true estimate truekx
      do i=1,101
        keff = B(1)+B(2)*T1(i+1)+B(3)*T1(i+1)**2+B(4)*T1(i+1)**3
     +         +B(5)*T1(i+1)**4+B(6)*T1(i+1)**5+B(7)*T1(i+1)**6
        area = area+keff*(T1(i+1)-T1(i))
        sumarea = sumarea+area
        betax = area/(T1(i+1)-T1(1))
        write(*,*)betax
        sumbeta = sumbeta+area**2/(T1(i+1)-T1(1))
      enddo
      truekx = sumbeta/sumarea
C
      write(*,*)'truekl = ',truekl
      write(*,*)'truekx = ',truekx
      stop
      end
```