EMBRY-RIDDLE
Aeronautical University™
SCHOLARLY COMMONS

Doctoral Dissertations and Master's Theses

Fall 11-10-2014

Inverse Volume-of-Fluid Meshless Method for Efficient Non-Destructive Thermographic Evaluation

Hussein Awad Kurdi Saad
Embry-Riddle Aeronautical University

Follow this and additional works at: https://commons.erau.edu/edt

Part of the Heat Transfer, Combustion Commons

INVERSE VOLUME-OF-FLUID MESHLESS METHOD FOR EFFICIENT NON-DESTRUCTIVE THERMOGRAPHIC EVALUATION

by

Hussein Awad Kurdi Saad

A Thesis Submitted to the College of Engineering Department of Mechanical Engineering in Partial Fulfillment of the Requirements for the Degree of Master of Science in Mechanical Engineering

Embry-Riddle Aeronautical University
Daytona Beach, Florida
November 10, 2014

INVERSE VOLUME-OF-FLUID MESHLESS METHOD FOR EFFICIENT NON-DESTRUCTIVE THERMOGRAPHIC EVALUATION
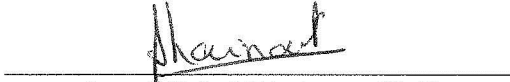
by

Hussein Awad Kurdi Saad

This thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. Eduardo A. Divo, Associate Professor, Daytona Beach Campus, and Thesis Committee Members Dr. Sandra K. Boetcher, Associate Professor, Daytona Beach Campus, and Dr. Jeff Brown, Associate Professor, Daytona Beach Campus, and has been approved by the Thesis Committee. It was submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

Thesis Review Committee:
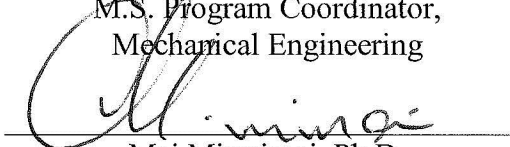
_____
Eduardo Divo, Ph.D.
Committee Chair
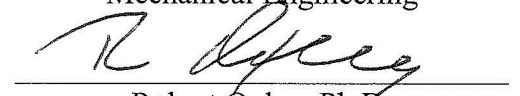
_____
Sandra K. Boetcher, Ph.D.
Committee Member

_____
Jean-Michel Dhainaut, Ph.D.
M.S. Program Coordinator,
Mechanical Engineering

_____
Maj Mirmirani, Ph.D.
Dean, College of Engineering

_____
Jeff Brown, Ph.D.
Committee Member

_____
Charles F. Reinholtz, Ph.D.
Department Chair,
Mechanical Engineering

_____
Robert Oxley, Ph.D.
Associate Vice President of Academics

1-19-14
Date

ACKNOWLEDGEMENTS

ABSTRACT

Researcher: Hussein Awad Kurdi Saad

Title: Inverse Volume-of-Fluid Meshless Method for Efficient Non-Destructive

Thermographic Evaluation

Institution: Embry-Riddle Aeronautical University

Degree: Master of Science in Mechanical Engineering

Year: 2014

A novel computational tool based on the Localized Radial-basis Function (RBF) Collocation (LRC) Meshless method coupled with a Volume-of-Fluid (VoF) scheme capable of accurately and efficiently solving transient multi-dimensional heat conduction problems in composite and heterogeneous media is formulated and implemented. While the LRC Meshless method lends its inherent advantages of spectral convergence and ease of automation, the VoF scheme allows to effectively and efficiently simulate the location, size, and shape of cavities, voids, inclusions, defects, or de-attachments in the conducting media without the need to regenerate point distributions, boundaries, or interpolation matrices. To this end, the Inverse Geometric problem of Cavity Detection can be formulated as an optimization problem that minimizes an objective function that computes the deviation of measured temperatures at accessible locations to those generated by the LRC-VoF Meshless method. The LRC-VoF Meshless algorithms will be driven by an optimization code based on the Genetic Algorithms technique which can efficiently search for the optimal set of design parameters (location, size, shape, etc.) within a predefined design space. Initial guesses to the search algorithm will be provided

by the classical 1D semi-infinite composite analytical solution which can predict the approximate location of the cavity. The LRC-VoF formulation is tested and validated through a series of controlled numerical experiments. This approach will allow solving the onerous computational inverse geometric problem in a very efficient and robust manner while affording its implementation in modest computational platforms, thereby realizing the disruptive potential of the multi-dimensional high-fidelity non-destructive evaluation (NDE) method.

NOMENCLATURE

RBF   Radial Based Function

LRC   Localized radial based function collection

VoF   Volume of Fluid

NDE   Non-destructive function

RC   Reinforced concentrate

FRP   Fiber-reinforced polymer

BEM   Boundary element method

AGP   Anchored grid patterns

FEM   Finite-element methods

FVM   Finite volume methods

DRBEM   Dual reciprocity boundary element method

LCMM   Localized collection Meshless method

NC Set of data centers

NB Points on the boundary

NI Points on the interior

Γ Boundary

Ω Domain

T Temperature [k]

x, y, z Cartesian axis directions

t Time [sec]

k Thermal conductivity [W/m.K]

c Specific heat capacity [J/Kg.K]

ρ Density [kg/m^3]

e Thermal effusivity [J/m^2·K.s$^{1/2}$]

$\hat{\beta}_j$ Boundary condition coefficients

NF Topology of influence points

α_j RBF expansion coefficients

$\chi_j(x)$ Radial-basis functions (RBF)

$r_j(x)$ Euclidean distance from x [m]

x_j Expansion point

d RBF shape parameter

x_c Topology data center

L Linear differential operator

$\{L_c\}$ Derivative expansion vector

$\{L\}$ Derivative interpolation vector

$\{T\}$ Derivative of the temperature field

MLS   Moving Least-Square

s Volume-of-Fluid parameter

T_i LRC-VoF computed temperature

q Heat flux [W/m^2]

$\hat{T}_i$ Temperatures acquired through IR measurements

N_m Finite number of measurement locations

N_r Number of cluster rays

$S(z)$ Objective function

M_i Value of the second derivative of the spline at the node i

i Node

$\Delta\theta_i$ Angular spread of each spline

z Number of geometric parameters

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The objective of this research is to formulate, test, and validate the Inverse VoF Meshless Method for Efficient Non-Destructive Thermographic Evaluation. To this end, a novel computational methodology based on the Localized Radial-basis Function (RBF) Collocation (LRC) Meshless method coupled with a Volume-of-Fluid (VoF) scheme will be implemented to accurately and efficiently solve transient multi-dimensional heat conduction problems in composite and heterogeneous media while offering the advantage of being able to simulate the presence, location, size, and shape of cavities, voids, inclusions, defects, or de-attachments in the conducting media without the need of domain or boundary remodeling, point distributions regeneration, or interpolation matrices recalculation. This highly automated technique can then be seamlessly integrated into an optimization framework formulated to search for such cavities, voids, inclusions, defects, or de-attachments by parameterizing their location, size, and shape through a series of design variables. The solution method will be validated by comparing its predictions to the actual setup of controlled laboratory experiments designed to acquire the surface thermal signatures through an IR camera from different heating conditions over a variety of composite conducting domains with different defect configurations.

The specific aims of the plan are:

i) To formulate and implement a novel computational paradigm based on the Localized Radial-basis Function (RBF) Collocation (LRC) Meshless method

coupled with a Volume-of-Fluid (VoF) scheme capable of accurately and efficiently solving transient multi-dimensional heat conduction problems in composite and heterogeneous media. While the LRC Meshless method lends its inherent advantages of spectral convergence and ease of automation, the VoF scheme allows the effective and efficient simulation of location, size, and shape of cavities, voids, inclusions, defects, or de-attachments in the conducting media without the need to regenerate point distributions, boundaries, or interpolation matrices.

ii) To formulate and implement the Inverse Geometric problem of Cavity Detection as an optimization problem that minimizes an objective function that computes the deviation of measured temperatures at accessible locations to those generated by the LRC-VoF Meshless method. The LRC-VoF Meshless algorithms will be driven by an optimization code based on the Genetic Algorithms which can efficiently search for the optimal set of design parameters (location, size, shape, etc.) within a predefined design space. Initial guesses to the search algorithm will be provided by the classical 1D semi-infinite composite analytical solution which can predict the approximate location but not the size or shape of the cavity.

iii) To test and validate the LRC-VoF Meshless Method Cavity Detection algorithms through a series of controlled numerical and laboratory experiments. A comprehensive sensitivity analysis and numerical tests will be conducted to quantify the robustness of the computational tool to error-induced measurements.

In addition, the solution method and tool will be further verified by comparing its predictions to the actual setup of a controlled laboratory experiment designed to acquire the surface thermal signatures through an IR camera from different heating conditions over a variety of composite conducting domains with different defect configurations, designed to simulate structural health monitoring scenarios.

The novel idea of integrating the LRC Meshless method coupled with a VoF scheme into an optimization framework formulated to search for cavities, voids, inclusions, defects, or de-attachments by parameterizing their location, size, and shape through a series of design variables will allow solving the onerous computational inverse geometric problem in a very efficient and robust manner while affording its implementation in modest computational platforms, thereby realizing the disruptive potential of the multi-dimensional high-fidelity non-destructive evaluation (NDE) method in displacing the current practice of 1D-based NDE.

CHAPTER 2

LITERATURE REVIEW

The Federal Highway Administration approximated in 2010 that there is a lot of money being used to replacement or rehabilitation for the bridges in the United States of America. The percentage of this money was 89.5% ($12.8 billion) [1] of the total capital outlay for the bridges. As the huge rate of the U.S. bridge inventory was established between 1950 and 1970 stages to age, with 3/2 of the all bridges in America build before 1964 [2], the total outlay required to keep the functionality of U.S. bridge infrastructure has grown an average of 7.3% per year between 2000 and 2008 [1]. For over thirty years, there is a considerable research that has been conducted in the field strengthening existing reinforced concrete (RC) structures with fiber-reinforced polymer (FRP) composite materials [3]. This study has been conducted in design standards and specification that engineers can use to acquire an extensive type of strengthening purposes [4]-[6], like growing the shear and flexural capacity of reinforced (RC) members and supplying extra confinement for RC columns. A main benefit of external strengthening with FRP composites is the on-site flexibility that is given by these materials (Figure 2-1). The wet lay-up method is used by reinforced concrete strengthening applications, and this method includes saturating dry fibers on-site with a polymer matrix material (usually epoxy) and applying the wetted composite to the concrete surface. Mechanical and chemical bond is founded between the concrete and the composite during curing and stresses are transported from the concrete to the composite via shear through the bondline as the structure is loaded.

This wet lay-up method gives the flexibility and puts these systems vulnerable to installation flaws. The extent to which installation defects clear as long-term durability concerns is not well understood, and non-destructive evaluation technique is not globally acceptable for observing durability of these systems. An enormous promise has been presented by thermal imaging technique for identifying the presence/absence of defects in a specific sense [7]-[9]. Moreover, quantitative methods have been used in laboratory settings to indicate the potential for defect characterization [10]-[11]. However, needing for a rapid, robust method is important, and this method can be used for completely characterizing the location, size, depth, and material composition of any anomalies that are happened during an infrared thermography inspection.

A simple FRP strengthening application is indicated in Figure 2-1 for an interstate overpass that was destroyed in a collision with an over-height vehicle out of Jacksonville, Florida. Great installation defects were shown by the qualitative thermal, but it is impossible to define the implicit cause of the flaws or their possible effects on the long-term efficiency of the repair. A rapidly deployable method supplies an overall description of the nature of encountered defects, and this method is considered as the first step towards advancing materials processing techniques to minimize the occurrence of defects in the first place. The methodology described in this research will drive to more efficient techniques for structural health for observing and ensuring that installed systems execute as expected.

(a)   (b)   (c)
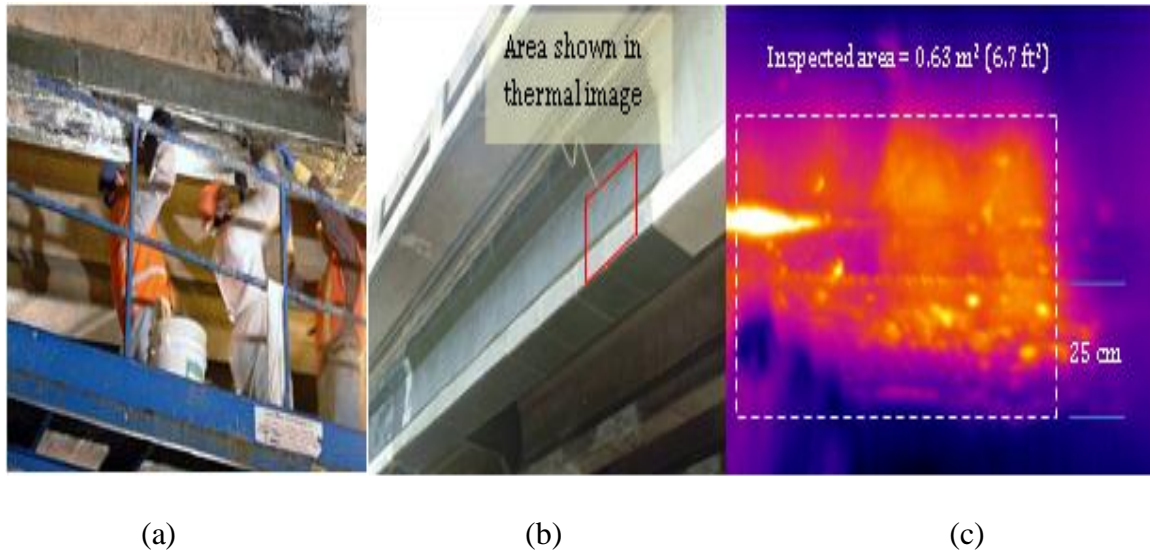
Figure 2-1. Application of FRP composite to strengthen existing interstate overpass. a) Workers applying carbon-fiber composite. b) Completed project. c) Qualitative thermal imaging results obtained during non-destructive evaluation.

Within the family of inverse heat transfer problems [12]-[16], the inverse geometric problem finds its application in the nondestructive evaluation of subsurface flaws and cavities. Here, the governing equation, the thermophysical properties, the initial condition, the boundary conditions, and the portion of the geometry which is exposed, are all known. However, the portion of the problem geometry that is hidden from view is unknown and to be determined with the help of an overspecified (Cauchy) condition at the exposed surface; see Figure 2-2. Specifically, the surface temperature and heat flux are given at the exposed surface and the geometry of the cavity(ies) that generated the measured temperature footprint is to be determined. The boundary condition at the cavity side is specified as either homogeneous or nonhomogeneous first, second, or third kind of boundary condition. Solution of the inverse geometric problem can be undertaken by considering either the transient or steady-state thermal response of the system subjected

6

to a thermal load. Consequently, there are two general categories of techniques for the solution of the inverse geometric problem: transient based (also known as thermal wave imaging methods) [17]-[21] and steady-state based (also known as infrared computerized axial tomography, IR CAT) [22]-[24]. In the case of the steady-state inverse geometric problem Ramm [25] demonstrates mathematically that the solution is unique for media with constant thermal conductivity.

The inverse geometric problem, which has been solved by a variety of numerical methods [26]-[33], and its closely related shape optimization problem [34]-[39], are arguably the most computationally intensive of all inverse heat transfer problems. This is due to their inherent nature, regardless of whether a numerical or analytical approach is taken to solve the associated direct problem, which requires a complete regeneration of the mesh as the geometry evolves. Moreover, the continuous evolution of the geometry itself poses certain difficulties in arriving at analytical or numerical sensitivity coefficients [40]-[42] for gradient-based optimization approaches and in the updates of the subsurface geometry(ies) and associated mesh(es), particularly in three dimensions, whether using domain-meshing methods such as finite-element or finite-volume methods, or boundary-meshing methods such as boundary elements [43]-[45], which have been developed extensively by Divo and Kassab along with their research [46]-[48].
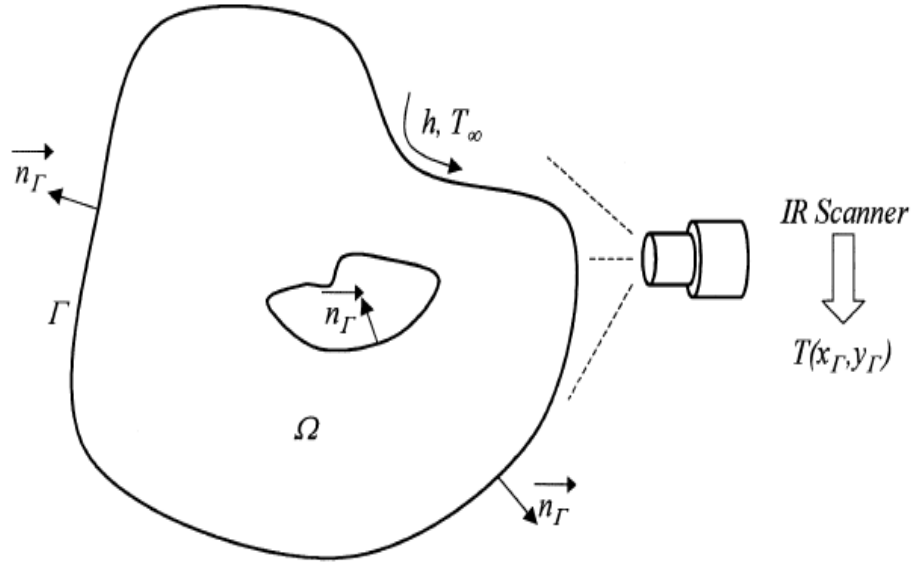
Figure 2-2. Problem setup using IR scanner to measure thermal footprint at the exposed boundary.

An efficient approach was introduced by Divo et al [49] where singularity clusters were employed in a boundary element method (BEM) heat conduction formulation to simulate the presence of subsurface cavities in 2D and 3D geometries. The efficiency of this approach comes from the fact that the problem geometry does not need to be regenerated during the search process. Instead, the search is performed for the location, distribution, and strength of singularity clusters that act as voids within the medium. This allowed for accurate and efficient identification of subsurface cavities without the need of regenerating geometries or BEM interpolation matrices. This technique was later extended by Ojeda, Divo, and Kassab [50] for biomechanical applications of cavity detection in cortical bones. In this case, the deformation field difference with respect to a measured field at the exposed boundaries was minimized by using an elastostatics BEM code and employing a variety of differently shaped anchored grid patterns (AGP) that adapt to the shape of the internal cavity using the efficient singularity superposition idea.

8

While finite-element methods (FEM), finite-volume methods (FVM), and boundary-element methods (BEM) have been developed to a mature stage such that they are now utilized routinely to model complex multi-physics problems, they require significant effort in mesh generation and problem setup. Meshless methods are a relative newcomer to the field of computational methods, and the term "Meshless Methods" refers to the class of numerical techniques that rely on either global or localized interpolation on non-ordered spatial point distributions. As such, there has been much interest in the development of these techniques as they have the hope of reducing the effort devoted to model preparation [51]-[57]. The approach finds its origin in classical spectral or pseudo-spectral methods [58]-[62] that are based on global orthogonal functions such as Legendre or Chebyshev polynomials requiring a regular nodal point distribution. In contrast, Meshless methods use a nodal or point distribution that is not required to be uniform or regular due to the fact that most such techniques rely on global radial-basis functions (RBF) [63]-[67]. RBF have proved quite successful in their application to an earlier mesh-reduction method, namely the dual reciprocity boundary element method (DRBEM). However, global RBF-based Meshless methods have some drawbacks, including poor conditioning of the ensuing algebraic set of equations, which can be addressed to some extent by domain decomposition and appropriate pre-conditioning [68]-[73]. Moreover, care must be taken in the evaluation of derivatives in global RBF-based Meshless methods. Although very promising, these techniques can also be computationally intensive. Recently, localized collocation Meshless methods [74]-[76] have been suggested to address many of the issues posed by global RBF Meshless methods.

In a series of recent publications [77]-[84], Divo, Kassab, and their group have developed a Localized Collocation Meshless Method (LCMM) based on Radial-Basis Function (RBF) interpolation for modeling of coupled viscous fluid flow, heat transfer problems, and fluid-structure interaction problems. The LCMM features Hardy Multiquadrics RBF augmented by polynomial expansions over a local topology of points for the sought-after unknowns with an efficient formulation for computing the interpolations in terms of vector products. This approach is applicable to explicit or implicit time marching schemes as well as steady-state iterative methods. The LCMM technique lends itself very well to parallel computations and has been shown to be computationally more efficient than a comparative finite volume method (FVM) code whilst affording the distinct advantage of solving the partial differential conservation field equations of fluid flow and heat transfer on a non-ordered set of points. The method has been extensively verified against benchmarks and validated finite volume codes for several cases. This technique has been implemented in the solution of inverse heat transfer problems [85] as well as shape optimization problems [86]. An alternative approach to the cavity detection problem using Meshless methods was proposed by Karageorghis et al [87]-[89] by formulating a moving pseudo-boundary method of fundamental solutions to detect voids and boundary locations.

The need arises for an efficient technique that avoids the implicit requirement of performing completely new solutions as the geometry is sought while offering the possibility for automation and the robustness of predicting the location, size, and shape of cavities, voids, inclusions, defects, or de-attachments within the conducting media.

An efficient numerical approach was introduced by Divo and Kassab [49] where singularity clusters were employed in a boundary element method (BEM) heat conduction formulation to simulate the presence of subsurface cavities in 2D and 3D geometries. The search was performed for the location, distribution, and strength of singularity clusters that act as voids within the medium and hence the formulation did not required remodeling or remeshing. This technique was later extended by Ojeda, Divo, and Kassab [50] for cavity detection in cortical bones. In this case, an elastostatics BEM formulation was employed to simulate the cavities as clusters of point loads within the domain. In these formulations, a Genetic Algorithm (GA) was used to optimize the objective function that measured the deviation between measured and BEM-generated field data. The technique, although proven to be accurate, efficient, and robust, was not capable of employing transient information due to the onerous nature of time-accurate formulation in BEM. However, very promising results were obtained validating the hypothesis that cavities, voids, and defects can be detected and modeled without the need for geometric reconstruction. Figure 2-3 shows a BEM-singularity cluster search for two cavities in a 2D hollow block after (a) first generation of the GA and (b) 3000 generations of the GA showing predicted cavity.

11
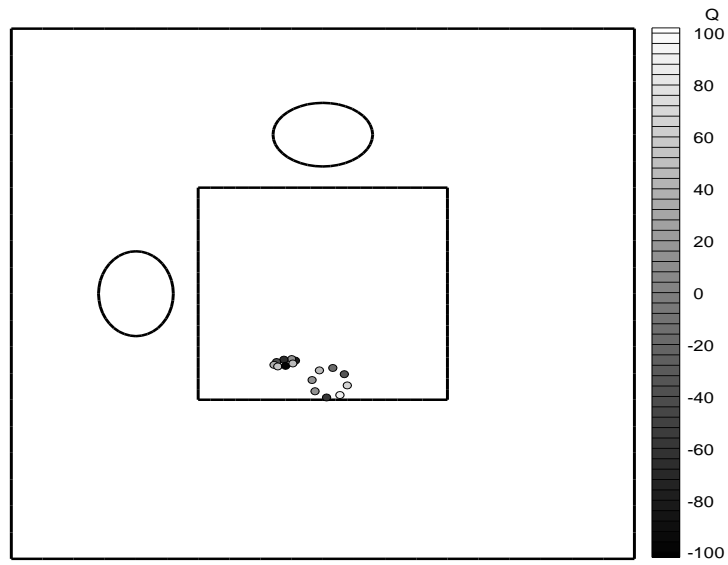
(a)
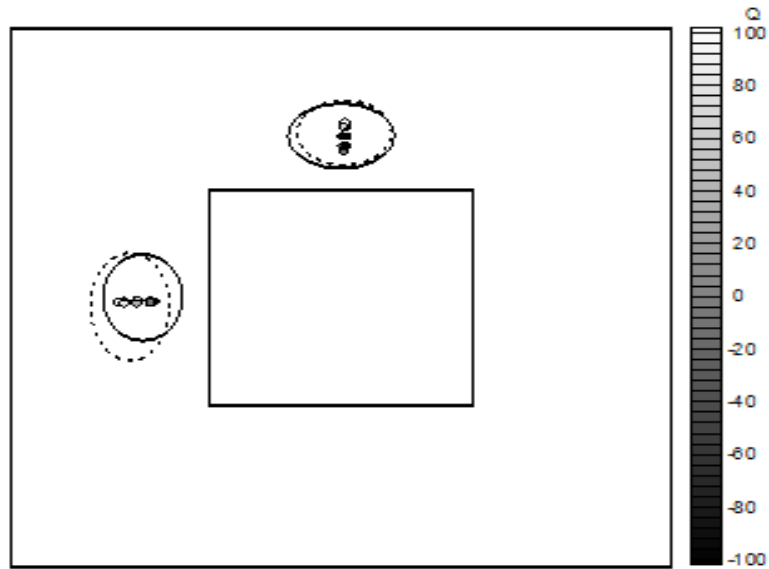
(b)

Figure 2-3. Singularity cluster search for two cavities in a 2D hollow block after (a) first generation of the GA and (b) 3000 generations of the GA showing predicted cavity.

Figure 2-4 shows a BEM-singularity cluster search for a cavity in a 3D hollow block after (a) first generation of the GA, (b) 2000 generations of the GA zoomed in at cavity, and (c) superimposed actual and retrieved cavities. Very good agreement was found between the actual and predicted cavities for the 2D and 3D examples using error-induced thermal footprint from heating through the exposed walls.
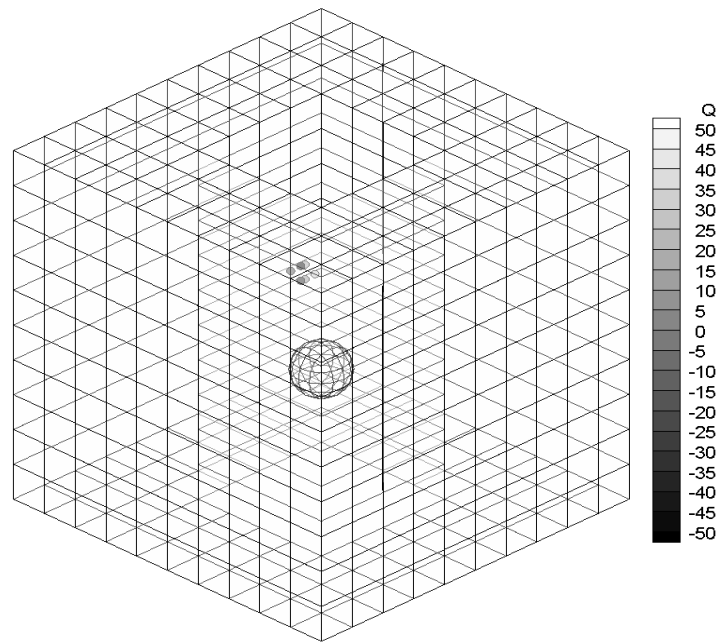
(a)
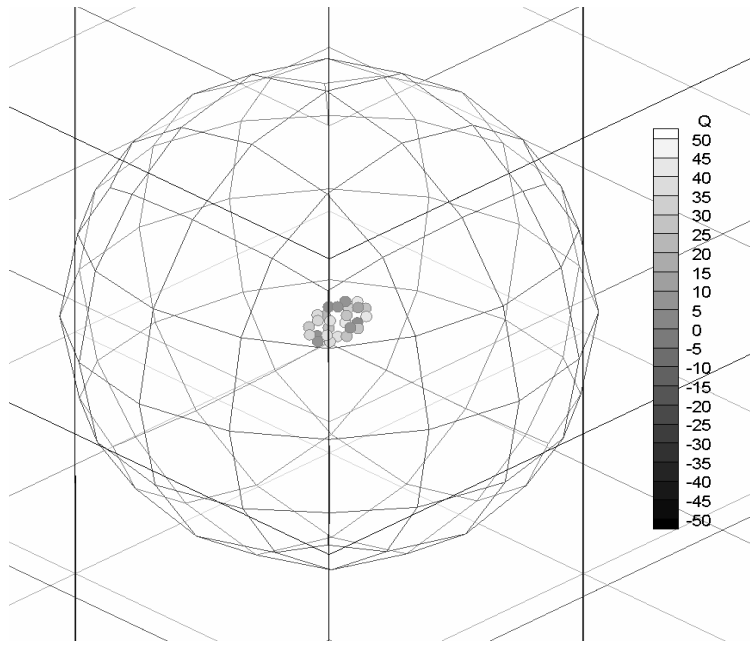
(b)
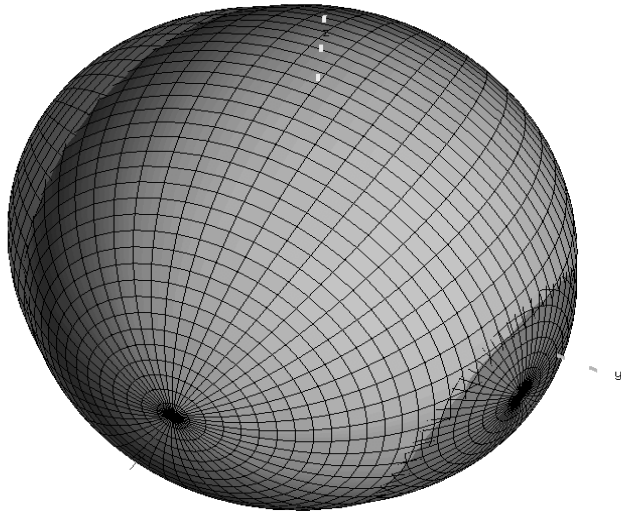
(c)

Figure 2-4. Singularity cluster search for a cavity in a 3D hollow block after (a) first generation of the GA, (b) 2000 generations of the GA zoomed in at cavity, and (c) superimposed actual and retrieved cavities.

CHAPTER 3
METHODOLOGY

3.1 THE LOCALIZED RBF COLLOCATION (LRC) MESHLESS METHOD

Traditional methods for the numerical solution of Fluid Flow and Heat Transfer problems contain Finite Difference Method (FDM), Finite Volume Method (FVM), Finite Element Method (FEM), and Boundary Element Method (BEM). In all these techniques, a Mesh or Grid is needed so as to create assumptions for the local approximation of the field variables and/or its derivatives on the boundary and in the interior of the domain of interest. The most time-consuming and man-power-demanding part of a numerical analysis is created by Meshing particularly for Fluid Flow problems where the numerical solution highly relies on the quality of the mesh.

A number of Meshless Methods have grown since the beginning of 1990's from the FEM community such as Diffuse Element Methods, Element-Free Galerkin Methods, Partition of Unity Methods, H-p Cloud Methods, Local Petrov-Galerkin Methods, and Reproducing Kernel Particle Methods. Even though all these methods are called Mesh-Free or Element-Free, it is necessary to mention that a mesh or shadow elements are important for integration goals in all situations. In parallel to the evolution of these methods, a various class of techniques appeared based on interpolation and collocation of global shape functions like the Trefftz Method, Method of Fundamental Solutions, and Radial-Basis Function Collocation Method. These techniques show the capability to globally clarify a field variable in a truly Meshless method, with no necessities for background meshes, point structure, or polygonalization.

However, while these techniques depend on global interpolation functions, large fully-populated, non-diagonally dominant, ill-conditioned matrices rise in their implementation. Therefore, particular care must be received in the selection and formulation of such interpolation functions as well as in the chosen of the resulting algebraic systems.

The Meshless formulation begins by defining a set of data centers, *NC*, comprised of points on the boundary, *NB*, and points on the interior, *NI*. These data centers will serve as collocation points for the localized expansion of the different field variables in the domain, Ω, and on the boundary, Γ, see Figure 3-1. The essential difference between boundary points and internal points is simply that boundary conditions will be applied at the first while governing equations will be applied at the last.
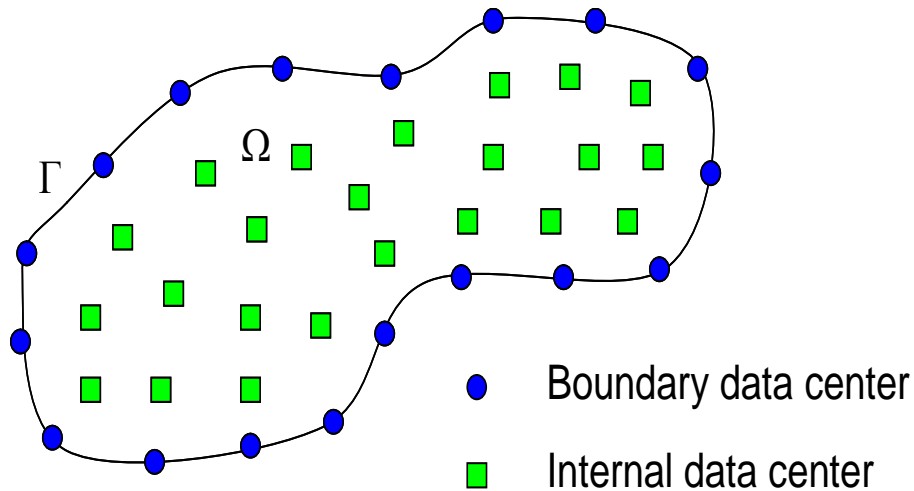
Figure 3-1. Scattered point distribution in a generalized domain

To illustrate the Meshless formulation the diffusion equation for the temperature, T, in a generalized coordinate system, x, and time, t, will be taken into consideration as

the governing equation valid in the domain, Ω, with constant conductivity, k, density, ρ, and specific heat capacity, c, as:

$$\frac{\partial T}{\partial t}(x,t) = \frac{k}{\rho c}\nabla^2 T(x,t) \tag{3-1}$$

In addition, a set of generalized boundary conditions on the boundary, Γ, are given by:

$$\hat{\beta}_1 \frac{\partial T}{\partial n} + \hat{\beta}_2 T = \hat{\beta}_3 \tag{3-2}$$

Where: $\hat{\beta}_1, \hat{\beta}_2, and\ \hat{\beta}_3$ are imposed coefficients of (x,t) that dictate the boundary condition type and constraint values. A linear localized expansion over a group or topology of influence points, *NF*, around each data center is sought such that:

$$T(x) = \sum_{j=1}^{NF} \alpha_j \chi_j(x) + \sum_{j=1}^{NP} \alpha_{j+NF} P_j(x) \tag{3-3}$$

The terms α_j represent the unknown expansion coefficients while the terms $\chi_j(x)$ are expansion functions defined a-priori. While *NP* is a number of additional polynomial functions, $P_j(x)$, added to the expansion to guarantee that constant and linear fields can be retrieved by the expansion exactly. Notice that the time dependency has been dropped as a different expansion will be performed for each time level and, therefore, the expansion coefficients, α_j, will vary as time progresses. The expansion functions $\chi_j(x)$ are selected as the Inverse Hardy Multiquadrics Radial-basis functions (RBF), defined as:

$$\chi_j(x) = \left[\left(\frac{r_j(x)}{d} \right)^2 + 1 \right]^{-\frac{1}{2}} \tag{3-4}$$

Here, the term $r_j(x)$ is the Euclidean distance from any point x to an expansion point x_j, while the term d is a shape parameter. The larger this shape parameter d the flatter the expansion function becomes and therefore the derivative field becomes smoother. However, the value of the shape parameter d cannot be increased indefinitely as the resulting coefficient matrix from the collocation process becomes ill-conditioned. A simple search process is performed to determine the optimal value of this shape parameter d for each localized expansion. The behavior of this RBF expansion function has been widely studied in the literature [64]-[73].

The selection of an influence region or localized topology of expansion around each data center is easily accomplished by a circular (spherical in 3D) search around each data center. The search is automated to guarantee that a minimum number of points is included and additional criteria, such as including all directions around internal data centers, are met. In addition, this search must guarantee that topologies around boundary data centers do not include opposing boundaries or points around a re-entry corner. Figure 3-2 shows a typical collocation topology for a non-uniform point distribution.
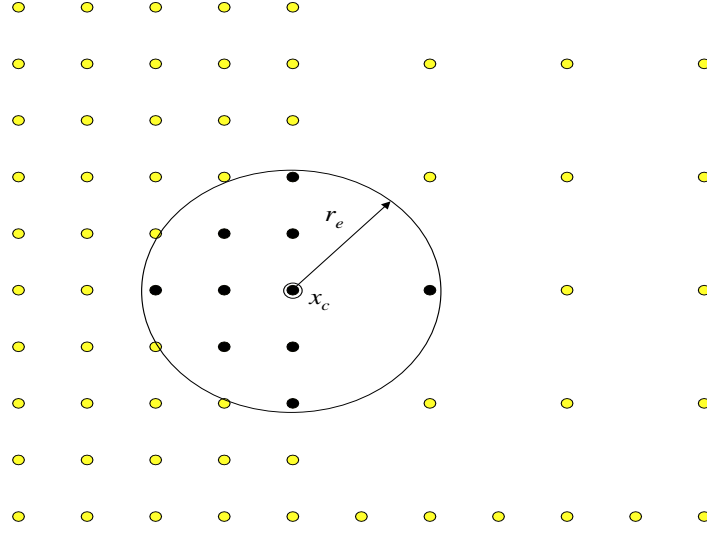
Figure 3-2. Collocation topology selection on a non-uniform point distribution.

The collocation of the known temperature field (from previous time level or iteration step) at the points within the localized topology, can be expressed in matrix-vector form as: $\{T\} = [C]\{\alpha\}$, and, therefore, the expansion coefficients can be determined as: $\{\alpha\} = [C]^{-1}\{T\}$. Where the resulting collocation matrix is given by:

$$[C] = \begin{bmatrix} \chi_1(x_1) & \cdots & \chi_{NF}(x_1) & P_1(x_1) & \cdots & P_{NP}(x_1) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \chi_1(x_{NF}) & \cdots & \chi_{NF}(x_{NF}) & P_1(x_{NF}) & \cdots & P_{NP}(x_{NF}) \\ P_1(x_1) & \cdots & P_1(x_{NF}) & 0 & \cdots & 0 \\ \cdots & \ddots & \cdots & \vdots & \ddots & \vdots \\ P_{NP}(x_1) & \cdots & P_{NP}(x_{NF}) & 0 & \cdots & 0 \end{bmatrix}_{NF+NP,NF+NP} \tag{3-5}$$

And the right-hand side known vector is augmented as:

$$\{T\} = \left\{ T(x_1)...T(x_{NF})\ 0...0 \right\}_{1,NF+NP}^{T} \tag{3-6}$$

19

Note that the polynomial-augmented matrix in Eqn. (3-5) guarantees constant and linear (or as high as the polynomial order employed) temperature fields to be expanded exactly. The augmentation of the temperature vector in Eqn. (3-6) with values of zero does not indicate zero temperatures but rather the dimensional consistency with the expansion matrix. The real advantage of the localized collocation approach is capitalized in the way the derivatives of the field variable are calculated at the data center, x_c of each topology. For instance, any linear differential operator L can be applied over the localized expansion equation as:

$$LT(x_c) = \sum_{j=1}^{NF} \alpha_j L\chi_j(x_c) + \sum_{j=1}^{NP} \alpha_{j+NF} LP_j(x_c) \qquad (3\text{-}7)$$

Or, in matrix-vector form: $LT_c = \{L_c\}^T\{\alpha\}$, where the derivative expansion vector $\{L_c\}$ is given as:

$$\{L_c\} = \left\{ L\chi_1(x_c)...L\chi_{NF}(x_c) \; LP_1(x_c)... \; LP_{NP}(x_c) \right\}^T_{1,NF+NP} \qquad (3\text{-}8)$$

Substitution of the expansion coefficients, $\{\alpha\}$, leads to:

$LT_c = \{L_c\}^T[C]^{-1}\{T\}$, and defining the derivative interpolation vector $\{L\}$ as:

$$\{L\}^T = \{L_c\}^T[C]^{-1} \qquad (3\text{-}9)$$

Leads to the final expression:

$$LT_c = \{L\}^T\{T\} \qquad (3\text{-}10)$$

The coefficients of the derivative interpolation vector $\{L\}$ of size $(NF,1)$ directly retrieve the value of the derivative of the temperature field $\{T\}$ at the data center of the topology x_c. Therefore, evaluation of the temperature derivatives at each of the data centers x_c is provided by a simple inner product of two small vectors: $\{L\}$ which can be

20

pre-built and stored at a setup stage of the problem as it is only dependent on geometry and point distribution, and $\{T\}$, which is the updated temperature field in the topology of the data center.

Furthermore, imposition of the generalized boundary conditions in Eqn. (3-2), at the boundary data centers, x_c, can be accomplished in a similar fashion. To aid the boundary interpolation an additional set of internal points that "shadow" each boundary point in the direction of the normal vector into the domain, as seen in Figure 3-3, are included in the point distribution and used to directly approximate the normal derivatives at each boundary data center.
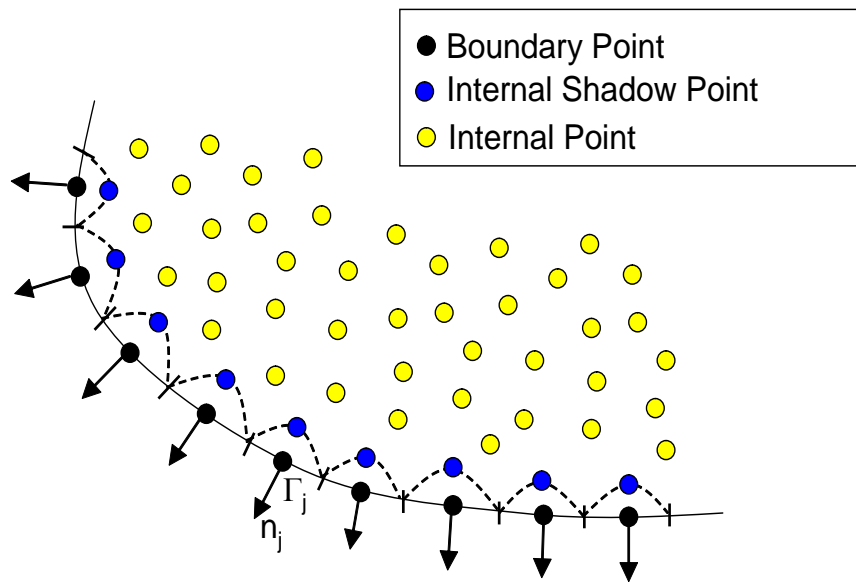
Figure 3-3. Distribution of internal shadow points to compute normal derivatives.

This localized expansion approach reduces the burden of the more common global interpolation methods [68]-[70] by expanding the field variable locally around each data center to obtain its derivatives that are then used in time-marching or iterative schemes. This approach yields the generation of multiple but small derivative interpolation vectors

21

that can be pre-built and stored at a setup stage of the problem as they are only dependent on geometry and point distribution. Additional interpolation vectors for Moving Least-Square (MLS) smoothing and Upwinding schemes can be pre-computed and stored in an analogous fashion, see [79]-[84].

3.2   THE VOLUME-OF-FLUID (VOF) METHOD

The Volume-of-Fluid (VoF) method was introduced by Hirt and Nichols [90] to approximate the behavior of two-phase non-mixing fluid flow problems by implicitly tracing the interface between the two dissimilar fluids through the transport of a continuous variable s that quantifies the absolute content of one of the fluids $(s = 1)$ or the absolute absence of it $(s = 0)$ as:

$$\frac{\partial s}{\partial t} + \left(\vec{V} \cdot \nabla \right) s = 0 \tag{3-11}$$

Therefore, the VoF parameter s is used to post-determine the location of the interface between the two phases, $(s = 0.5)$. This approach offers the great advantage that a two-phase flow problem can be modeled in a single domain through a single set of governing equations while the VoF parameter s is used as a weighting factor for the thermo-physical properties of the two fluids as, for instance, in the case of the thermal conductivities k^1 and k^2:

$$k = \left(1 - s \right) k^1 + s k^2 \tag{3-12}$$

One of the premises of this research is that the presence of a cavity within a thermally conducting medium can be simulated and approximated using a static version

22

of the VoF method, where the parameter s is not transported through the static field $\vec{V}=0$ but simply fixed at a value $s=0$ at the hypothetical location of the cavity and $s=1$ elsewhere. And, therefore, there is no need to model the actual geometry of the cavity. This technique was implemented by the authors in the setting of Meshless methods in [92].

3.3 THE INVERSE PROBLEM FORMULATION

The inverse problem of determining the location, size, and shape of the cavity may be formulated as an optimization problem whose objective is to minimize a function that computes the standard deviation between the LRC-VoF-computed temperatures T_i at the exposed boundaries and the temperatures acquired through IR measurements $\hat{T}_i$. This can be expressed as a least-squares function over a finite number of measurement locations N_m, or:

$$S(z) = \sqrt{\frac{1}{N_m} \sum_{i=1}^{N_m} \left[T_i(z) - \hat{T}_i \right]^2}$$

(3-13)

This objective function S depends on a number of geometric parameters z that define the location, size, and shape of the cluster of LRC Meshless points that are imposed with a VoF value $s=0$, i.e. a simulated cavity. For instance, in 2D, the cluster may be generally defined by an anchored Cubic spline set centered at (x_o, y_o) with a number of rays N_r each extending a distance r_i from the center, expressed in polar coordinates as:

$$r(\theta) = M_{i-1} \frac{(\theta_i - \theta)^3}{6\Delta\theta_i} + M_i \frac{(\theta - \theta_{i-1})^3}{6\Delta\theta_i} + \left(r_{i-1} - \frac{M_{i-1}\Delta\theta_i^2}{6} \right) \frac{\theta_i - \theta}{\Delta\theta_i} + \left(r_i - \frac{M_i \Delta\theta_i^2}{6} \right) \frac{\theta - \theta_{i-1}}{\Delta\theta_i} \quad (3\text{-}14)$$

Where M_i is the value of the second derivative of the spline at the node i, and $\Delta\theta_i$ is the angular spread of each spline, i.e. $\Delta\theta_i = \pi/4$ for $N_r = 8$. Requiring that the first and second derivatives are continuous at all nodes of the spline set results in a simple 8-dof tri-diagonal system for the values of M_i.

A sample anchored Cubic spline set is shown in Figure 3-4 for $N_r = 8$. All the LRC Meshless points that lie within the resulting anchored Cubic spline set will then be imposed with a VoF parameter value of $s = 0$.
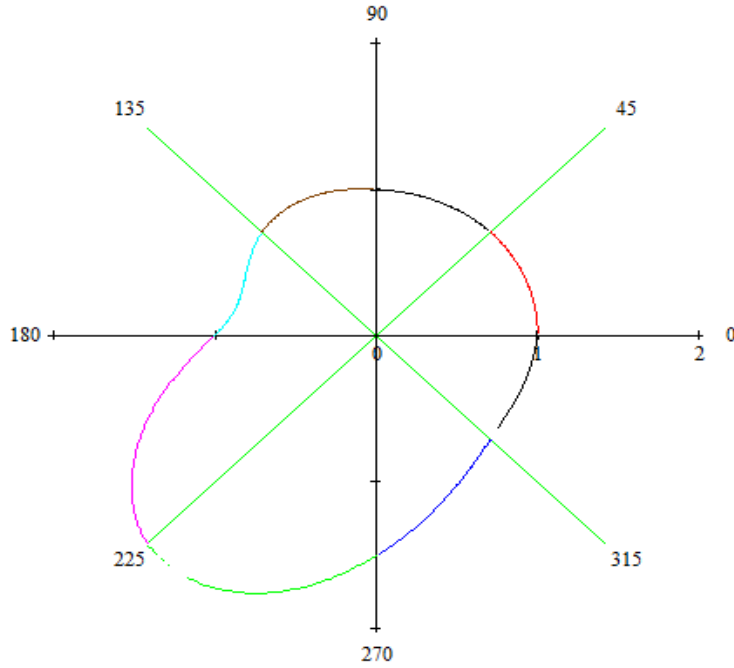
Figure 3-4. Anchored Cubic spline set for $N_r = 8$

Therefore, the parameters z in the objective function are the coordinates of the center and the size r_i of the N rays of the Cubic (in 2D) or bi-Cubic (in 3D) spline set.

3.4   GENETIC ALGORITHMS

The minimization of the objective function $S(z)$ in Eqn. (3-13) to approximate the size, shape, and location of the cavity may be accomplished by a non-gradient based method such as the Genetic Algorithms (GA), see [91]. GA are robust adaptive search techniques that mimic the idea of Darwinian evolution using rules of natural selection to investigate highly complex multidimensional problems. As a non-gradient-based optimization technique the use of GA is advantageous for this until a best-fit is found that application. The parameters that characterize the existence of the cavity may be progressively adjusted by the operators of the GA maximizes a fitness function. This fitness function can be easily and directly defined as the inverse of the least-square functional $S(z)$ as:

$$Z(z) = \frac{1}{S(z)}$$

(3-15)

The GA optimization process begins by setting a random set of possible solutions, called the population, with a fixed initial size or number of individuals. Note that one of these initial possible solutions may be set to that provided by the 1D initial guess described in the previous section. Each individual is defined by optimization variables and is represented as a bit string or a chromosome, see Figure 3-5. An objective function, Z_{GA}, is evaluated for every individual in the current population defining the fitness or their probability of survival. At every iteration of the GA, the processes of selection,

25

cross-over, and mutation operators are used to update the population of designs. A selection operator is first applied to the population in order to determine and select the individuals that are going to pass information in a mating process with the rest of the individuals in the population. This mating process is called the crossover operator, and it allows the genetic information contained in the best individuals to be combined to form offspring. Additionally, a mutation operator randomly affects the information obtained by the mating of individuals. This is a crucial step for continuous improvement.
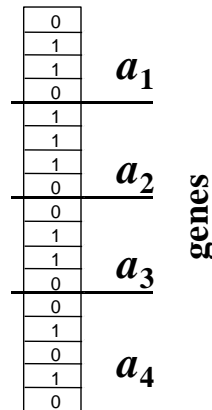
$$Z_{GA}(a_1, a_2, a_3, a_4) = F_1$$
chromosome

Figure 3-5. Example of an individual in the population characterized by four parameters (genes) encoded in a chromosome yielding the individual's fitness value *F1*.

A series of parameters are initially set in the GA code, and these determine and affect the performance of the genetic optimization process. The number of parameters per individual or optimization variables, the size of the bit string or chromosome that defines each individual, the number of individuals or population size per generation, the number of children from each mating, the probability of crossover, and the probability of

mutation are among the parameters that control the optimization process. This set of operations is carried out generation after generation until either a convergence criterion (a preset level of acceptable fitness) is satisfied or a maximum number of generations is reached. It is also important to point out that three important features distinguish GA from the others evolutionary algorithms, namely: (1) binary representation of the solution, (2) the proportional method of selection, and (3) mutation and crossover as primary methods of producing variations.

In nature, the properties of an organism are described by a string of genes in the chromosomes. Therefore, if one is trying to simulate nature using computers one must encode the design variable in a convenient way. We adopt a haploid model using a binary vector to model a single chromosome. The length of the vector is dictated by the number of design variables and the required precision of each design variable. Each design variable has to be bounded with a minimum and a maximum value and in the process the precision of the variable is determined. The number of divisions used in the discretization has to be integer power of two. This procedure allows an easy mapping from real numbers to binary strings and vice versa. This coding process represented by a binary string is one of the distinguishing features of GA and differentiates them from other evolutionary approaches. The haploid GA place all design variables into one binary string, called a chromosome or off-spring. The information contained in the string of vectors comprising the chromosome characterizes an individual in a population. In turn, each individual is equipped with a given set of design variables to which corresponds a value of the objective function. This value is the measure of "fitness" of the individual

design. In GA, poorly fit designs are not discarded, rather they are kept, as in nature, to provide genetic diversity in the evolution of the population. This genetic diversity is required to provide forward movement of the population during the mating, cross-over, and mutation processes which characterize the GA.

The initial population size may grow or diminish to mimic actual biological systems. However, in the GA used here, the population size is not allowed to change while the program is running. Once the population size is fixed, the algorithm initializes all of the chromosomes. This operation is carried out by assigning a random value of 0 or 1 for each bit contained in each of the chromosomes. After initializing the population, evaluation of the fitness of each individual is performed by computing the objective (or fitness) which of course represents a set of possible solutions. Having the values of the objective function for each individual, the selection process can be started. First values of the fitness function for each individual have to be added, and then the probability of being a selected individual is calculated as the ratio between the value of the fitness function of each individual and the sum of all objectives function values. This is given by:

$$P_{selected_i} = \frac{Z(z_i)}{\sum_{i=1}^{pop-size} Z(z_i)}$$
(3-16)

Where z_i is the ith member of the population, and $Z(z_i)$ is the measure of the fitness of that member under its currently evolved parameter set configuration. A weighted roulette wheel is generated, where each member of the current population is assigned a portion of the wheel in proportion to its probability of selection. The wheel is spun as many times as

there are individuals in the population to select which members mate. Obviously, some chromosomes would be selected more than once, where the best chromosomes get more copies, the average stay even, and the worst die off. Once selection has been applied, cross-over and mutation occur to the surviving individuals. These operations further expand genetic diversity in the current population. All other probabilities referred to in the description of the GA adopted in this research are computed in an analogous fashion as the selection probability.

The probability of crossover P_c is an important parameter that defines the expected population size of chromosomes which undergoes crossover operation. This is a mating process that allows individuals to interchange intrinsic information contained in the chromosomes. The operation may be implemented in two steps: (1) a random selection based on the probability of crossover is performed to obtain pairs of individuals, and (2) a random number is generated between the first position of the binary vector and the last one to indicate the location of the crossing point which delineates the location about which genetic information is interchanged between two chromosomes.

The mutation operator is the final operator implemented. The probability of mutation P_m gives the expected number of mutated bits and every bit in all chromosomes in the whole population has an equal chance to undergo mutation: switch of a bit from 0 to 1 or vice-versa. This process is implemented by generating a random number within the range (0...1) for each bit within the chromosome. If the generated number is smaller than P_m the bit is mutated. When the mutation is done on a bit-by-bit basis is called the

creep mutation. Another type of mutation is the jump mutation which is applied to an individual selected to be mutated from this perspective. In this case all bits within the chromosome are switched from 0 to 1 and vice-versa. Following selection, crossover and mutation the new population is ready for its next evolution until the convergence criteria "fitness" is reached. It is the very nature of the binary representation of the design variables of the objective function and the random search process which provide yet another but implicit degree of regularization in this optimization process. The sensitivity of the objective function can be tuned depending on the size of each element of the chromosome. Thus, low bit representation is insensitive to large variations in input (regularized but may lead to poor solution due to low resolution), while high bit representation is sensitive to large variations in input (not regularized and therefore may lead to poor solution as well). There is a range of bit size which produces a regularized and sensitive response leading to stable solutions.

In the GA employed in this research, the following parameters are chosen: population size of *20* individuals per generation, with strings of *8* bits for the *x* and *y* location of the anchored grid pattern as well as for the *8* rays of the pattern. The mating process produces one offspring per mating using uniform crossover which produces a higher level of diversity than single point crossover, a *4%* probability of jump mutation, *20%* probability of creep mutation, and *50%* probability of crossover. The population is not allowed to grow (static population) and elitist generation (the best parent survives to the next generation). The population is completely eliminated after 50 generations if there is no further improvement, keeping the best member of the population (restart). This

combination of GA parameters has been shown by the authors to provide robust results in

cavity detection problems as seen in [49].

CHAPTER 4
RESULTS AND DISCUSSION

4.1 DIRECT PROBLEM EXAMPLE

To illustrate this approach, the LRC Meshless method coupled with the VoF method is tested in a composite domain made of a ($1m \times 1m$) Concrete block ($\rho = 2300 kg / m^3$, $c = 880 J / kgK$, and $k = 1.4W / mK$) with an attached ($0.1m \times 1m$) epoxy layer ($\rho = 1000 kg / m^3$, $c = 1000 J / kgK$, and $k = 14W / mK$) with perfect thermal contact. The composite domain is at an initial temperature $T_i = 20°C$ and heated with a constant heat flux $q_0 = 1000W / m^2$ through the exposed epoxy wall while insulated elsewhere. A uniform LRC Meshless point distribution with $\Delta x = \Delta y = 0.01m$ was employed to model this problem as shown in Figure 4-1 along with the resulting temperature field after *1000s* of heating.
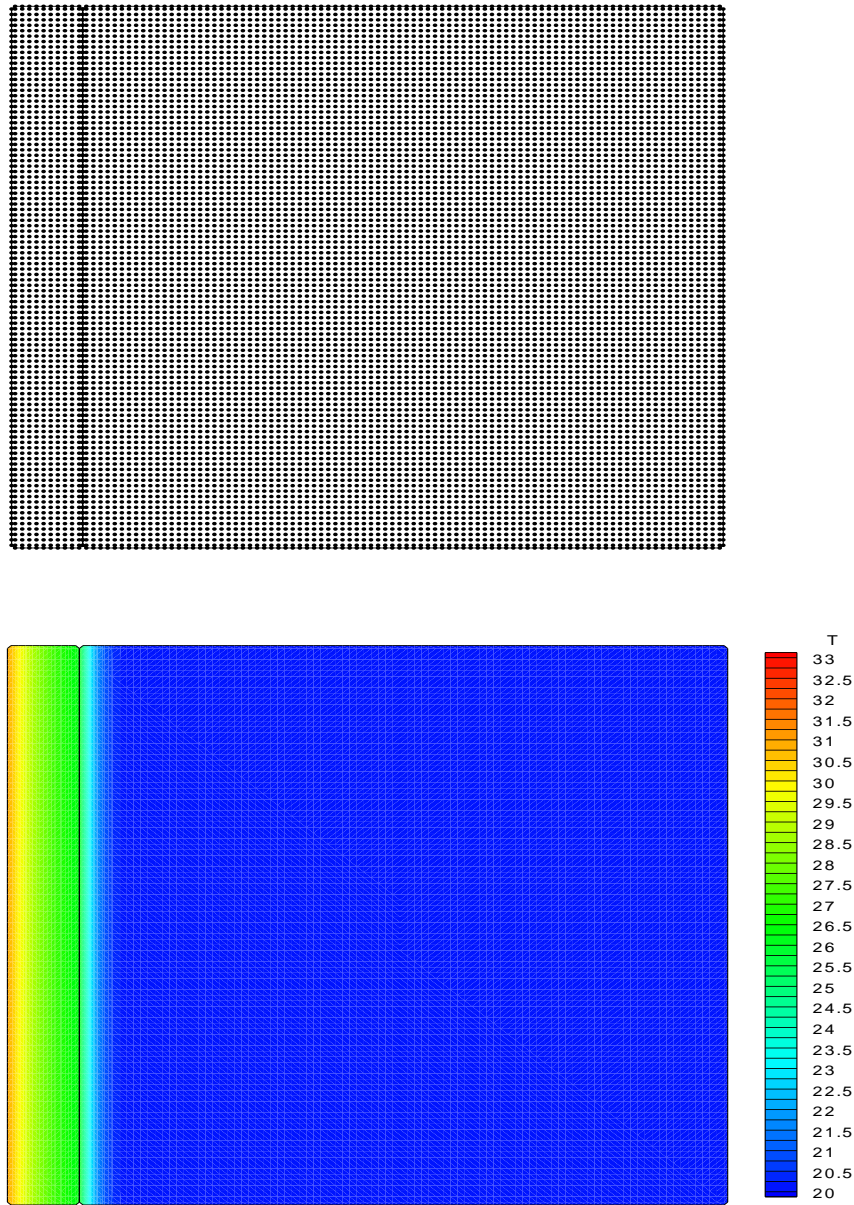
Figure 4-1. Meshless point distribution and resulting temperature field after *1000s* of heating.

Furthermore, a cavity is modeled by a $(0.01m \times 0.2m)$ sliver centered at the contact mid-point of the composite. In order to simulate this cavity using the VoF approach, the LRC Meshless points at the location of the expected cavity were imposed with a VoF

parameter $s = 0$, while a value of $s = 1$ was imposed elsewhere. The VoF parameter s essentially weights the thermo-physical properties of the epoxy with those of Air ($\rho = 1.2 kg / m^3$, $c = 1000 J / kgK$, and $k = 0.05 W / mK$) which acts as an almost perfect insulator ($k \sim 0$) but capable of diffusing energy ($\alpha = k / \rho c$) better than the epoxy. The resulting LRC Meshless temperature field after 1000s of heating is shown in Figure 4-2 for the case of (a) the actual cavity and (b) the VoF-simulated cavity. Notice that the temperature footprint provided by the VoF-simulated cavity is qualitatively very similar to that provided by the actual cavity. This feature is revealed in more detail in Figure 4-3 where the temperature profiles at the exposed wall are compared after 1000s of heating. Notice that the profiles are in very close agreement, demonstrating the capacity of the VoF method to simulate not only the presence of a cavity but also its location, size, and shape, as these features are captured by the cluster of LRC Meshless points that were imposed with the VoF parameter $s = 0$. Figure 4-3 also shows the constant temperature profile produced by the attached domain (or composite in perfect thermal contact), revealing that there is sufficient sensitivity ($\sim 2°C$) to detect the thermal footprint produced by a de-attachment or cavity by standard measurement techniques such as infrared (IR) thermography.
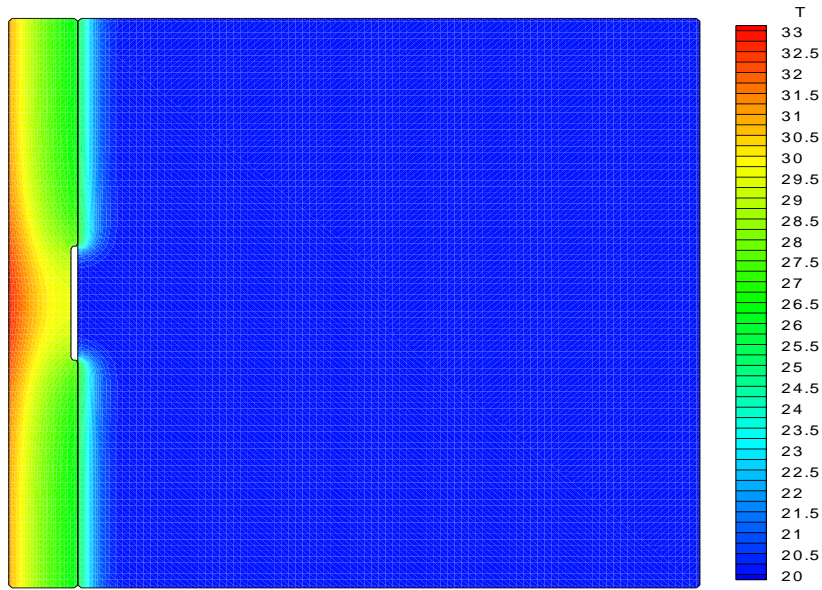
(a)
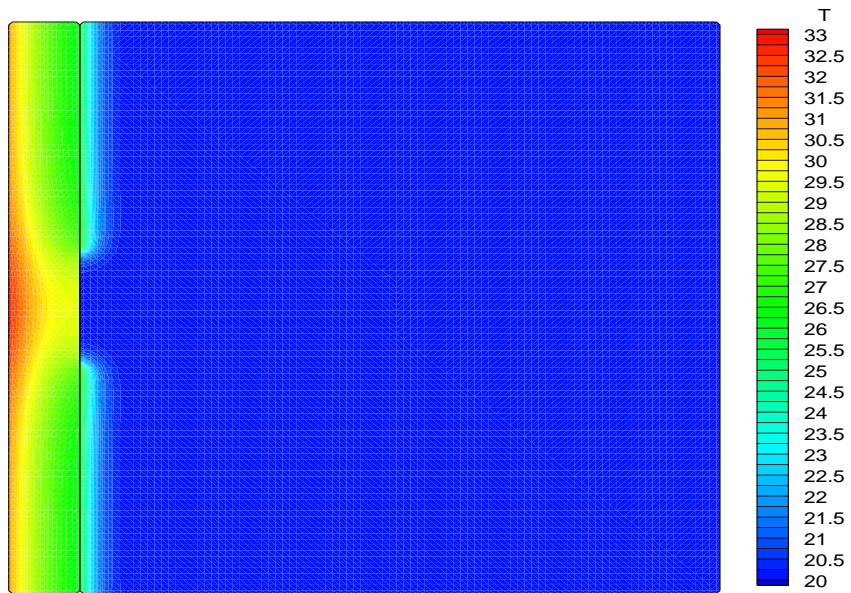
(b)

Figure 4-2. LRC Meshless temperature field after *1000s* of heating. (a) Actual cavity and
(b) VoF-simulated cavity.

35
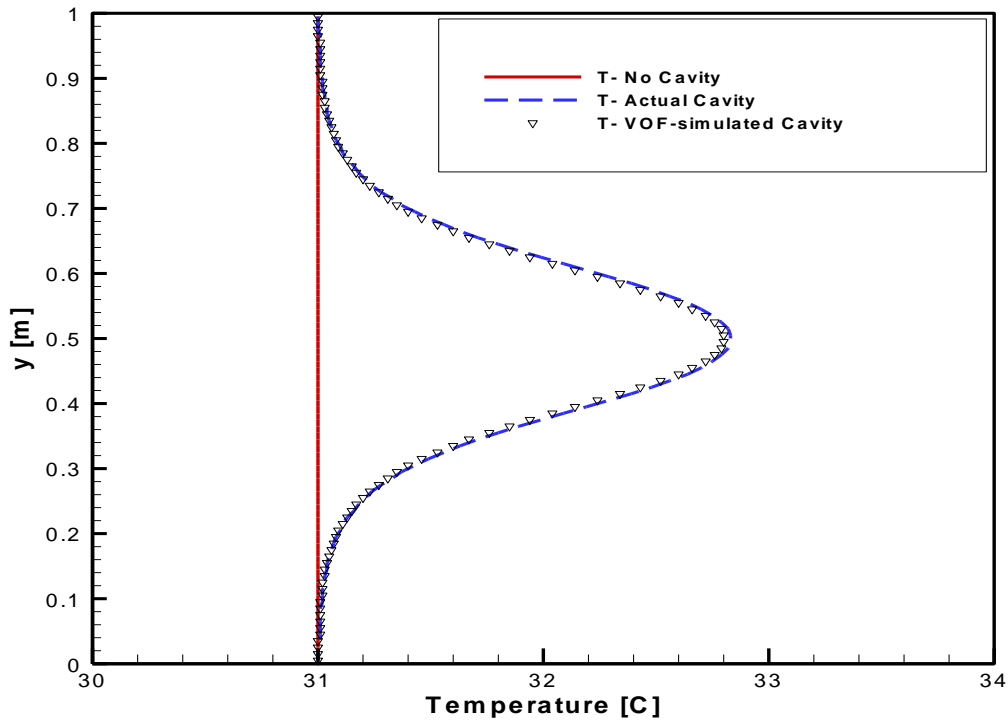
Figure 4-3. Temperature distribution on left-hand side wall after *t=1000s* of heating.

4.2 Circular Cavity Example

This is another example of the direct problem that shows the LRC Meshless method coupled with the VoF method is examined in a composite domain made of a ($1m \times 1m$) Concrete block ($\rho = 2300kg / m^3$, $c = 880J / kgK$, and $k = 1.4W / mK$) with a circular cavity centered *x=0.2m* and *y=0.2m* from the lower left-hand side walls with a radius of the circular cavity *r=0.1m* with perfect thermal contact. The composite domain is set with an initial temperature $T_i = 20°C$ and heated with a constant heat flux ($q_0 = 1000W / m^2$) through the hand left side and bottom side while the other sides are insulated. The total time was 10 hours to allow for thermal penetration and thermal

signature of the cavity to be discernible on the boundaries. Figure 4-4 shows the Meshless point collocation *(100×100)*.
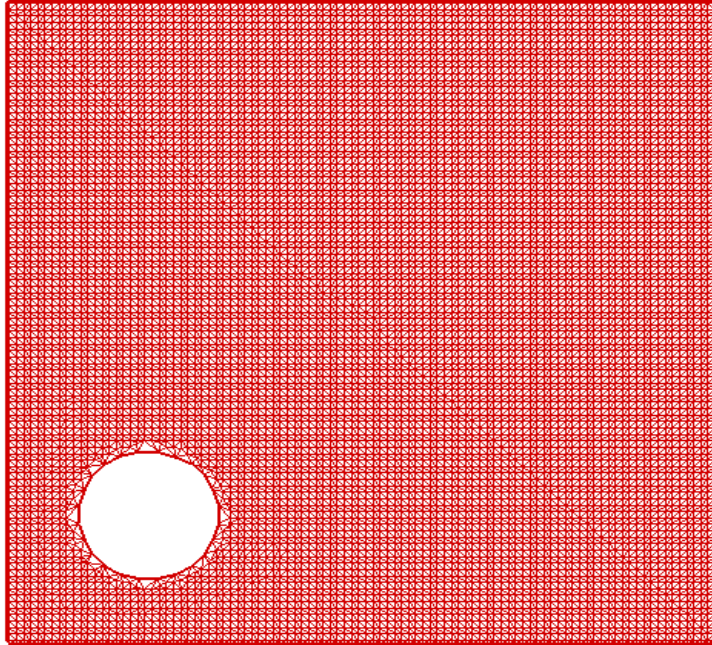
Figure 4-4. Meshless point collocation *(100x100)*

In addition, in order to simulate circular cavity utilizing the VoF approach, the LRC Meshless points at the location of the expected cavity were imposed with a VoF parameter $s = 0$, while a value of $s = 1$ was imposed elsewhere. The VoF parameter s basically weights the thermo-physical properties of Air ($\rho = 1.2 kg / m^3$, $c = 1000 J / kgK$, and $k = 0.05 W / mK$) which acts as an almost perfect insulator ($k \sim 0$). The resulting LRC Meshless temperature field after *10hrs* of heating is shown in Figure 4-5 for the case of (a) the actual circular cavity and (b) the VoF-simulated circular cavity.
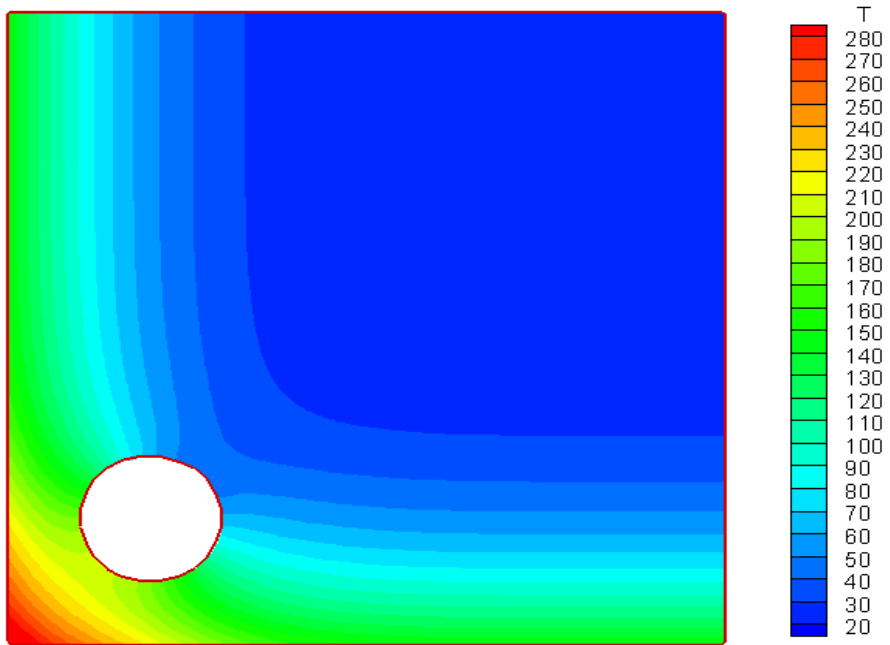
(a) Actual circular cavity
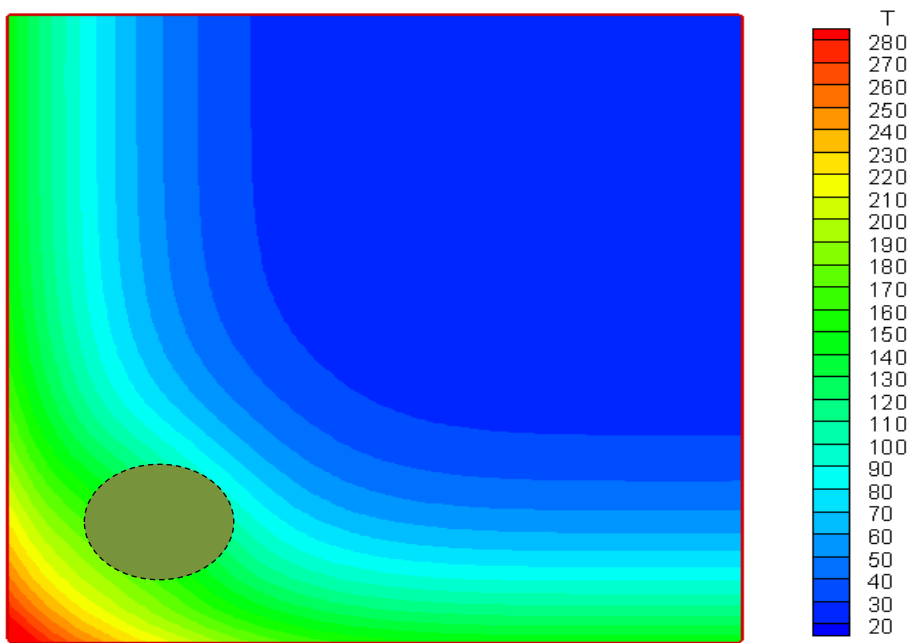
(b) VoF-simulated circular cavity

Figure 4-5. The resulting LRC Meshless temperature field after *10hrs* of heating

Notice that the temperature footprint provided by the VoF-simulated cavity is qualitatively almost similar to that provided by the actual cavity. Notice that the profiles are in very close agreement, demonstrating the capacity of the VoF method to simulate not only the presence of a cavity but also its location, size, and shape, as these features are captured by the cluster of LRC Meshless points that were imposed with the VoF parameter $s = 0$. Figure 4-6 also demonstrates the comparison of the temperature distribution for one hour (between *9hr* and *10hr*) of heating with (a) bottom side and (b) left-hand side.
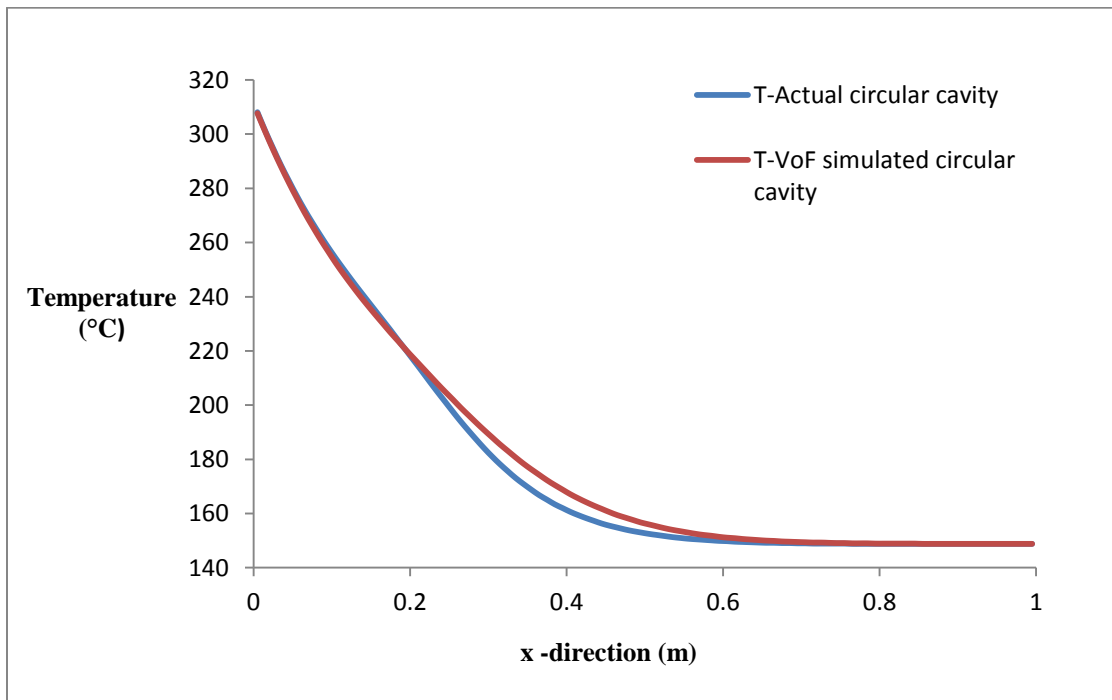
(a) Bottom side
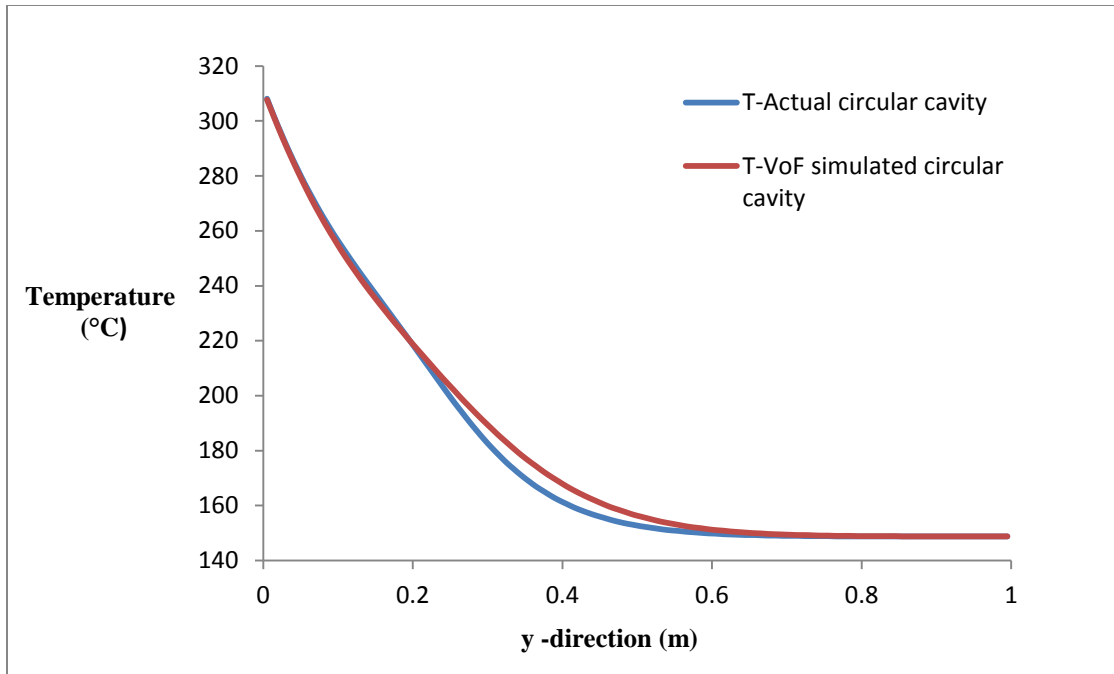
(b) Left-hand side

Figure 4-6. The temperature distribution for one hour (between *9hr* and *10hr*) of heating

4.3 1D INITIAL GUESS FOR CAVITY LOCATION

The search process to determine the size, shape, and location of the cavity may be aided by a good initial guess provided by the classical 1D temperature distribution solution. For instance, for the case of the composite domain studied in the previous section, the 1D temperature evolution at the heated wall of a finite layer of length L and properties k_0, ρ_0, and c_0, attached to a semi-infinite substrate with properties k_1, ρ_1, and c_1 is given by (recall that the thermal diffusivity $\alpha = k / \rho c$ and the thermal effusivity $e = \sqrt{k \rho c}$):

$$\Delta T(t) = \frac{2q_0}{k_0} \sqrt{\frac{\alpha_0 t}{\pi}} \left\{ 1 - 2 \sum_{n=1}^{\infty} \left(\frac{e_1 - e_0}{e_1 + e_0} \right)^n \left[\exp\left(-\frac{n^2 L^2}{\alpha_0 t} \right) - \frac{nL\sqrt{\pi}}{\sqrt{\alpha_0 t}} \, erfc\left(\frac{nL}{\sqrt{\alpha_0 t}} \right) \right] \right\} \quad (4\text{-}1)$$

To predict the temperature response produced by a de-attachment, the properties of the substrate material index 1 can be substituted for those of air (for example). The temperature responses for the problem described in the previous section with $q_0 = 1000W / m^2$ are shown in Figure 4-7. Here, the transient response at the center point of the heated wall ($y = 0.5m$) provided by the LRC Meshless solution with the actual cavity and the VoF-simulated cavity are shown in comparison with the temperature response provided by the 1D solution in Eqn. (4-1). Notice that the 1D solution tends to under-predict the actual temperature response due to its inability to factor in the actual size of the de-attachment. However, the 1D solution provides a good initial approximation for the cavity location which can be used to feed the search algorithm.
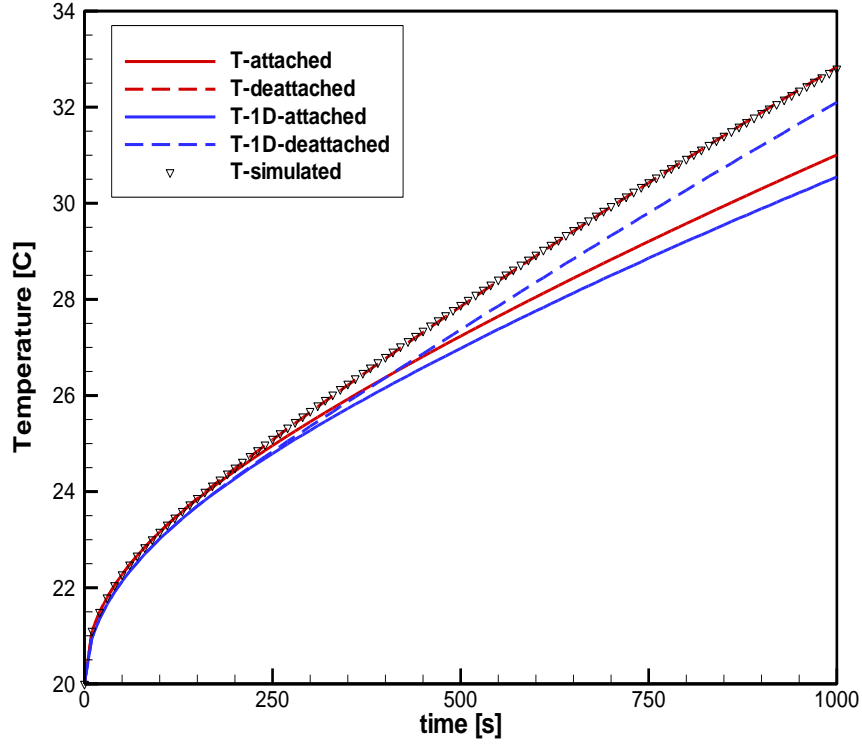
Figure 4-7. Temperature response at the mid-point of the heated wall provided by the LRC-VoF Meshless and the 1D solutions.

4.4 INVERSE PROBLEM EXAMPLE

A numerical example is now devised as a simulated experiment to approximate the size, shape, and location of a known cavity. The domain is a *1m×1m* concrete *(ρ=2300kg/m³, k=1.4W/mK, c_p=880J/kgK)* block with an elliptical cavity centered *x=0.1m* and *y=0.2m* from the lower left corner, with a horizontal radius r_x=0.025m and a vertical radius r_y=0.05m. The bottom and left-hand side walls are heated with a uniform flux *(q=1000W/m²)* while the other two walls are kept insulated. The initial temperature of the block is set to *20℃* and heating is continued for a total time of *10hrs* to allow for thermal penetration and for the thermal signature of the cavity to be discernible on the

boundaries. Figure 4-8 shows the Meshless point collocation *(100×100)* and the resulting

temperature contours after *10hrs* of heating.
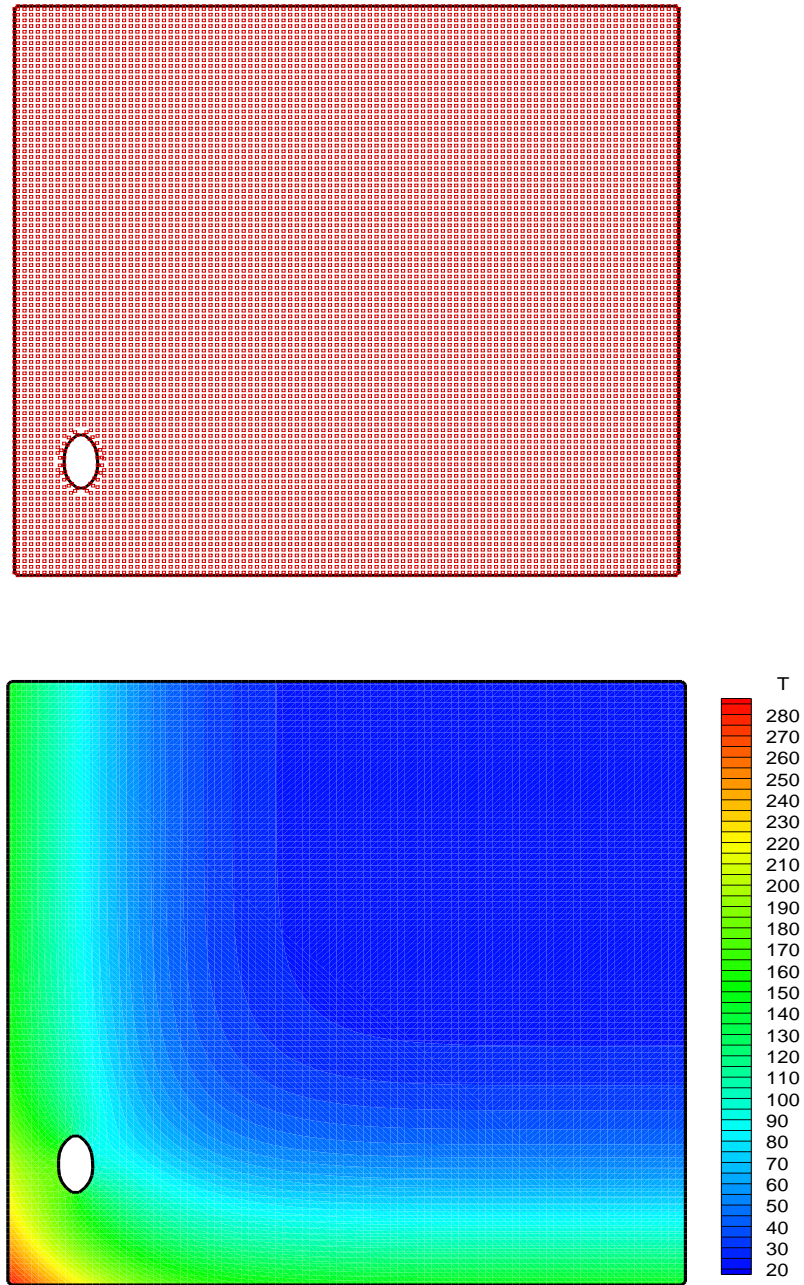
Figure 4-8. Meshless point collocation and temperature contours of cavity detection numerical experiment.

Next, the temperatures at the heated boundaries (bottom and left-hand side) were recorded every hour throughout the *10hrs* of heating to be used as the temperature measurements for the simulated inverse problem. These temperatures were rounded to the first decimal place to simulate a uniform error distribution of *±0.05°C*. Figure 4-9 displays the hourly evolution of the boundary temperatures over *10hrs* of heating along the bottom boundary and along the left-hand side boundary.
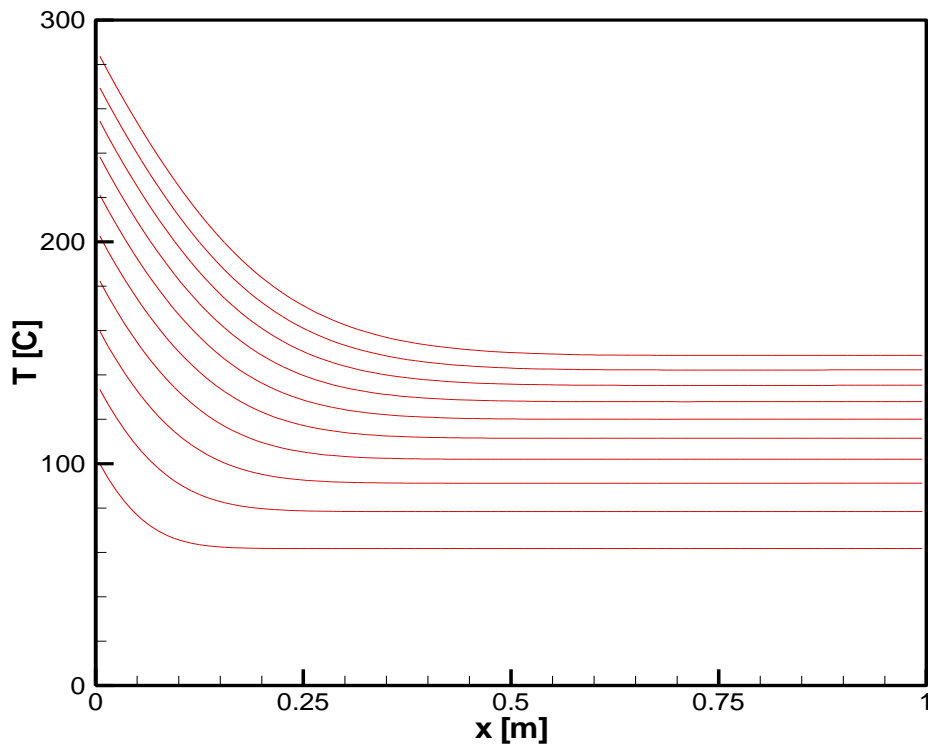
(a)
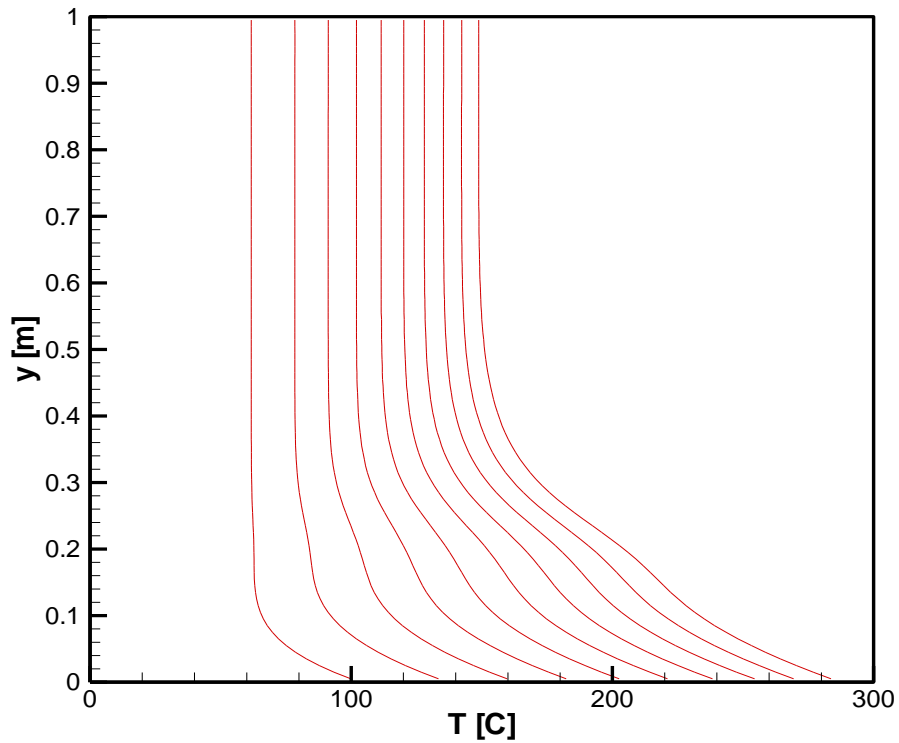
(b)

Figure 4-9. Evolution of the boundary temperatures over *10hrs* of heating: (a) bottom boundary and (b) left-hand side boundary

The GA was then executed using the parameters detailed in the previous section. The rounded *(±0.05°C)* boundary temperatures obtained from the simulated experiment (shown in (b)

Figure 4-9) where used as the measurements values for the objective function in Eqn. (3-13). A linear weighting factor equal to the elapsed time in *hrs* was used to impose higher weights on later measurements than on earlier ones. This was done to ensure that the objective function is tilted towards later measurements which are more sensitive to the presence of the cavity. The *1m×1m* concrete solid block is heated for *10hrs* with LRC Meshless points that are imposed with VoF values $s=1$ everywhere except for values of

$s = 0$ within the location of the anchored grid pattern produced by each GA individual in the population. This leads to each GA individual to yield its own boundary temperatures to be compared to those obtained by simulated measurements in the fitness function in Eqn. (3-15). A plot of the evolution of the fitness function $Z(z)$ for the best individual of each of the first 200 GA generations is shown in Figure 4-10.
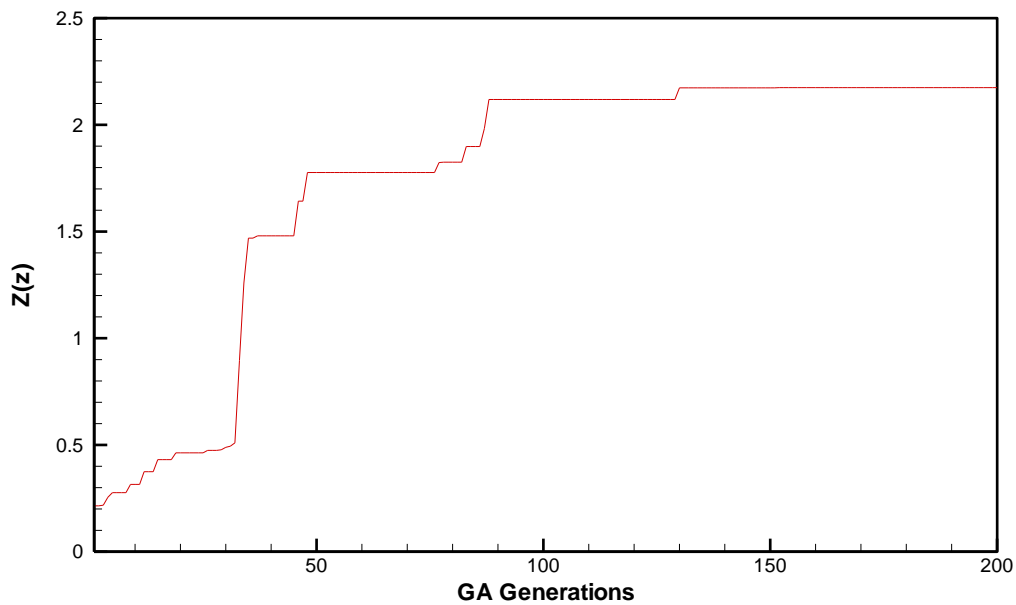
Figure 4-10. Evolution of GA fitness function.

The temperature contour plots provided by the actual cavity of the numerical experiment and the VoF-simulated cavity found by the GA search process are provided in Figure 4-11. Notice that the cavity found by the GA search process approximates very well the one used in the direct problem to generate the numerical experiment measurements. This is seen in more detailed in Figure 4-12 with a close-up plot of the actual cavity superimposed with the GA-found VoF-simulated cavity, revealing a very good approximation in just 200 generations of the GA.

46
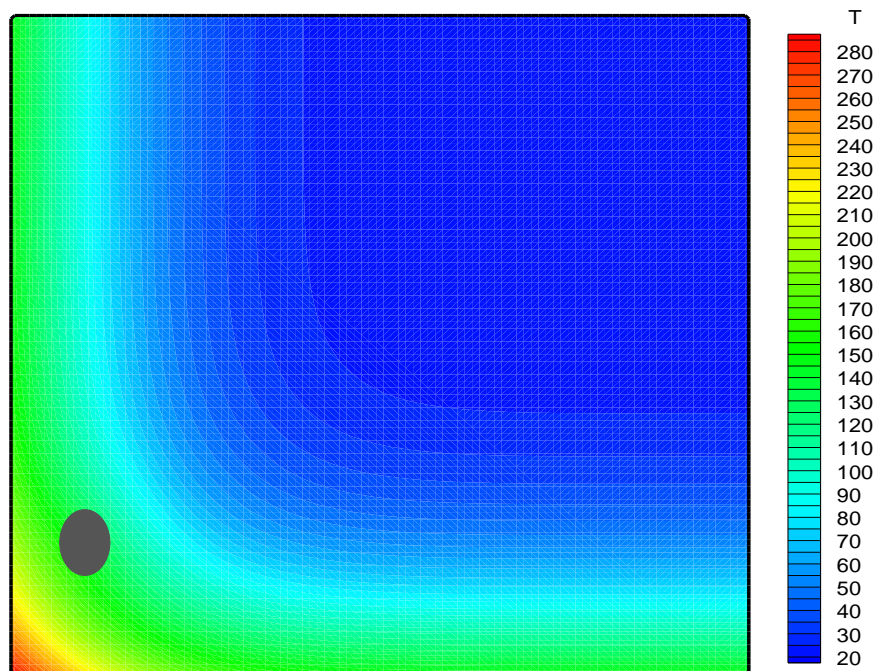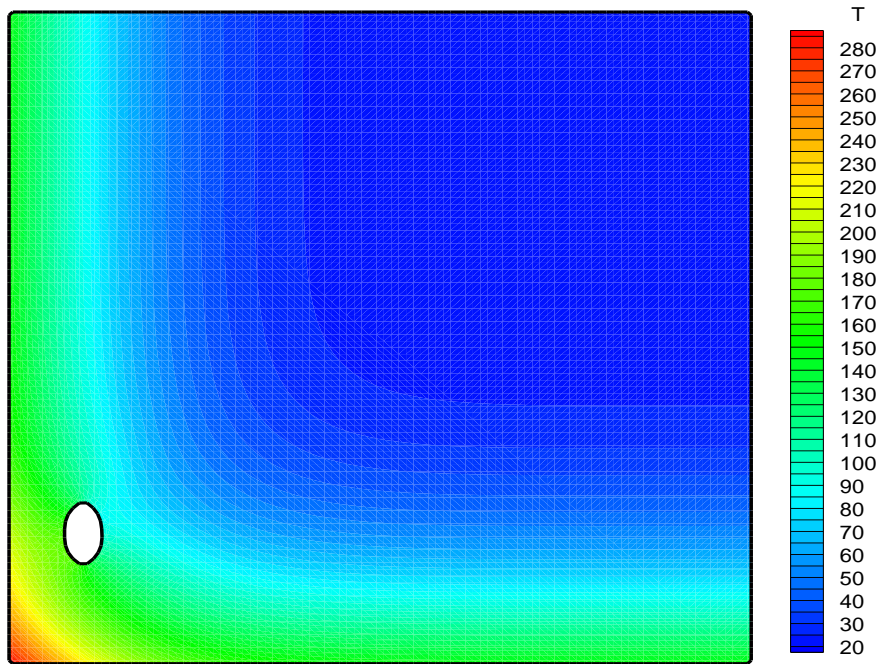
Figure 4-11. Temperature contour plots provided by the actual cavity of the numerical experiment and the VoF-simulated cavity found by the GA search process.
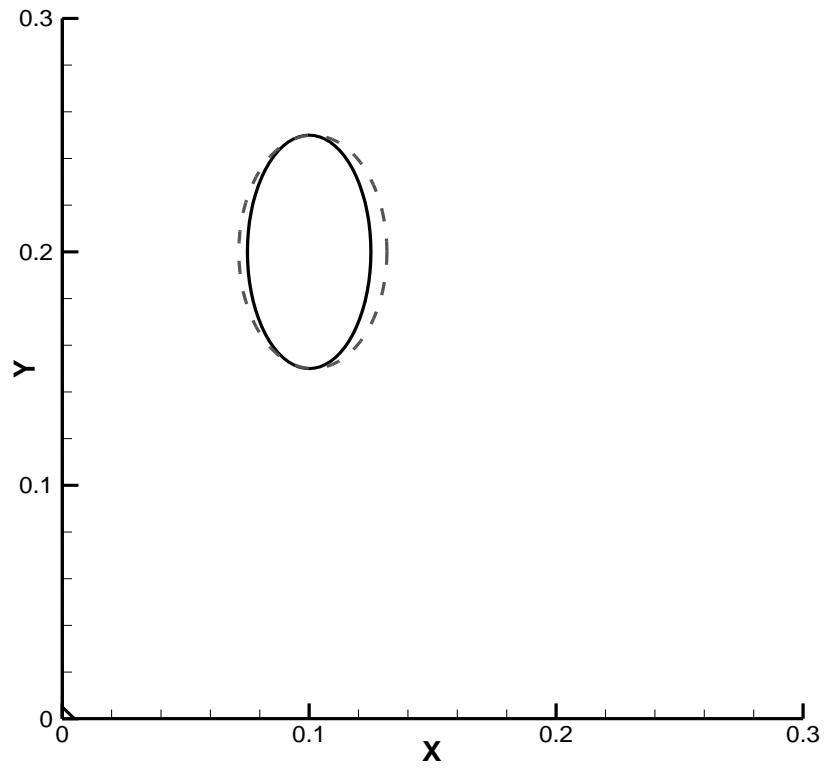
Figure 4-12. Close-up plot of the actual cavity (solid line) superimposed with the GA-found VoF-simulated cavity (dashed line).

CHAPTER 5

FUTURE WORK

An experiment can be designed to further validate the methodology. Concrete block samples can be manufactured in the Civil Engineering Materials Testing Lab at ERAU. There are many concrete mix designs which can account for the variability in material properties encountered across the concrete industry. Specific changes such as coarse aggregate type and water-cement ratio can be examined. Moreover, the industrial and application techniques for the FRP composites used in this research can be investigated. Primary stages of the experimental work can comprise of bonding high-quality, pre-cured laminates with familiar fiber volume fractions to the concrete substrate utilizing an epoxy-based adhesive. This is considered the best situation for replicating the material properties utilized in the numerical simulations. Wet lay-up FRP composites can be used during later stages to determine how the model's ability might be reduced by manufacturing defects and fiber volume fraction variation to precisely characterize sub-surface defects.

Furthermore, there are various manufactured defect types that can be investigated. Drilling holes can simulate natural defects in the finished surface of actual concrete structures with changing depth (0.125in-0.25in) and diameter (0.25in-0.5in). Utilizing teflon inserts can be simulated de-bonding between the FRP and the concrete substrate that happens after the composite has completely treated. Changing the diameter of the support ring which is put on the surface of the composite can control the size of delamination. Conceptual design for test fixture to generate simulated delamination within the concrete substrate is shown in Figure 5-1.
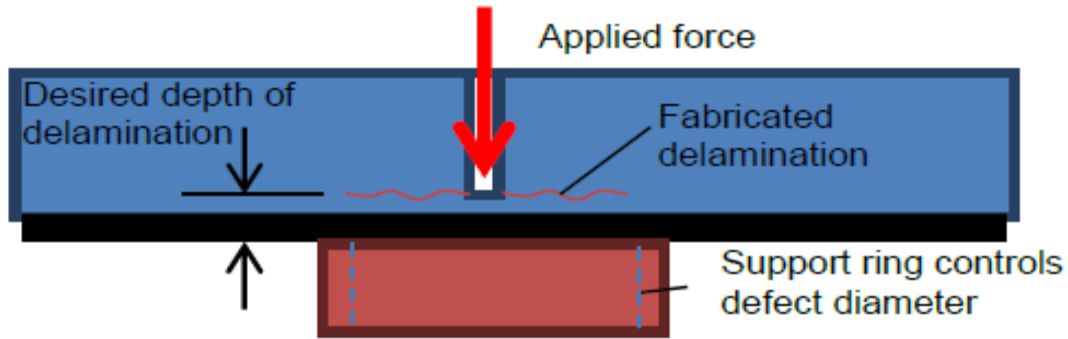
Figure 5-1. Conceptual design for test fixture to create simulated delamination within the concrete substrate.

Utilizing thermal paste in a heat flux sensor and flexible rubber heater can be added to one side of the concrete block for opposite of the FRP side. The block on all five sides which are going away the side with the FRP-exposed can be insulated. Initial tests can be implemented in the 1492 ft^2 Clean Energy Systems (CES) Laboratory at Embry-Riddle Aeronautical University. The tests can possess a dedicated Dell Precision Workstation T3500 Quad Core Intel® Xeon® E5506 2.8Ghz 4 GB Ram computer. Data recording of temperatures utilizing type E thermocouples and heat flux sensors can be implemented with an Agilent 34970A Data Acquisition/Data Logger Switch Unit with an Agilent 34901A 20 Channel Multiplexer. Presently, a Flir E40 49001-2001 IR camera with an uncooled micro-bolometer detector and a spectral band of 7.5–13μm is obtainable at the CES Lab. The resolution of the IR camera is 160x120 pixels. For the other properties of this camera, the thermal sensitivity is 70mK, the field of view is 0.4m, and the image frequency is 60Hz. A higher resolution Flir A655sc can be gained if this award is achieved. This camera contains a higher resolution of 640x480 pixels and a thermal sensitivity of 50mK, letting for a more accurate acquisition of the thermal footprint. The experimental setup for block heating and data acquisition is indicated in Figure 5-2
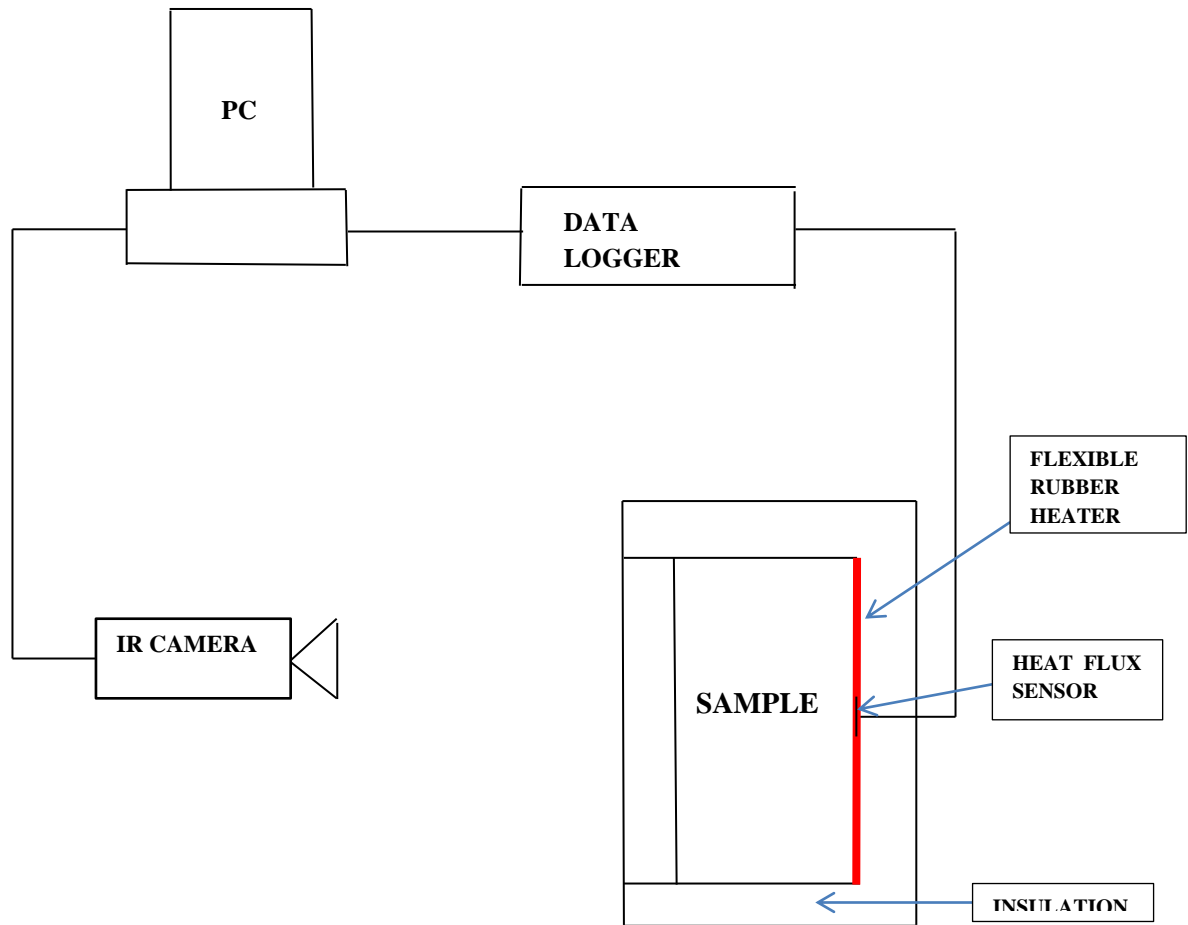
Figure 5-2. Experimental setup for block heating and data acquisition.

CHAPTER 6
CONCLUSIONS AND RECOMMENDATIONS

CONCLUSIONS

A novel computational tool based on the Localized Radial-basis Function (RBF) Collocation (LRC) Meshless method coupled with a Volume-of-Fluid (VoF) scheme capable of accurately and efficiently solving transient multi-dimensional heat conduction problems in composite and heterogeneous media is formulated and implemented. While the LRC Meshless method lends its inherent advantages of spectral convergence and ease of automation, the VoF scheme allows to effectively and efficiently simulate the location, size, and shape of cavities, voids, inclusions, defects, or de-attachments in the conducting media without the need to regenerate point distributions, boundaries, or interpolation matrices. To this end, the Inverse Geometric problem of Cavity Detection is formulated as an optimization problem that minimizes an objective function that computes the deviation of measured temperatures at accessible locations to those generated by the LRC-VoF Meshless method. The LRC-VoF Meshless algorithms is driven by an optimization code based on the Genetic Algorithms technique which efficiently searches for the optimal set of design parameters (location, size, shape, etc.) within a predefined design space provided by an anchored grid pattern. Initial guesses to the search algorithm are provided by the classical 1D semi-infinite composite analytical solution which can predict the approximate location of the cavity. The LRC-VoF formulation is tested using numerical experiments that reveal a high degree of accuracy and serve to validate the approach.

RECOMMENDATIONS

The following recommendations can be considered for future research:

- Inverse heat transfer problem

  ➢ Using square concrete block (1mx1m) with heat flux in two sides and heat convection in the other sides.

- Implementation Method

  ➢ Using an optimization code based on the Simplex Linear Programming algorithm to build the code of the LRC-VoF Meshless algorithms.

REFERENCES

[1] U.S. House of Representatives Committee on Transportation and Infrastructure, Memo to House members of Subcommittee on Highways and Transit, June 4, 2008.

[2] Federal Highway Administration, "2010 Status of the Nation's Highways, Bridges, and Transit: Conditions and Performance," Report to the United States Congress, 2010.

[3] L. C. Bank, Composites for Construction. Hoboken, New Jersey: John Wliey and Sons, Inc., 2006.

[4] American Concrete Institute (ACI) Committee 440, "Guide for the design and construction of externally bonded FRP systems for strengthening concrete structures," American Concrete Institute, ACI 440.2R-08, Farmington Hills, Michigan, 2008.

[5] International Federation for Structural Concrete (FIB), "Retrofitting of concrete structures by externally bonded FRPs, with emphasis on seismic applications," International Federation for Structural Concrete, FIB Bulletin 35, Lausanne, Switzerland, 2006.

[6] Darby, A., Ibell, T. and Clarke, J., TR55 "Design guidance for strengthening concrete structures using fibre composite materials." London: The Concrete Society, 2004.

[7] Brown, J. and Hamilton III, H.R.. "Investigation of Heating Methods and Detection Limits for Infrared Thermography Inspection of FRP Composites Used to Strengthen Reinforced Concrete", ACI Materials Journal, vol. 104 no. 5, pp 481-490, 2007.

[8] Hamilton III, H.R. and Brown, J., "CFRP Repair of Impact-Damaged Bridge Girders Volume II - Infrared Thermographic Inspection of FRP Composite Repairs," FDOT Report BC354 RPWO #55, Florida Department of Transportation, 2004.

[9] Kharbari, V., Kaiser, H., Navada, R., Gosh, K., and Lee, L., "Methods for Detecting Defects in Composite Rehabilitated Concrete Structures," Final Report, SPR 336, Oregon Department of Transportation, 2005.

[10] Brown, J. and Hamilton III, H.R., "Quantitative Infrared Thermography Inspection for FRP Applied to Concrete Using Single Pixel Analysis," Construction & Building Materials, ACI Committee 228 25th Anniversary Special Edition, Vol. 38, January 2013, pp. 1292-1301.

[11] Starnes, M. A., N. J. Carino, and Kausel, E.A., "Preliminary thermography studies for quality control of concrete structures strengthened with fiber-reinforced polymer composites." Journal of Materials in Civil Engineering 15(3): 266-273, (2003).

[12] A. N. Thikonov and V. Arsenin, Solution of Ill-Posed Problems, Wiley, New York, 1994.

[13] N. M. Ozisik and H. Orlande, Inverse Heat Transfer, Taylor & Francis, New York, 2001.

[14] K. Kurpisz and A. J. Nowak, Inverse Thermal Problems, Computational Mechanics Publications, Southampton and Boston, 1995.

[15] J. Beck, B. Blackwell, and C. R. St. Clair, Inverse Heat Conduction: Ill-Posed Problems, Wiley Interscience, New York, 1985.

[16] A. O. Alifanov, Inverse Heat Transfer Problems, Springer-Verlag, Berlin, 1994.

[17] J. C. Krapez and P. Cielo, Thermographic Nondestructive Evaluation: Data Inversion Procedures, Part I: 1-D Analysis, Res. Nondestr. Eval., vol. 3, pp. 81–100, 1991.

[18] J. C. Krapez, X. Maldague, and P. Cielo, Thermographic Nondestructive Evaluation: Data Inversion Procedures, Part II: 2-D Analysis and Experimental Results, Res. Nondestr. Eval., vol. 3, pp. 101–124, 1991.

[19] X. Maldague, Non Destructive Evaluation of Materials by Infrared Thermography, Springer-Verlag, New York, 1992.

[20] A. Mandelis, Nondestructive Evaluation, Prentice-Hall, Englewood Cliffs, NJ, 1994.

[21] L. Favro, J. D. Crowther, P. K. Kuo, and R. L. Thomas, Inversion of Pulsed Thermal Wave Images for Defect Sizing and Shape Recovery, SPIE, vol. 1682, Thermosense XIV, pp. 178–181, 1992.

[22] C. K. Hsieh and K. C. Su, A Methodology of Predicting Cavity Geometry Based on the Scanned Surface Temperature Data-Prescribed Surface Temperature at the Cavity Side, J. Heat Transfer, vol. 102, pp. 324–329, 1980.

[23] C. K. Hsieh and K. C. Su, A Methodology of Predicting Cavity Geometry Based on the Scanned Surface Temperature Data-Prescribed Heat Flux at the Cavity Side, J. Heat Transfer, vol. 103, pp. 42–46, 1981.

[24] C. K. Hsieh, X. A. Wang, and S. L. Yang, Infrared Scanning Thermography for a Quantitative Detection of Cavities in a Plane Slab and Rectangular Prism, J. Nondestr. Eval., vol. 3, pp. 99–109, 1982.

[25] A. G. Ramm, A Geometrical Inverse Problem, Inverse Problems, vol. 2, pp. L19–L21, 1986.

[26] C. H. Huang and B.-H. Chao, Inverse Geometry Problem in Identifying Irregular Boundary Configurations, Int. J. Heat Mass Transfer, vol. 40, no. 9, pp. 2045–2053, 1997.

[27] A. J. Kassab, C. K. Hsieh, and J. Pollard, Solution of the Inverse Geometric Problem for the Detection of Subsurface Cavities by the IR CAT Method, in L. C. Wrobel and D. B. Ingham (eds.), Boundary Integral Formulations in Inverse Analysis, chap. 2, Computational Mechanics, Boston, 1997.

[28] A. J. Kassab, F. A. Moslehy, and A. Daryapurkar, Nondestructive Detection of Cavities by an Inverse Elastostatics Method, Eng. Anal. Boundary Elements, vol. 13, no. 1, pp. 45–56, 1994.

[29] A. J. Kassab and J. Pollard, Automated Cubic Spline Anchored Grid Pattern Algorithm for the High Resolution Detection of Subsurface Cavities by the IR-CAT Method, Numer. Heat Transfer B, vol. 26, no. 1, pp. 63–78, 1994.

[30] A. J. Kassab and J. Pollard, Automated Algorithm for the Nondestructive Detection of Subsurface Cavities by the IR-CAT Method, J. Nondestr. Eval., vol. 12, no. 3, pp. 175–187, 1993.

[31] S. Das and A. Mitra, An Algorithm for the Solution of Inverse Lapalce Problems and Its Application in Flaw Identification in Materials, J. Comput. Phys., vol. 99, pp. 99–105, 1992.

[32] A. J. Kassab and C. K. Hsieh, Application of Infrared Scanners and Inverse Heat Conduction Problems to Infrared Computerized Axial Tomography, Rev. Sci. Instrum., vol. 58, no. 1, pp. 89–95, 1987.

[33] C. K. Hsieh and A. J. Kassab, A General Method for the Solution of Inverse Heat Conduction Problems with Partially Unknown System Geometries, Int. J. Heat Mass Transfer, vol. 29, no.1, pp. 47–58, 1985.

[34] T. Burczynski, W. Beluch, A. Dlugosz, W. Kus, M. Nowakowski, and P. Orantek, Evolutionary Computation in Optimization and Identification, Comput. Assist. Mech. Eng. Sci., vol. 9, no. 1, pp. 3–20, 2002.

[35] R. S. Nordlund and A. J. Kassab, A Conjugate BEM Optimization Algorithm for the Design of Turbine Vane Cooling Channels, BETECH96: Proc. 12th Int. Boundary Element Technical Conf., Hawaii, in E. K. Ertekin, M. Tanaka, R. P. Shaw, and C. A. Brebbia (eds.), pp. 237–246, Computational Mechanics, Southampton and Boston, 1996.

[36] T. Burczinski, J. H. Kane, and C. Balakrishna, Shape Design Sensitivity Analysis via Material Derivative-Adjoint Variable Technique for 3D and 2D Curved Boundary Elements, Int. J. Numer. Meth. Eng., vol. 38, pp. 2839–2866, 1995.

[37] C. H. Cheng and C. Y. Wu, An Approach Combining Body-Fitted Grid Generation and Conjugate Gradient Methods for Shape Design in Heat Conduction Problems, Numer. Heat Transfer B, vol. 37, pp. 69–83, 2000.

[38] A. Meric, Shape Optimization and Identification of Solid Geometries Considering Discontinuities, Trans. ASME, vol. 110, pp. 544–550, 1988.

[39] G. S. Dulikravich and J. M. Martin, Geometrical Inverse Analysis Problems in Three- Dimensional Non-linear Steady Heat Conduction, Eng. Anal. Boundary Elements, vol. 15, pp. 161–169, 1995.

[40] A. Meric, Differential and Integral Sensitivity Formulations and Shape Optimization by BEM, Eng. Anal. Boundary Elements, vol. 15, pp. 181–188, 1995.

[41] R. Bialecki, E. Divo, A. J. Kassab, and R. Ait Maalem Lahcen, Explicit Calculation of Smoothed Sensitivity Coefficients for Linear Problems, Int. J. Numer. Meth. Eng., vol. 57, no. 2, pp. 143–167, 2003.

[42] B. Blackwell, K. Dowding, and R. J. Chochran, Development and Implementation of Sensitivity Coefficient Equations for Heat Conduction Problems, Numer. Heat Transfer B, vol. 36, pp.15–32, 1999.

[43] C. A. Brebbia, J. C. F. Telles, and L. C. Wrobel, Boundary Element Techniques, Springer- Verlag, Berlin, 1984.

[44] L. C. Wrobel, The Boundary Element Method—Applications in Thermo-Fluids and Acoustics, vol. 1, Wiley, New York, 2002.

[45] E. Divo and A. J. Kassab, Boundary Element Method for Heat Conduction with Applications in NonHomogeneous Media, Wessex Institute of Technology Press, Southampton and Boston, 2003.

[46] Kassab, A.J., Wrobel, L.C., Bialecki, R.A., and Divo, E., "Boundary Elements in Heat Transfer," Chapter 4 in *Handbook of Numerical Heat Transfer*, Minkowycz, W. and Sparrow, E.M. (eds.), John Wiley and Sons, New York, NY, John Wiley and Sons, Vol. 1, 2nd Edition, pp. 125-166, 2005.

[47] Divo, E., Kassab, A.J., and Erhart, K., "Domain Decomposition Techniques for Boundary Elements. Applications to Fluid Flow," Chapter 5 in *Parallel BEM and Mesh Reduction Methods*, Popov, V., Power, H., and Skerget, L. (eds.), WIT Press, Billerica, MA, pp. 147-186, 2007.

[48] Kassab, A. and Divo, E., "Application of the Boundary Element Method to Non-Homogeneous Media: heat conduction and thermoelasticity," Chapter in *Recent Developments in the Boundary Element Method*, E.J. Sapountzakis (ed.), WIT Press, Southampton, UK, 2010.

[49] E. Divo, A. J. Kassab and F. Rodriguez, An Efficient Singular Superposition Technique for Cavity Detection and Shape Optimization, *Numerical Heat Transfer B*, vol. 46, pp. 1-30, 2004.

[50] D. Ojeda, E. Divo and A. J. Kassab, Cavity Detection in Biomechanics by an Inverse Evolutionary Point Load BEM Technique, *Inverse Problems in Science and Engineering*, vol. 16, no. 8, pp. 981-993, 2008.

[51] Belytscho, T., Lu, Y.Y., and Gu, L., "Element-free Galerkin methods, *Int. J. Num. Methods*," Vol. 37, 1994, pp. 229-256.

[52] Atluri, S.N. and Shen, S., *The Meshless Method*, Tech. Science Press, Forsyth, 2002.

[53] Atluri, S.N. and Zhu, T., "A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics," Computational Mechanics, Vol. 22, 1998, pp. 117-127.

[54] Onate, E., Idehlson, S., Zienkiewicz, O.C., Taylor, R.L., and Sacco, C.,"A stabilized finite point method for analysis of fluid mechanics problems," Computer Methods in Applied Mechanics and Engineering, Vol. 139, 1996, pp. 315-346.

[55] Liu, G.R., Mesh Free Methods: Moving Beyond the Finite Element Method., CRC Press, Boca Raton, 2002.

[56] Melenk, J.M. and Babuska, I., "The partition of unity finite element method: basic theory and application," Comp. Meth. Appl. Mechanics and Eng., Vol. 139, 1996, pp. 289- 316.

[57] Gu, Y.T., and Liu, G.R., "Meshless techniques for convection dominated problems," Computational Mechanics, Vol. 38, 2006, pp. 171–182.

[58] Gottlieb, D. and Orzag, S.A., Numerical Analysis of Spectral Methods: theory and applications, Society for Industrial and Applied Mathematics, Bristol, England, 1977.

[59] Maday, Y. and Quateroni, A., "Spectral and Pseudo-Spectral Approximations of the Navier-Stokes Equations," SIAM J. Numerical Analysis, 1982, Vol. 19, No. 4, pp. 761-780.

[60] Patera, A., "A Spectral Element Method of Fluid Dynamics: laminar flow in a channel expansion," J. of Computational Physics, 1984, Vol. 54, pp. 468-488.

[61] Macaraeg, M. and Street, C.L., "Improvement in Spectral Collocation Discretization Through a Multiple Domain Technique," Applied Numerical Mathematics, 1986, Vol. 2, pp. 95-108.

[62] Hwar, C.K., Hirsch, R., Taylor, T., and Rosenberg, A.P., "A Pseudo-Spectral Matrix Element Method for Solution of Three Dimensional Incompressible Flows and its Parallel Implementation," J. of Computational Physics, 1989, Vol. 83, pp. 260-291.

[63] Fasshauer, G, "RBF Collocation Methods as Pseudo-Spectral Methods," Boundary Elements XVII, Kassab, A., Brebbia, C.A. and Divo, E. (eds.), WIT Press, 2005, pp. 47-57.

[64] Powell, M.J.D., "The Theory of Radial Basis Function Approximation," in Advances in Numerical Analysis, Vol. II, Light, W., ed., Oxford Science Publications, Oxford, 1992. pp. 143-167.

[65] Buhmann, M.D., Cambridge Radial Basis Functions: Theory and Implementation University Press, Cambridge, 2003.

[66] Dyn, N., Levin, D., and Rippa, S., "Numerical Procedures for Surface Fitting of Scattered Data by Radial Basis Functions," SIAM J. of Sci. Stat. Computing, 1986, Vol. 7, No. 2, pp. 639-659.

[67] Hardy, R.L., Multiquadric Equations of Topography and Other Irregular Surfaces, Journal of Geophysical Research, Vol. 176, pp. 1905-1915.

[68] Kansa, E.J., "Multiquadrics- a scattered data approximation scheme with applications to computational fluid dynamics I-surface approximations and partial derivative estimates," Comp. Math. Appl., Vol. 19, 1990, pp. 127-145.

[69] Kansa, E.J., "Multiquadrics- a scattered data approximation scheme with applications to computational fluid dynamics II-solutions to parabolic, hyperbolic

and elliptic partial differential equations," Comp. Math. Appl., Vol. 19, 1990, pp. 147-161.

[70] Kansa, E.J. and Hon, Y.C., "Circumventing the Ill-Conditioning Problem with Multiquadric Radial Basis Functions: Applications to Elliptic Partial Differential Equations," Comp. Math. Appl., 2000, Vol. 39, pp. 123-137.

[71] Franke, R., "Scattered data interpolation: Test of some methods," Math. Comput., Vol. 38, 1982, pp. 181-200.

[72] Mai-Duy, N. and Tran-Cong, T., "Mesh-Free Radial Basis Function Network Methods with Domain Decomposition for Approximation of Functions and Numerical Solution of Poisson's Equation," Engineering Analysis with Boundary Elements, 2002, Vol. 26, pp. 133-156.

[73] Cheng, A.H.-D., Golberg, M.A., Kansa, E.J., Zammito, G., "Exponential Convergence and H-c Multiquadric Collocation Method for Partial Differential Equations," Numerical Methods in P.D.E., Vol. 19, No. 5, 2003, pp. 571-594.

[74] Sarler, B. and Vertnik, R., "Local Explicit Radial Basis Function Collocation Method for Diffusion Problems," Computers and Mathematics with Applications,Vol. 51, No. 8, 2006, pp. 1269-1282.

[75] Vertnik, R. and Sarler, B.,"Meshless local radial basis function collocation method for convective-diffusive solid-liquid phase change problems," International Journal of Numerical Methods for Heat and Fluid Flow, Vol. 16, No. 5, 2006, pp. 617-640.

[76] Vertnik, R., Zaloznik, M. and Sarler, B., "Solution of transient direct-chill aluminium billet casting problem with simultaneous material and in9terphase moving boundaries by a meshless method," Engineering Analysis with Boundary Elements, Vol. 30, No. 10, 2006, pp. 847-855.

[77] Divo, E. and Kassab, A.J., "A Meshless Method for Conjugate Heat Transfer Problems," Engineering Analysis, Vol. 29, No. 2, 2005, pp. 136-149.

[78] Divo, E. and Kassab, A.J., "Iterative Domain Decomposition Meshless Method Modeling of Incompressible Flows and Conjugate Heat Transfer," Engineering Analysis, Vol. 30, No. 6, 2006, pp. 465-478.

[79] Divo, E. and Kassab, A.J., "An Efficient Localized RBF Meshless Method for Fluid Flow and Conjugate Heat Transfer," ASME Journal of Heat Transfer, 2007, Vol. 129, pp. 124-136.

[80] Divo, E. and Kassab, A.J., "Localized Meshless Modeling of Natural Convective Viscous Flows," Numerical Heat Transfer, Part A: Fundamentals, Volume 53, Issue 6, June 2008, pages 487-509.

[81] El Zahab, Z., Divo, E., and Kassab, A.J., "A Localized Collocation Meshless Method (LCMM) for Incompressible Flows CFD Modeling with Applications to Transient Hemodynamics," Engineering Analysis with Boundary Elements, Vol. 33, No. 8-9, pp. 1045-1061, 2009.

[82] Kelly, J., Divo, E., and Kassab, A.J., "Numerical Solution of the Two-Phase Incompressible Navier-Stokes Equations Using a GPU-Accelerated Meshless Method," Engineering Analysis with Boundary Elements, 2012 (in review).

[83] Gerace, S., Erhart, K., Kassab, A.J., and Divo, E., "A Model-Integrated Localized Collocation Meshless Method for Large Scale Three-Dimensional Heat Transfer Problems," Engineering Analysis with Boundary Elements, 2012, (in press).

[84] Gerace, S., Erhart, K., Divo, E., and Kassab, A.J., "Adaptively Refined Hybrid FDM-RBF Meshless Scheme with Applications to Laminar and Turbulent Viscous Fluid Flows," Computer Modeling in Engineering and Science, Vol. 81, No.1, pp.35-67, 2011.

[85] Erhart, K. Kassab, A.J. and Divo, E., "An Inverse Localized Meshless Technique for the Determination of Non-Linear Heat Generation Rates in Living Tissues," International Journal of Numerical Methods for Heat and Fluid Flow, Vol. 18, No. 3/4, 2008, pp. 401-414.

[86] El Zahab, Z., Divo, E., and Kassab, A.J., "A Meshless CFD Approach for Evolutionary Shape Optimization of Bypass Grafts Anastomoses," Journal Inverse Problems in Science and Engineering, Vol. 17, No. 3, 411-435, 2009.

[87] A. Karageorghis, D. Lesnic and L Marin, A Survey of the MFS to Inverse Problems, Inverse Problems in Science and Engineering, vol. 19, no. 3, pp. 309-336, 2011.

[88] A. Karageorghis, D. Lesnic and L Marin, A Moving Pseudo-boundary Method of Fundamental Solutions for Void Detection, Wiley Online Library, 10.1002/num.21739, 2012.

[89] A. Karageorghis, D. Lesnic and L Marin, The Method of Fundamental Solutions for the Detection of Rigid Inclusions and Cavities in Plane Linear Elastic Bodies, Computers and Structures, vol. 106-107, pp. 176-188, 2012.

[90] Hirt, C.W. and Nichols, B.D., Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries, *Journal of Computational Fluids*, 1981, Vol. 39, pp. 201-225.

[91] Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, MA, 1989.

[92] Pepper, D.W, Kassab, A.J., and Divo, E., (2014), *Introduction to Finite Element, Boundary Element, and Meshless Methods: with applications to heat transfer and fluid flow*. ASME Press.

APPENDIX

Main Code Listing (FORTRAN)

```fortran
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC   PROGRAM NDE_iVoF_MMGA
CCC
CCC   Version 1.0: Parallel (mpich)
CCC
CCC   Non-Destructive Evaluation
CCC
CCC   Inverse Volume-of-Fluid
CCC
CCC   Meshless Method
CCC
CCC   Genetic Algorithm
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC   Embry-Riddle Aeronautical University
CCC
CCC   MDBL: Multi-Disciplinary Bioengineering Lab
CCC
CCC   Eduardo Divo
CCC
CCC   Hussein Saad
CCC
CCC   September 23, 2014
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC                               MAIN PROGRAM
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE 'PGA_Parameters.h'
      INCLUDE 'mpif.h'
C
      COMMON/PGAINFO/NGEN,NPOP,NPAR,NBIT(NPARMAX)
      COMMON/PGAMUTA/PJMU,PCMU
      COMMON/PARINFO/PARMIN(NPARMAX),PARMAX(NPARMAX),PARRES(NPARMAX)
      COMMON/PARAMET/PARAM(NPARMAX,NPOPMAX)
```

```fortran
      COMMON/NEWGENE/CHILD(NPARMAX,NPOPMAX)
      COMMON/FITNESS/FITNESS(NPOPMAX),PSEL(NPOPMAX),JBEST
C
      REAL*8 STARTTIME,ENDTIME
      REAL*8 TIMEMY,TIMETO
      REAL*8 TIM(MAXPROC),FRA(MAXPROC)
      INTEGER ID,IDMASTER,NPRO,IERR
      INTEGER ILOAD(MAXPROC,NPOPMAX)
C
      INTEGER IGEN,IG
      INTEGER JP1,JP2,IC
      INTEGER IK
      INTEGER UNDAT,UNOUT
C
C**********************************************************************
C
C
C     START OF PARALLEL BENCHMARKING
C
      CALL MPI_INIT(IERR)
C
      CALL MPI_COMM_RANK(MPI_COMM_WORLD,ID,IERR)
C
      CALL MPI_COMM_SIZE(MPI_COMM_WORLD,NPRO,IERR)
C
C     SPECIFY MASTER PROCESS
C
      IDMASTER=0
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'PGA: MULTIVARIABLE PARALLEL GENETIC ALGORITHM'
       WRITE(*,*) '     OPTIMIZATION TOOL'
       WRITE(*,*)
       WRITE(*,*) 'VERSION 1.1'
       WRITE(*,*)
       WRITE(*,*) 'EMBRY-RIDDLE AERONAUTICAL UNIVERSITY'
       WRITE(*,*)
       WRITE(*,*)
       WRITE(*,*) 'NUMBER OF PROCESSORS....................:',NPRO
       WRITE(*,*)
      END IF
C
C**********************************************************************
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*)
'*************************************************'
       WRITE(*,*)
       WRITE(*,*) 'CLUSTER BENCHMARKING PROCESS STARTED..........:
[OK]'
       WRITE(*,*)
       WRITE(*,*)
      END IF
C
      CALL BENCHMARK(TIM,FRA,ID,IDMASTER,NPRO)
```

```fortran
C
      IF (ID.EQ.IDMASTER) THEN
       DO N=1,NPRO
        WRITE (*,'("   PROCESS, BENCHMARK TIME & FRACTION...: ",I3,
     &           2X,F6.2,2X,F6.4)') N,TIM(N),FRA(N)
       END DO
       WRITE(*,*)
       WRITE(*,*) 'CLUSTER BENCHMARKING PROCESS ENDED............:
     [OK]'
       WRITE(*,*)
       WRITE(*,*)
'**************************************************'
      END IF
C
C***********************************************************************
C
C
C     INPUT DATA FILE AND INITIAL PARAMETERS
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'INPUT GENETIC ALGORITHM PARAMETERS STARTED....:
     [OK]'
       WRITE(*,*)
      END IF
C
      CALL INPUT(ID,IDMASTER)
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,'("   NUMBER OF GENERATIONS.............: ",I5)') NGEN
       WRITE(*,'("   POPULATION SIZE..................: ",I5)') NPOP
       WRITE(*,'("   PROBABILITY OF JUMP MUTATION......:  ",F4.2)')
     PJMU
       WRITE(*,'("   PROBABILITY OF CREEP MUTATION.....:  ",F4.2)')
     PCMU
       WRITE(*,*)
       WRITE(*,*) 'INPUT GENETIC ALGORITHM PARAMETERS ENDED......:
     [OK]'
       WRITE(*,*)
       WRITE(*,*)
'**************************************************'
      END IF
C
C***********************************************************************
C
C     LOAD BALANCING ALGORITHM
C
C
C     INITIALIZE RANDOM GENERATOR
C
      CALL RANDOM(-1000-100*ID,R)
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'LOAD BALANCING PROCESS STARTED................:
     [OK]'
C
```

63

```
         CALL LOAD(IDMASTER,ILOAD,FRA,NPRO)
C
         WRITE(*,*)
         WRITE(*,*) '  FINAL POPULATION BALANCE OVER PROCESSORS....:'
         WRITE(*,*)
         DO N=1,NPRO
          WRITE(*,'("  PROCESS & FRACTION: ",I3,1X,F6.4,2X,100(I1))')
     @                         N,FRA(N),(ILOAD(N,K),K=1,NPOP)
         END DO
C
         WRITE(*,*)
         WRITE(*,*) 'LOAD BALANCING PROCESS ENDED..................:
[OK]'
         WRITE(*,*)
         WRITE(*,*)
'**************************************************'
      END IF
C
C     BROADCASTING CLUSTER WORK LOAD
C
      CALL MPI_BCAST(ILOAD,MAXPROC*NPOPMAX,MPI_INTEGER,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
C
C**********************************************************************
C
C     INPUT DATA FOR FUNCTION EVALUATION
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'FUNCTION EVALUATION DATA INPUT STARTED........:
[OK]'
      END IF
C
      CALL FUNCTIONSETUP(ID,IDMASTER,NPRO)
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,'("   NUMBER OF PARAMETERS..............: ",I5)') NPAR
       WRITE(*,*)
       WRITE(*,*) 'FUNCTION EVALUATION DATA INPUT ENDED..........:
[OK]'
       WRITE(*,*)
       WRITE(*,*)
'**************************************************'
      END IF
C
C**********************************************************************
C
C     GENERATE INITIAL POPULATION
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'INITIAL POPULATION GENERATION STARTED.........:
[OK]'
      END IF
C
C     DATA FILES UNIT NUMBERS
```

```fortran
C
      UNDAT=8
      UNOUT=9
C
      CALL INITIAL(ID,IDMASTER,IGEN,UNDAT,UNOUT)
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'INITIAL POPULATION GENERATION ENDED...........:
[OK]'
       WRITE(*,*)
       WRITE(*,*)
'************************************************'
      END IF
CCC
C     PARAM(1,1)=0.1
C     PARAM(2,1)=0.2
C     PARAM(3,1)=0.025
C     PARAM(4,1)=0.05
CCC
C
C*********************************************************************
C
C     MAIN OPTIMIZATION LOOP
C
      IF (ID.EQ.IDMASTER) THEN
       STARTTIME=MPI_WTIME()
       WRITE(*,*)
       WRITE(*,*) 'MAIN GENETIC OPTIMIZATION LOOP STARTED........:
[OK]'
       FITNESSMAX=0.
       IK=0
      END IF
C
      CALL FUNCTIONEVALUATE(ID,IDMASTER,NPRO,ILOAD)
C
      DO IG=1,NGEN
C
       DO IC=1,NPOP-1
C
        IF (ID.EQ.IDMASTER) CALL SELECTION(JP1,JP2)
C
        IF (ID.EQ.IDMASTER) CALL REPRODUCT(JP1,JP2,IC)
C
       END DO
C
       IF (ID.EQ.IDMASTER) THEN
        IK=IK+1
        IF (FITNESS(JBEST).GT.FITNESSMAX) THEN
         WRITE(*,010) IGEN+IG,FITNESS(JBEST)
         FITNESSMAX=FITNESS(JBEST)
         IK=0
        END IF
        IF (IK.GE.50) THEN
         CALL KILLGEN()
         WRITE(*,010) IGEN+IG,FITNESS(JBEST)
         IK=0
```

```fortran
        END IF
        END IF
C
       CALL NEWGEN(ID,IDMASTER,NPRO,IGEN,IG,ILOAD,UNDAT,UNOUT)
C
       CALL FUNCTIONEVALUATE(ID,IDMASTER,NPRO,ILOAD)
C
      END DO
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,010) IGEN+IG-1,FITNESS(JBEST)
  010  FORMAT ('  GENERATION: ',I6,'    BEST FITNESS: ',E10.4)
       WRITE(*,*)
       WRITE(*,*) 'MAIN GENETIC OPTIMIZATION LOOP ENDED..........:
[OK]'
       CLOSE (UNOUT)
       ENDTIME=MPI_WTIME()
      END IF
C
C***********************************************************************
C
C     OUTPUT FUNCTION DATA
C
      IF (ID.EQ.IDMASTER) THEN
       WRITE(*,*)
       WRITE(*,*) 'FUNCTION DATA OUTPUT STARTED..................:
[OK]'
C
       CALL FUNCTIONOUTPUT()
C
       WRITE(*,*)
       WRITE(*,*) 'FUNCTION DATA OUTPUT ENDED....................:
[OK]'
       WRITE(*,*)
       WRITE(*,*)
'**************************************************'
      END IF
C
C***********************************************************************
C
C     OUTPUT TIMES
C
      IF (ID.EQ.IDMASTER) THEN
       TOTALTIME=ENDTIME-STARTTIME
       OPEN (21,FILE='Model/PGA.time')
       WRITE (21,*) 'NUMBER OF GENERATIONS PERFORMED..: ',NGEN
       WRITE (21,*) 'TOTAL TIME ELAPSED...............: ',TOTALTIME
       TOTALTIME=TOTALTIME/DBLE(NGEN)
       WRITE (21,*) 'TIME ELAPSED PER GENERATION......: ',TOTALTIME
       CLOSE (21)
      END IF
C
C***********************************************************************
C
      CALL MPI_FINALIZE(IERR)
C
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC
CCC                          SUBROUTINE INPUT
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
      SUBROUTINE INPUT(ID,IDMASTER)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE 'PGA_Parameters.h'
      INCLUDE 'mpif.h'
C
      COMMON/PGAINFO/NGEN,NPOP,NPAR,NBIT(NPARMAX)
      COMMON/PGAMUTA/PJMU,PCMU
C
      INTEGER ID,IDMASTER,IERR
C
C**********************************************************************
****
C
      IF (ID.EQ.IDMASTER) THEN
C
        OPEN (14,FILE='model/PGA.inp',STATUS='OLD')
C
C       MAXIMUM NUMBER OF GENERATIONS
C
        READ (14,*) NGEN
C
C       POPULATION SIZE
C
        READ (14,*) NPOP
C
C       PROBABILITY OF JUMP MUTATION
C
        READ (14,*) PJMU
C
C       PROBABILITY OF CREEP MUTATION
C
        READ (14,*) PCMU
C
        CLOSE (14)
C
      END IF
C
C     BROADCAST GENETIC ALGORITHM DATA OVER CLUSTER
C
      CALL MPI_BCAST(NGEN,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL MPI_BCAST(NPOP,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL MPI_BCAST(PJMU,1,MPI_DOUBLE_PRECISION,
     &               IDMASTER,MPI_COMM_WORLD,IERR)
```

67

```
C
      CALL MPI_BCAST(PCMU,1,MPI_DOUBLE_PRECISION,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
C
C********************************************************************
****
C
      RETURN
C
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC
CCC                    SUBROUTINE INITIAL
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
      SUBROUTINE INITIAL(ID,IDMASTER,IGEN,UNDAT,UNOUT)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE 'PGA_Parameters.h'
      INCLUDE 'mpif.h'
C
      COMMON/PGAINFO/NGEN,NPOP,NPAR,NBIT(NPARMAX)
      COMMON/PGAMUTA/PJMU,PCMU
      COMMON/PARINFO/PARMIN(NPARMAX),PARMAX(NPARMAX),PARRES(NPARMAX)
      COMMON/PARAMET/PARAM(NPARMAX,NPOPMAX)
C
      INTEGER ID,IDMASTER,IERR
C
      REAL*8 PAR(NPARMAX)
      INTEGER IPAR(NPARMAX*NBITMAX)
      INTEGER IGEN
      INTEGER IO1,IO2,IO3
      INTEGER UNDAT,UNOUT
C
      REAL*8 B2D
C
C*********************************************************************
****
C
      IF (ID.EQ.IDMASTER) THEN
       OPEN (UNOUT,FILE='Model/PGA.out')
       OPEN (UNDAT,FILE='Model/PGA.dat',STATUS='OLD',IOSTAT=IO1)
C
       IF (IO1.EQ.0) THEN
C
        READ (UNDAT,*) IGEN
        DO J=1,NPOP
         READ (UNDAT,*,IOSTAT=IO2) JJ,(PARAM(I,J),I=1,NPAR)
        END DO
        CLOSE (UNDAT)
C
        DO J=1,NPOP
         DO I=1,NPAR
          IF (PARAM(I,J).GT.PARMAX(I)) PARAM(I,J)=PARMAX(I)
          IF (PARAM(I,J).LT.PARMIN(I)) PARAM(I,J)=PARMIN(I)
         END DO
        END DO
C
        DO I=1,IGEN
         READ (UNOUT,*,IOSTAT=IO3) II,FIT,(PAR(J),J=1,NPAR)
        END DO
```

```fortran
C
      ELSE
C
         IGEN=0
        DO J=1,NPOP
         III=0
         DO I=1,NPAR
          DO II=1,NBIT(I)
           III=III+1
           CALL RANDOM(1,R)
           IF (R.LT.0.5) THEN
            IPAR(III)=0
           ELSE
            IPAR(III)=1
           END IF
          END DO
         END DO
         DO I=1,NPAR
          PARAM(I,J)=B2D(IPAR,I)
         END DO
        END DO
C
       END IF
C
      END IF
C
C     BROADCAST PARAMETERS OVER CLUSTER
C
      DO J=1,NPOP
       DO I=1,NPAR
        PAR(I)=PARAM(I,J)
       END DO
       CALL MPI_BCAST(PAR,NPAR,MPI_DOUBLE_PRECISION,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
       DO I=1,NPAR
        PARAM(I,J)=PAR(I)
       END DO
      END DO
C
C*********************************************************************
****
C
      RETURN
C
      END
```

70

```fortran
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC
CCC                        SUBROUTINE FUNCTIONINPUT
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
      SUBROUTINE FUNCTIONSETUP(ID,IDMASTER,NPRO)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE 'PGA_Parameters.h'
      INCLUDE 'mpif.h'
C
C     USE THE FOLLOWING SPACE TO INCLUDE THE FILES FOR THE FUNCTION
INPUT
C
      INCLUDE '../Include/Parameters.for'
      INCLUDE '../Include/Information.for'
      INCLUDE '../Include/Geometry.for'
      INCLUDE '../Include/Materials.for'
      INCLUDE '../Include/Field-S.for'
      INCLUDE '../Include/Field-M.for'
      INCLUDE '../Include/Field-E.for'
      INCLUDE '../Include/Field-P.for'
      INCLUDE '../Include/Topology.for'
      INCLUDE '../Include/Interpolation-C.for'
      INCLUDE '../Include/Interpolation-U.for'
      INCLUDE '../Include/Triangulation.for'
C
C     USE THE FOLLOWING SPACE TO INCLUDE THE COMMON BLOCKS FOR FUNCTION
VARIABLES
C
      PARAMETER (NTMAX=20,NMMAX=200)
      COMMON/MEASURE/NMN,NMT,MT(NTMAX),MR(NMMAX),MN(NMMAX)
      COMMON/MEASURT/TM(NMMAX,NTMAX)
      COMMON/HOLELOC/XCH,YCH,RXH,RYH
C
C     STANDARD COMMON BLOCKS AND VARIABLES
C
      COMMON/PGAINFO/NGEN,NPOP,NPAR,NBIT(NPARMAX)
      COMMON/PGAMUTA/PJMU,PCMU
      COMMON/PARINFO/PARMIN(NPARMAX),PARMAX(NPARMAX),PARRES(NPARMAX)
C
      INTEGER ID,IDMASTER,NPRO,IERR
      CHARACTER*120 TITLE
      INTEGER IOS
C
C*********************************************************************
****
C
C     INPUT AND PROCESS THE NECESSARY DATA FOR FUNCTION EVALUATION
C     INCLUDE THE INPUT DATA AND PROBLEM SETUP ROUTINES
```

71

```fortran
C       USE FUNCTION INPUT DATA TO STABLISH NUMBER OF PARAMETERS TO
OPTIMIZE
C       AND THE MINIMUM AND MAXIMUM ALLOWABLE VALUES
C
        IF (ID.EQ.IDMASTER) THEN
C
         NPAR=4
C
C        MEASUREMENT DATA
C
         OPEN
(20,FILE='Model/ALMA_a_measure.txt',STATUS='OLD',IOSTAT=IOS)
C
         IF (IOS.EQ.0) THEN
          READ (20,*) TITLE
C
C         NUMBER OF MEASURMENT NODES AND TIMES
C
          READ (20,*) NMN,NMT
C
C         XMIN,YMIN,XMAX,YMAX
C
          READ (20,*) XMIN,YMIN,XMAX,YMAX
C
C         X-HOLE LOCATION
C
          PARMIN(1)=XMIN+(XMAX-XMIN)/10.D+000
          PARMAX(1)=XMAX-(XMAX-XMIN)/10.D+000
          NBIT(1)=8
C
C         Y-HOLE LOCATION
C
          PARMIN(2)=YMIN+(YMAX-YMIN)/10.D+000
          PARMAX(2)=YMAX-(YMAX-YMIN)/10.D+000
          NBIT(2)=8
C
C         HOLE X-RADIUS
C
          PARMIN(3)=(XMAX-XMIN)/1.D+002
          PARMAX(3)=(XMAX-XMIN)/2.D+001
          NBIT(3)=8
C
C         HOLE Y-RADIUS
C
          PARMIN(4)=(YMAX-YMIN)/1.D+002
          PARMAX(4)=(YMAX-YMIN)/2.D+001
          NBIT(4)=8
C
C         MEASUREMENT REGION AND BOUNDARY NODE
C
          DO I=1,NMN
           READ (20,*) MR(I),MN(I)
          END DO
C
          DO NT=1,NMT
C
C          TIME-STEP
```

```fortran
C
          READ (20,*) MT(NT)
C
C        MEASURED TEMPERATURE
C
          DO I=1,NMN
           READ (20,*) TM(I,NT)
          END DO
C
         END DO
C
        ELSE
C
         WRITE (*,*) '...MISSING INFORMATION FILE.'
         STOP
C
        END IF
C
        CLOSE (20)
C
C       INPUT PARAMETERS, GEOMETRY, CONNECTIVITY AND INTERPOLATION
C
        WRITE (*,*) 'READING PROBLEM DATA............................'
        CALL MESHLESS_INPUT
        WRITE (*,*) '........................................DONE'
C
C       VALIDATING THERMOPHYSICAL QUANTITIES
C
        WRITE (*,*) 'VALIDATING THERMOPHYISICAL QUANTITIES...........'
        CALL VALIDATE
        WRITE (*,*) '........................................DONE'
C
C       INPUT LEVEL-SET
C
        IF (MSL.EQ.1) THEN
         WRITE (*,*) 'READING LEVEL-SET DATA.........................'
         CALL INPUTLEVELSET
          WRITE (*,*) '........................................DONE'
        END IF
C
C       INPUT MOMENTUM
C
        IF (MSM.EQ.1) THEN
         WRITE (*,*) 'READING MOMENTUM DATA..........................'
         CALL INPUTMOMENTUM
          WRITE (*,*) '........................................DONE'
        END IF
C
C       INPUT ENERGY
C
        IF (MSE.EQ.1) THEN
         WRITE (*,*) 'READING ENERGY DATA............................'
         CALL INPUTENERGY
          WRITE (*,*) '........................................DONE'
        END IF
C
C       INPUT PORE PRESSURE
```

```fortran
C
        IF (MSP.EQ.1) THEN
         WRITE (*,*) 'READING PORE PRESSURE DATA.......................'
         CALL INPUTPOREPRESSURE
         WRITE (*,*) '............................................DONE'
        END IF
C
C       INPUT STRUCTURAL
C
        IF (MSS.EQ.1) THEN
         WRITE (*,*) 'READING STRUCTURAL DATA.........................'
         CALL INPUTSTRUCTURAL
         WRITE (*,*) '............................................DONE'
        END IF
C
      END IF
C
C     BROADCAST MESHLESS DATA OVER CLUSTER
C
C      CALL MPI_BCAST(XXX,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C      ...
C      ...
C      ...
C      ...
C
C
C***********************************************************************
****
C
C     COMPUTE PARAMETER RESOLUTION AND BROADCAST OVER CLUSTER
C
      IF (ID.EQ.IDMASTER) THEN
       DO I=1,NPAR
         PARRES(I)=(PARMAX(I)-PARMIN(I))/(2.**NBIT(I)-1.)
       END DO
      END IF
C
      CALL MPI_BCAST(NPAR,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL
MPI_BCAST(NBIT,NPAR,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL MPI_BCAST(PARMIN,NPAR,MPI_DOUBLE_PRECISION,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL MPI_BCAST(PARMAX,NPAR,MPI_DOUBLE_PRECISION,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
C
      CALL MPI_BCAST(PARRES,NPAR,MPI_DOUBLE_PRECISION,
     &                IDMASTER,MPI_COMM_WORLD,IERR)
C
C***********************************************************************
****
C
      RETURN
C
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
CCC
CCC
CCC                      SUBROUTINE FUNCTIONEVALUATE
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCC
      SUBROUTINE FUNCTIONEVALUATE(ID,IDMASTER,NPRO,ILOAD)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE 'PGA_Parameters.h'
      INCLUDE 'mpif.h'
C
C     USE THE FOLLOWING SPACE TO INCLUDE THE FILES FOR THE FUNCTION
EVALUATION
C
      INCLUDE '../Include/Parameters.for'
      INCLUDE '../Include/Information.for'
      INCLUDE '../Include/Geometry.for'
      INCLUDE '../Include/Materials.for'
      INCLUDE '../Include/Field-S.for'
      INCLUDE '../Include/Field-M.for'
      INCLUDE '../Include/Field-E.for'
      INCLUDE '../Include/Field-P.for'
      INCLUDE '../Include/Topology.for'
      INCLUDE '../Include/Interpolation-C.for'
      INCLUDE '../Include/Interpolation-U.for'
      INCLUDE '../Include/Triangulation.for'
C
C     USE THE FOLLOWING SPACE TO INCLUDE THE COMMON BLOCKS FOR FUNCTION
VARIABLES
C
      PARAMETER (NTMAX=20,NMMAX=200)
      COMMON/MEASURE/NMN,NMT,MT(NTMAX),MR(NMMAX),MN(NMMAX)
      COMMON/MEASURT/TM(NMMAX,NTMAX)
      COMMON/HOLELOC/XCH,YCH,RXH,RYH
C
C     STANDARD COMMON BLOCKS AND VARIABLES
C
      COMMON/PGAINFO/NGEN,NPOP,NPAR,NBIT(NPARMAX)
      COMMON/PGAMUTA/PJMU,PCMU
      COMMON/PARINFO/PARMIN(NPARMAX),PARMAX(NPARMAX),PARRES(NPARMAX)
      COMMON/PARAMET/PARAM(NPARMAX,NPOPMAX)
      COMMON/FITNESS/FITNESS(NPOPMAX),PSEL(NPOPMAX),JBEST
C
      INTEGER ID,IDMASTER,NPRO,IERR
      INTEGER ILOAD(MAXPROC,NPOPMAX)
C
      REAL*8 FIT
      INTEGER STATUS(MPI_STATUS_SIZE)
      INTEGER ISENDER,IND
C
      INTEGER IDUMMY
```

75

```
C
C***********************************************************************
****
C
      DO J=1,NPOP
       IF (ILOAD(ID+1,J).EQ.1) THEN
C
         FITNESS(J)=0.D+000
C
C        TRANSLATE PARAMETERS INTO SOURCE CHARACTERISTICS
C
         XCH=PARAM(1,J)
         YCH=PARAM(2,J)
         RXH=PARAM(3,J)
         RYH=PARAM(4,J)
C
C        INITIALIZE TIME STEPPING AND MEASUREMENT TIME
C
         ITP=0
         CTIME=0.D+000
         NT=1
C
C        INITIALIZE MODEL SETUP
C
         CALL INITIALIZE
C
C        LOOP OVER MAXIMUM ITERATIONS
C
         DO WHILE (ITP.LT.MAXITER)
C
C         INCREASE ITERATION AND ELAPSED TIME
C
          ITP=ITP+1
          CTIME=CTIME+DT
C
C         SOLVE LEVEL-SET FIELD
C
          IF (MSL.EQ.1) CALL SOLVELEVELSET
C
C         SOLVE MOMENTUM FIELD
C
          IF (MSM.EQ.1) CALL SOLVEMOMENTUM
C
C         SOLVE ENERGY FIELD
C
          IF (MSE.EQ.1) CALL SOLVEENERGY
C
C         SOLVE PORE PRESSURE FIELD
C
          IF (MSP.EQ.1) CALL SOLVEPOREPRESSURE
C
C         SOLVE STRUCTURAL FIELD
C
          IF (MSS.EQ.1) CALL SOLVESTRUCTURAL
C
C         EVALUATE THE FITNESS OF THE Jth INDIVIDUAL GIVEN ITS
PARAMETERS
```

```fortran
C
            IF (ITP.EQ.MT(NT)) THEN
C
C            CALCULATE RMS
C
              DO I=1,NMN
               FITNESS(J)=FITNESS(J)+(TM(I,NT)-TCC(MN(I),MR(I)))**2.D+000
CCC
CCC             WRITE (*,*) ITP,TM(I,NT),TCC(MN(I),MR(I))
CCC
              END DO
C
              NT=NT+1
C
            END IF
C
          END DO
C
C        INVERT RMS TO CALCULATE FITNESS
C
          IF (FITNESS(J).GT.EPS) THEN
           FITNESS(J)=DSQRT(DBLE(NMN*NMT)/FITNESS(J))
          ELSE
           FITNESS(J)=1.D+020
          END IF
CCCCC
          WRITE (*,*) "INDIVIDUAL: ",J
          WRITE (*,*) XCH,YCH
          WRITE (*,*) RXH,RYH
          WRITE (*,*) "FITNESS: ",FITNESS(J)
CCCCC
         END IF
C
       END DO
C
C********************************************************************
****
C
C     SEND FITNESS TO MASTER COMPUTER
C
      CALL MPI_BCAST(IDUMMY,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
      DO J=1,NPOP
       IF (ID.NE.IDMASTER) THEN
        IF (ILOAD(ID+1,J).EQ.1) THEN
         FIT=FITNESS(J)
         CALL MPI_SEND(FIT,1,MPI_DOUBLE_PRECISION,IDMASTER,J,
     @                 MPI_COMM_WORLD,IERR)
        END IF
       ELSE
        IF (ILOAD(ID+1,J).EQ.0) THEN
         CALL MPI_RECV(FIT,1,MPI_DOUBLE_PRECISION,MPI_ANY_SOURCE,
     @                 MPI_ANY_TAG,MPI_COMM_WORLD,STATUS,IERR)
         ISENDER=STATUS(MPI_SOURCE)
         IND=STATUS(MPI_TAG)
         FITNESS(IND)=FIT
        END IF
```

```fortran
       END IF
       END DO
C
       CALL MPI_BCAST(IDUMMY,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
C      ACUMMULATE PROBABILITY OF SELECTION
C
       IF (ID.EQ.IDMASTER) THEN
        FITMIN=FITNESS(1)
        DO J=2,NPOP
         IF (FITNESS(J).LT.FITMIN) FITMIN=FITNESS(J)
        END DO
        IF (FITMIN.GE.0.) FITMIN=0.
        FIT=0.
        DO J=1,NPOP
         FIT=FIT+(FITNESS(J)-FITMIN)
        END DO
        DO J=1,NPOP
         PSEL(J)=(FITNESS(J)-FITMIN)/FIT
        END DO
        DO J=2,NPOP
         PSEL(J)=PSEL(J)+PSEL(J-1)
        END DO
       END IF
C
C      SELECT BEST FITNESS
C
       IF (ID.EQ.IDMASTER) THEN
        FIT=FITNESS(1)
        JBEST=1
        DO J=2,NPOP
         IF (FITNESS(J).GT.FIT) THEN
          FIT=FITNESS(J)
          JBEST=J
         END IF
        END DO
       END IF
C
C      BROADCAST BEST INDIVIDUAL
C
       CALL MPI_BCAST(JBEST,1,MPI_INTEGER,IDMASTER,MPI_COMM_WORLD,IERR)
C
C***********************************************************************
****
C
C      ASSIGN THE BEST PARAMETERS TO FUNCTION EVALUATION VARIABLE
C
       XCH=PARAM(1,JBEST)
       YCH=PARAM(2,JBEST)
       RXH=PARAM(3,JBEST)
       RYH=PARAM(4,JBEST)
C
C***********************************************************************
C
       RETURN
       END
```

```fortran
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
CCC
CCC
CCC                       SUBROUTINE INPUT
CCC
CCC
CCC
CCC    PURPOSE:
CCC
CCC
CCC
CCC    READS INPUT, DATA, VECTOR, AND TRIANGULATION
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
      SUBROUTINE MESHLESS_INPUT
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE '../../Include/Parameters.for'
      INCLUDE '../../Include/Information.for'
      INCLUDE '../../Include/Geometry.for'
      INCLUDE '../../Include/Materials.for'
      INCLUDE '../../Include/Topology.for'
      INCLUDE '../../Include/Interpolation-C.for'
      INCLUDE '../../Include/Interpolation-U.for'
      INCLUDE '../../Include/Triangulation.for'
C
C-----------------------------------------------------------------------
C
      CHARACTER*120 TITLE
      INTEGER IOS
C
C-----------------------------------------------------------------------
C
C     READ ITERATION PARAMETERS
C
      OPEN (12,FILE='Model/ALMA_i_inf.txt',STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.EQ.0) THEN
C
C       SOLVE MOMENTUM, ENERGY, STRUCTURAL, PORE PRESSURE, LEVEL-SET
C
        READ (12,*) TITLE
        READ (12,*) MSM,MSE,MSS,MSP,MSL
C
C       TIME STEP, NUMBER OF STEPS, OUTPUT FREQUENCY, RESIDUAL
FREQUENCY, SUB-LEVEL ITERATIONS
C
        READ (12,*) TITLE
        READ (12,*) DT,MAXITER,IWRITE,IRES,ISUB
C
C       RELAXATION FOR: POTENTIAL, UPWIND, INTERFACE
C
```

```fortran
      READ (12,*) TITLE
      READ (12,*) THP,THU,THI
C
C     GRAVITY
C
      READ (12,*) TITLE
      READ (12,*) GX,GY
C
      ELSE
C
       WRITE (*,*) '...MISSING INFORMATION FILE.'
       STOP
C
      END IF
C
      CLOSE (12)
C
C------------------------------------------------------------------------
C
C     VALIDATE ITERATION PARAMETERS
C
      IF (DT.LT.EPS) THEN
       WRITE (*,*)
     & '...TIME STEP IS TOO SMALL....................'
       STOP
      END IF
C
      IF (IWRITE.LT.1) IWRITE=MAXITER
      IF (IRES.LT.1) IRES=1
      IF (IRES.GT.IWRITE) IRES=IWRITE
      IF (ISUB.LT.1) ISUB=1
C
      IF (THP.LT.0.0D+000) THP=0.0D+000
      IF (THP.GE.1.0D+000) THP=1.0D+000
C
      IF (THU.LT.0.0D+000) THU=0.0D+000
      IF (THU.GT.1.0D+000) THU=1.0D+000
C
      IF (THI.LT.0.0D+000) THI=0.0D+000
      IF (THI.GT.1.0D+000) THI=1.0D+000
C
C------------------------------------------------------------------------
C
C     READ GEOMETRY
C
      OPEN (11,FILE='Model/ALMA_d_geo.txt',STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.EQ.0) THEN
C
       READ (11,*) TITLE
C
C      NUMBER OF REGIONS
C
       READ (11,*) NR
C
       DO K=1,NR
C
```

```fortran
C         NUMBER OF BOUNDARY POINTS AND INTERNAL POINTS
C
          READ (11,*) KK,NB(K),NI(K)
C
C         BOUNDARY GEOMETRY
C
          DO I=1,NB(K)
           READ (11,*) II,XC(I,K),YC(I,K),AR(I,K),XN(I,K),YN(I,K)
          END DO
C
C         INTERNAL POINTS
C
          NC(K)=NB(K)+NI(K)
          DO I=NB(K)+1,NC(K)
           READ (11,*) II,XC(I,K),YC(I,K)
          END DO
C
         END DO
C
      ELSE
C
       WRITE (*,*) '...MISSING GEOMETRIC DATA FILE.'
       STOP
C
      END IF
C
      CLOSE (11)
C
C-----------------------------------------------------------------
C
C     READ MATERIALS FILE
C
      OPEN (11,FILE='Model/ALMA_i_mat.txt',STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.NE.0) THEN
C
       WRITE (*,*) '...MISSING MATERIALS FILE.'
       STOP
C
      ELSE
C
       DO K=1,NR
C
C       READ REGION NUMBER AND MATERIAL TYPE
C
        READ (11,*) KK,KR(K)
C
C       KR = 0 : SOLID
C
        IF (KR(K).EQ.0) THEN
C
         READ (11,*) TITLE
         READ (11,*) DES(K)
         READ (11,*) VIS(K),POS(K)
         READ (11,*) TCS(K),SHS(K)
         READ (11,*) BES(K),TRS(K)
C
```

```fortran
            DEE(K)=DES(K)
            VIE(K)=VIS(K)
            POE(K)=POS(K)
            TCE(K)=TCS(K)
            SHE(K)=SHS(K)
            BEE(K)=BES(K)
            TRE(K)=TRS(K)
C
          END IF
C
C         KR = 1 : FLUID
C
          IF (KR(K).EQ.1) THEN
C
            READ (11,*)  TITLE
            READ (11,*)  DEF(K)
            READ (11,*)  VIF(K),POF(K)
            READ (11,*)  TCF(K),SHF(K)
            READ (11,*)  BEF(K),TRF(K)
C
            DEE(K)=DEF(K)
            VIE(K)=VIF(K)
            POE(K)=POF(K)
            TCE(K)=TCF(K)
            SHE(K)=SHF(K)
            BEE(K)=BEF(K)
            TRE(K)=TRF(K)
C
          END IF
C
C         KR = 2 : POROUS MEDIUM
C
          IF (KR(K).EQ.2) THEN
C
            READ (11,*)  TITLE
            READ (11,*)  DES(K)
            READ (11,*)  VIS(K),POS(K)
            READ (11,*)  TCS(K),SHS(K)
            READ (11,*)  BES(K),TRS(K)
            READ (11,*)  PER(K),POR(K)
C
            READ (11,*)  TITLE
            READ (11,*)  DEF(K)
            READ (11,*)  VIF(K),POF(K)
            READ (11,*)  TCF(K),SHF(K)
            READ (11,*)  BEF(K),TRF(K)
            READ (11,*)  COF(K)
C
            DEE(K)=DES(K)
            VIE(K)=VIS(K)
            POE(K)=POS(K)
            TCE(K)=TCS(K)
            SHE(K)=SHS(K)
            BEE(K)=BES(K)
            TRE(K)=TRS(K)
C
          END IF
```

```
C
       END DO
C
      END IF
C
      CLOSE (11)
C
C----------------------------------------------------------------------
C
C     READ INTERPOLATION VECTORS
C
      OPEN (21,FILE='Model/ALMA_d_vec.bin',FORM='UNFORMATTED'
     &          ,STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.EQ.0) THEN
C
       READ (21) NRC
       IF (NRC.EQ.NR) THEN
        DO K=1,NR
         READ (21) NCC
         IF (NCC.EQ.NC(K)) THEN
          DO I=1,NC(K)
           READ (21) NCONN(I,K)
           READ (21) RXA(I,K),RYA(I,K)
           DO II=1,NCONN(I,K)
            READ (21) ICONN(I,II,K),
     &                FXC(I,II,K),FYC(I,II,K),
     &                FXX(I,II,K),FYY(I,II,K),FXY(I,II,K),
     &                FXE(I,II,K),FXW(I,II,K),FYN(I,II,K),FYS(I,II,K),
     &                SXE(I,II,K),SXW(I,II,K),SYN(I,II,K),SYS(I,II,K)
           END DO
          END DO
         ELSE
          IOS=1
         END IF
        END DO
       ELSE
        IOS=1
       END IF
      END IF
C
      IF (IOS.NE.0) THEN
       WRITE (*,*) '...MISSING OR CORRUPTED INTERPOLATION VECTOR FILE.'
       STOP
      END IF
C
      CLOSE (21)
C
C----------------------------------------------------------------------
C
C     READ TRIANGULATION FROM FILE
C
      OPEN (35,FILE='Model/ALMA_d_tri.txt',STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.EQ.0) THEN
C
      READ (35,*) NRR
```

```fortran
       IF (NRR.NE.NR) IOS=1
       DO K=1,NRR
        READ (35,*) NCC,NMESH(K)
        IF (NCC.NE.NC(K)) IOS=1
        DO I=1,NMESH(K)
         READ (35,*) (MESH(I,II,K),II=1,3)
        END DO
       END DO
C
      END IF
C
      IF (IOS.NE.0) THEN
       WRITE (*,*) '...MISSING OR CORRUPTED TRIANGULATION FILE.'
       DO K=1,NR
        NMESH(K)=0
       END DO
      END IF
C
      CLOSE (35)
C
C----------------------------------------------------------------------
C
C     OUTPUT INFORMATION
C
C      OPEN (12,FILE='Model/ALMA_o_inf.txt',STATUS='OLD',IOSTAT=IOS)
C
C      IF (IOS.EQ.0) THEN
C
C       READ (12,*) ITP,CTIME
C
C      ELSE
C
       ITP=0
       CTIME=0.D+000
C
C      END IF
C
C      CLOSE (12)
C
C----------------------------------------------------------------------
C
C     INCREASE MAXIMUM ITERATIONS TO ACCOUNT FOR PREVIOUS RESULTS
C
      MAXITER=MAXITER+ITP
C
C----------------------------------------------------------------------
C
      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
CCC
CCC
CCC                         SUBROUTINE INPUTENERGY
CCC
CCC
CCC
CCC    PURPOSE:
CCC
CCC
CCC
CCC    READS ENERGY CONDITIONS AND SOLUTION
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
      SUBROUTINE INPUTENERGY
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE '../../Include/Parameters.for'
      INCLUDE '../../Include/Geometry.for'
      INCLUDE '../../Include/Materials.for'
      INCLUDE '../../Include/Field-E.for'
      INCLUDE '../../Include/Field-M.for'
      INCLUDE '../../Include/Topology.for'
      INCLUDE '../../Include/Interpolation-C.for'
C
C----------------------------------------------------------------------
C
      INTEGER IOS
      REAL*8 TCV(NCMAX)
C
C----------------------------------------------------------------------
C
C     READ ENERGY BOUNDARY CONDITIONS
C
      OPEN (11,FILE='Model/ALMA_d_ene.txt',STATUS='OLD',IOSTAT=IOS)
C
      IF (IOS.NE.0) THEN
       WRITE (*,*) '...MISSING ENERGY BOUNDARY CONDITION FILE.'
       STOP
      END IF
C
      DO K=1,NR
C
C       READ REGION NUMBER
C
       READ (11,*) KK
C
C       READ INITIAL VALUES
C
       READ (11,*) TINI(K)
C
C       READ BODY FORCES
```

```fortran
C
        READ (11,*) UBG(K)
C
C       BOUNDARY CONDITIONS
C       KE=0:INSULATED, KE=1:DIRICHLET, KE=2:NEUMANN, KE=3:ROBIN
C       KE<-1000:INTERFACE TEMPERATURE, KE<-2000:INTERFACE FLUX
C
        DO I=1,NB(K)
         READ (11,*) II,KE(I,K),TB,QB,HB
C        INSULATED WALL
         IF (KE(I,K).EQ.0) THEN
          GT(I,1,K)=0.D+000
          GT(I,2,K)=1.D+000
          GT(I,3,K)=0.D+000
         END IF
C        DIRICHLET (TEMP)
         IF (KE(I,K).EQ.1) THEN
          GT(I,1,K)=1.D+000
          GT(I,2,K)=0.D+000
          GT(I,3,K)=TB
         END IF
C        NEUMANN (FLUX)
         IF (KE(I,K).EQ.2) THEN
          GT(I,1,K)=0.D+000
          GT(I,2,K)=-TCE(K)
          GT(I,3,K)=QB
         END IF
C        ROBIN (CONVECTION)
         IF (KE(I,K).EQ.3) THEN
          GT(I,1,K)=HB
          GT(I,2,K)=TCE(K)
          GT(I,3,K)=HB*TB
         END IF
C
        END DO
C
       END DO
C
       CLOSE (11)
C
C-----------------------------------------------------------------------
C
C     ENERGY OUTPUT
C
       OPEN (13,FILE='Model/ALMA_o_ene.bin',STATUS='OLD',IOSTAT=IOS
     &          ,FORM='UNFORMATTED')
C
       IF (IOS.EQ.0) THEN
C
        DO K=1,NR
         DO I=1,NC(K)
          READ (13) TCC(I,K),DXT(I,K),DYT(I,K),D2T(I,K)
         END DO
        END DO
C
       ELSE
C
```

```fortran
        DO K=1,NR
         DO I=1,NC(K)
          TCC(I,K)=TINI(K)
          DXT(I,K)=0.D+000
          DYT(I,K)=0.D+000
          D2T(I,K)=0.D+000
         END DO
        END DO
C
      END IF
C
      CLOSE (13)
C
C----------------------------------------------------------------------
C
C     INTERFACE ENERGY BOUNDARY CONDITIONS
C
      DO K=1,NR
C
C      INTERFACE BOUNDARY CONDITIONS
C      KE<-1000:INTERFACE TEMPERATURE, KE<-2000:INTERFACE HEAT FLUX
C
       DO I=1,NB(K)
C       TYPE 1 INTERFACE: IMPOSE TEMPERATURE
        IF ((KE(I,K).LT.(-1000)).AND.(KE(I,K).GT.(-2000))) THEN
         GT(I,1,K)=1.D+000
         GT(I,2,K)=0.D+000
         GT(I,3,K)=TCC(I,K)
        END IF
C       TYPE 2 INTERFACE: IMPOSE HEAT FLUX
        IF ((KE(I,K).LT.(-2000)).AND.(KE(I,K).GT.(-3000))) THEN
         GT(I,1,K)=0.D+000
         GT(I,2,K)=-TCE(K)
         GT(I,3,K)=-TCE(K)*(DXT(I,K)*XN(I,K)+DYT(I,K)*YN(I,K))
        END IF
C
       END DO
C
      END DO
C
C----------------------------------------------------------------------
C
      RETURN
      END
```

```fortran
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
CCC
CCC
CCC                      SUBROUTINE SOLVEENERGY
CCC
CCC
CCC
CCC    PURPOSE:
CCC
CCC
CCC
CCC    SOLVES ENERGY TRANSPORT EQUATION
CCC
CCC
CCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCC
      SUBROUTINE SOLVEENERGY
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      INCLUDE '../../Include/Parameters.for'
      INCLUDE '../../Include/Information.for'
      INCLUDE '../../Include/Geometry.for'
      INCLUDE '../../Include/Materials.for'
      INCLUDE '../../Include/Field-S.for'
      INCLUDE '../../Include/Field-M.for'
      INCLUDE '../../Include/Field-E.for'
C
C----------------------------------------------------------------------
C
C     USE THE FOLLOWING SPACE TO INCLUDE THE COMMON BLOCKS FOR FUNCTION
VARIABLES
C
      PARAMETER (NTMAX=20,NMMAX=200)
      COMMON/MEASURE/NMN,NMT,MT(NTMAX),MR(NMMAX),MN(NMMAX)
      COMMON/MEASURT/TM(NMMAX,NTMAX)
      COMMON/HOLELOC/XCH,YCH,RXH,RYH
C
C----------------------------------------------------------------------
C
      REAL*8 DIFF(NCMAX)
      REAL*8 RHS(NCMAX)
      REAL*8 QFT(NCMAX)
C
C----------------------------------------------------------------------
C
C     INITIALIZE RESIDUAL
C
      RESE=0.D+000
C
C----------------------------------------------------------------------
C
C     LOOP OVER SUBREGIONS FOR FIELD SOLUTION
C
      DO K=1,NR
```

```
C
C      DIFFUSIVITY
C
       DO I=1,NC(K)
        DIFF(I)=TCE(K)/(DEE(K)*SHE(K))
CCC
CCC
CCC
        ELX=(XC(I,K)-XCH)/(1.33D+000*RXH)
        ELY=(YC(I,K)-YCH)/(1.33D+000*RYH)
        IF ((K.EQ.1).AND.((ELX*ELX+ELY*ELY).LT.1.D+000)) THEN
         DIFF(I)=30.D+000*DIFF(I)
C          WRITE (*,*) '*'
         END IF
CCC
CCC
CCC
       END DO
C
C      BOUNDARY CONDITIONS
C
       DO I=1,NB(K)
        RHS(I)=GT(I,3,K)
       END DO
C
C      BODY FORCE
C
       DO I=NB(K)+1,NC(K)
        RHS(I)=UBG(K)/(DEE(K)*SHE(K))
       END DO
C
C      CONVECTIVE ENERGY FLUX
C
       IF (KR(K).EQ.0) THEN
        DO I=1,NC(K)
         QFT(I)=0.D+000
        END DO
       END IF
C
cc        IF (KR(K).EQ.1) THEN
cc         CALL UPWIND(UCC,VCC,TCC,QFT,DIFF,K)
cc        END IF
C
cc        IF (KR(K).EQ.2) THEN
cc         CALL UPWIND(VFX,VFY,TCC,QFT,DIFF,K)
cc         DO I=1,NC(K)
cc          QFT(I)=DEF(K)*SHF(K)*QFT(I)/(DEE(K)*SHE(K))
cc         END DO
cc        END IF
C
C      TRANSPORT ENERGY
C
       CALL TRANSPORT(TCC,D2T,DXT,DYT,QFT,DIFF,RHS,GT,DT,REST,K)
C
C      ACCUMULATE ENERGY RESIDUAL
C
cc        RESE=RESE+REST
```

```fortran
C
C      END SUBREGION LOOP
C
      END DO
C
C----------------------------------------------------------------------
C
C      AVERAGE INTERFACE ENERGY
C
      IF (NR.GT.1) THEN
       CALL INTERFACEENERGY
      END IF
C
C----------------------------------------------------------------------
C
      RETURN
      END
```