

P. R. M. Lyra  
R. de C. F. de Lima  
and C. S. C. Guimarães

Departamento de Engenharia Mecânica – UFPE  
Av. Acadêmico Hélio Ramos, S/N  
50740-530 Recife, PE, Brazil  
prmlyra@demec.ufpe.br  
ritallima@ufpe.br  
carla@demec.ufpe.br

D. K. E. de Carvalho

Departamento de Engenharia Civil – UFPE  
Av. Acadêmico Hélio Ramos, S/N  
50740-530 Recife, PE, Brazil  
darlan@demec.ufpe.br

# An Edge-Based Unstructured Finite Volume Procedure for the Numerical Analysis of Heat Conduction Applications

*In recent years, there has been a significant level of research on the application of unstructured mesh methods to the simulation of a variety of engineering and scientific problems. Great progress has been achieved in such area and one of the most successful methodologies consists on the use of the Finite Volume Method (FVM). The unstructured FV formulation is very flexible to deal with any kind of control volume and therefore any kind of unstructured meshes, which are particularly important when complex geometries or automatic mesh adaptation are required. In this article, an unstructured finite volume vertex centered formulation, which was implemented using an edge-based data structure, is deduced and detailed for the solution of heat conduction problems. The numerical formulation is initially described considering a tri-dimensional model and latter particularized for bi-dimensional applications using triangular meshes. The presented procedure is very flexible and efficient to solve potential problems. It can also be extended to deal with a broader class of applications, such as models involving convection-diffusion-reaction terms, after considering the appropriate discretization of the convection-type term. In order to demonstrate the potentiality of the method, some model problems are investigated and the results are validated using analytical or other well-established numerical solutions.*

**Keywords:** Finite volume method, unstructured mesh, edge based data structure, heat transfer

## Introduction

Whenever analyzing numerical applications that involve complex geometries the adoption of methods capable to deal with unstructured meshes is very attractive and highly recommended (AGARD Report 787, 1992). Within such class of methods the most frequently used are the finite element method (FEM), (Zienkiewicz and Morgan, 1983) and the finite volume method (FVM), (Barth, 1992). The FVM is very flexible to deal with any kind of control volumes and, consequently, any kind of unstructured meshes including dual meshes. Finite volume methods are usually either a node/vertex centered, where the unknowns are defined at the nodes of the mesh, or element/cell centered where the unknowns are defined within the element, usually at the element's centroid. Both options have advantages and disadvantages, but in two-dimensional applications all of them have basically the same computational cost, which is proportional to the number of edges of the mesh. However, the node-centered formulation has a strong connection with an edge-based finite element formulation, when linear triangular (tetrahedral) elements are used, and requires less memory and computations when extended for three-dimensional tetrahedral meshes, (Barth, 1992, Peraire et al., 1993 and Sorensen et al., 2001). The adoption of unstructured grids implies the storage of mesh topology information (connectivities), increasing the use of computer memory and indirect addressing to retrieve local information required by the solver. In order to reduce indirect addressing, CPU time and memory requirements (Barth, 1992, Peraire et al., 1993 and Lyra, 1994), an edge-based data structure is adopted. This data structure also enables a straightforward implementation of different types of finite difference discretization, such as upwind-biased schemes in the context of unstructured algorithms, when dealing with CFD (Computational Fluid Dynamics) problems, (Peraire et al., 1993 and Lyra et al., 1994).

In this paper, the vertex centered finite volume formulation using median dual control volumes is implemented using an edge-based data structure, (Barth, 1992, Sorensen et al., 1999 and Lyra et al., 2002). The complete formulation is deduced and detailed for the solution of heat conduction problems. In two dimensional (2-D) models, triangular, quadrilateral or mixed meshes can be directly used, and the same happens when dealing with three dimensional (3-D) models, where tetrahedral, hexahedral, pyramids, prisms and mixed meshes can be adopted. We derive the FV discretization of a transient potential problem subject to different types of boundary conditions (Dirichlet, Neumann, and Cauchy) and to some non-conventional loads, considering also systems with multi-materials.

This paper is organized as follows: after these initial considerations, the physical-mathematical model considered is described. Then, the discrete spatial formulation adopted is fully presented. Some important implementation aspects are discussed and several simple model problems are analyzed to validate and to study the performance of the whole procedure. Finally, some concluding comments are presented and the potentiality of the described approach is highlighted.

## Mathematical Model

Using the energy conservation law we can derive the partial differential equation that governs transient heat transfer in a stationary continuous medium,

$$\rho c \frac{\partial T}{\partial t} = -\frac{\partial q_j}{\partial x_j} + S \quad \text{in } \Omega \times T. \quad (1)$$

In previous equation, the product  $\rho c$  represents the heat capacity, with  $\rho$  being the mass density and  $c$  being the specific heat,  $T$  is the temperature,  $q_j$  is the heat flux in  $x_j$  direction and  $S$  represents the source (or sink) terms. The spatial domain of the problem is represented by  $\Omega$ , with  $x_j$  being the spatial independent variable with  $j$

varying from one to the number of spatial dimensions, and  $T = [t^i, t^f]$  represents the time interval of integration.

The constitutive relation between the conductive heat flux and the temperature gradient is given by Fourier's law,

$$q_j = -k_j \frac{\partial T}{\partial x_j}, \quad (2)$$

in which  $k_j$  is the thermal conductivity in  $x_j$  direction. For simplicity, the medium is considered either orthotropic or isotropic with  $\rho$ ,  $c$  and  $k_j$  constants.

Equation (1) represents a boundary-initial value problem and must be subjected to boundary and initial conditions. The boundary conditions of interest can be of three different types:

a) A prescribed temperature  $\bar{T}$  over a portion of the boundary  $\Gamma_D$ , i.e., Dirichlet boundary condition:

$$T = \bar{T} \quad \text{in } \Gamma_D \times T, \quad (3)$$

b) A prescribed normal heat flux  $\bar{q}_n$  over  $\Gamma_N$ , also known as Neumann boundary condition:

$$-q_j n_j = \bar{q}_n \quad \text{in } \Gamma_N \times T, \quad (4)$$

in which  $n_j$  are the outward normal direction cosines.

c) A mixed type boundary condition over  $\Gamma_C$ , called Cauchy or Robin boundary condition:

$$-q_j n_j = \bar{q}_n + \alpha_f (T - T_a) \quad \text{in } \Gamma_C \times T, \quad (5)$$

with  $\alpha_f$  representing the film coefficient and  $T_a$  being the bulk fluid temperature.

Finally, an initial distribution of the temperature  $\bar{T}^i$  must be known at an initial time stage  $t^i$ , and the initial condition is expressed by:

$$T = \bar{T}^i \quad \text{in } \Omega \text{ and } t = t^i. \quad (6)$$

Equations (1) to (6) fully describe our mathematical model, which governs heat conduction in a stationary medium.

### Finite Volume Formulation

In this section, the majority of the numerical formulation adopted is presented without reference to a particular type of mesh or spatial dimension. The formulation is completed by assuming a two dimensional computational domain discretized into an unstructured assembly of triangular elements. The time discretization adopted is the simple first-order accurate Euler-forward procedure. Such scheme is just first order accurate in time and the  $\Delta t$  must be chosen according to a stability condition (Zienkiewicz and Morgan, 1983). Other alternatives, such as the generalized trapezoidal method (Zienkiewicz and Morgan, 1983 and Lyra, 1994), multi-stage Runge-Kutta scheme (Lyra, 1994) or schemes involving more than two time intervals (Sorensen, 2001) can be implemented if higher-order time accuracy is required.

The mathematical model presented in previous section represents an example of potential problem and therefore a broad variety of applications governed by similar models can be solved using the numerical procedure to be described in this section.

### Spatial Discretization

The integral form of the potential problem given by Eq. (1) is written as,

$$\int_{\Omega} \rho c \frac{\partial T}{\partial t} d\Omega = - \int_{\Omega} \frac{\partial q_j}{\partial x_j} d\Omega + \int_{\Omega} S d\Omega, \quad (7)$$

or alternatively by the use of the divergence theorem,

$$\int_{\Omega} \rho c \frac{\partial T}{\partial t} d\Omega = - \int_{\Gamma} q_j n_j d\Gamma + \int_{\Omega} S d\Omega, \quad (8)$$

where  $\Omega$  denotes an arbitrary control volume, with closed boundary  $\Gamma$ .

The computational domain is discretized into an unstructured assembly of elements. Then Eq. (8) is applied over each control volume in the mesh. So the volume integrals of (8) can be computed over the control volume surrounding node  $I$  as

$$\rho c \int_{\Omega_I} \frac{\partial T}{\partial t} d\Omega \cong \rho c \frac{\partial T_I}{\partial t} V_I \cong \rho c \frac{\partial \hat{T}_I}{\partial t} V_I, \quad (9)$$

and

$$\int_{\Omega_I} S d\Omega \cong S_I V_I, \quad (10)$$

where  $V_I$  is the volume of the control volume,  $\hat{T}_I$  and  $S_I$  represent the numerically calculated temperature and source term at node  $I$ , respectively. The assumption of  $\rho$  and  $c$  being constants was used in previous approximations.

The boundary integral presented in Eq. (8) is computed over the boundary of the control volume that surrounds node  $I$  using an edge-based representation of the mesh, i.e.,

$$\int_{\Gamma_I} q_j n_j d\Gamma \cong \sum_L C_{IL}^j \hat{q}_{IL}^{j(\Omega)} + \sum_L D_{IL}^j \hat{q}_{IL}^{j(\Gamma)}, \quad (11)$$

for a general flux  $q_j$ . In Equation (11)  $C_{IL}^j$  denotes the coefficient

that must be applied to the approximate edge value of the flux  $\hat{q}_{IL}^{j(\Omega)}$  in the  $x_j$  direction to obtain the contribution made by the edge  $L$  to node  $I$ .  $J_L$  represents the node connected to node  $I$  through edge  $L$ . In addition,  $D_{IL}^j$  represents the boundary edge coefficient that must

be applied to the boundary edge flux  $\hat{q}_{IL}^{j(\Gamma)}$  when the edge  $L$  lies on the boundary. These coefficients can be readily computed and this will be detailed afterwards. The first summation in Eq. (11) extends over all edges  $L$  in the mesh which are connected to node  $I$ , and the second summation is non-zero only when node  $I$  is on the boundary and extends over all boundary edges that are connected to node  $I$ .

Considering the approximations given by Eqs. (9), (10) and (11), the semi-discrete formulation of Eq. (8) can be conveniently expressed as:

$$\rho c \frac{d\hat{T}_I}{dt} V_I = - \left( \sum_L C_{IL}^j \hat{q}_{IL}^{j(\Omega)} + \sum_L D_{IL}^j \hat{q}_{IL}^{j(\Gamma)} \right) + S_I V_I \quad (12)$$

The approximation of the value of the edge flux  $\hat{q}_{IJ_L}^{j(\Omega)}$  is computed using the midpoint rule, or simple arithmetic average:

$$\hat{q}_{IJ_L}^{j(\Omega)} = \frac{\hat{q}_I^j + \hat{q}_{J_L}^j}{2} \quad (13)$$

To compute the boundary edge flux  $\hat{q}_{IJ_L}^{j(r)}$  we have considered a linear variation of the flux over edge  $IJ_L$ :

$$\hat{q}_{IJ_L}^{j(r)} = \frac{(3\hat{q}_I^j + \hat{q}_{J_L}^j)}{4} \quad (14)$$

Alternatively, a finite volume fashion approximation could be used, which would consider the value of the flux constant inside the control volume and compute  $\hat{q}_{IJ_L}^{j(r)}$  as the nodal value  $\hat{q}_I^j$ . Both options gave good results for the test cases studied and further investigation is required to favor one of them.

In order to compute the edge fluxes described by Eqs. (13) and (14) we need to know the nodal values of the fluxes and so the nodal values of the temperature gradients. By adopting the divergence theorem and the approximation used to compute volume integrals over a control volume surrounding node  $I$  (e.g. Eq. (10)), we have:

$$\int_{\Omega_I} \frac{\partial T}{\partial x_j} d\Omega \cong \int_{\Gamma_I} T n_j d\Gamma \quad \text{and} \quad \int_{\Omega_I} \frac{\partial T}{\partial x_j} d\Omega \cong \frac{\partial \hat{T}_I}{\partial x_j} V_I \quad (15)$$

From previous expressions and using the same approximation adopted to compute the boundary integral in Eq. (11), we get the approximate nodal gradients through:

$$\frac{\partial \hat{T}_I}{\partial x_j} V_I \cong \int_{\Gamma_I} \hat{T} n_j d\Gamma \cong \sum_L C_{IJ_L}^j \hat{T}_{IJ_L}^{(\Omega)} + \sum_L D_{IJ_L}^j \hat{T}_{IJ_L}^{(r)} \quad (16)$$

Similarly to the computation of the edge fluxes described in Eqs. (13) and (14), the edge values of the temperature,  $\hat{T}_{IJ_L}^{(\Omega)}$  and  $\hat{T}_{IJ_L}^{(r)}$ , are calculated by:

$$\hat{T}_{IJ_L}^{(\Omega)} = \frac{\hat{T}_I + \hat{T}_{J_L}}{2} \quad (17)$$

$$\hat{T}_{IJ_L}^{(r)} = \frac{(3\hat{T}_I + \hat{T}_{J_L})}{4}$$

The use of Eq. (16) to compute the gradients implies that the discretization of the diffusion term in Eq. (12) involves information from two layers of points surrounding the point  $I$  under consideration. Furthermore, if uniform structured quadrilateral (or hexahedral) meshes are adopted, the values computed at a given node are uncoupled from the values of those nodes directly connected to it. This fact may cause “checker-boarding” or “odd-even” oscillations (Lyra, 1994 and Sorensen, 2001). When computing the diffusive term in non-uniform unstructured meshes, the adoption of an extended stencil and a weak coupling with the directly connected nodes may lead to some loss of robustness and reduction of convergence rate of the resulting scheme. In order to overcome such weak-

nesses, the gradients must be computed in an alternative way. Following the procedure suggested in the literature (Crumpton et al., 1997 and Sorensen, 2001) a better approach can be developed as follows.

The edges values of the temperature gradient can be approximately computed by:

$$\frac{\partial T_{IJ_L}}{\partial x_j} \cong \frac{\partial \hat{T}_{IJ_L}}{\partial x_j} = \frac{1}{2} \left( \frac{\partial \hat{T}_I}{\partial x_j} + \frac{\partial \hat{T}_{J_L}}{\partial x_j} \right) \quad (18)$$

Using a local frame of reference, in which one axis is along the edge direction (P) and another axis (N) is in the plane orthogonal to direction (P), (see Fig. 1), the edge gradient can be alternatively computed as:

$$\frac{\partial \hat{T}_{IJ_L}}{\partial x_j} = \frac{\partial \hat{T}_{IJ_L}^P}{\partial x_j} + \frac{\partial \hat{T}_{IJ_L}^N}{\partial x_j} \quad (19)$$

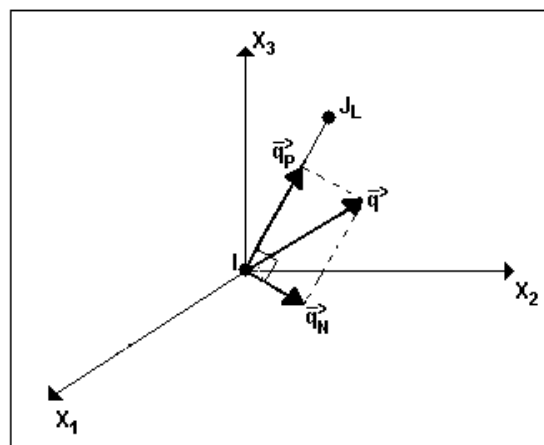


Figure 1. Local frame of reference.

Once the nodal gradients are known the corresponding fluxes can be directly obtained by the use of the Fourier Constitutive Law given in Eq. (2), and, similarly to the edge values of the temperature gradients, the edge fluxes are given by:

$$\hat{q}_{IJ_L}^{j(\Omega)} = \frac{(\hat{q}_I^j + \hat{q}_{J_L}^j)}{2} = \left( \hat{q}_{IJ_L}^{j(\Omega)} \Big|_P + \hat{q}_{IJ_L}^{j(\Omega)} \Big|_N \right) \quad (20)$$

Using a second-order central finite difference approximation, the temperature derivative over the edge in direction (P) can be calculated by:

$$\frac{\partial \hat{T}_{IJ_L}}{\partial x_{P^*}} = \frac{\hat{T}_{J_L} - \hat{T}_I}{\Delta X_{IJ_L}} \mathbf{L}_{IJ_L} \quad (21)$$

Where  $\Delta X_{IJ_L}$  is the size of the edge  $IJ_L$ , i.e.:

$$\Delta X_{IJ_L} = |X_{J_L} - X_I| \quad \text{with} \quad X_I = (x_I^1, x_I^2, x_I^3), \quad (22)$$

and

$$\mathbf{L} = \mathbf{L}_{IJ_L} = \frac{\mathbf{X}_{J_L} - \mathbf{X}_I}{\Delta X_{IJ_L}} \quad \text{and} \quad L_j = \frac{x_{J_L}^j - x_I^j}{\Delta X_{IJ_L}}, \quad (23)$$

where  $\mathbf{L}$  represents the unitary vector defined in the edge direction from  $I$  to  $J_L$ , and  $L_j$  are the director cosines.

The Cartesian components of the derivative on the edge direction are given by,

$$\frac{\partial \hat{T}_{IJ_L}^{p^*}}{\partial x_j} = L_j \frac{\partial \hat{T}_{IJ_L}}{\partial x_{p^*}}, \quad (24)$$

and the Cartesian components of the portion of the gradient orthogonal (or normal) to the edge direction is then,

$$\frac{\partial \hat{T}_{IJ_L}^N}{\partial x_j} = \frac{\partial \hat{T}_{IJ_L}}{\partial x_j} - \frac{\partial \hat{T}_{IJ_L}^{p^*}}{\partial x_j}, \quad (25)$$

where  $\partial \hat{T}_{IJ_L} / \partial x_K$  is calculated in a finite volume fashion given by Eq. (18).

In short, the edge temperature gradient given by Eq. (19) is computed using the edge direction quantity calculated by Eq. (24) and the normal one using Eq. (25). Similarly, the edge fluxes components given by Eq. (20) are now replaced by Eq. (26), which is computed using the gradients as described previously and the Fourier Constitutive Law Eq. (2), i.e.,

$$\hat{q}_{IJ_L}^{j(\Omega^*)} = \frac{(\hat{q}_I^j + \hat{q}_{J_L}^j)}{2} \cong \left( \hat{q}_{IJ_L}^{j(\Omega)} \Big|_{P^*} + \hat{q}_{IJ_L}^{j(\Omega)} \Big|_N \right), \quad (26)$$

where the heat fluxes in the edge direction  $\hat{q}_{IJ_L}^{j(\Omega)} \Big|_P$  are replaced by

$$\hat{q}_{IJ_L}^{j(\Omega)} \Big|_{P^*}.$$

The final semi-discrete scheme is then given by Eq. (12) after replacing  $q_{IJ_L}^{j(\Omega)}$  by  $q_{IJ_L}^{j(\Omega^*)}$ , i.e.:

$$\rho c \frac{d\hat{T}_I}{dt} V_I = - \left( \sum_L C_{IJ_L}^j \hat{q}_{IJ_L}^{j(\Omega^*)} + \sum_L D_{IJ_L}^j \hat{q}_{IJ_L}^{j(r)} \right) + S_I V_I. \quad (27)$$

### Weighting Coefficients Definitions

The definition of the weighting coefficients is presented here just for the two-dimensional model, which will be adopted for validating the formulation and computational system developed. However, similar definitions apply for the 3-D spatial model (Sorensen, 2001).

For two-dimensional problems, we have the mesh on a flat middle plane defined by coordinates  $x_1$  and  $x_2$ . Then, Eq. (8) must first be integrated over  $x_3$  and then over a 2-D space. In such model, we have the nodal volume computed as  $V_I = A_I E_I$ , where  $E_I$  refers to the thickness of the domain at point  $I$ , and  $A_I$  is the area of the control volume surrounding node  $I$ . The 2-D weighting coefficients  $C_{IJ_L}^j$  and  $D_{IJ_L}^j$  are defined by

$$C_{IJ_L}^j = \sum_K A_K n_K^j, \quad (28)$$

$$D_{IJ_L}^j = A_L n_L^j$$

where  $A_K = L_K E_K$  with  $E_K = (E_I + E_{J_L})/2$  and  $L_K$  is the length of each interface  $K$  associated to edge  $IJ_L$ . Each interface connects the element centroid ( $C$ ) to the middle point ( $MP$ ) of one of the edges that belongs to such element and  $A_L = L_L E_L$ , where  $L_L$  is half the size of the boundary edge under consideration and  $E_L$  is defined similarly to  $E_K$ .  $n_K^j$  and  $n_L^j$  are the outward normal direction cosines to the interface  $K$  and  $L$ , respectively. The geometric parameters required for computing the weighting coefficients are detailed in Figs. 2 and 3.

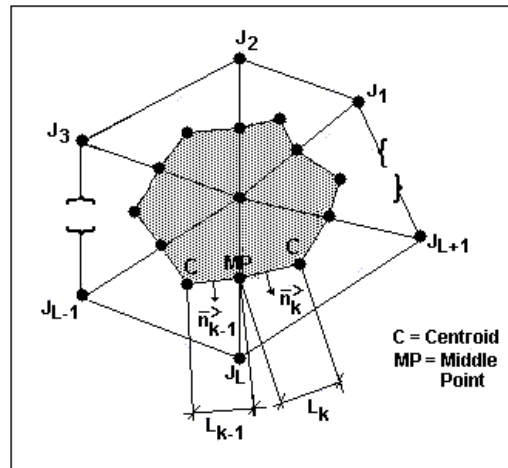


Figure 2. 2-D control volume and its geometric parameters.

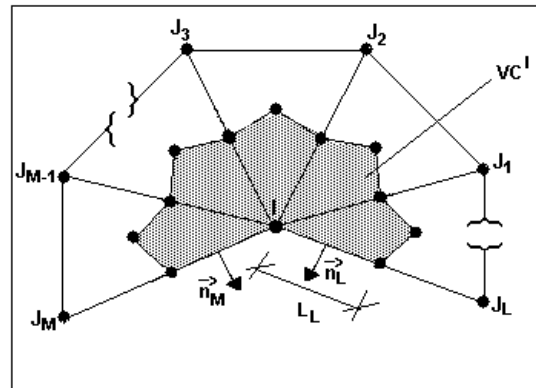


Figure 3. 2-D boundary control volume and its geometric parameters.

### Thermal Loads, Boundary Conditions and Multi-Materials Domain

We will consider only the two-dimensional model however, the three-dimensional extension is straightforward. The implementation of certain discrete terms relies on some features inherent to our system for two-dimensional mesh generation (Lyra and Carvalho, 2000 and Carvalho, 2001). The discretization of the different thermal loads represented by  $S$  in Eq. (1) and different boundary conditions (Eqs. (3) to (5)) are now considered. Finally, the treatment of multi-material problems is also described.

### Thermal Loads

For the 2-D model, the thermal load  $S$  over the domain is represented here as:

$$S = Q + [\alpha_{\Omega}(T_a - T)]/E, \quad (29)$$

with

$$Q = \bar{Q}^P + Q^C + Q^R, \quad (30)$$

where the superscripts  $P, C, R$  accounts for thermal sources (or “sinks”) acting on a point, a curve or a region, respectively. On the right hand side of Eq. (29) the first term represents the thermal sources (or sinks) described in Eq. (30). The second term accounts for convection over each face of the two dimensional domain,  $\alpha_{\Omega}$  is the film coefficient where the subscript  $\Omega$  is used to emphasize that it acts over the domain, while in Eq. (5),  $\alpha_r$  acts over the boundary with mixed boundary condition and  $E$  refers to the thickness of the domain. It should be observed that the second term does not exist in a three-dimensional model and the convection on the surface is accounted in the boundary conditions.

The integral of the thermal loads  $Q$  described by Eq. (30) is given by:

$$\int_{\Omega} Q d\Omega = Q^P + \int_{\Gamma_C} Q^C d\Gamma + \int_{\Omega_R} Q^R d\Omega. \quad (31)$$

In Equation (31), the term  $Q^P = \int_{\Omega} \bar{Q}^P d\Omega$  considers the total

value of heat source for a unity volume and therefore  $Q_I^P$  is just a point heat source computed at a given node  $I$ . If the point source is not applied at a nodal point its value is distributed to the nodes of the triangle that contains it, using a linear approximation (Lyra et al., 2000). The ADT (Alternate Digital Tree), (Bonet and Peraire, 1990), data structure and corresponding searching algorithm is used to determine the element that contains the position of the point load.

The flexibility of our two-dimensional mesh generator (Lyra and Carvalho, 2000) is exploited using the possibility to build a fictitious boundary along the curve where we want to apply a heat source. The term containing  $Q^C$  considers the heat source value for a strip of unity width over the plane and along the surface  $\Gamma_C$ , which represents the transversal section along the line heat source. Then, the boundary integral in (31) is easily approximated by each portion of the fictitious boundary associated to node  $I$  ( $\Gamma_{C_I}$ ) as:

$$\int_{\Gamma_{C_I}} Q^C d\Gamma \cong \sum_L Q_I^C A_L. \quad (32)$$

The summation extends over the two edges connected to node  $I$  that belong to the fictitious boundary and  $A_L$  is an area computed as previously defined.

If the heat source per unit of volume  $Q^R$  is distributed over a region  $\Omega_R$ , the third integral in Eq. (31) is then approximated in the same fashion as the transient term given in Eq. (9), i.e. for each control volume surrounding node  $I$ ,  $\forall I \in \Omega_{R_I}$ , we have:

$$\int_{\Omega_{R_I}} Q^R d\Omega \cong Q_I^R V_I. \quad (33)$$

Finally, the convective type source term is computed for each node  $I$ ,  $\forall I \in \Omega_{R_I}$ , by:

$$\int_{\Omega_{R_I}} [\alpha_{\Omega_R}(T_a - T)]/E d\Omega \cong \alpha_{\Omega_R}(T_a - \hat{T}_I) V_I / E_I. \quad (34)$$

For the previous loads given in Eqs. (33) and (34) a specific region covering  $\Omega_R$  is built with the help of our mesh generator, which allows for the generation of consistent multi-regions meshes. In Equation (34), for the explicit time integration adopted here, we can use the value of  $\hat{T}_I$  known from the previous time level.

### Boundary Conditions

To compute the Dirichlet boundary condition, Eq. (3), it is enough to substitute  $T_I$  by  $\bar{T}_I$  whenever required, i.e.  $\forall I \in \Gamma_D$ . To impose the Neumann boundary condition, Eq. (4), the total boundary edge flux that appears in the boundary loop in Eq. (27) must be projected on the directions parallel and normal to the edge under consideration. The normal portion must then be replaced by the prescribed flux  $\bar{q}_n$ .

For Cauchy (or Robin) mixed boundary condition, Eq. (5), the value  $(-\bar{q}_n + \alpha_r T_a)$  is known and computed in the same fashion as implemented to compute the Neumann boundary condition. The remaining term is computed for each  $\Gamma_{R_I}$  according to:

$$\int_{\Gamma_{R_I}} \alpha_r T d\Gamma \cong \sum_L \alpha_r \hat{T}_I A_L, \quad (35)$$

with  $\Gamma_{R_I}$  being the portion of the  $\Gamma_R$  boundary associated to node  $I$  and the summation extends over the two boundary edges connected to node  $I$ . Following the procedure adopted in Eq (34),  $\hat{T}_I$  of Eq. (35) is taken as the temperature of node  $I$  obtained at the previous time level.

### Multi-Materials Domain

Whenever addressing heat transfer problems which involve different material properties on different portions of the domain we need to build proper meshes for each sub-region and to perform consistently the discretization of the governing equation in order to guarantee the correct solution through the interface of the sub-regions. As already mentioned, our mesh generator has the flexibility to generate consistent meshes over multi-region domains.

For each boundary and interior edges apart from the value of the weighting coefficients we identify and keep in memory the region that the edge belongs to, during the pre-processing stage. This is necessary to recovery the material properties used in the computations during the processing stage. For each edge at the interface of two regions, the edge coefficient is computed independently for each region with the identification of the region also kept in memory. Referring to Fig. 4, the edge  $IJ_L$  would have two coefficients defined by:

$$C_{IJ_L}^{j(R_1)} = A_{k-I} n_{k-I}^j \quad \text{and} \quad C_{IJ_L}^{j(R_2)} = A_k n_k^j. \quad (36)$$

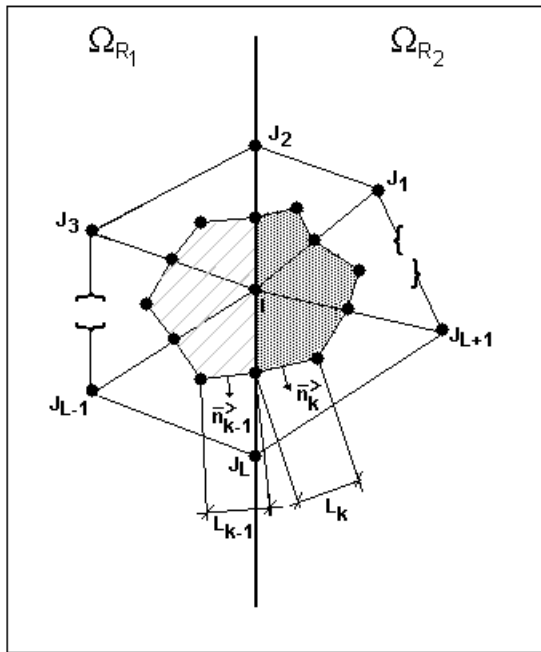


Figure 4. Control volumes at an interface between two regions and their geometric parameters.

For the gradients and associated fluxes computations the procedure basically consists on three steps: first a loop over all edges of the mesh except the interface edges; second, a loop over the boundary edges and third, a double loop over the interface edges. In each of these loops we use the corresponding edge coefficients and associated material properties. In the computation of the “final” discrete equation, we can proceed similarly, with the corresponding properties and thermal loads. The semi-discrete Eq. (27) is now replaced by:

$$\rho c \frac{dT}{dt} V_I = - \left( \sum_L C_{IJ}^j q_{IJ}^{j(\varrho^*)} + \sum_L D_{IJ}^j q_{IJ}^{j(r)} + \sum_{k=1}^2 \sum_L C_{IJ}^{j(R_k)} q_{IJ}^{j(r^*)} \right) + S_I V_I, \quad (37)$$

with the third term on the right hand side of Eq. (37) being non-zero only when node  $I$  is on the interface between two or more regions ( $\Gamma_I$ ) with different material properties. The external loop is necessary to consider twice each interface edges that are connected to node  $I$ , i.e. once for each region. The interface edges fluxes  $q_{IJ}^{j(r^*)}$  are computed similarly to  $q_{IJ}^{j(\varrho^*)}$ , see Eq. (26).

It should be observed that the proposed procedure is general, so that node  $I$  can belong to the interface of several regions, not just two and also that the interface does not need to be a straight line.

If the properties of the material, loads or boundary conditions vary in space, the middle point rule is adopted. For instance, the heat conductivity  $k$  of edge  $IJ_L$  is computed by:

$$k_{IJ_L} = \frac{k_I + k_J}{2}. \quad (38)$$

If the material has non-linear behavior [e.g.  $k = f(T)$ ] it is important to use an iterative procedure, such as the Newton-Raphson method, but such feature has not yet been attempted in the present formulation.

## Some Important Numerical and Implementation Issues

In this section, several aspects referring to the generation and use of unstructured meshes, the adoption of an edge-based data structure and other related issues are addressed.

### Unstructured Mesh Generation

In order to discretize arbitrary two-dimensional domains, we have developed a computational system (Lyra and Carvalho, 2000 and Carvalho, 2001) which can generate triangular, quadrilateral and mixed consistent meshes. The computational system deals with several connected domains and both isotropic and anisotropic meshes (i.e. directionally stretched meshes). In this work, only isotropic triangular meshes were considered and generated through the advancing front technique (Peraire et al., 1987), with the concepts of using a background grid to define the spacing, gradation and directionality of the mesh and of generating simultaneously points and triangles during the triangulation. As any conventional unstructured mesh generator the mesh data delivered consists of the physical coordinates simply listed by node numbers and a list of the connectivity of each element (topological information). Our mesh generator gives also a list of boundary edges connectivities, which are very helpful for the implementation of certain boundary conditions.

### Edge-Based Data Structure and Its Implementation

A significant reduction in gather/scatter costs and memory requirements in the solver can be obtained by going from an element-based to an edge-based data structure (Lyra et al., 1998). This reduction is more pronounced in three-dimensional simulations (Peraire et al., 1993). The adopted mesh generator, as any conventional unstructured mesh generator, provides the mesh data in an element-based data structure and the implementation of the finite volume solver described requires a pre-processing stage to convert the element-based data structure into edge-based. After the pre-processor stage is finished, the element-based data structure can be discarded.

The pre-processing stage consists basically on the following steps (Lyra, 1994):

1. Build the arrays with the mesh and boundary topology, which are lists of edges and boundary edges with their respective connectivities;
2. Compute and store the edge and boundary weighting coefficients;
3. Transfer loads, material properties, boundary and initial conditions, which are associated to the geometry, to the mesh topological entities.

To extract the edges from the original data structure in an efficient way, a hash table searching technique is used (Lyra et al., 1998). The weighting coefficients are computed as described previously. In our mesh generator all mesh topological entities (nodes, edges and elements) are associated to their correspondent geometric entity (point, curve or sub-domain). In this way, loads, material properties, boundary and initial conditions are initially associated to the geometry and then transferred to the topological data to get the finite volume model for the analysis. This is of particular importance in an adaptive procedure as the mesh changes dynamically during the analysis and a new finite volume model has to be built for each new mesh.

After the pre-processing stage, all the terms that appear in the finite volume formulation are calculated using loops over all edges of the mesh, except the interface ones, loops over the boundary edges and double loops over the interface edges with the contribu-

tions to the nodes (or vertices) being accumulated during the process. The operations performed inside these loops, which take place in the edge-based solver algorithm are:

1. Gather Information the nodes of each edge;
2. Operate on this information;
3. Scatter the results back to the nodes of the edges and add them to the nodal quantities.

### Numerical Results

In this section, some simple, though representative academic, examples are presented in order to validate and show some of the capabilities of the numerical scheme previously discussed. All steady-state solutions were obtained using the transient algorithm, which is stopped when the time derivative of the temperature is not changing, i.e. the residual is below a pre-assigned tolerance, with the adopted value of  $10^{-7}$ . For all steady-state solution we adopted the density  $\rho = 1.0 \text{ kg/m}^3$  and the specific heat  $c = 1.0 \text{ J/kg K}$ . These values guarantee that the equation is non-stiff and were not chosen taking into account any aspects related to computational efficiency. The results are post-processed using the scientific visualization tool "Mtool" (Tecgraf-PUC-Rio).

### Steady-State Heat Conduction Problem

The first application refers to a steady state solution of a two-dimensional heat transfer problem in a rectangular plate of 0.6 m by 1.0 m and uniform unity thickness. The left face of the plate is insulated (zero heat flux), while the bottom edge is at a fixed temperature of 100 °C and the right and top edges are under convection to ambient temperature of 0 °C. The thermal conductivity of the plate is  $k = 52.0 \text{ W/m}^\circ\text{C}$  and the convective heat transfer coefficient is  $\alpha_r = 750.0 \text{ W/m}^2\text{C}$ .

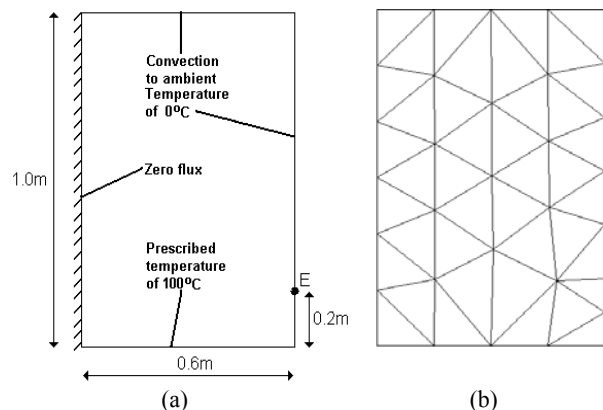


Figure 5. 2-D heat conduction problem: (a) domain representation; (b) coarse mesh with 32 nodes.

The initial condition is an uniform temperature of  $T^i = 0.0 \text{ }^\circ\text{C}$ . This example was extracted from the NAFEMS selected FE benchmarks in structural and thermal analysis (Barlow and Davies, 1987). Figure 5a shows the domain representation for this problem and Fig. 5b shows the triangular coarse mesh utilized. The target value to be achieved is a temperature of 18.3 °C in point E (see Fig. 5a).

In Table 1 we show the obtained results with two different uniform meshes. The first one is a coarse mesh with 32 nodes, and the second one is a finer mesh with 793 nodes. It can be observed that the final results are in good agreement with the expected target value.

Table 1. Temperatures at node (E) obtained with two different mesh densities.

TEMPERATURE AT NODE E ( $^\circ\text{C}$ )		
NAFEMS	Coarser Mesh	Finer Mesh
18.30	18.14	18.29

Figure 6 shows the contours of temperature for both meshes. It is important to note that even the coarse mesh provided a good result if compared with the NAFEMS results. As expected, for the finer mesh, the contours of temperature are much smoother than those obtained with the coarser one. Only uniform meshes have been used up to now, but the results can certainly be much better, even for meshes with the size of the coarse mesh, if the knowledge of the expected solution is used to build the mesh or an automatic error analysis and adaptive procedure is incorporated (Lyra et al., 2000).

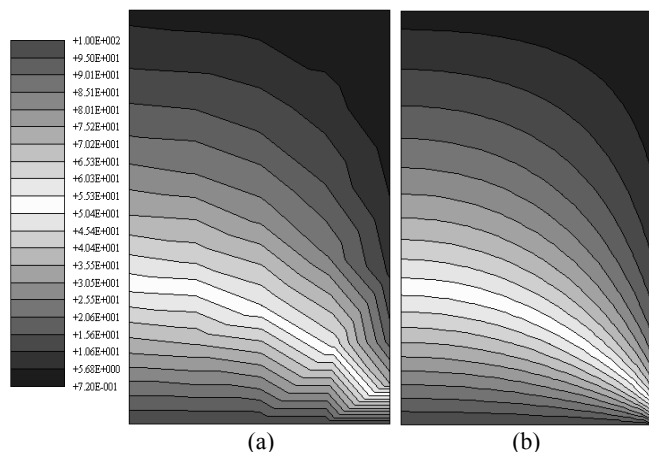


Figure 6. 2-D heat conduction problem. Temperature contours: (a) For the coarser mesh (32 nodes); (b) For the finer mesh (793 nodes).

### Multi-Material Steady-State Heat Conduction Problem

In the second example, we present a steady state problem of a rectangular plate of unity thickness compounded by two materials with different conductivities. The 2-D domain representing the plate was subdivided into two sub-domains where the triangular mesh was built independently for each sub-domain (representing each material), keeping the consistency of the mesh between them. First we analyze the plate with an interface between the materials which is placed in the middle of the plate and it is perpendicular to its base as shown in Fig. 7. Then we analyze a plate in which the interface between the two materials is semicircular as shown in Fig. 8.

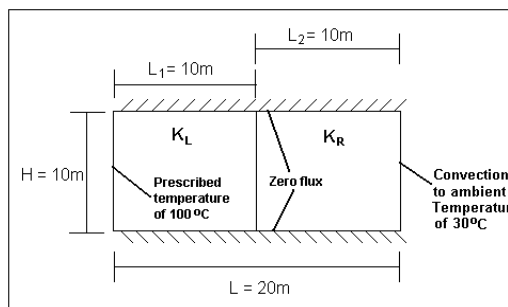


Figure 7. Domain for multi-material heat conduction problem with a perpendicular interface between the two materials.

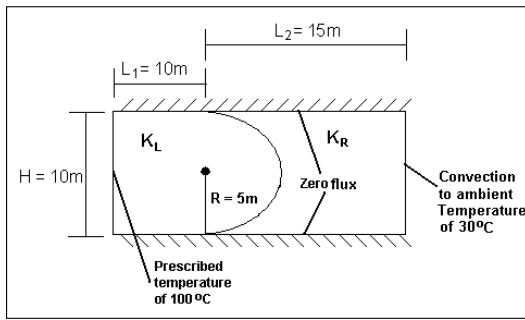


Figure 8. Domain for multi-material heat conduction problem with a semi-circular interface between both materials.

In both cases, with perpendicular and curvilinear interfaces, the plate is subjected to a prescribed temperature ( $T = 100\text{ }^\circ\text{C}$ ) on its left side and, the top and the bottom sides of the plate are insulated. On the right side, the plate is under convection, with the convection heat transfer coefficient  $\alpha_f = 100.0\text{ W/m}^2\text{C}$  and the ambient temperature of  $T_a = 30\text{ }^\circ\text{C}$ . The thermal conductivity of the left and right parts of the plate are, respectively,  $k_L = 50.0\text{ W/m}^\circ\text{C}$ ,  $k_R = 15.0\text{ W/m}^\circ\text{C}$ . The initial condition considered is a uniform temperature of  $T^i = 30\text{ }^\circ\text{C}$ .

For the first case (with a perpendicular interface between the two materials), a uniform triangulation with 216 nodes and 410 elements was adopted. Figures 9 and 10 show, respectively, the contours of temperature for this problem, and the temperature distribution for  $y = 0.0$  and  $0.0 \leq x \leq 20.0$ . In Figure 10, we can note the abrupt change in the slope of the temperature distribution due to the change in the conductivity coefficient of the two adjacent materials. In Table 2 we can see the good agreement achieved when we compare the nodal temperatures at the right side and at the interface between the two different materials, with the 1-D analytical solution.

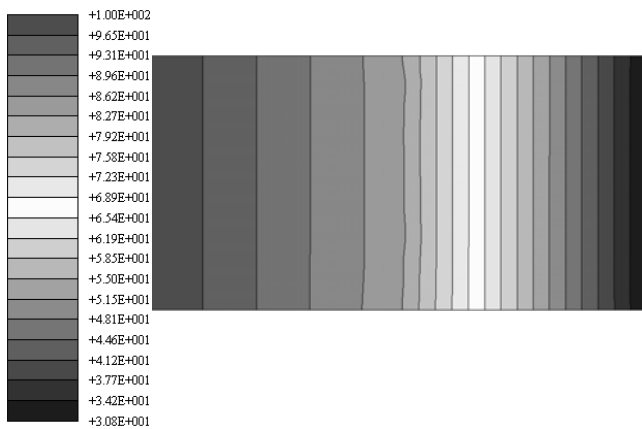


Figure 9. Temperature contours for multi-material heat conduction problem.

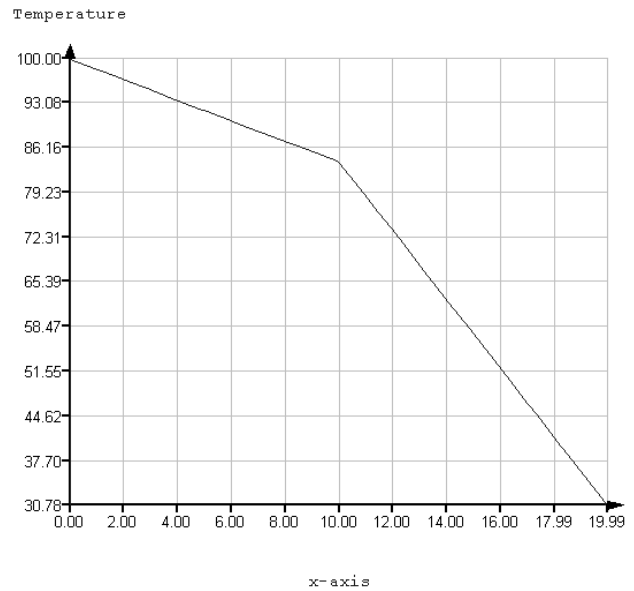


Figure 10. Temperature distribution over the  $y = 0.0$  axes for  $0.0 \leq x \leq 20.0$ , for the multi-material heat conduction problem with a perpendicular interface.

Table 2. Nodal temperatures at the right side and at the perpendicular interface between the two different materials.

x (m)	Analytical Solut. ( $^\circ\text{C}$ )	Numerical Solut. ( $^\circ\text{C}$ )
10	84.03	84.04
20	30.79	30.79

Figure 11 shows the heat flux vectors and the isolines of temperature for the case with a semi-circular interface between the two materials. In this figure we can readily verify the small value of the  $y$  component of the fluxes at the interface and the essentially 1-D behavior of the heat transfer when the distance from the interface increases. For this analysis a uniform triangular mesh with 303 nodes and 534 elements was used.

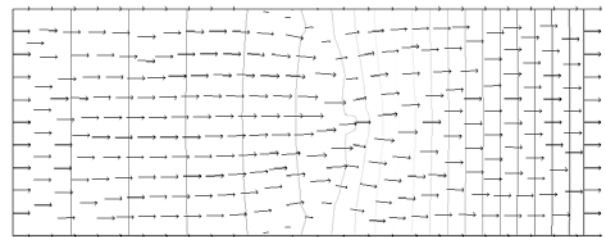


Figure 11. Triangular mesh and heat flux vectors for the multi-material heat conduction problem with a semi-circular interface.

In this latter case, the energy conservation (integral of the fluxes) was checked. For this purpose we computed the integral of the fluxes at the inflow (left) and outflow (right) faces of the plate, and the relative difference obtained was of 0.53% between the two faces.

### Transient Heat Transfer Through an Infinite Slab

In the final example, we show a transient one dimensional heat transfer problem in which the domain is represented by an infinite slab of unit thickness with length  $L$  in the  $x$  direction and of “infinite” length in the  $y$  direction. The slab is initially at a temperature of  $T^i = 15.6\text{ }^\circ\text{C}$ . A uniform distributed heat source,  $Q = 4.14 \times 10^5$



$W/m^3$  is applied over the entire domain. The thermal conductivity of the medium is  $k = 34.61 W/m^{\circ}C$ , its density is  $\rho = 8009.25 kg/m^3$ , and its specific heat is  $c = 8373.34 J/kg^{\circ}C$ . The surfaces in  $x = 0$  e  $x = L$  are kept at a temperature  $T_S = 0^{\circ}C$ . Figure 12 shows a sketch of the domain for this problem.

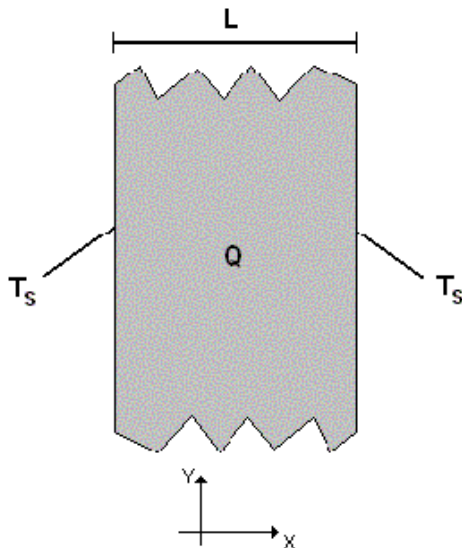


Figure 12. Slab of length  $x = L$  and "infinite" length in the  $y$  direction subjected to a uniform distributed heat source  $Q$ .

In order to represent the infinite slab and after some numerical experimentation we use a computational domain with  $2L$  length in the  $y$  direction. Due to the symmetry of the problem we simulate only a quarter of the slab. On the two symmetry axis (right vertical and lower horizontal faces of the computational domain) an adiabatic flux condition ( $q_n = 0.0 W/m^2$ ) is prescribed. On the left vertical face the prescribed temperature is  $T_S = 0^{\circ}C$ . Finally, on the upper horizontal face, which represents the surface of the domain truncated for numerical purposes, we also set a prescribed flux  $q_n = 0.0 W/m^2$ . The triangular mesh used in the simulation has 72 nodes and 112 elements. Figure 13 shows the mesh and the contours of temperature at instant  $t = 720$  sec.

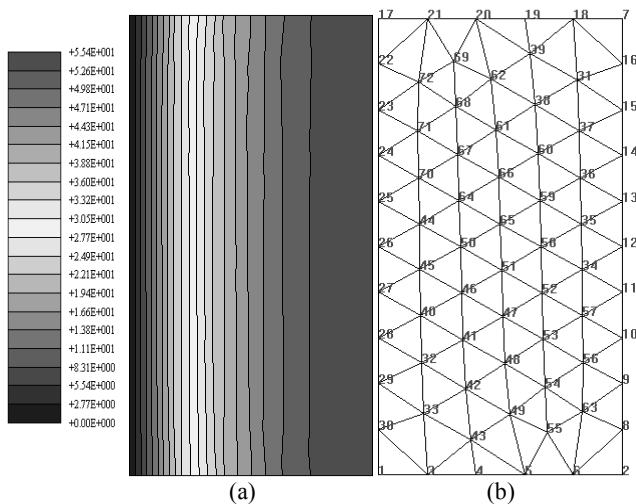


Figure 13. Infinite slab problem at  $t = 720$  sec.: (a) Contours of temperature; (b) Triangular mesh.

In order to verify the accuracy of our results, we compared them to the solution obtained with the commercial software "ANSYS", with a similar mesh density, and we also compared our results with the analytical solution obtained using up to three terms of the infinite series solution given in ANSYS verification manual (see ANSYS Internet address).

Tables 3 and 4 show, respectively, the temperatures and the temperatures ratios, obtained dividing ANSYS and finite volume solutions by the analytic one, i.e. (ANSYS/Analytic.) and (FVM/Analytic.). The results were obtained at time  $t = 720$  sec, for the collinear nodes 3, 4, 5, 6 and 2 localized at the left bottom side of Fig. 13b. As it can be observed, the results obtained with the FVM show very good agreement with the analytical solution and a comparable performance to that obtained with the finite element method (FEM) attained using the ANSYS software.

Table 3. Temperatures at five collinear nodes, obtained through different methods: Analytic, FEM (ANSYS), Finite Volume (FVM).

Nodes	Temperature ( $^{\circ}C$ ) at time $t = 720$ s		
	Analytic.	ANSYS	FVM
3	24.31	24.10	23.76
4	40.00	39.60	39.36
5	49.33	48.77	49.06
6	54.14	53.48	53.86
2	55.61	54.92	54.95

Table 4. Temperature ratios at five collinear nodes between ANSYS/Analytical and FVM/Analytical solutions.

Nodes	Temperature ratios at time $t = 720$ s	
	ANSYS/Analytic.	FVM/Analytic.
3	0.991	0.977
4	0.990	0.984
5	0.988	0.994
6	0.988	0.995
2	0.988	0.988

### Concluding Remarks

An unstructured finite volume formulation was fully derived to deal with potential problems involving different types of boundary conditions, thermal loads and multi-material properties. The adoption of an edge-based data structure renders the procedure very flexible, easy to implement, efficient and directly extendible to deal with any number of spatial dimension and for a broader class of CFD problems. Most of the formulation was presented in a general form and is valid for any spatial dimension and type of meshes. The whole system was validated for simple bidimensional model state-steady and transient problems using triangular meshes. However, the real potentiality of the developed numerical procedure must be exploited when solving more realistic problems. We are currently using this system to simulate the temperature distribution of bioheat transfer applications, such as hyperthermic treatment of inoperable tumors using laser heat sources (Lima et al., 2002 and Guimarães, 2003). In these applications the flexibilities for dealing with complex geometries, multi-materials, different thermal loads and boundary conditions are of paramount importance in order to have a good model of the physical features involved in the process. Several implementation and related numerical issues have been briefly discussed and further investigation will be pursued in the future referring to the incorporation of mesh adaptation and parallel implementation, which are of fundamental importance to address practical engineering applications.

## Acknowledgements

The authors would like to acknowledge the Brazilian Research Council CNPq and the National Petroleum Agency (ANP) for the financial support provided during the development of this research.

## References

- AGARD Report 787, 1992, "Special Course on Unstructured Grid Methods for Advection Dominated Flows", France.
- ANSYS Verification Manual. Internet Address: [http://www1.ansys.com/customer/content/documentation/60/Hlp\\_V\\_VMTOC.html](http://www1.ansys.com/customer/content/documentation/60/Hlp_V_VMTOC.html)
- Barlow, J. and Davies, G.A.O., 1987, "Selected FE Benchmarks in Structural and Thermal Analysis", NAFEMS (National Agency for Finite Element Methods & Standards), National Engineering Laboratory, Glasgow, U. K.
- Barth, T.J., 1992, "Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations", AGARD Report 787, pp. 6.1-6.61.
- Bonet, J. and Peraire, J., 1990, "An Alternating Digital Tree (ADT) Algorithm for 3-D Geometric Searching and Intersection Problems", International Journal for Numerical Methods in Engineering, Vol. 31, pp.1-17.
- Carvalho, D.K.E., 2001, "A Computational System for the Generation and Adaptation of Bidimensional Unstructured Meshes", Master Dissertation, Department of Mechanical Engineering, Federal University of Pernambuco, Brazil, DEMEC-UFPE, (In Portuguese).
- Crumpton, P.I., Moinier, P. and Giles, M.B.T.J., 1997, "An Unstructured Algorithm for High Reynolds Number Flows on Highly Stretched Grids". In: C. Taylor and J. T. Cross, (editors), Numerical Methods In: Laminar and Turbulent Flow, Pineridge Press, pp.561-572.
- Guimarães, C.S.C., 2003, "Computational Modeling of the Bioheat Transfer During the Hyperthermic Treatment of Duodenum Tumors Using an Unstructured Mesh Finite Volume Method". Master Dissertation, Mechanical Engineering, DEMEC-UFPE, (In Portuguese).
- Lima, R. de C.F. de, Lyra, P.R.M. and Guimarães, C.S.C., 2002, "A Technique to Compute Temperatures in Abdominal Tumors by the Solution of the Bioheat Transfer Equation Through the Use of The Finite Volume Method on Unstructured Meshes", LATCYM2002 – Nono Congreso Latino Americano de Transferencia de Calor y Materia, Porto Rico, november, pp. 1-13, in CD-ROM (In Portuguese).
- Lyra, P.R.M., 1994, "Unstructured Grid Adaptive Algorithms for Fluid Dynamics and Heat Conduction", Ph.D. thesis C/PH/182/94, University of Wales - Swansea.
- Lyra, P.R.M., Morgan, K., Peraire, J. and Peiró, J., 1994, "TVD Algorithms for the Solution of the Compressible Euler Equations on Unstructured Meshes", International Journal for Numerical Methods in Fluids, Vol. 19, pp. 827-847.
- Lyra, P.R.M., Willmersdorf, R.B., Martins, M.A.D. and Coutinho, A.L.G.A., 1998, "Parallel Implementation of Edge-Based Finite Element Schemes for Compressible Flows on Unstructured Grids", VECPAR'98 – Third International Meeting on Vector and Parallel Processing, Porto, Portugal, pp. 99-111.
- Lyra, P.R.M. and Carvalho, D.K.E., 2000, "A Flexible Unstructured Mesh Generator for Transient Anisotropic Remeshing", ECCOMAS 2000 – European Congress on Comp. Meth. in Applied Sciences and Eng., Barcelona-Espanha, september, published in CD-ROM.
- Lyra, P.R.M, Almeida, F.P.C., Willmersdorf, R. B., and Afonso, S.M.B., 2000, "An Adaptive FEM Integrated System for the Solution of Transient Thermal-Structural Problems", CILAMCE 2000 - Iberian Latin American Congress on Computational Methods in Engineering, Rio de Janeiro-Brasil, december, pp. 1-18, published in CD-ROM.
- Lyra, P.R.M., Lima, R.C.F., Guimarães, C.S.C. and De Carvalho, D.K.E., 2002, "An Edge-Based Unstructured Finite Volume Method for the Solution of Potential Problems", MECOM'2002 - First South American Congress on Computacional Mechanics, Parana and Santa Fé, Argentina, November, 2002, pp. 1-19, published in CD-ROM;
- Peraire, J., Vahdati, M., Morgan, K. and Zienkiewicz, O.C., 1987, "Adaptive Remeshing for Compressible Flow Computations", Journal of Computational Physics, Vol. 72, pp. 449-66.
- Peraire, J., Peiró, J. and Morgan, K., 1993, "Finite Element Multigrid Solution of Euler Flows Past Installed Aero-Engines", V.11.433-451, J. Computational Mechanics.
- Sorensen, K.A., Hassan, O., Morgan, K. and Weatherill, N. P., 1999, "An Agglomeration Unstructured Hybrid Mesh Method for 2D Turbulent Compressible Flows", ISCFD'99, Bremen, published in CD-ROM.
- Sorensen, K.A., 2001, "A Multigrid Procedure for the Solution of Compressible Fluid Flows on Unstructured Hybrid Meshes", Ph.D. thesis C/PH/251/01, University of Wales - Swansea.
- Sorensen, K.A., Hassan, O., Morgan, K. and Weatherill, N., 2001, "Agglomeration Multigrid on Hybrid Meshes for Compressible Viscous Flows", ECCOMAS European Computational Fluid Dynamics Conference, published in CD-ROM.
- TeCGraf– PUC-Rio. Eletronic Adress: <http://www.tecgraf.puc-rio.br>.
- Zienkiewicz, O.C. and Morgan, K., 1983, "Finite Element and Approximation", John Wiley & Sons, Inc.