Hittite Journal of Science and Engineering, 2015, 2 (2) 127-144
ISSN NUMBER: 2149-2123
DOI: 10.17350/HJSE19030000017

Numerical Solution of Two Phase Solidification Problem Using Dynamic Substructuring Based on Adaptive Error Estimation

Ozgur Uyar and Ata Mugan
Istanbul Technical University, Department of Mechanical Engineering, Gumussuyu, Istanbul, TURKEY

ABSTRACT

Numerical solution of solidification of metals with a sharp front, in particular solidification of lead, is investigated. Considering the fact that the associated CPU time and memory requirement may be costly for large domains, alternatives are searched. It is observed that using a substructuring technique with a local mesh refinement is promising. Following, by the use of an adaptive error estimation algorithm to find the location of solidification front and mushy zone, dynamic substructuring technique is developed to decrease the computational cost and to increase the accuracy of results. Superconvergent patch recovery technique is used to obtain the heat fluxes to evaluate the error energy norm of elements at each analysis step. Solidification front, mushy zone and elements having errors above a threshold value are captured with the error estimator. Then, elements having errors above the threshold value are refined by creating a substructure which is independent from the original global mesh. Equations of the global coarse mesh are augmented with the equations of the substructure. Employing the equations of the original coarse mesh help reduce the computational cost. Numerical solutions are presented and it is shown that the proposed approach has advantages over the alternative methods and, by the virtue of the adaptive error estimation algorithm, significantly decreases the CPU time of numerical solutions while it increases the accuracy of solutions and locates precisely the solidification front and mushy zone.

Key Words:
Solidification; Finite Element Methods; Substructuring; Stefan Problem; Error Estimation

Article History:
Received: 2015/08/09
Accepted: 2015/09/16
Online: 2015/12/30

***Correspondence to**: Ozgur Uyar, Istanbul Technical University, Faculty of Engineering, Department of Mechanical Engineering, Istanbul, Turkey*
Tel: +90 (312) 497-46 19
Fax: +90 (312) 497-43 01
E-Mail: ozuyar78@hotmail.com

INTRODUCTION

Solidification with a sharp front is a moving boundary value problem, i.e. a Stefan problem which is very important for several engineering applications. When the historical background of numerical solutions of solidification problems are examined, relevant works in literature are summarized as follows.

Lynch and O'Neill [1] developed a moving mesh technique for finite element phase change simulation. Lynch and Sullivan [2] calculated the heat flux at the phase boundary. Kuang and Atluri [3] developed another moving mesh method. Tamma and Saw [4] developed an adaptive mesh refinement for the p-version finite element method (FEM) to improve the solutions locally. Zabaras et al. [5] used front tracking FEM for calculation of temperature and stress fields in a solidfying pure metal to understand the formation of cracks due to the induced thermal stress field. Ro [6] investigated the development of heat transfer in phase change processes. Ghosh and Moorthy [7] proposed an alternative Arbitrary Lagrangian-Eulerian (ALE) approach to solidification problems. Wang et al. [8] examined a class of phase-field models for crystallization of pure substances from its melt. Gandin and Rappaz [9] developed a new algorithm based upon a 2-dimensional cellular automation technique for the simulation of dendritic grain formulation during solidification. Franca and Haghigi [10] developed a new adaptive finite element procedure for the solution of transient heat conduction problems. Juric and Tryggvason [11] presented a front-tracking method to simulate time dependent two-dimensional dendritic solidification of pure substances based on finite difference approximation of the heat equation and for explicit tracking of the liquid-solid interface. Chen et al. [12] developed a two-dimensional

finite element program which can be applied to solve the solidification of pure metals and alloys, that includes the effects of natural convection in liquid using temperature recovery scheme. Provatas et al. [13] studied the evolution of solidification microstructures using a phase-field model computed on an adaptive finite element grid. Lewis and Ravindran [14] studied the coupling of the solidification analysis based on the heat conduction equation with fluid flow and thermal analysis for metal casting where various algorithms available in literature for modeling of solidification problems are discussed. Merle and Dolbow [15] applied the eXtended finite element method (X-FEM) to thermal problems with moving heat sources and phase boundaries which provides accurate solutions to transient thermal and phase change problems on fixed finite element meshes. Chessa et al. [16] applied the X-FEM to multi-dimensional Stefan problems whose approximation represents the phase interface and associated discontinuity in temperature gradient within an element, and the phase interface can be evolved without re-meshing or the use of artificial heat capacity techniques. Ji et al. [17] presented the application of the hybrid XFEM/Level Set Method (LSM) to two dimensional solidification problems and used a new approach with the XFEM for this class of problems whereby the partition of unity is constructed with $\mathcal{C}^1(\Omega)$ polynomials and enriched with a $\mathcal{C}^0(\Omega)$ function. Zhao et al. [18] presented a two-dimensional model for simulation of the directional solidification of dendritic alloys and solved the transient energy and solute conservation equations using FEM discretizations; the energy equation was solved by a fixed mesh of bilinear elements in which the interface is tracked and the solute conservation equation is solved by an independent, variable mesh of quadratic triangular elements in the liquid phase only while the triangular mesh used in analyses is regenerated at each time step to accommodate the changes in the interface position using a Delaunay triangulation. Zang and Xu [19] developed a FEM model to compute the thermal and thermomechanical phenomena during pulsed laser induced melting and solidification where they used element removal and reactivation method in order to release and retrieve the stress and strain during melting and solidification. Takaki et al. [20] performed phase-field simulations during solidification of a binary alloy and used adaptive mesh refinement techniques during the FEM analysis in order to conduct the phase-field simulations effectively. Zabaras et al. [21] studied dendritic solidification of pure materials from an undercooled melt using the XFEM/LSM for modeling of the thermal problem and a volume averaged stabilized FEM formulation for modeling the fluid flow where they presented a dimension-independent methodology to simulate the growth of dendrites in the presence of convection. Their formulation is based on the XFEM/LSM to simulate the temperature evolution and a volume-averaged stabilized FEM formulation for the velocity evolution. Zhang et al. [22] developed an integrated meshless thermal–mechanical analysis system with a meshless solidification model based on Finite Point Method and the Meshless Local Petrov-Galerkin method based elastic–plastic analysis model where they calculated heat transfer and solidification using the finite point meshless method. Wang et al. [23] applied moving grid method for the solution of a phase-field model for dendritic growth in two- and three-dimensions where moving mesh technique is used and the mesh redistribution is realized by solving an elliptic boundary control problem together with a nonlinear multi-grid algorithm. Hu et al. [24] proposed a multi-mesh adaptive finite element method for simulating the dendritic growth in two- and three-dimensions and implemented the multi-mesh h-adaptive mesh refinement algorithm to enhance the computational efficiency. Lee and Sundararaghavan [25] studied a multi-scale analysis scheme for solidification based on two-scale computational homogenization and used a non-linear coupled macro-micro FEM model for addressing the fluid solidification problems and tracked solidifying interface using an adaptive meshing strategy. Li and Shopple [26] developed a new FEM level set approach to simulate the interface motion where they applied the method to the classical solidification problem to locate the dendrites whose key feature is the construction of an interface-fitted mesh and its unrefinement with respect to a fixed base mesh at each time step of evolution. O'Hara et al. [27] presented the application of the generalized finite element method (GFEM) with global-local enrichments to problems of transient heat transfer involving localized features where the GFEM is utilized in order to numerically construct general, specially-tailored shape functions yielding high levels of accuracy on coarse FEM meshes. Chen et al. [28] coupled the macro and micro analysis to predict the microstructure growth of magnesium alloys in directional solidification process and used the FEM to calculate undercooling temperature, by which the macro analysis results were coupled into the micro analysis where very fine mesh is used in order to calculate temperatures. Chen et al. [29] proposed a coupled Cellular Automation – FEM model to predict the grain structure formation during Gas Tungsten Arc Welding where the FEM is used to solve the heat flow problem based on an adaptive meshing. Ghoneim [30] used a new meshfree interface-finite element method for numerical modeling of isothermal solutal melting and solidification in binary systems where the implicitly represented liquid-solid interface is allowed to arbitrarily intersect the finite elements where meshfree solid-liquid interface interface nodes are generated automatically based on the distribution of the signed distance function.

When the above listed studies are examined, it is observed that computational methods used to solve the

Stefan problem can be classfied into the five classes such as the moving mesh method, adaptive remeshing method, XFEM, generalized FEM and meshless methods. Moving mesh methods update the mesh in order to conform the element edges to the solidification front; the mesh is updated totally or locally and mesh regeneration for complex interfaces is difficult. Adaptive remeshing methods use error estimators to determine the errors around solidification front and these algorithms refine the mesh by remeshing to reduce the error levels. On the other hand, the XFEM, generalized FEM and meshless methods are very popular in recent years for interface problems. These methods capture the discontinuities such as the solidification front by using LSM. Then, extra degrees of freedom (DOF) are added to the elements or extra nodes are added to the mesh in order to obtain improved solutions around the solidification front. Advantage of these methods is that the original mesh is not updated when the extra DOF or extra nodes are added to the original mesh; however, the associated matrices are to be updated at each analysis step that slows down the solution procedure.

All of the methods mentioned above require the update of associated matrices at each analysis step due to the updated mesh, refined mesh, adding extra DOF or adding nodes; as a result, the CPU time and memory requirement of these methods increase significantly.

Motivated by the drawbacks of above methods, a new technique using FEM is proposed called dynamic substructuring approach based on adaptive error estimation. In this method, the original FEM mesh is not changed during the solution steps and accurate solutions can be obtained using very coarse meshes. In particular, two phase Stefan problem of lead material is considered in this study. The computational domain is meshed with a coarse mesh and propagation of solidification front, mushy zone and elements having errors above a threshold value are captured with an error energy norm estimator at each analysis step. Then, a substructure independent from the global model is generated by refining the elements having errors above the error threshold; thus, remeshing of the entire original mesh at each analysis step is avoided that helps reduce the CPU time considerably. Substructure boundary conditions are obtained from the nodes of global model which are the neighbours to the elements used for the substructure. Following, the original global equations are augmented by the substructure equations and then the coupled equations of original and substructure equations are solved simultaneously. The mesh refinement is adaptive based on the adaptive error estimator calculations and the refined regions are removed in the substructure if their error level reduces to below the threshold value during analysis steps.

The main advantage of the proposed approach is that the substructure generation does not affect the original global matrices and a small portion of the coarse global mesh is refined automatically; thus, there is no need for remeshing the original coarse mesh at each analysis step. In sum, the original global mesh is kept unchanged during the whole analysis steps. Therefore, the proposed method does not employ any remeshing algorithm that slows down the numerical solutions that are followed in the moving mesh method, adaptive remeshing method, XFEM, generalized FEM and meshless methods in literature. Moreover, the initial global equations can be used throughout the solution procedure if associated FEM matrices are unchanged that further reduces the CPU time of the proposed approach. Numerical solutions are presented and it is shown that the proposed approach has advantages over the alternative methods and, by the virtue of the adaptive error estimation algorithm, significantly decreases the CPU time of numerical solutions while it increases the accuracy of solutions and locates precisely the solidification front and mushy zone.

Solidification and Mathematical Equations

During the solidification process, liquid metal changes its phase, conduction heat transfer occurs conduction heat transfer occurs in both solid phase and liquid phase. It also occurs between these two phases. Convective heat transfer occurs in liquid phase and at solidification front. In this study, heat convection motion of liquid metal is ignored. The mushy zone is also considered in all analyses. Solidification process is described mathematically with heat conduction and heat convection equations. Equations used in FEM analyses are presented below.

The transient heat transfer by conduction can be described by the following equation [10]

$$\rho C_p \frac{\partial T}{\partial t} = \nabla(\lambda \nabla \mathrm{T}) + Q \tag{1}$$

where ρ is the density, C_p specific heat, T temperature, t time, λ thermal conductivity and Q external heat source.

The heat transfer between solidification front and liquid metal occurs by convection associated with the release of latent heat of solidification that is represented by the following equation [31]

$$\lambda \frac{\partial T}{\partial x} = L\rho \frac{\partial \xi}{\partial t} + h\nabla \mathrm{T} \tag{2}$$

At the solidification front, temperature $\mathrm{T}(t, \xi(t))$ will satisfy the following condition [31]

$$T\left(t,\xi(t)\right)=T_c \tag{3}$$

where L is the latent heat of crystallization, ξ position of solidification front, h coefficient of heat convection at the solidification front interface, ∇T overheating and T_c crust temperature at which solidification starts.

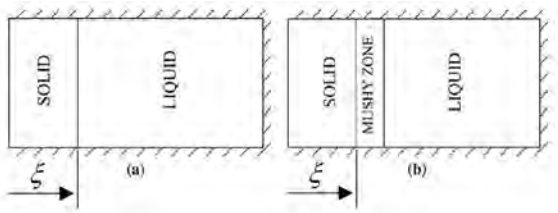

Figure 1. Solidification scheme: (a) pure metals and (b) alloys [32].

Figure 1 shows the process of solidification for pure metals and alloys schematically [32]. Solidification starts when the temperature of liquid metal (T_l) decreases to melting temperature (T_m) for pure metals. When the alloys are considered, a mushy zone starts to develop at melting temperature and solidification starts at the crust temperature. Mushy zone temperature varies between the crust temperature and melting temperature. In this zone, the liquid and solid phases are mixed. Figure 2 shows the phase change depending on temperature for alloys.

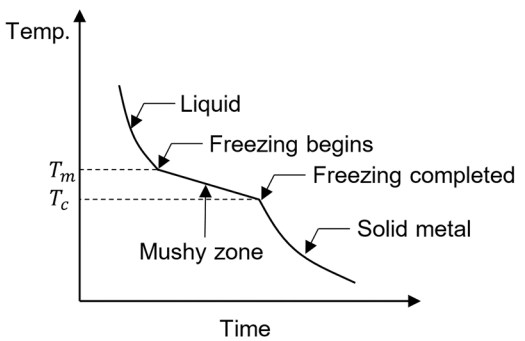

Figure 2. Phase change depending on temperature for alloys.

In this study, solidification characteristics of lead is studied with the proposed method. The experimental studies on the solidification of lead are completed using an experimental equipment by Bratu [31] where it is reported that the crust temperature of lead is 320°C and melting temperature of lead is 327°C.

Temperature dependent thermodynamic and thermophysical quantities of lead can be determined using the relationships given by Equations (4) to (6)

For $T \le 600.2\ K$

$$\lambda(T)=(-6E-8)T^3+(8E-5)T^2-0.0428T+42.89 \tag{4}$$

Equation (4) is obtained with the least squares curve fit to the tabular values given in [33]. This equation is valid from 123.2 K to 600.2 K. In addition, for $T \le 600.2\ K$, C_p and ρ are given by [34]

$$C_p=0.156391T^{-0.32702}e^{0.001089T}e^{-31.14/T} \tag{5}$$

$$\rho=11340\,kg/m^3 \tag{6}$$

Moreover, for $T > 600.2\ K$, we have [35]

$$\lambda(T)=0.011T+9.2 \tag{7}$$

$$C_p=176.2-(4.923E-2)T+(1.544E-5)T^2-(1.524E6)T^{-2} \tag{8}$$

$$\rho=11441-1.2795T \tag{9}$$

2-D Finite Element Formulation for Solidification Problem

The semi-discrete FEM equations for linear transient heat transfer problems can be cast into the following form [36]

$$[C]\{\dot{T}(t)\}+([K_c]+[K_h])\{T(t)\}=\{R_Q(T,t)\}+\{R_q(t)\}+\{R_h(t)\} \tag{10}$$

In this study, heat generation and heat flux are assumed to be zero and heat convection is considered only at the solidification front. Thus, the above formulation reduces to

$$[C]\{\dot{T}(t)\}+([K_c]+[K_h])\{T(t)\}=\{R_h(t)\} \tag{11}$$

where $[C]$ is the capacity matrix, $[K_c]$ conductivity matrix, $[K_h]$ convection matrix and $\{R_h(t)\}$ convective heat vector.

Time Integration

Equation (11) is a parabolic semi-discrete equation, and time dependent numerical solution of this equation can be obtained with the generalized trapezoidal family of methods which consists of the following equations [37]

$$C\dot{T}_{n+1}+KT_{n+1}=R_{n+1} \tag{12}$$

$$T_{n+1}=T_n+\Delta t\dot{T}_{n+\alpha} \tag{13}$$

$$\dot{T}_{n+\alpha}=\left[(1-\alpha)\dot{T}_n+\alpha\dot{T}_{n+1}\right] \tag{14}$$

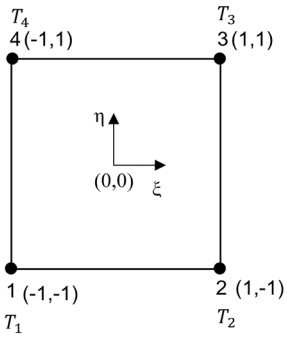

Figure 3. Four noded quadrilateral element in normal coordinates.

where T_n and $\dot{T}_n$ are respectively the FEM approximations of $T(t_n)$ and $\dot{T}(t_n)$ at discrete time steps, $R_{n+1} = R(t_{n+1})$, Δt is the time step assumed to be constant and α is a parameter in the interval [0, 1]. Note that α is chosen as 0.5 which corresponds to trapezoidal rule [37].

Finite Element Formulation

In this study, 4 noded quadrilateral elements in normal coordinates shown in Figure 3 are used for element formulations.

Bilinear shape functions of the element are given below [38]

$$N_1 = \frac{1}{4}(1-\xi)(1-\eta) \tag{15}$$

$$N_2 = \frac{1}{4}(1+\xi)(1-\eta) \tag{16}$$

$$N_3 = \frac{1}{4}(1+\xi)(1+\eta) \tag{17}$$

$$N_4 = \frac{1}{4}(1-\xi)(1+\eta) \tag{18}$$

The temperature at an arbitrary point inside an element can be approximated by

$$T(\xi,\eta) = N_1(\xi,\eta)T_1 + N_2(\xi,\eta)T_2 + N_3(\xi,\eta)T_3 + N_4(\xi,\eta)T_4 \tag{19}$$

When steady-state conditions do not prevail, temperature change in a unit volume of material is resisted by thermal mass that depends on the mass density ρ of the material and its specific heat C_p. The capacity matrix $[C]$ is built by assembling the element heat capacity matrices $[C_e]$ [39]; namely,

$$[C] = \sum_{(assemble)} [C]_e \text{ where } [C]_e = \int N^T N \rho C_p dV \tag{20}$$

In this study, lumped capacity matrix formulation is used in the developed FEM code. For a rectangular element, lumped capacity matrix is obtained by the row-sum technique [37]; namely, we simply divide the summation of heat capacity matrix $[C]_e$ components in a row by four and put the result along the diagonal of the same row of $[C]_e$. Thus, if we employ 4 noded rectangular elements, we may use the following for an internal node [40]

$$c_{ii} = \frac{\rho C_p A}{4} \tag{21}$$

in the main diagonal of the capacity matrix $[C]_e$. Thus, the lumped capacity matrix $[C]_e$ has the following diagonal form [40]

$$[C]_e = \frac{\rho c A}{4}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{22}$$

Such a diagonal capacity matrix provides considerable computational advantage because they are easy to store and invert. Then, element conductivity matrix $[K_c]_e$ can be evaluated as follows

$$[K_c]_e = \int_{A_e} [B]^T [\lambda][B] dA \tag{23}$$

where $[B]$ is the temperature differentiation matrix, $[\lambda]$ thermal conductivity matrix and A_e area of element as follows

$$[B] = \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x} & \frac{\partial N_4}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} & \frac{\partial N_4}{\partial y} \end{bmatrix} \tag{24}$$

$$[\lambda] = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \tag{25}$$

Following, element convection matrix can be expressed as follows [36]

$$[K_h]_e = \int_{S_e} h[N]^T [N] dS \tag{26}$$

where N is shape function matrix, h coefficient of convection and S_e length of convection edge of the element #e. The convective heat vector $\{R_h\}_e$ can be

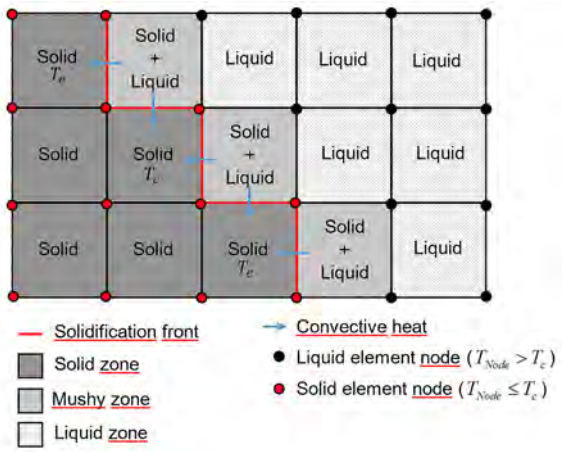

Figure 4. 2-D FEM solution scheme of two phase solidification problem.

calculated by [36]

$$\{R_h\}_e = \int_{S_e} hT_e [N]^T \, dS \tag{27}$$

where T_e is the environment temperature.

Note that heat convection equations are only considered at the solidification front. Hence, T_e value is equal to the average temperature of the neighboring solid element at the solidification front during analysis steps.

The 2-D FEM solution scheme of two phase solidification problem is shown in Figure 4. Note that all elements are liquid at the first analysis step. Temperatures of all nodes are checked after the calculations at each analysis step according to the algorithm described below and the location of solidification front can be determined by this methodology.

When the four nodes of an element are equal or below T_c that element is considered as solid. Solid elements, liquid elements and mixed phase elements are shown in Figure 4. Mixed phase elements located in the mushy zone are considered as liquid due to [31] and the location of solidification front is shown with red lines. Convective heat transfer is only considered at the solidification front (i.e., red edges of mixed phase elements) and it is shown with blue arrows in Figure 4. The environment temperature T_e for heat convection calculations is equal to the average temperature of solid element at the solidification front.

Thermodynamic and thermophsical properties (i.e., ρ, C_p and λ) of the material are temperature dependent which are calculated at the beginning of each analysis step depending on the average temperature of each element.

Adaptive Finite Element Recovery and A Posteriori Error Estimators

It is important to obtain the correct location of solidification front during the analyses due to the effect of heat flux jump at the front. The location of solidification front is assumed to be at element boundaries in this study. Therefore, refined meshes will give more accurate results than the coarse meshes.

The two phase solidification problems involve evolution of surfaces coupled with flux jump boundary conditions across the interfaces [25]. It is known that heat flux jumps cause errors during the numerical solution and these errors can be decreased to allowable levels by refining the mesh around the solidification front. To this end, adaptive finite element recovery techniques can be used. The adaptive error estimator developed by Zienkiewicz and Zhu allows the global error energy norm to be well estimated and also gives a good evaluation of local errors [41].

In an optimal mesh, it is desirable that the distribution of element error energy norm (i.e., $\|e\|_K$) should be almost constant for all elements. Thus, the total permissible error level is determined (assuming that it is found by numerical solutions) as follows [42]

$$\text{Permissible error} \equiv \bar{\eta}\|u\| \approx \bar{\eta}\left(\|q\|^2 + \|e\|^2\right)^{1/2} \tag{28}$$

Then, we could pose a requirement that the error norm in any element #*k* should satisfy

$$\|e\|_K < \bar{\eta}\left(\frac{\|q\|^2 + \|e\|^2}{m}\right)^{1/2} \equiv \bar{e}_m \tag{29}$$

where $\bar{\eta}$ is the permissible error percentage, $\|q\|$ heat flux energy norm, $\|e\|$ heat flux error energy norm and m number of elements.

Since no analytical solution is available for most of the practical problems, an estimated error is calculated based on a recovered solution. The exact error can be written as [43]

$$e_u^{ex} = u^{ex} - u^h \tag{30}$$

where u^{ex} and u^h are the exact and numerical solutions, respectively. Similarly, the heat flux error can be defined as follows

$$e = q^{ex} - q^h \tag{31}$$

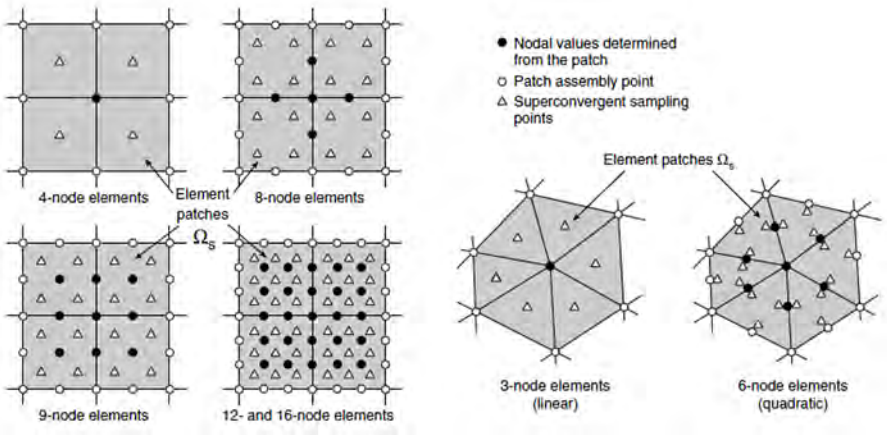

Figure 5. Interior superconvergent patches for quadrilateral elements (i.e., linear, quadratic and cubic) and triangles (i.e., linear and quadratic) [42].

where q^{ex} and q^{h} are the exact and numerical solutions of heat fluxes. Then, temperature error energy norm can be calculated by the following equation

$$\left\|e_u^{ex}\right\| = \left[\int_\Omega \left(u^{ex}-u^h\right)^T \lambda^{-1}\left(u^{ex}-u^h\right) d\Omega\right]^{1/2} \tag{32}$$

and heat flux error energy norm expression is as follows

$$\left\|e\right\| = \left[\int_\Omega \left(q^{ex}-q^h\right)^T \lambda^{-1}\left(q^{ex}-q^h\right) d\Omega\right]^{1/2} \tag{33}$$

Numerical heat fluxes at any point inside an element can be calculated by

$$q^h = \begin{Bmatrix} q_x \\ q_y \end{Bmatrix} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x} & \frac{\partial N_4}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} & \frac{\partial N_4}{\partial y} \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{Bmatrix} \tag{34}$$

Accurate heat flux values are to be calculated to determine the heat flux error. In this study, superconvergent patch recovery (SPR) technique is used for obtaining accurate heat flux values. The concept of superconvergence is that the approximate solutions at some points are more accurate, or in other words, the rate of convergence at those points is higher than those of other points [44]. The SPR technique is employed to recover the heat fluxes from Gauss integration points [37] by using element patches. On each patch, a polynomial expansion for each component of the recovered heat flux field is expressed in the following form [45].

$$\mathrm{q}_i^{ex} = \mathrm{p}(\mathrm{x},\mathrm{y})\,\mathrm{a}_i \tag{35}$$

where i denotes the axes (i.e., x or y axis), $\mathrm{p}(\mathrm{x},\mathrm{y})$ represents a polynomial basis and a_i are unknown coefficients. Usually, the polynomial basis is chosen equal to the non-extended FEM basis [37] for temperature field. A least squares approximation to the values of q^{ex} is evaluated at Gauss integration points of the elements within the patch. Detailed description of the SPR technique is given as follows [42]. Then,

$$p(x,y) = \left[1, \quad \bar{x}, \quad \bar{y}, \quad \ldots, \quad \bar{y}^p\right] \tag{36}$$

$$a_i = \left[a_1, \quad a_2, \quad a_3, \quad \ldots, \quad a_m\right]^T \tag{37}$$

with $\bar{x} = x - x_c$, $\bar{y} = x - y_c$ where x_c and y_c are the coordinates of the interior vertex node describing the patch, e.g., see Figure 5.

For each element patch, a least squares functional is minimized with n sampling points [42].

$$\Pi = \frac{1}{2}\sum_{k=1}^{n}\left[q^h(x_k,y_k) - p_k a_i\right]^2 \tag{38}$$

where

$$p_k = p(x_k,y_k) \tag{39}$$

and $[x_k,y_k]$ corresponds to the coordinates of the superconvergent sampling point k, that yields immediately the coefficients a_i as follows [42].

$$a_i = A^{-1}b_i \tag{40}$$

where

$$A = \sum_{k=1}^{n} p_k^T p_k \text{ and } b_i = \sum_{k=1}^{n} p_k^T q^h(x_k,y_k) \tag{41}$$

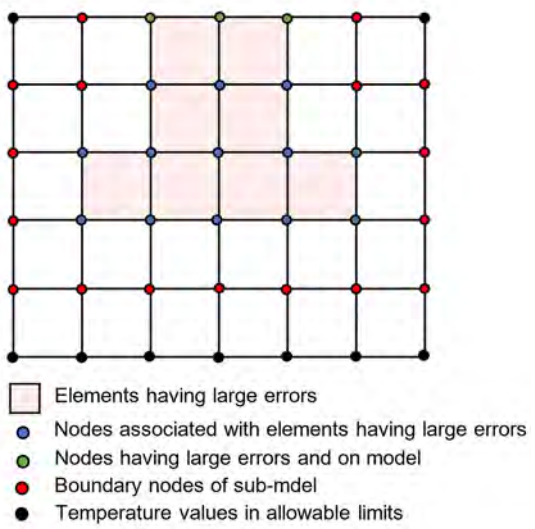

Figure 6. Global model with coarse mesh and elements having large errors.

In this study, center points of elements are selected as superconvergent sampling points and accurate heat flux values q^{ex} at nodes are determined using element patches. Numerical heat flux values q^{h} are determined via extrapolation of the heat flux values at Gauss integration points to the nodes.

Note that the value of permissible error percentage denoted by $\bar{\eta}$ is very important for adaptive finite element recovery. If $\bar{\eta}$ is selected smaller than an optimum value, more elements are captured by the error algorithm and total solution time increases. If $\bar{\eta}$ is selected larger than an optimum value, less number of elements are captured and accuracy of solution decreases.

Dynamic Substructuring Alagorithm

Approximation errors caused by the heat flux jump at the solidification front and discretization errors due to large element size can be decreased to allowable levels by mesh refinement around the solidification front. Refining all elements can be a remedy for this problem but its computational cost will be very high. In this study, only the elements having errors larger than the permissible error percentage $\bar{\eta}$ are refined to reduce the computational cost of problem. This refinement is performed by preparing a substructure and the original mesh of the global model is not affected by the mesh refinement. The original global equations are augmented with the substructure equations and this process is repeated at each analysis step. If the error level reduces to below the threshold value for certain elements at an analysis step, associated refined elements are removed in the substructure. Thus, the substructure is very effective for accurate numerical solutions.

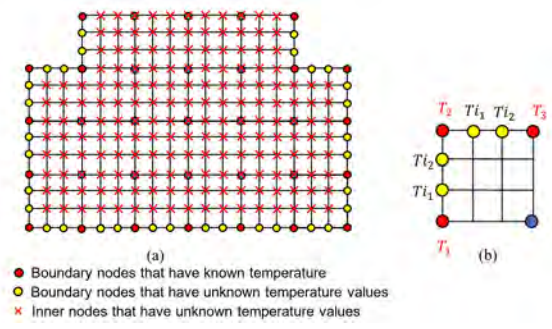

Figure 7. (a) Substructure and (b) interpolation of boundary temperatures..

Figures 6 and 7 show a representation of substructuring method. The global model with a coarse mesh is shown in Figure 6. At each solution step, errors of elements are calculated with the methodology given in Section 3.3. Elements having errors higher than permissible error percentage $\bar{\eta}$ are shown with red dotted texture in Figure 6. Nodes of these elements are shown with blue and green dots. Green dots indicate the nodes of elements having errors higher than permissible error percentage $\bar{\eta}$ on model boundary. Temperature values at blue and green nodes contain errors due to the coarse mesh and heat flux jump exists at the solidification front. Temperature values of black nodes and red nodes are in the allowable limits. Thus, red nodes which are neighboring nodes to elements having errors higher than permissible error percentage $\bar{\eta}$ are used as the boundary nodes of substructure and temperature values of these red nodes are applied to the substructure as boundary conditions. Our algorithm determines the elements having errors higher than permissible error percentage $\bar{\eta}$ and neighboring nodes to these elements at every analysis step.

The substructure given in Figure 7 (a) is created by the substructuring algoritm developed in this study. All codes are developed using Matlab environment. Number of element division in mesh refinement is selected by the user. In this example, each element is divided by 3x3 and 9 sub-elements are created inside each global element if that element has an error higher than permissible error percentage $\bar{\eta}$. Red nodes and yellow nodes represent the boundary nodes. The boundary conditions of substructure are applied to these nodes. Temperature values of red nodes are obtained from the global model. Temperature values of yellow nodes are determined by linear interpolation. Figure 7 (b) shows an element which has two shared edges with a substructure boundary. The temperatures T_1, T_2 and T_3 are obtained from the global model. Firstly, the algoritm determines the red nodes which are around yellow nodes. For example, yellow nodes on vertical edge are located between the red nodes having the temperatures of T_1 and T_2 , and yellow nodes on horizontal edge are located between

the red nodes having the temperatures of T_2 and T_3. Then, the algoritm calculates the temperatures Ti_1 and Ti_2 of the nodes located on vertical and horizontal edges with the following interpolation equations

$$Ti_i = \min(T_1, T_2) + \left[\frac{abs(T_1 - T_2)}{n}\right] * i \tag{42}$$

$$Ti_i = \min(T_2, T_3) + \left[\frac{abs(T_1 - T_2)}{n}\right] * i \quad i = 1, 2 \tag{43}$$

where n is the number of divisions on the corresponding edge of the element.

Temperature initial conditions are to be applied to the substructure nodes except the nodes lying on the boundaries before the solution process. At the first analysis step, global model initial temperatures are applied to the nodes and problem is solved. At the following steps, the algoritm checks the substructure of previous step. If there are common global elements at succeeding steps, calculated common nodal temperatures of the previous step are applied as the initial temperature to the common nodes of the current step.

Figure 9 (a) shows the global model and Figure 9 (b) shows the augmented model at the analysis step #1. Elements having errors higher than permissible error percentage $\bar{\eta}$

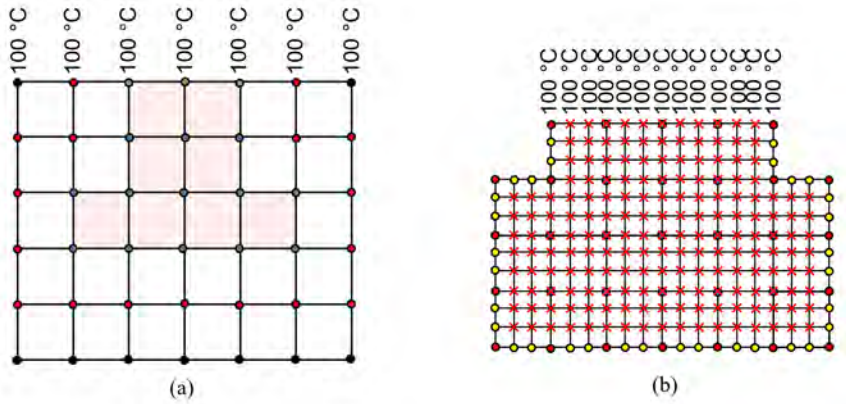

Figure 8. (a) Global model with boundary conditions and (b) substructure with global model boundary conditions.

After obtaining boundary conditions of the substructure, the capacity and stiffness matrices and heat vectors of the substructure are calculated. Following, the global model equations are augmented with substructure equations which is explained in Section 3.5.

The red cross symbols in Figure 7 (a) indicate the nodes having unknown temperature values. Temperatures of inner nodes are calculated by the generalized trapezoidal method at each analysis step. During the solution process, the location of solidification front is determined at each analysis step and heat convection is applied to elements which are neighbour to the solidification front.

Another important issue in the substructuring emerge when the substructure boundary nodes coincide with the global model boundary nodes. When such a condition occurs, the algoritm applies the global model boundary conditions to the coincident nodes of the substructure. Such a case is shown in Figure 8 where 100 °C temperature is applied to the top nodes of global model as boundary conditions. When the substructure is created, the same boundary condition is applied to the coincident nodes of the substructure.

and substructure boundary are determined with the global model solution (i.e., see Figure 9 (a)) at the end of analysis step #1. Then, the substructure is generated to reduce the error levels by following the method described above. Initial temperature of the global model is applied to internal nodes of the substructure as initial conditions because this is the first step of analyses. After the substructure is generated, the global model is augmented with it as shown in Figure 9 (b). This augmented model is solved in analysis step #1 again and unknown substructure temperature values shown by blue nodes are obtained.

After solving the first analysis step, the program increases the time step number and obtains the global model solution; then, elements having errors higher than permissible error percentage $\bar{\eta}$ and substructure boundary for the second analysis step are calculated (e.g., see Figure 10 (a)). Then, new substructure for the second analysis step is prepared. The method to obtain the initial temperatures of substructure is different for step #1 and for other steps. The program checks the existence of common elements having errors higher than permissible error percentage $\bar{\eta}$ between the current and previous analysis steps. If such elements are found, then the program automatically maps

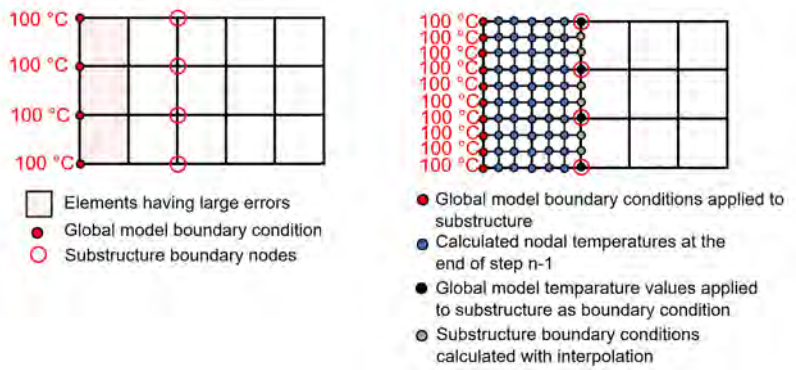

Figure 9. (a) Global model and elements having large errors at the end of step #1. (b) Substructure and calculated temperatures at the end of step #1.

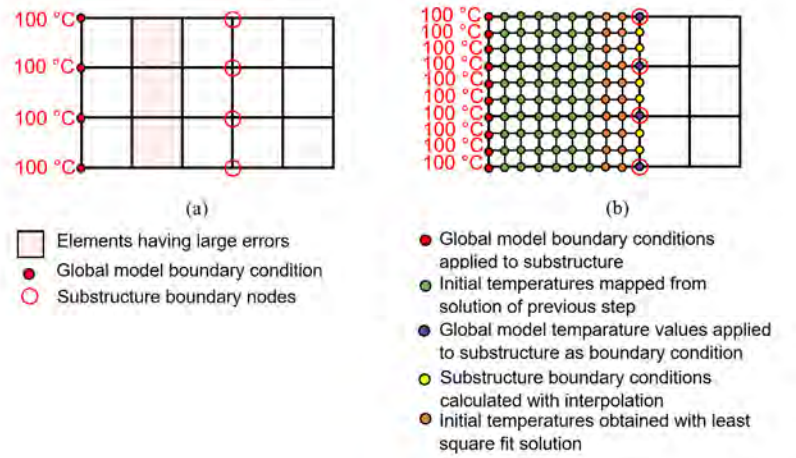

Figure 10. (a) Global model and elements having large errors at the end of step #2. (b) Substructure and initial temperatures at the beginning of step #2..

the temperatures calculated by the previous augmented model onto the common nodes of the current substructure. Figure 10 (b) shows the substructure at analysis step #2. It is seen that the green nodes in Figure 10 (b) are the common nodes with blue, black and grey nodes of the substructure in Figure 9 (b) at analysis step #1. Thus, temperature values of blue, black and grey nodes are mapped onto green nodes of the new substructure at the beginning of anaysis step #2.

Initial temperatures of orange nodes are also to be determined before the solution of augmented equation system. Temperature values of green nodes, purple nodes and yellow nodes in Figure 10 (b) are known. Subsequently, initial temperature values of orange nodes can be found by interpolation with the least square fit method given by

$$\{T\} = [P]\{\alpha\} \tag{46}$$

$$\{\alpha\} = [P]^{-1}\{T\} \tag{47}$$

where $T(x,y)$ is a polynominal curve fitting function for the temperature, the coefficients α_i are unknown, and x_i and y_i are the nodal coordinates. Equation (45) shows the matrix equation of polynominal temperature curve fitting function. The unknown coefficients α_i can be determined using Equation (47) and known temperature values which are those of the neighbouring nodes to orange nodes in Figure 10 (b) . After finding the coefficients α_i, unknown temperature values of orange nodes are calculated using Equation (44).

$$T(x,y) = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 x^2 + \alpha_4 xy + \alpha_5 y^2 + \alpha_6 x^3 + \alpha_7 x^2 y + \alpha_8 xy^2 + \alpha_9 y^3 \tag{44}$$

$$\begin{Bmatrix} T_1 \\ T_1 \\ \vdots \\ T_n \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1y_1 & y_1^2 & x_1^3 & x_1^2y_1 & x_1y_1^2 & y_1^3 \\ 1 & x_2 & y_2 & x_2^2 & x_2y_2 & y_2^2 & x_2^3 & x_2^2y_2 & x_2y_2^2 & y_2^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_n^2 & x_ny_n & y_n^2 & x_n^3 & x_n^2y_n & x_ny_n^2 & y_n^3 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{Bmatrix} \tag{45}$$

Augmentation of Substructuring Equations

Semi-discrete FEM equations of the original global model for transient two phase solidification problem can be written in the following form

$$[C]\{\dot{T}(t)\}+([K_c]+[K_h])\{T(t)\}=\{R_h(t)\} \quad (48)$$

Semi-discrete FEM equations for the substructuring can be writen in a similar fashion as follows

$$[C]_{sub}\{\dot{T}_{sub}(t)\}+([K_c]_{sub}+[K_h]_{sub})\{T_{sub}(t)\}=\{R_{hsub}(t)\} \quad (49)$$

These two matrix equations can be augmented as follows:

$$\begin{bmatrix}[C] & [0]\\ -[G] & [C]_{sub}\end{bmatrix}\begin{Bmatrix}\{\dot{T}(t)\}\\ \{\dot{T}_{sub}(t)\}\end{Bmatrix}+\begin{bmatrix}[K_c]+[K_h] & [0]\\ -[G] & [K_c]_{sub}+[K_h]_{sub}\end{bmatrix}\begin{Bmatrix}\{T(t)\}\\ \{T_{sub}(t)\}\end{Bmatrix}=\begin{Bmatrix}\{R_h(t)\}\\ \{R_{hsub}(t)\}\end{Bmatrix} \quad (50)$$

where $[C]$ is the global model capacity matrix, $[C]_{sub}$ is the substructure capacity matrix, $\{\dot{T}(t)\}$ is the global model temperature derivative vector, $\{\dot{T}_{sub}(t)\}$ is the substructure temperature derivative vector, $[K_c]$ and $[K_h]$ are the global model stiffness matrices, $[K_c]_{sub}$ and $[K_h]_{sub}$ are the substructure stiffness matrices, $\{T(t)\}$ is the global model temperature vector, $\{T_{sub}(t)\}$ is the substructure temperature vector, $\{R_h(t)\}$ is the global heat vector, $\{R_{h_{sub}}(t)\}$ is the substructure heat vector and $[G]$ is the substructure boundary condition matrix. All matrices and vectors except $[G]$ in Equation (50) can be calculated with the formulas given in Section 3.2. The matrix $[G]$ is formed considering the substructure boundaries, corresponding global model nodes and element refinement of global model elements. How to construct the matrix $[G]$ is below described for a simple domain.

Figure 11 shows the global model, substructure and augmented model for a rectangular domain where the substructure is located between the global model nodes 6, 7, 10 and 11. Then, substructure nodes 1, 4, 13 and 16 are coincident with the global model nodes 6, 7, 10 and 11. Before creating the matrix $[G]$, its size should be determined, that depends on the global model DOF and substructure DOF. Number of rows of $[G]$ is equal to the substructure DOF (m) and number of columns of $[G]$ is equal to the global model DOF (n); then, the size of $[G]$ will be mxn. It is shown in Figure 11 that total global model DOF is 16 and total substructure DOF is 16; then, the size of $[G]$ is 16x16 for this domain.

Internal nodes of the substructure have no connection with the boundary and nodes of the global model. Hence, the rows of the matrix $[G]$ which match with the internal node numbers will be zero and subsequently all matrix components which are in the rows 6, 7, 10 and 11 are zero.

The rows of the matrix $[G]$ which match with the corner node numbers will also be zero except for the columns of global model node numbers coincident with the corresponding substructure node numbers that will be equal to unity. The substructure nodes 1, 4, 13 and 16 are coincident with the global model nodes 6, 7, 10 and 11; thus, the components $G(1,6)$, $G(4,7)$, $G(13,10)$, $G(16,11)$ are equal to 1.

The rows of the matrix $[G]$ which match with the mid node numbers on the boundary will be zero except for the columns of global model node numbers coincident with the corresponding substructure corner node numbers. The substructure nodes 2 and 3 are located between the global model corner nodes 6 and 7. Thus, the components $G(2,6)$, $G(2,7)$, $G(3,6)$, $G(3,7)$ are not equal to zero. The substructure nodes 8 and 12 are located between the global model corner nodes 7 and 11. Thus, the components $G(8,7)$, $G(8,11)$, $G(12,7)$, $G(12,11)$ are not equal to zero. The

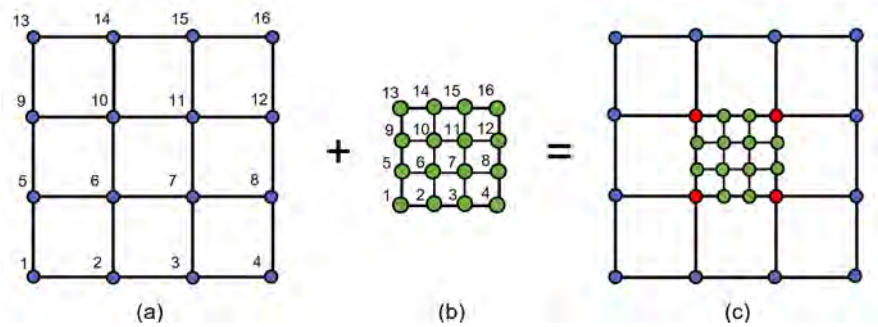

Figure 10. (a) Global model, (b) substructure and (c) augmented model.

substructure nodes 14 and 15 are located between the global model corner nodes 10 and 11. Thus, the components $G(14,10)$, $G(14,11)$, $G(15,10)$, $G(15,11)$ are not equal to zero. The substructure nodes 5 and 9 are located between the global model corner nodes 6 and 10. Thus, the components $G(5,6)$, $G(5,10)$, $G(9,6)$, $G(9,10)$ are not equal to zero. The values of non-zero components of $[G]$ are found with the following formula which is an interpolation equation

$$G(m,n)=1-\frac{distance\ of\ sub\ model\ node\ m\ to\ global\ model\ node\ n}{distance\ between\ global\ model\ corner\ nodes} \tag{51}$$

For example, the substructure node 2 is located between the global model corner nodes 6 and 7. Following, the components $G(2,6)$ and $G(2,7)$ can be calculated as follows

$$G(2,6)=1-\frac{distance\ of\ sub\ model\ node\ 2\ to\ global\ model\ node\ 6}{distance\ between\ global\ model\ corner\ nodes\ 6\ and\ 7}=1-\frac{1}{3}=\frac{2}{3} \tag{52}$$

$$G(2,7)=1-\frac{distance\ of\ sub\ model\ node\ 2\ to\ global\ model\ node\ 7}{distance\ between\ global\ model\ corner\ nodes\ 6\ and\ 7}=1-\frac{2}{3}=\frac{1}{3} \tag{53}$$

Then, the matrix $[G]$ for the sample domain shown in Figure 11 is given by

$$[G]=\begin{bmatrix}
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&2/3&1/3&0&0&0&0&0&0&0&0&\\
0&0&0&0&0&1/3&2/3&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&2/3&0&0&0&1/3&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&2/3&0&0&0&1/3&0&0&0&0&0\\
0&0&0&0&0&1/3&0&0&0&2/3&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1/3&0&0&0&2/3&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&2/3&1/3&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1/3&2/3&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0
\end{bmatrix} \tag{54}$$

Numerical Examples

In this section, two phase solidification problem of lead is solved in 2D using the proposed approach. Firstly, it is solved using a fine mesh and a coarse mesh to be used for comparisons. Then, dynamic substructuring technique is applied to the coarse mesh along with the adaptive error estimation algorithm. All examples are solved with a developed Matlab code on a PC having an Intel Core i7-4820 @ 3.70 GHz CPU, 16 GB RAM and Windows 8 operating system. Only one core of the CPU is utilized for the computations.

The computational domain has the length of 0.5 m and width of 0.5 m. Initial temperature of liquid lead is 337°C (T_i) and computational domain is cooled by applying 300°C temperature (T_b) to the left and bottom edges of the domain. Other edges of the domain are adiabatic (e.g., see Figure 12). Temperature variations of thermal conductivity, specific heat and density of the material are given by Equations (4) to (9), respectively. These properties of the lead material are calculated depending on the element average temperature at every analysis step.

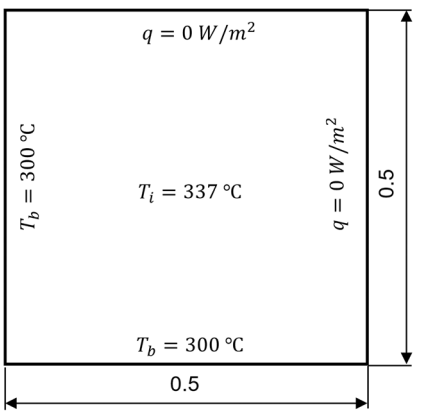

Figure 12. Geometry of 2-D computational domain with the boundary and initial conditions.

Total solution time is chosen as 2350 seconds in real time, the solution is completed at 2350 steps and constant time step of 1 second is used in numerical solutions. Figure 13 shows the temperature distribution in the coarse mesh (having the element division of 25x25) at the last analysis step and Figure 14 shows the temperature distribution in the fine mesh (having the element division of 75x75) at the last analysis step. The crust temperature of lead alloy is 320 °C and melting temperature is 327°C. Therefore, the green contours in both figures show the location of mushy zone and the boundary between the green contour and cyan contour shows the location of solidification front. If the solutions are examined, it is observed in the coarse mesh solution that almost all of the liquid metal is solidified at the time instant of 2350 seconds and the position of solidification front is not very clear. However, when the fine mesh is used, it is observed that the mushy zone is larger and position of solidification front is very clear.

Following, dynamic substructuring approach proposed in this paper is applied to the same solidification problem having the coarse mesh shown in Figure 13. Figures 15 (a) to (g) show the temperature distributions at different time instants during solidification. Each element is divided into 9 elements when generating the substructure, the error threshold $\bar{\eta}$ is selected to be 1.5 %, the density ρ is assumed to be constant for the solid phase and the coefficient of convection h is assumed to be constant at the solid-liquid interface whose value is obtained from Bratu, et al. [31] as 1600 $W/(m^2°C)$.

Note that total solution time of the fine mesh is 11,773 seconds. When dynamic substructuring approach based on adaptive error estimation is used with 1.5 % permissible error percentage $\bar{\eta}$, total solution time reduced to 9,141 s. The reduction in CPU times will be even further if the element matrices of the original global model are constant. Besides, the proposed dynamic substructuring approach requires less memory than the conventional fine mesh analysis. Moreover, the locations of solidification front and mushy zone are found very precisely by the proposed approach.

Temperature difference percentage between dynamic substructure results and fine mesh results at the time instant of 2350 seconds is given in Figure 16 in which it is observed that the maximum temperature difference percentage is 0.29 %. This accuracy level shows that the error estimation algoritm based dynamic substructuring technique works very well and this technique provides very good accuracy by decreasing the total CPU time.

The same problem is solved using different permissible error values $\bar{\eta}$ to compare the effect of this parameter on the accuracy and CPU times. To this end, selected $\bar{\eta}$ values are

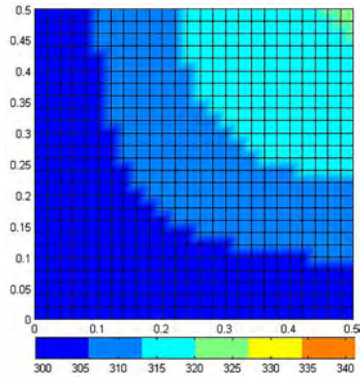

Figure 13. Coarse mesh temperature distribution at t=2350 seconds.

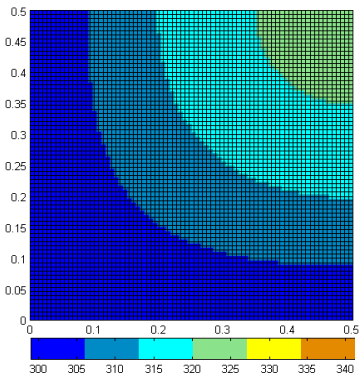

Figure 14. Fine mesh temperature distribution at t=2350 seconds.

1 %, 1.5 %, 2 % and 3 % and the corresponding CPU times are found as 13,913 sec., 9,141 sec., 7,312 sec., 5,813 sec., respectively. Figure 17 shows the temperature distributions as $\bar{\eta}$ changes at the time instant of 2350 sec. Note that the locations of solidification front and mushy zone can be found very precisely by the proposed approach for all values of $\bar{\eta}$. It is observed that permissible error percentage $\bar{\eta}$ value of 3 % yields acceptable results while the associated CPU time is almost half of that of the fine mesh.

DISCUSSION

Under the light of numerical results, it is observed that dynamic substructuring technique is more advantageous when compared with the other methods such as moving mesh method, adaptive remeshing method, XFEM method, GFEM method and meshless methods whose main drawbacks are listed below.

In moving mesh method, firstly solid liquid interface is tracked explicitly and then the mesh is deformed dynamically to align the element edges with the solidification front. Main advantage of this method is that number of nodes and elements are fixed; however, element sizes in the deformed region become larger during rearrangment of the nodes and this condition causes high element aspect ratios. New elements need to be created to decrease the element

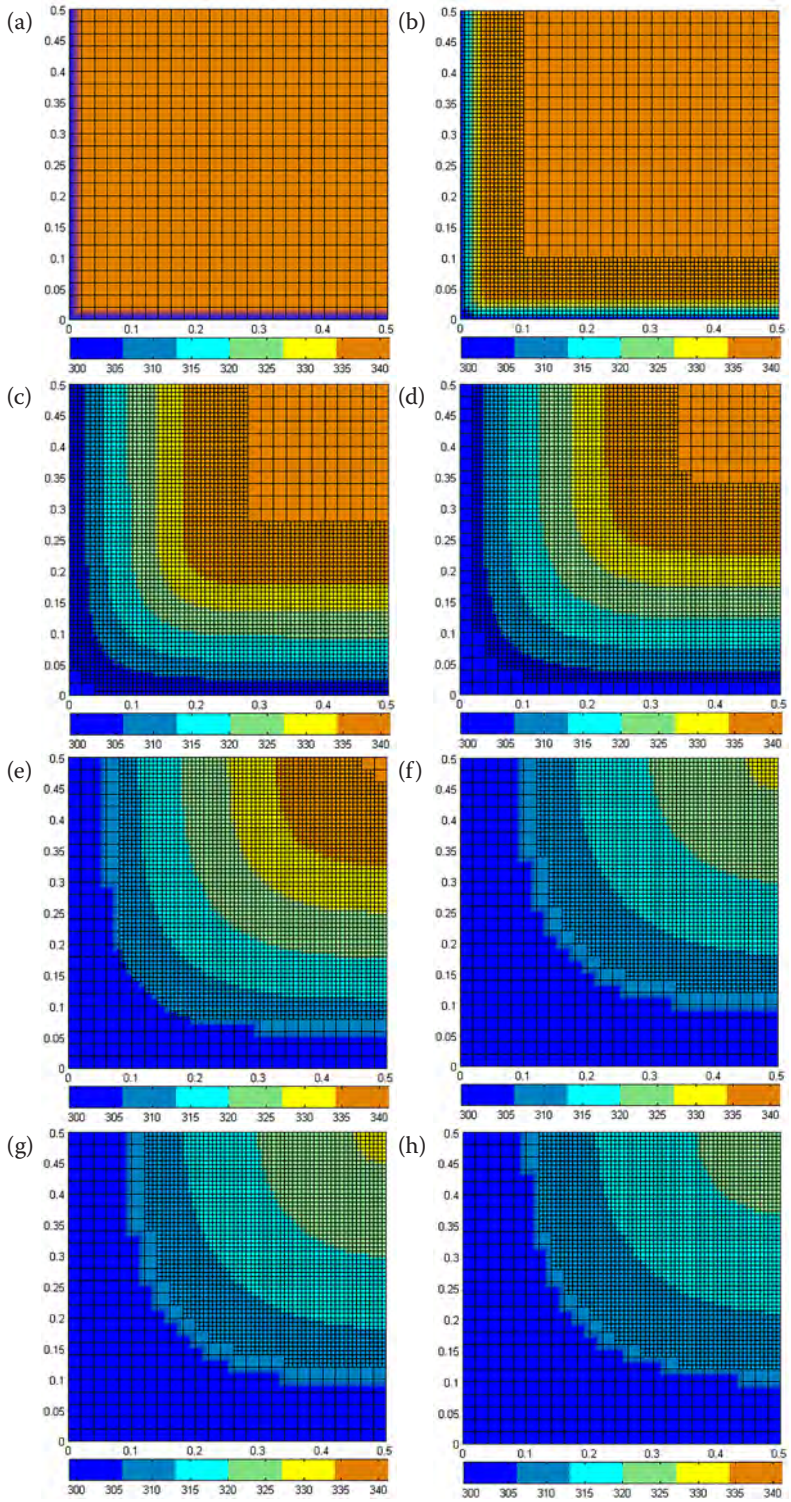

Figure 15. Temperature distributions found by dynamic substructuring approach at time instants of (a) 1 sec., (b) 10 sec., (c) 300 sec., (d) 500 sec., (e) 1000 sec., (f) 1500 sec., (g) 2000 sec., (h) 2350 sec.

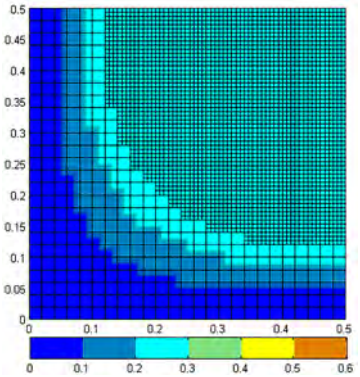

Figure 16. Temperature difference percentage (ΔT %) between dynamic substructure results and fine mesh results at the time instant of 2350 sec.

aspect ratios. During this process, global matrices are to be updated at each analysis step due to rearrangment of the node coordinates and also element sizes are to be checked at each analysis step, e.g., see [1], [2], [3], [5], [11], [12], [26]. In addition, this method is not suitable for alloys with finite freezing range and it is applicable to simple geometry and interface shapes only; appearing or disappearing phases and multiple interfaces cannot be handled by this method [16].

In adaptive remeshing method, the location of solidification front is determined with error estimators or interface tracking methods. After that the whole global domain is remeshed to align the element edges with the solidification front and a fine mesh is used at interface. During remeshing, old mesh data are to be mapped onto new mesh data for which special algorithms are required to obtain accurate solutions. In this method, global matrices are to be updated due to remeshing operation and also element numbers increase drastically because the whole domain is remeshed, e.g., see [10], [13], [18], [20], [24], [43].

The GFEM and XFEM determine the location of solidification front with interface tracking methods such as level set methods. After finding the location of solidification front, imaginary nodes are placed on interface and elements are split with these nodes. Mesh of the domain is not affected during these methods but global matrices are to be updated at each step due to imaginary nodes and extra DOF, e.g., see [15], [16], [17], [21], [27], [45].

Meshless methods determine the location of solidification front with interface tracking methods and then place points on interface to obtain accurate solutions at phase change location. Global matrices are to be updated at each analysis step due to the extra points placed on the solidification interface, e.g., see [22], [30].

All of these methods mentioned above focus on aligning the nodes to the solidification front except for error estimator based adaptive remeshing methods. Thus, very

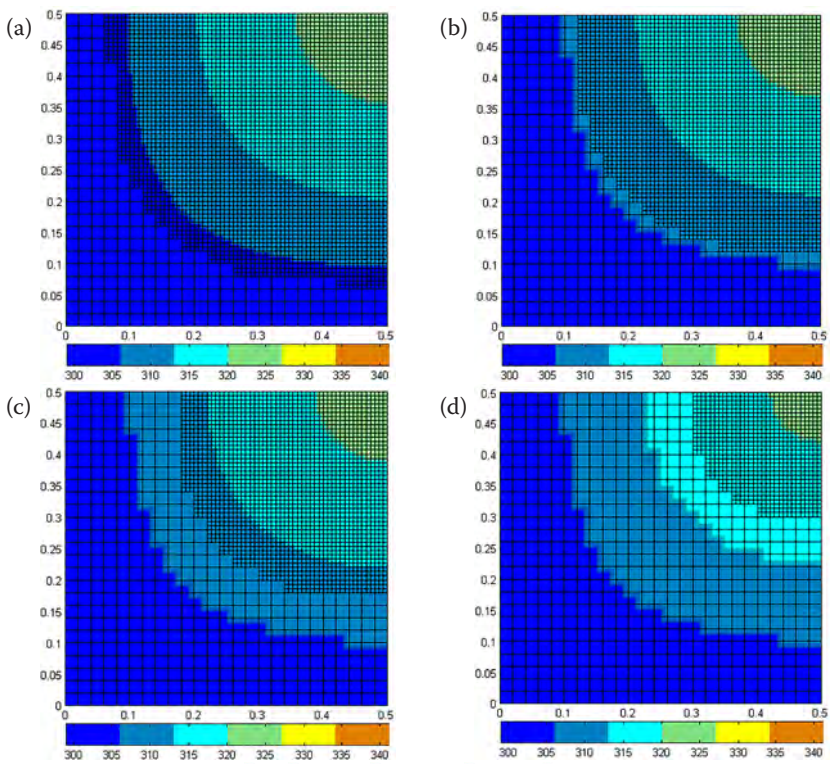

Figure 17. Temperature distributions depending on $\bar{\eta}$: (a) $\bar{\eta}$=1 %, (b) $\bar{\eta}$=1.5 %, (c) $\bar{\eta}$=2 %, (d) $\bar{\eta}$=3 %

accurate solutions around the solidification front can be obtained with these methods. However, all of these methods ignore mushy zone except for error estimator based adaptive remeshing method. One of the advantages of dynamic substructuring method arise at this point; error estimation algorithm automatically captures the solidification front and mushy zone due to high error ratios of these elements and this condition causes both the solidification front and mushy zone to be refined automatically and to get more accurate results at these zones.

On the other hand, error estimator based adaptive remeshing methods can also capture the mushy zone but their disadvantage is high CPU times because the whole domain is remeshed with refined elements in these methods. When the size of model geometry increases, the solution time also increases drastically. Another disadvantage of these methods compared with dynamic substructuring method is mesh compatibility. In these alternative methods during remeshing, old mesh data needs to be mapped onto new mesh data and special algorithms are required to obtain accurate solutions. If the local remeshing is applied to the domain, it is hard to connect the new generated refined elements to the coarse mesh of domain. This phenomenon does not occur at dynamic substructuring approach because the substructure is created explicitly and independent from the original global model. Thus, substructure domain elements can be refined freely.

There is no clear CPU time comparisions of alternative methods in literature given above except for meshless methods. Meshless methods are slower than classical FEM per reference [30]. But, when the alternative methods are considered, node numbers and element numbers of the global domain are increased to improve the accuracy and this condition causes the global matrices to be updated at each analysis step. When the computation domain gets larger, solution time will increase drastically. Another advantage of dynamic substructuring arises at this point. The original global model is not affected when dynamic substructuring approach is used. The substructure which is a smaller portion of the global model is prepared at each step. Thus, CPU times for substructure matrix preparation will be less time consuming when compared with alternative methods.

CONCLUSIONS

We presented a dynamic substructuring technique based on adaptive error estimation applied to solution of two phase solidification problems. The key feature of this method is the construction of a substructure using a locally refined mesh in the computational domain by the virtue of error estimates and augmentation of the governing equations by the equations of this substructure. One of the advantages of using such a dynamic substructuring technique is that high accuracy in solutions can be achieved while determining the location of solidification front and mushy zone with coarse meshes and calculating temperature distribution of computational domain. Another advantage of the proposed method is to reduce the CPU times significantly while obtaining very accurate results. The reduction in CPU times will be even further if the element matrices of the original global model are constant. To this end, the parameter of permissible error percentage $\bar{\eta}$ should be selected appropriately.

The proposed technique is applied to simulate the solidification of lead in a 2-dimensional domain. The same problem is also solved with the fine and coarse meshes for comparisons. Numerical results show that the error estimator accurately captures the numerical errors with the coarse meshes and the same accuracy of fine mesh can be achieved with the dynamic substructuring technique.

In comparison with the numerical results of fine mesh, dynamic substructuring technique has the advantage of fast computation as well as obtaining the same accuracy when solving the solidification problems, in particular in detection of solidification front and mushy zone. It is concluded that the proposed dynamic substructuring method based on adaptive error estimation is a potential numerical analysis tool for the analysis of solidification problems.

REFERENCES

1. O'Neill K and Lynch DR. A finite element solution for freezing problems, using a continuously deforming coordinate system. Numerical Methods in Heat Transfer. (1981) 215–231.

2. Lynch DR and Sullivan JM. Heat conversation in deforming element change simulation. Journal of Computational Physics. 57 (1985) 303–317.
3. Kuang Z and Atluri S. Temperature fields due to a moving heat source: A moving mesh finite element analysis. J. Appl. Mech. 52(2) (1985) 274-280.
4. Tamma K and Saw K. Hierarchical p–version finite elements and adaptive a posteriori computational formulations for two–dimensional thermal analysis. Comp. Struct. 32(5) (1989) 1183-1194.
5. Zabaras N, Ruan Y, Richmond O. Front tracking thermomechanical model for hypoelastic–viscoplastic behaviour in a solidifying body. Computer Methods in Applied Mechanics and Engineering. 81 (1990) 333–364.
6. Ro ST. Recent progress in heat transfer during melting and solidification processes. Transport Phenomena in Heat and Mass Transfer. (1992) 587–604.
7. Ghosh S and Moorthy S. An arbitrary Lagrangian–Eulerian finite element model for heat transfer analysis of solidification processes. Numer. Heat Trans. Part B. 23 (1993) 327-350.
8. Wang SL, Sekerka RF, Wheeler AA, Murray BT, Coriell SR, Braun RJ and McFadden GB. Thermodynamically–consistent phase–field models for solidification. Physica D 69 (1993) 189–200.
9. Gandin ChA and Rappaz M. A coupled finite element cellular automaton model for the prediction of dendritic grain structures in solidification processes. Acta Metall. Mater. 42 (1994) 2233–2246.
10. Franca AS and Haghigi K. Adaptive finite element analysis of transient thermal problems. Numerical Heat Transfer Part B. 26 (1994) 273–292.
11. Juric D and Tryggvason G. A front–tracking method for dendritic solidification. Journal of Computational Physics. 123 (1996) 127–148.
12. Chen Y, Im Y–T, Yoo J. Finite element analysis of solidification of aluminum with natural convection. Journal of Materials Processing Technology 52 (1995) 592–609.
13. Provatas N, Goldenfeld N and Dantzig J. Adaptive mesh refinement computation of solidification microstructures using dynamic data structures. Journal of Computational Physics. 148 (1999) 265–290.
14. Lewis RW and Ravindran K. Finite element simulation of metal casting. International Journal for Numerical Methods in Engineering. 47 (2000) 29–59.
15. Merle R and Dolbow J. Solving thermal and phase change problems with the extended finite element method. Computational Mechanics 28 (2002) 339–350.
16. Chessa J, Smolinski P and Belytschko T. The extended finite element method (xfem) for solidification problems. International Journal for Numerical Methods in Engineering. 53 (2002) 1959–1977.
17. Ji H, Chopp D and Dolbow JE. A hybrid extended finite element/level set method for modeling phase transformations. International Journal For Numerical Methods in Engineering. 54 (2002) 1209–1233.
18. Zhao P, Venere M, Heinrich JC, Poirier DR. Modeling dendritic growth of a binary alloy. Journal of Computational Physics. 188 (2003) 434–461.
19. Zhang XR and Xu X. Finite element analysis of pulsed laser bending: the effect of melting and solidification. Journal of Applied Mechanics 71 (2004) 321–326.
20. Takaki T, Fukuoka T, Tomita Y. Directional solidification of a binary alloy using adaptive finite element method. Journal of Crystal Growth. 283 (2005) 263–278.
21. Zabaras N, Ganapathysubramanian B, Tan L. Modelling dendritic solidification with melt convection using the extended finite element method. Journal of Computational Physics. 218 (2006) 200–227.
22. Zhang L, Shen H–F, Rong Y, Huang T–Y. Numerical simulation on solidification and thermal stress of continuous casting billet in mold based on meshless methods. Materials and Science Engineering. A466 (2007) 71–78.
23. Wang H, Li R, Tang T. Efficient computation of dendritic growth with r–adaptive finite element methods. Journal of Computational Physics. 227 (2008) 5984–6000.
24. Hu X, Li R and Tang T. A multi–mesh adaptive finite element approximation to phase field models. Communications in Computational Physics. 5 (2009) 1012–1029.
25. Lee S and Sundararaghavan V. Multi–scale homogenization of moving ınterface problems with flux jumps: Application to solidification. Comput Mech. 44 (2009) 297–307.
26. Bo Li and John Shopple. An Interface–Fitted Finite Element Level Set Method With Application to Solidification and Solvation. Commun. Comput. Phys. 10 (2011) 32–56.
27. P. O'Hara CA and Eason DT. Transient analysis of sharp thermal gradients using coarse finite element meshes. Comput. Methods Appl. Mech. Engrg. 200 (2011) 812–829.
28. Chen M, Hu X–D, Ju D–Y, Zhao H–Y. The microstructure prediction of magnesium alloy crystal growth in directional solidification. Computational Materials Science. 79 (2013) 684–690.
29. Chen S, Guillemot G and Gandin C–A. 3D coupled cellular automaton (ca) – finite element (fe) modeling for solidification grain structures in gas tungsten arc welding (GTAW). ISIJ International. 54 (2014) 401–407.
30. Ghoneim A. A meshfree ınterface–finite element method for modelling ısothermal solutal melting and solidification in binary systems. Finite Elements in Analysis and Design. 95 (2015) 20–41.
31. Bratu V, Mortici C, Oros C, Ghiban N. Mathematical model of solidification process in steel continuous casting into account the convective heat transfer at liquid–solid ınterface. Computational Materials Science. 94 (2014) 2–7.
32. Skrzypczak T, Wegrzyn–Skrzypczak E. Mathematical and numarical model of solidification process of pure metals. International Journal of Heat and Mass Transfer. 55 (2012) 4276–4284.
33. Ho CY, Powell RW andLiley PE. Thermal conductivity of the elements. J. Phys. Chem. Ref. Data 1 No. 2 (1972) 279–421.
34. Abu–Eishah SI, Haddad Y, Solieman A and Bajbouj A. A new correlation for the specific heat of metals, metal oxides and metal fluorides as a function of temperature. Latin American Applied Research. 34 (2004) 257–265.
35. Sobolev V. Database of thermophysical properties of liquid metal coolants for GEN–IV. Scientific Report of the Belgian Nuclear Research Centre. (2011).
36. Nikishkov G. Programming finite elements in java. Springer–Verlag, London. (2010).

37. Hughes TJR. The finite element method linear static and dynamic finite element analysis. Prentice–Hall. (1987).
38. Liu GR and Quek SS. The finite element method a practical course. Butterworth–Heinemann. (2003).
39. Cook RD. Finite element modeling for stress analysis, John Wiley and Sons. (1995). Rudolph Szilard. Theories and Applications of Plate Analysis. John Wiley and Sons. (2004).
40. Zienkiewicz OC and Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. International Journal For Numerical Methods in Engineering. 24 (1987) 337–357.Thomas J. R. Hughes. The Finite Element Method Linear Static and Dynamic Finite Element analysis, Prentice–Hall, 1987.
41. Zienkiewicz OC, Taylor RL and Zhu JZ. The finite element method: its basis and fundamentals, seventh ed. Butterworth–Heinemann is an imprint of Elsevier. (2013).
42. Rabizadeh E, Bagherzadeh AS, Rabczuk T. Adaptive thermo–mechanical finite element formulation based on goal–oriented error estimation. Computational Materials Science. 102 (2015) 27–44.
43. Khoei AR, Gharehbaghi SA. Three–dimensional data transfer operators in large plasticity deformations using modified–spr technique. Applied Mathematical Modelling. 33 (2009) 3269–3285.
44. Gonzales–Estrada OA, Rodenas JJ, Bordas SPA, Nadal E, Kerfriden P, Fuenmayor FJ. Locally equilibrated stress recovery for goal oriented error estimation in the extended finite element method. Computers and Structures. 152 (2015) 1–10.