University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

Mechanical & Materials Engineering Faculty
Publications

Mechanical & Materials Engineering,
Department of

9-2008

Finite Element Analysis Using Nonconforming Mesh

Ashok V. Kumar
University of Florida

Ravi Buria
University of Florida

Sanjeev Padmanabhan
Spatial Corp., Westminster, CO

Linxia Gu
University of Nebraska-Lincoln, gul@fit.edu

Follow this and additional works at: <https://digitalcommons.unl.edu/mechengfacpub>

Part of the [Mechanical Engineering Commons](#)

Kumar, Ashok V.; Buria, Ravi; Padmanabhan, Sanjeev; and Gu, Linxia, "Finite Element Analysis Using Nonconforming Mesh" (2008). *Mechanical & Materials Engineering Faculty Publications*. 44.
<https://digitalcommons.unl.edu/mechengfacpub/44>

This Article is brought to you for free and open access by the Mechanical & Materials Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in Mechanical & Materials Engineering Faculty Publications by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

Ashok V. Kumar
Associate Professor

Ravi Burla
Graduate Student

Department of Mechanical and Aerospace
Engineering,
University of Florida,
Gainesville, FL 32611

Sanjeev Padmanabhan
Spatial Corp.,
10955 Westmoor Drive,
Suite 425,
Westminster, CO 80021

Linxia Gu
South Dakota School of Mines and Technology,
501 East Saint Joseph Street,
Rapid City, SD 57701

Finite Element Analysis Using Nonconforming Mesh

A method for finite element analysis using a regular or structured grid is described that eliminates the need for generating conforming mesh for the geometry. The geometry of the domain is represented using implicit equations, which can be generated from traditional solid models. Solution structures are constructed using implicit equations such that the essential boundary conditions are satisfied exactly. This approach is used to solve boundary value problems arising in thermal and structural analysis. Convergence analysis is performed for several numerical examples and the results are compared with analytical and finite element analysis solutions to show that the method gives solutions that are similar to the finite element method in quality but is often less computationally expensive. Furthermore, by eliminating the need for mesh generation, better integration can be achieved between solid modeling and analysis stages of the design process.

[DOI: 10.1115/1.2956990]

1 Introduction

The finite element method (FEM) is widely used in industry as well as research for solving the problems arising in the engineering analysis. Traditional FEM requires a conforming mesh (Fig. 1(a)) and mesh generation algorithms have been developed that work acceptably for most 2D problems but are still unreliable for complex 3D geometries, often resulting in poor or distorted elements in some regions. In a typical mechanical component design cycle, a solid model of the design concept is created in computer-aided design (CAD) software and then exported to mesh generation or preprocessing software to create a finite element mesh. Often due to data incompatibility, the imported model contains only geometric information such as curves and surfaces and does not include all the topological information needed to define a solid model unambiguously. As a result, additional manual input is needed or sometimes the solid model is recreated in the mesh generation software. Several iterations of the above process are sometimes needed to come up with a satisfactory design for an engineering component. Mesh generation is therefore often the most time consuming process in the analysis, where significant amount of user intervention may be needed to create a mapped mesh and to correct problems related to distorted elements. A method for eliminating the need for a conforming mesh is presented here that uses a structured grid consisting of rectangular or regular hexahedral elements.

A number of meshless or meshfree analysis techniques have been proposed in the past two decades [1]. Meshless methods use a scattered set of nodes for the analysis but the nodes are not connected to form elements (Fig. 1(b)). The difference between the various meshless methods arises due to differences in the method of constructing the meshless approximations for the trial and test functions or due to the nature of the weak form used [2–13]. One of the popular meshless approximation schemes is based on moving least squares method. This and some other methods used to represent trial functions for the meshless approach do

not possess Kronecker delta properties. As a result, boundary conditions are difficult to apply precisely along the entire boundary.

An alternative approach to reduce mesh generation difficulties is to use nonconforming mesh, often a structured grid, to interpolate functions in the analysis domain. The geometry of the analysis domain must then be independently represented using the equations of the boundary curves or surfaces (Fig. 1(c)). It is easy to automate the generation of a structured grid since it does not have to conform to the boundaries of the geometry. If the equations of the boundaries curves/surfaces are available as implicit equations then they can be used to construct step function of solids that has a value of 1 inside the solid and 0 outside. Such a step function can be used to perform volume integration [14]. Furthermore, implicit equations of the boundaries of the domain can be used to construct solution structures that satisfy boundary conditions accurately. This approach was first proposed by Kantorovich and Krylov in 1958 [15]. They proposed a solution structure for applying essential boundary conditions as $u(x,y)=f(x,y)U(x,y)+u_0$, where $f(x,y)=0$ is the implicit equation of the boundary of the domain and is the essential boundary condition. $U(x,y)$ is the unknown function that is interpolated piecewise over a grid. This solution structure guarantees the satisfaction of essential boundary conditions. Shapiro and Tsukanov [16] described a method to construct a single implicit equation to represent the entire boundary of a solid using R -function technique, which was developed by Rvachev and Shieko [17]. Solution structures were constructed to satisfy all kinds of prescribed boundary conditions including essential, natural, and convection boundary conditions.

Belytschko et al. [18] proposed extended finite element method (X-FEM) based on a structured grid and implicit boundary representation. Approximate implicit function of the solid was constructed by fitting a set of sample points on the boundary. Radial basis functions were used for the implicit equation construction. Clark and Anderson [19] used the penalty method to perform analysis using a nonconforming mesh by constraining the weak form with a penalty factor to satisfy the prescribed essential boundary conditions.

In this paper, an analysis method is presented that uses a structured grid where the geometry of the analysis domain is represented using approximate step functions. The volume integrals in the weak form are evaluated using these step functions. A solution

Contributed by the Engineering Informatics (EIX) Committee of ASME for publication in the JOURNAL OF COMPUTING INFORMATION SCIENCE AND ENGINEERING. Manuscript received July 29, 2007; final manuscript received March 14, 2008; published online August 19, 2008. Assoc. Editor: J. Michopoulos.

Fig. 1 Analysis domain and boundary representations. (a) Conforming mesh in FEM (b) Scattered nodes in meshless methods. (c) Nonconforming structured grid methods.

structure that uses step functions of the boundary curves/surfaces is developed that enforces essential boundary conditions. This approach guarantees that all internal elements have identical stiffness matrix, thus eliminating the need for computing stiffness matrix individually for all elements as in traditional finite element method. Furthermore, since the elements in the grid are undistorted, numerical integration errors associated with Gauss quadrature are eliminated for internal elements. In addition to these advantages, a primary motivation for the work is to directly use geometry created in CAD systems for analysis and eliminate the need for replacing this accurate geometric model with a finite element mesh. By using a structured grid the need for generating such a conforming mesh is eliminated, significantly reducing the cost of building numerical models. In Sec. 2, a brief summary of the method for representing geometry and constructing step functions of solids is presented. Section 3 describes the method for constructing a solution structure so that the essential boundary conditions are exactly satisfied. Finite element formulation and evaluation of various terms in the weak form are discussed in Sec. 4. Several numerical examples involving thermal and structural analyses are presented in Sec. 5, which demonstrate the accuracy and efficiency of the method. The conclusions and inferences are provided in Sec. 6.

2 Representation of Geometry

Implicit equations, of the form $\Phi(\mathbf{x})=0$, provide a convenient way to represent the geometry of the boundaries of the analysis domain in structured grid methods. The function $\Phi(\mathbf{x})$ is referred to as the characteristic function. Belytschko et al. [18] used radial basis functions to define implicit equations of the boundary with the zero level set of the characteristic function representing the boundary. Shapiro and Tsukanov [16] used R -functions to represent the boundaries of the solid and used this definition for imposition of boundary conditions. In the level set method [20], it is a common practice to construct a signed distance function as the implicit function where the characteristic function $\Phi(\mathbf{x})$ is the distance from the boundary curve/surface and $\Phi(\mathbf{x}) < 0$ inside the domain. These distance functions are used to construct step functions for computing volume integrals.

Also in this paper, distance functions are used to define implicit equations of the boundary of the domain. Distance functions are needed only for elements through which a boundary passes and therefore they need to be constructed only within these elements by computing the distance from the point of interest to the nearest point on the boundary. A grid that overlaps the domain is created, as shown in Fig. 1(c). The grid elements are classified as internal, external, or boundary elements. External elements, whose nodes are all outside the analysis domain, are removed from the grid. All nodes of internal elements are inside the analysis domain, while the boundary elements have at least one boundary passing through

them. The characteristic equation of the boundary passing through a boundary element is defined by determining the nodal values of the distance from the boundary and then interpolating these values within the element. The shape functions used for interpolating the characteristic function of the boundary could be different from the shape functions used for interpolating the trial and test functions. Using distance function approximation, one can create an implicit equation for any type of curve or surface even if it is a free-form spline curve/surface. Since the distance function is computed directly from the exact geometry defined in the CAD system, it can be made as accurate as needed by increasing the order of the interpolation or increasing the resolution of the grid.

Distance functions provide a convenient method for representing individual boundaries but a typical analysis domain is defined using many boundaries. Each individual boundary is represented as a half-space with the characteristic function defined such that it has a positive value in the interior of the domain and a negative value outside. For elements that contain multiple boundaries the half-spaces represented by each boundary must be combined using boolean operations to define the region within the element that is interior to the analysis domain. Solid models of the analysis domain can be defined using constructive solid geometry (CSG) tree [21]. In a CSG tree, solids are defined as the intersection, union, or difference between sets of points defined by the half-spaces. As an example, the Boolean tree corresponding to a boundary element that contains three boundary curves is shown in Fig. 2. The interior of the analysis domain or the solid region is defined by the CSG tree shown in the figure as the intersection of these half-spaces.

To facilitate the construction of a single implicit function that represents the result of Boolean operations between several half-spaces, step functions are used [14]. Step function of a solid is defined to have a value of 1 inside the solid and 0 outside. An approximate step function of a half-space represented by the implicit equation $\Phi(x) \geq 0$ can be defined as follows:

Fig. 2 Boundary element with multiple curves and associated Boolean tree

$$h(\Phi) = \begin{cases} 1, & \Phi \geq \varepsilon, \\ \frac{1}{2} + \frac{\Phi}{2\varepsilon}, & -\varepsilon \leq \Phi \leq \varepsilon \\ 0, & \Phi \leq -\varepsilon \end{cases} \quad (1)$$

This step function can be used as a characteristic function to define the half-space as $h(\Phi) - 0.5 \geq 0$. The approximate step function tends to the exact Heaviside step function as the step size $\varepsilon \rightarrow 0$. If Φ_{Ai} , $i=1, \dots, n_A$, are half-spaces that define the boundaries of a convex solid A , then the step function of this solid can be defined as the product of the step functions of the half-spaces because the product of step functions yield the step function of their intersection.

$$h_A = \prod_{i=1}^{n_A} h(\Phi_{Ai}) \quad (2)$$

When two solids are combined using a Boolean operation, a new step function can be constructed to represent the Boolean result. Let h_A and h_B be the step functions of two solids A and B which are combined by Boolean operations. The step function of the Boolean result can be constructed as follows.

$$H_{A \cup B} = h_A + h_B - h_A h_B$$

$$H_{A - B} = h_A - h_A h_B \quad (3)$$

$$H_{A \cap B} = h_A h_B$$

In the above equations, $H_{A \cup B}$, $H_{A - B}$, and $H_{A \cap B}$ represent the step functions of the union, subtraction, and intersection, respectively, of the solids A and B . The step function of a solid has a value equal to 1 inside and 0 outside. The volume of the solid can therefore be computed by integrating the step function of the solid over a region Ω in the real space that fully encloses the solid. An arbitrary function can be integrated over the volume of the solid as follows.

$$V_F = \int_{\Omega} F(x, y) H_V d\Omega \quad (4)$$

The integral V_F is the volume integral of $F(x, y)$ over a volume, V , H_V is the step function of the domain of analysis V , and Ω is an arbitrary domain that includes the volume V . This method of computing volume integrals can be used to integrate the weak form over any arbitrary volume.

3 Implicit Boundary Method for Elliptic Boundary Value Problems

In this section, we discuss the solution structure for boundary value problems arising in steady state heat transfer and linear elasticity. The governing equation and boundary conditions for both these problems are similar, so a discussion for heat transfer problems is presented first and then extended for elasticity problems.

3.1 Steady State Heat Transfer. The governing differential equation for general steady state heat transfer problem shown in Fig. 3 is given as $\nabla \cdot (k \nabla T) + f = 0$ in Ω subjected to $T = \bar{T}$ on Γ_g and $k \nabla T \cdot \hat{\mathbf{n}} = q_h$ on Γ_h and the corresponding weak form is

$$\begin{aligned} \int_{\Omega} \nabla \delta T \cdot k \cdot \nabla T d\Omega &= \int_{\Omega} f \delta T d\Omega + \int_{\Gamma_h} q_h \delta T d\Gamma \\ &- \int_{\Gamma_c} h(T - T_{\infty}) \delta T d\Gamma \end{aligned} \quad (5)$$

In the above equation, Γ_g is the portion of the boundary with prescribed temperature while Γ_h is the portion with heat flux con-

Fig. 3 A general steady state heat transfer problem

ditions, Γ_c represents the portion with convection conditions, and T_{∞} is the ambient temperature. h is the convection coefficient, k is the thermal conductivity of the material, which is assumed to be isotropic for simplicity, and f is the heat source.

In order to use a structured grid that may not have nodes along the boundary of the domain, we construct a solution structure for temperature such that the boundary conditions are satisfied exactly. Although the following discussion is provided for 2D steady state heat transfer problems, the extension to 2D linear elasticity and 3D boundary value problems is straightforward. The solution structure for temperature is constructed as

$$T = T_s + T_a = DT_g + T_a \quad (6)$$

T_g is the grid variable defined by piecewise interpolation within the elements of the grid, $T_s = DT_g$ and T_a is a *boundary value function*, which has the specified value of boundary condition along the boundary. The function, D , referred to as the *essential boundary function* is defined using the distance function of the boundary such that

$$\begin{aligned} D &= 0 \quad \text{and } |\nabla D| \neq 0 \quad \text{on } \Gamma_g \\ D &> 0 \quad \text{elsewhere} \end{aligned} \quad (7)$$

This ensures that on the boundary Γ_g , $T_s = DT_g = 0$ and $T = T_a$. A method for constructing the essential boundary function is provided in Sec. 3.3. The function T_a is referred to as the boundary value function and it needs to be constructed such that on Γ_g , $T_a = \bar{T}$. By choosing the test function $\delta T = \delta T_a = D \delta T_g$ such that it vanishes on Γ_g and upon substituting the solution structure into Eq. (5), we get the following modified weak form.

$$\begin{aligned} \int_{\Omega} \nabla \delta T_s \cdot k \cdot \nabla T_s d\Omega &+ \int_{\Gamma_c} \delta T_s h T_s d\Gamma \\ &= - \int_{\Omega} \nabla \delta T_s \cdot k \cdot \nabla T_a d\Omega + \int_{\Omega} f \delta T_s d\Omega + \int_{\Gamma_h} q_h \delta T_s d\Gamma \\ &- \int_{\Gamma_c} h(T_a - T_{\infty}) \delta T_s d\Gamma \end{aligned} \quad (8)$$

The finite element formulation and the method for the evaluation of volume and boundary integrals are presented in a later section.

3.2 Linear Elasticity. The equation governing the linear elastic problem over a domain $\Omega \subset R^2$ bounded by Γ as shown in Fig. 4 is stated as $\nabla \cdot \mathbf{q} + \mathbf{b} = 0$ in Ω subjected to $\boldsymbol{\sigma} \cdot \hat{\mathbf{n}} = \mathbf{T}$ on Γ , and $\mathbf{u} = \bar{\mathbf{u}}$ on Γ_u . Here $\boldsymbol{\sigma}$ is the stress tensor, $\mathbf{b}$ is the body force vector, $\hat{\mathbf{n}}$ is the normal vector to the boundary of the domain, $\mathbf{T}$ is the applied traction vector on the boundary Γ , and $\bar{\mathbf{u}}$ is the prescribed displacement on Γ_u .

The weak form for this problem using the principle of virtual work is expressed as

Fig. 4 A general elastostatic problem

$$\int_{\Omega} \{\delta \boldsymbol{\varepsilon}\}^T \{\boldsymbol{\sigma}\} d\Omega = \int_{\Gamma_t} \{\delta \mathbf{u}\}^T \{\mathbf{T}\} d\Gamma + \int_{\Omega} \{\delta \mathbf{u}\}^T \{\mathbf{b}\} d\Omega \quad (9)$$

The virtual displacement and virtual strain vectors are represented in the above equation by $\delta \mathbf{u}$ and $\delta \boldsymbol{\varepsilon}$, respectively. The constitutive equation relating stress and strain is expressed as $\{\boldsymbol{\sigma}\} = [\mathbf{C}]\{\boldsymbol{\varepsilon}\}$, where $\boldsymbol{\varepsilon}$ is the strain expressed as a column matrix and $[\mathbf{C}]$ represents the stress-strain constitutive relation tensor.

Using the approach described in the previous subsection, the solution structure for displacement and virtual displacement is expressed below.

$$\mathbf{u} = \mathbf{u}_s + \mathbf{u}_a = \mathbf{D}\mathbf{u}_g + \mathbf{u}_a \quad (10)$$

$$\delta \mathbf{u} = \mathbf{D} \delta \mathbf{u}_g \quad (11)$$

In the preceding equations, $\mathbf{u}_s = \mathbf{D}\mathbf{u}_g$, $\mathbf{u}_g$ is the grid variable that is defined by interpolating nodal values within the elements of the grid and $\mathbf{u}_a$ is the boundary value function which is a vector field whose value at the boundary is equal to the prescribed boundary conditions. $\mathbf{D}$ is a diagonal matrix such that its diagonal components D_{ii} are essential boundary functions that satisfy all the properties listed in Eq. (7) on any boundary Γ_{ui} on which the i th component of displacement u_i is specified. The stress and strain tensors can be decomposed in the following form.

$$\{\boldsymbol{\sigma}\} = [\mathbf{C}]\{\boldsymbol{\varepsilon}_g\} + [\mathbf{C}]\{\boldsymbol{\varepsilon}_a\} = \{\boldsymbol{\sigma}_g\} + \{\boldsymbol{\sigma}_a\} \quad (12a)$$

For the 2D case, if the displacement vectors are $\mathbf{u}_s = \{u_s, v_s\}^T$ and $\mathbf{u}_a = \{u_a, v_a\}^T$, then the strain vectors can be defined as follows.

$$\{\boldsymbol{\varepsilon}_s\} = \begin{Bmatrix} \frac{\partial u_s}{\partial x} \\ \frac{\partial v_s}{\partial y} \\ \frac{\partial u_s}{\partial y} + \frac{\partial v_s}{\partial x} \end{Bmatrix} \quad \text{and} \quad \{\boldsymbol{\varepsilon}_a\} = \begin{Bmatrix} \frac{\partial u_a}{\partial x} \\ \frac{\partial v_a}{\partial y} \\ \frac{\partial u_a}{\partial y} + \frac{\partial v_a}{\partial x} \end{Bmatrix} \quad (12b)$$

When the decomposed stress and virtual strain tensors are substituted in the weak form it takes the following form, which forms the basis of the finite element implementation presented here.

$$\int_{\Omega} \{\delta \boldsymbol{\varepsilon}_s\}^T \{\boldsymbol{\sigma}_s\} d\Omega = \int_{\Gamma_t} \{\delta \mathbf{u}\}^T \{\mathbf{T}\} d\Gamma + \int_{\Omega} \{\delta \mathbf{u}\}^T \{\mathbf{b}\} d\Omega - \int_{\Omega} \{\delta \boldsymbol{\varepsilon}_a\}^T \{\boldsymbol{\sigma}_a\} d\Omega \quad (13)$$

3.3 Essential Boundary Function. The essential boundary function D must have a zero value on any boundary Γ_g that has an essential boundary condition specified. It must be constructed such that $|\nabla D| \neq 0$ on Γ_g to ensure that $|\nabla T|$ (or $|\nabla u|$) is not con-

Fig. 5 Essential boundary function

strained to be zero along the boundaries. Furthermore, it is necessary to ensure that $D(\mathbf{x}) > 0$, $\forall \mathbf{x} \in \Omega$ so that T (or u_i) is not constrained to be equal to $\bar{T}$ (or $\bar{u}_i$) anywhere within the domain of analysis.

A method for constructing essential boundary functions that satisfy these properties is described below.

$$D(\mathbf{x}) = \begin{cases} \prod_{j=1}^{n_b} d(\phi_j) & \text{if } H_{\Omega} < 1.0 \\ 1.0 & \text{if } H_{\Omega} = 1.0 \end{cases} \quad (14)$$

In the above definition, $\phi_j = 0$ is the implicit equation of the j th boundary on which essential boundary conditions are specified, n_b is the total number of such boundaries, and H_{Ω} is the approximate step function of the solid region in the boundary element. The functions $d(\phi_i)$ are constructed as

$$d(\phi) = \begin{cases} 1, & \phi > \varepsilon \\ -\frac{\phi^2}{\varepsilon^2} + 2\frac{\phi}{\varepsilon}, & 0 < \phi < \varepsilon \\ -\frac{\phi^2}{\varepsilon^2} - 2\frac{\phi}{\varepsilon}, & -\varepsilon < \phi < 0 \\ 1, & \phi < -\varepsilon \end{cases} \quad (15)$$

Figure 5 shows the plot of $d(\phi)$, which shows that this function is defined such that it is equal to zero at the boundary $\phi = 0$ and its value becomes equal to 1 outside the band $-\varepsilon < \phi < \varepsilon$. It has a non-negative value everywhere, a discontinuous nonzero slope at $\phi = 0$, and has a continuous zero slope at $\phi = \pm \varepsilon$. These properties of $d(\phi)$ ensure that D_i satisfies the requirements of essential boundary function listed in Eq. (7).

Note that the parameter ε , which defines the range over which $d(\phi)$ varies, is the same that is used to define step function $h(\phi)$ in Eq. (1) so that both D_i and H_{Ω} reach unit values at the same distance from the boundary. In addition, when $\varepsilon \rightarrow 0$, the D_i function reaches unit values very close to the boundary. This implies that for all the internal elements D_i has a value equal to unity and therefore the stiffness matrix of these elements is exactly identical to that of traditional finite elements. Furthermore, since a regular structured grid is used here for the analysis, all internal elements are of the same shape and size and have identical stiffness matrices. Therefore, it is sufficient to compute stiffness for any one internal element and use the same for all other internal elements, thereby saving computational time.

3.4 Boundary Value Function. The boundary value function T^a (or u_i^a) must be defined such that they have a value equal to the imposed essential boundary conditions on the appropriate bound-

Imposed Essential Boundary Value = ξ

Fig. 6 Boundary value function

aries [22]. While there is no unique method for constructing boundary value function, it is beneficial to ensure that this function is a polynomial of the same order as the shape functions used for interpolating the grid variable. This can be accomplished by constructing the boundary value function by piecewise interpolation using the element shape functions. The nodal values of T^a can be interpolated similar to the grid variable u^g using the grid element shape functions. The boundary value function within any element is interpolated as

$$T_a = \sum_i N_i T_i^a \quad (16)$$

In Eq. (16) N_i are the shape functions of the grid elements and T_i^a are the nodal values of T_a . The main advantage of constructing the boundary value function using the same shape functions is that within the domain, the grid variable can compensate for T_a in the solution structure to better approximate the exact solution. The solution structure will not be able to accurately represent constant strains, if T_a and T_g are not constructed using the same shape functions.

The nodes of the boundary elements should be assigned nodal values of T_a such that upon interpolating these values, the resulting function will have the desired values at the boundary passing through the element. This is very easy to achieve when the assigned value is constant or even linearly varying along the boundary. The rest of the nodes that are not part of any boundary element can have any arbitrary value and therefore can be set equal to zero. Figure 6 shows a structured grid for an arbitrary analysis domain. To apply the essential boundary condition $T=\xi$ on the portion of the boundary shown as bold, the nodal values of T_a is set equal to ξ at all the nodes shown in black color while at all other nodes the nodal values are set equal to zero. The boundary value function T_a contributes to the load computation on the right hand side of Eq. (8) for all those elements in which the gradient of T_a has a nonzero magnitude.

4 Finite Element Formulation

A finite element formulation is constructed by using the solution structure explained above in the weak forms. The grid variables are interpolated within each element using typical finite element shape functions that are based on Lagrange interpolation of the nodal values.

4.1 Steady State Heat Transfer. The nodal degrees of freedom for temperature are interpolated as $T_g = [N_1, \dots, N_N] \times \{T_{g1}, \dots, T_{gN}\}^T = [\mathbf{N}]\{\mathbf{T}_g\}$ where N_i are the shape functions used for the interpolation with the elements. The homogeneous part of the solution is therefore $\mathbf{T}_s = D[\mathbf{N}]\{\mathbf{T}_g\}$, where D is the essential boundary function. The boundary value function is constructed as $T_a = [N_1, \dots, N_N]\{T_{a1}, \dots, T_{aN}\}^T = [\mathbf{N}]\{\mathbf{T}_a\}$. The gradients of temperature, which are used in the weak form, are constructed as

Fig. 7 Representation of band in boundary elements having essential boundary

$$\nabla T_s = \left\{ \begin{matrix} \frac{\partial T_s}{\partial x} \\ \frac{\partial T_s}{\partial y} \end{matrix} \right\} = \left[\begin{matrix} D \frac{\partial N_1}{\partial x} + N_1 \frac{\partial D}{\partial x} & \dots & D \frac{\partial N_N}{\partial x} + N_N \frac{\partial D}{\partial x} \\ D \frac{\partial N_1}{\partial y} + N_1 \frac{\partial D}{\partial y} & \dots & D \frac{\partial N_N}{\partial y} + N_N \frac{\partial D}{\partial y} \end{matrix} \right] \left\{ \begin{matrix} T_{g1} \\ \vdots \\ T_{gN} \end{matrix} \right\} = [\bar{\mathbf{B}}]\{\mathbf{T}_g\} \quad (17)$$

The weak form takes the following form when the above approximations are incorporated into Eq. (8):

$$\begin{aligned} & \sum_{i=1}^{NE} \{\delta \mathbf{T}_g\}^T \int_{\Omega_e} [\bar{\mathbf{B}}]^T k [\bar{\mathbf{B}}] \{\mathbf{T}_g\} H_v d\Omega_e \\ & + \sum_{i=1}^{NBE} \{\delta \mathbf{T}_g\}^T \int_{\Gamma_c} [\mathbf{N}]^T h [\mathbf{N}] \{\mathbf{T}_g\} d\Gamma \\ & = - \sum_{i=1}^{NE} \{\delta \mathbf{T}_g\}^T \int_{\Omega_e} [\bar{\mathbf{B}}]^T k [\mathbf{B}] \{\mathbf{T}_a\} H_v d\Omega_e \\ & + \sum_{i=1}^{NE} \{\delta \mathbf{T}_g\}^T \int_{\Omega_e} [\mathbf{N}]^T f H_v d\Omega_e + \sum_{i=1}^{NBE} \{\delta \mathbf{T}_g\}^T \int_{\Gamma_h} [\mathbf{N}]^T q_0 d\Gamma \\ & - \sum_{i=1}^{NBE} \{\delta \mathbf{T}_g\}^T \int_{\Gamma_c} [\mathbf{N}]^T h (T_a - T_\infty) d\Gamma \end{aligned} \quad (18)$$

In the above expression, all the known quantities are moved to the right hand side and the unknown quantities are placed in the left hand side. NE is the total number elements in the grid and NBE is the number of boundary elements. The first term on the right hand side is the contribution to the load due to applied essential boundary conditions.

Very small value for the range parameter ε implies that the D_i function has very large gradients near the boundary. This requires special treatment for evaluating the stiffness matrix of the boundary elements that contain a boundary on which an essential boundary condition is applied. The following discussion illustrates the method for the computation of stiffness matrix for these elements using 2D heat transfer problem as an example and the corresponding boundary is shown in Fig. 7.

The gradient of temperature can be decomposed as follows:

$$\nabla T_s = ([\bar{\mathbf{B}}_1] + [\bar{\mathbf{B}}_2]) \left\{ \begin{matrix} T_{g1} \\ \vdots \\ T_{gN} \end{matrix} \right\} \quad \text{with}$$

$$[\bar{\mathbf{B}}_1] = \left[\begin{matrix} D \frac{\partial N_1}{\partial x} & \dots & D \frac{\partial N_N}{\partial x} \\ D \frac{\partial N_1}{\partial y} & \dots & D \frac{\partial N_N}{\partial y} \end{matrix} \right] \quad \text{and} \quad (19)$$

$$[\bar{\mathbf{B}}_2] = \begin{bmatrix} N_1 \frac{\partial D}{\partial x} & \cdots & N_N \frac{\partial D}{\partial x} \\ N_1 \frac{\partial D}{\partial y} & \cdots & N_N \frac{\partial D}{\partial y} \end{bmatrix}$$

Using the above decomposition, the stiffness matrix becomes the following:

$$[\mathbf{K}_e] = \int_{\Omega} [\bar{\mathbf{B}}_1]^T [\mathbf{C}] [\bar{\mathbf{B}}_1] H_V d\Omega + \int_{\Omega} [\bar{\mathbf{B}}_2]^T [\mathbf{C}] [\bar{\mathbf{B}}_2] H_V d\Omega + \int_{\Omega} ([\bar{\mathbf{B}}_1]^T [\mathbf{C}] [\bar{\mathbf{B}}_2] + [\bar{\mathbf{B}}_2]^T [\mathbf{C}] [\bar{\mathbf{B}}_1]) H_V d\Omega \quad (20a)$$

$$[\mathbf{K}_e] = [\mathbf{K}_1] + [\mathbf{K}_2] + [\mathbf{K}_3] \quad (20b)$$

Considering the $[\mathbf{K}_2]$ matrix alone and expanding the terms, we get

$$[\mathbf{K}_2] = \int_{\Omega} \begin{bmatrix} N_1 k N_1 & N_1 k N_2 & \cdots \\ N_2 k N_1 & N_2 k N_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \left(\left(\frac{\partial D}{\partial x} \right)^2 + \left(\frac{\partial D}{\partial y} \right)^2 \right) H_V d\Omega \quad (21)$$

Any (i, j) th term of $[\mathbf{K}_2]$ matrix can be written as

$$K_2(i, j) = \int_{\Omega} \left(\left(\frac{\partial D}{\partial x} \right)^2 + \left(\frac{\partial D}{\partial y} \right)^2 \right) k N_i N_j H_V d\Omega \quad (22)$$

Figure 7 shows the boundary of the domain within an element. The value of the essential boundary function varies from 0 to 1 over a narrow band near the boundary. When the parameter $\varepsilon \leq 10^{-5}$, it can be assumed that the element shape functions N_i 's are constant within the width of this narrow band. However, along the boundary, the shape functions vary. So the integral shown in Eq. (22) can be rewritten as

$$K_2(i, j) = \int_{\Gamma} k N_i N_j H_V \left(\int_{\Gamma} \left(\left(\frac{\partial D}{\partial x} \right)^2 + \left(\frac{\partial D}{\partial y} \right)^2 \right) dn \right) d\Gamma = \int_{\Gamma} k N_i N_j H_V A_{(\nabla D)^2} d\Gamma \quad (23)$$

Where dn is an infinitesimal increment in a direction normal to the boundary. In Eq. (23), the volume integral corresponding to the stiffness is converted into a surface integral term. The term $A_{(\nabla D)^2}$ is an integral of the square of the magnitude of ∇D over the narrow band. This term $(A_{(\nabla D)^2})$ can be evaluated analytically and is constant along the curve of the boundary. Similarly all terms in $[\mathbf{K}_3]$ matrix involve a combination of D function and its gradient and can be rewritten as surface integrals. Upon multiplying and expanding the $[\mathbf{K}_3]$ matrix, any (i, j) th term of $[\mathbf{K}_3]$ matrix can be written as

$$K_3(i, j) = \int_{\Omega} k \left(N_i \frac{\partial N_j}{\partial x} + N_j \frac{\partial N_i}{\partial x} \right) H_V D \frac{\partial D}{\partial x} d\Omega + \int_{\Omega} k \left(N_i \frac{\partial N_j}{\partial y} + N_j \frac{\partial N_i}{\partial y} \right) H_V D \frac{\partial D}{\partial y} d\Omega \quad (24)$$

Equation (24) can be written as surface integral containing terms involving area integrals of D and its gradient function.

$$K_3(i, j) = \int_{\Gamma} k \left(N_i \frac{\partial N_j}{\partial x} + N_j \frac{\partial N_i}{\partial x} \right) H_V A_{DD_x} d\Gamma + \int_{\Gamma} k \left(N_i \frac{\partial N_j}{\partial y} + N_j \frac{\partial N_i}{\partial y} \right) H_V A_{DD_y} d\Gamma \quad (25)$$

where $A_{DD_x} = \int D(\partial D / \partial x) dn$ and $A_{DD_y} = \int D(\partial D / \partial y) dn$. In this manner $[\mathbf{K}_2]$ and $[\mathbf{K}_3]$ matrices are evaluated as surface integrals. Their contributions are then added to the $[\mathbf{K}_1]$ matrix computed as volume integral using regular Gauss quadrature. This technique is efficient, effective and involves a lot less computational effort than directly computing the stiffness matrix $[\mathbf{K}_e]$ by volume integration.

The first term in the right hand side of Eq. (18) is also evaluated according to the above procedure. The surface integrals are computed using parametric integration technique, where the boundary is approximated using linear (lines in 2D or triangles in 3D) and the integration is performed on each segment using Gauss quadrature.

4.2 Linear Elasticity. In the following discussion, finite element formulation is developed for 2D elasticity problems using the solution structure described in Sec. 3. The extension of this formulation to 3D problems is straightforward. The nodal values of the grid variables are used to construct the homogeneous part of the solution, as shown in Eq. (26a). The boundary value functions are constructed, as shown in Eq. (26b). Using these interpolation schemes, the strain is decomposed as shown in Eq. (27) and the discretized weak form is shown in Eq. (28).

$$\begin{Bmatrix} u_s \\ v_s \end{Bmatrix} = \begin{bmatrix} D_u N_1 & 0 & \cdots & D_u N_N & 0 \\ 0 & D_v N_1 & \cdots & 0 & D_v N_N \end{bmatrix} \{ \mathbf{X}_g \} = [\bar{\mathbf{N}}] \{ \mathbf{X}_g \} \quad (26a)$$

$$\begin{Bmatrix} u^a \\ v^a \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & \cdots & N_N & 0 \\ 0 & N_1 & \cdots & 0 & N_N \end{bmatrix} \{ \mathbf{X}_a \} = [\mathbf{N}] \{ \mathbf{X}_a \} \quad (26b)$$

Where $\{ \mathbf{X}_g \}$ and $\{ \mathbf{X}_a \}$ are the nodal values of the grid variable and the boundary value functions, respectively.

$$\{ \mathbf{X}_g \} = \begin{Bmatrix} u_1^g \\ v_1^g \\ \vdots \\ u_N^g \\ v_N^g \end{Bmatrix} \quad \text{and} \quad \{ \mathbf{X}_a \} = \begin{Bmatrix} u_1^a \\ v_1^a \\ \vdots \\ u_N^a \\ v_N^a \end{Bmatrix} \quad (26c)$$

$$\varepsilon_s = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{Bmatrix} = ([\bar{\mathbf{B}}_1] + [\bar{\mathbf{B}}_2]) \begin{Bmatrix} u_1^g \\ v_1^g \\ \vdots \\ u_N^g \\ v_N^g \end{Bmatrix} = [\bar{\mathbf{B}}] \{ \mathbf{X}_g \} \quad (27)$$

$$[\bar{\mathbf{B}}_1] = \begin{bmatrix} D \frac{\partial N_1}{\partial x} & 0 & \cdots & D \frac{\partial N_N}{\partial x} & 0 \\ 0 & D \frac{\partial N_1}{\partial y} & \cdots & 0 & D \frac{\partial N_N}{\partial y} \\ D \frac{\partial N_1}{\partial y} & D \frac{\partial N_1}{\partial x} & \cdots & D \frac{\partial N_N}{\partial y} & D \frac{\partial N_N}{\partial x} \end{bmatrix} \quad (28)$$

Fig. 8 Rectangular plate with a constant heat flux

$$[\bar{\mathbf{B}}_2] = \begin{bmatrix} N_1 \frac{\partial D}{\partial x} & 0 & \cdots & N_N \frac{\partial D}{\partial x} & 0 \\ 0 & N_1 \frac{\partial D}{\partial y} & \cdots & 0 & N_N \frac{\partial D}{\partial y} \\ N_1 \frac{\partial D}{\partial y} & N_1 \frac{\partial D}{\partial x} & \cdots & N_N \frac{\partial D}{\partial y} & N_N \frac{\partial D}{\partial x} \end{bmatrix} \quad (29)$$

$$\sum_{i=1}^{NE} \{\delta \mathbf{x}_g\}^T \int_{\Omega_i} [\bar{\mathbf{B}}]^T [\mathbf{C}] [\bar{\mathbf{B}}] \{\mathbf{x}_g\} d\Omega = \sum_{i=1}^{NBE} \{\delta \mathbf{x}_g\}^T \int_{\Gamma_{ii}} \{\bar{\mathbf{N}}\} \{\mathbf{t}\} d\Gamma + \sum_{i=1}^{NE} \{\delta \mathbf{x}_g\}^T \int_{\Omega_i} \{\bar{\mathbf{N}}\} \{\mathbf{b}\} d\Omega - \sum_{i=1}^{NE} \{\delta \mathbf{x}_g\}^T \int_{\Omega_i} [\bar{\mathbf{B}}] \{\sigma_a\} d\Omega \quad (30)$$

In the above expression, NE is the number of elements in the grid and NBE is the number of boundary elements. The known quantities are moved to the right hand side while the unknown quantities are in the left hand side. The first term on the right hand side is the load due to applied traction. The left hand side represents the stiffness matrix, which is decomposed as in the heat transfer problem, and the integrals are evaluated in the same way as explained in the previous subsection.

5 Results and Discussion

The implicit boundary finite element method described in this paper was implemented by modifying a traditional finite element program. The shape of the analysis domain was defined using distance functions constructed directly from geometric/solid models. These distance functions, which are implicit equations of the boundary, were used for constructing the solution structure and for the computation of the stiffness matrix and load vectors. In this section we present few of the examples that were used to validate this approach. The first example is a plate under constant heat flux. While the problem itself is trivial, it is an important example that illustrates a few important aspects of the implicit boundary finite element method. One could think of this example as being equivalent to the patch test used for nonstandard finite elements to verify whether they satisfy completeness condition. However, unlike in the patch test, only structured grids are used for the analysis consistent with our objective of eliminating the need for conforming mesh that requires distorted elements

Example 5.1. A rectangular region of dimensions $1.0 \times 0.5 \text{ m}^2$ is subjected to a constant heat flux of 100 W/m^2 and is modeled as in Fig. 8. The conductivity of the plate k is assumed to be $1.0 \text{ W/m}^\circ\text{C}$. The left end of the plate is set at a constant temperature of 0°C . Since the heat flux is constant, temperature distribution is linear along the length of the plate and this problem can be modeled using a single four node bilinear finite element using the traditional approach. However, using implicit boundary method and its nonlinear solution structure it is not obvious that a single element can, in fact, provide a reasonable solution. To study this aspect, first, a model consisting of a single four node quad ele-

Fig. 9 Effect of range parameter on the solution

ment was constructed for analysis using the implicit boundary approach.

The essential boundary function D is constructed such that the value of D is equal to zero on the left edge so that the temperature is fixed along this edge. Since only homogeneous boundary conditions are applied, the boundary value function T_a is set equal to zero over the entire domain.

The analytical solution for this simple problem is a linear temperature field with a constant heat flux of $q_x = 100 \text{ W/m}^2$ in the X direction and $q_y = 0.0 \text{ W/m}^2$ in the Y direction. The temperature varies linearly from zero to the maximum temperature of $T_{\max} = 100^\circ\text{C}$ in the X direction. Figure 9 shows the analytical solution and the numerical solution obtained using implicit boundary method for various values of the range parameter ϵ used in computing D . The value of this parameter was varied from 1.0 to 0.0001 to study its effect on the accuracy of the computed temperature. Since this parameter determines the range over which the essential boundary functions vary nonlinearly from 0 to 1, large values of this parameter cause large errors. As the value of the range parameter is reduced, the nonlinearity of the solution is restricted to a small region and the results obtained are closer to the theoretical results. For values equal to or less than 0.001 the error within the element was found to be negligible. For the rest of the examples in this paper, $\epsilon = 1.0 \times 10^{-5}$ was used.

Example 5.2. The convergence behavior of the implicit boundary finite element method is studied using the example of a hollow cylinder subjected to temperature boundary conditions. This problem has been selected because an analytical solution is known, which can be used to determine the error in the computed solution. The inner surface has a fixed temperature of $T_0 = 30^\circ\text{C}$ while the outer surface has a fixed temperature of $T_1 = 50^\circ\text{C}$. Symmetry of the geometry is exploited to model only a quarter of the cylinder, as shown in Fig. 10, with the heat flux set equal to zero at the two symmetric edges. The analytical solution for this problem is $T = T_0 + [(T_1 - T_0) / \ln(R_2/R_0)] \ln(r/R_0)$. Three grids with varying densities as shown in Fig. 11 are considered for analyzing this model. The computed temperature distribution in the radial direction is plotted in Fig. 12. The percentage errors along the radial direction for each grid are plotted in Fig. 13. It can be seen that the solution obtained by our method converges to the analytical solution as the grid density is increased.

Example 5.3. This example involves the convergence study of four node quadrilateral elements in our method and its comparison with traditional FEM. A cantilever beam of length $2.54 \times 10^{-2} \text{ m}$ (1.0 in.) and of thickness $5.08 \times 10^{-3} \text{ m}$ (0.2 in.) is fixed at the one end and a uniform shear load of $-6.894757 \times 10^6 \text{ Pa}$

Fig. 10 Quarter cylinder used for modeling

(−1000 psi) is applied at the other end. Since the beam is very short, shear locking is not an issue and shear energy is not negligible as assumed in Bernoulli beam theory. The exact solution for this problem is the Timoshenko beam solution [23] shown in Eqs. (31a)–(31c) and a typical structured grid used for analysis is shown in Fig. 14. The geometry of the beam is very simple and therefore one could have used a conforming grid for this analysis. However, to study the convergence properties with nonconforming elements, a grid that extends beyond the geometry of the beam is used.

Exact solution for Timoshenko beam:

$$u = \frac{-Py}{6EI} \left[(6L - 3x)x + (2 + \nu) \left(y^2 - \frac{1}{4}D^2 \right) \right] \quad (31a)$$

$$v = \frac{P}{6EI} \left[3\nu y^2(L - x) + \frac{1}{4}(4 + 5\nu)D^2x + (3L - x)x^2 \right] \quad (31b)$$

Fig. 11 Grids used for convergence analysis

Fig. 12 Plot of temperature variation along the radial direction

Fig. 13 Error in computed temperature in the radial direction for different grids

$$\sigma_x = \frac{P}{I}(L - x)y, \quad \sigma_{xy} = \frac{P}{2I} \left(\frac{D^2}{4} - y^2 \right), \quad \sigma_y = 0 \quad (31c)$$

The L_2 error norm and energy error norms are defined in Eqs. (32a) and (32b) where u , ε , and σ represent displacement, strain, and stress, respectively, while the superscript e represents exact solution and h represents approximate solution.

$$\|L_2\| = \left(\int_{\Omega} (u^e - u^h)^T (u^e - u^h) d\Omega \right)^{1/2} \quad (32a)$$

$$\|E\| = \left(\frac{1}{2} \int_{\Omega} (\sigma^e - \sigma^h)^T (\varepsilon^e - \varepsilon^h) d\Omega \right)^{1/2} \quad (32b)$$

The L_2 error norm and energy error norms are computed for various grid densities and shown in Figs. 15 and 16, respectively. The plots are shown in a log scale, where the y-axis represents the error norm and the x-axis represents the number of nodes in the model. We can see that the convergence behavior of the four node quadrilateral (Q4) elements using implicit boundary method is comparable to that of Q4 elements using traditional FEM. Figure 17 shows that the analysis time taken using the implicit boundary finite element method (shown as IBM Q4) is smaller when compared to the analysis time for FEM. This is due to the fact that, the stiffness matrix need not be computed for each internal element when implicit boundary method is used because all internal ele-

Fig. 14 Cantilever beam with end loading and the analysis grid

Fig. 15 Plot of L_2 error norm with respect to the number of nodes

ments have the same stiffness matrix, whereas for FEM the stiffness matrix has to be computed separately for each element in the mesh.

Example 5.4. A support bracket with a riblike reinforcement, shown in Fig. 18, is clamped on one end while the other end is subjected to pressure in the Y direction. The magnitude of the pressure applied is 6.895×10^6 Pa (1000 psi). Young's modulus is assumed to be 2.068×10^{11} Pa (30×10^6 psi) and Poisson's ratio equal to 0.3.

A solid model for this structure was created using commercial solid modeling software and was used to construct the implicit representation of the geometry using distance functions. The results of this structural analysis obtained using eight-node hexahedral element and implicit boundary method (shown as IBM H8) are compared with the solution obtained by FEM (ABAQUS FE H8). Eight-noded hexahedral elements are used for the grid as well as for the finite element mesh. Convergence analysis for this problem is performed by computing strain energy for a series of models with varying grid/mesh densities. The results are plotted in Fig. 19. The plot is shown on a semilog scale, where the x -axis

Fig. 16 Plot of energy error norm with respect to the number of nodes

Fig. 17 Comparison of analysis time

represents the number of nodes and the y -axis represents the strain energy. The strain energy is computed by the following formula.

$$e = \sum_{N_e} \left(\int_{V_e} \frac{1}{2} (\boldsymbol{\sigma} \cdot \boldsymbol{\epsilon}) dV \right) \quad (33)$$

When the number of nodes is made larger both methods converge to the same solution. The plots for bending stresses computed using implicit boundary finite element method (IBM H8) are shown in Fig. 20 while the plot of bending stress obtained using FEM is shown in Fig. 21. The stress distribution obtained using both methods is identical and the corresponding maximum and minimum values match closely.

6 Conclusions

In this paper, a method for applying essential boundary conditions using implicit equations of the boundary is discussed that enables finite element type analysis using a nonconforming structured grid. The primary motivation for the method is the desire to eliminate the mesh generation process and use the solid model itself to represent the geometry instead of using a conforming mesh to approximate it. Generating a uniform structured grid that encloses the geometry is a very straightforward process. The method holds the potential to enable design engineers to perform analysis directly using solid models instead of first generating a mesh and then applying boundary conditions on the mesh. The internal elements are identical to each other and therefore have the

Fig. 18 A 3D rib structure

Fig. 19 Strain energy convergence comparison with FEM

same stiffness matrix. This provides computational efficiency compared to traditional FEM. Boundary element methods (BEMs) have been developed that are highly efficient for linear problems but no comparison has been made in this paper with BEM. Several numerical examples were solved, which demonstrate that the accuracy and convergence capabilities of the implicit boundary method are comparable to the traditional FEM. The method for applying boundary conditions here can be used with any interpolation or approximation scheme. Even though some meshless approximation methods [24] provide Kronecker's delta property,

most of these schemes as well as *B*-spline approximations lack this property. The implicit boundary method presented in this paper can be used with any of these approximation schemes to apply boundary conditions. In problems involving localized high stress gradients (or stress concentration) it is beneficial to have smaller elements in that neighborhood to provide higher resolution. This requires localized grid refinement while maintaining continuity of the displacements. A solution structure that enables such local refinement is currently being developed and will be the subject of a future paper.

Fig. 20 Plot of bending stresses using implicit boundary finite element method

Fig. 21 Plot of bending stresses using traditional FEM

Acknowledgment

The authors would like to acknowledge support from Air Force Research Laboratory, Munitions Directorate and the Department of Mechanical and Aerospace Engineering at the University of Florida.

References

- [1] Belytschko, T., Krongauz, Y., Organ, D., Fleming, M., and Krysl, P., 1996, "Meshless Methods: An Overview and Recent Developments," *Comput. Methods Appl. Mech. Eng.*, **139**, pp. 3–47.
- [2] Nayroles, B., Touzat, G., and Villon, P., 1992, "Generalizing the Finite Element Method: Diffuse Approximation and Diffuse Elements," *Comput. Mech.*, **10**, pp. 307–318.
- [3] Belytschko, T., Lu, Y. Y., and Gu, L., 1994, "Element Free Galerkin Methods," *Int. J. Numer. Methods Eng.*, **37**, pp. 229–256.
- [4] Atluri, S. N., and Zhu, T. L., 2000, "The Meshless Local Petrov-Galerkin (MLPG) Approach for Solving Problems in Elasto-Statics," *Arch. Technol.*, **25**, pp. 169–179.
- [5] Liu, G. R., and Gu, Y. T., 2001, "A Point Interpolation Method for Two-Dimensional Solids," *Int. J. Numer. Methods Eng.*, **50**, pp. 937–951.
- [6] Melenk, J. M., and Babuska, I., 1996, "The Partition of Unity Finite Element Method: Basic Theory and Applications," *Comput. Methods Appl. Mech. Eng.*, **139**, pp. 289–314.
- [7] De, S., and Bathe, K. J., 2000, "Method of Finite Spheres," *Comput. Mech.*, **25**, pp. 329–345.
- [8] De, S., and Bathe, K. J., 2001, "The Method of Finite Spheres with Improved Numerical Integration," *Comput. Struct.*, **79**, pp. 2183–2196.
- [9] Sukumar, N., Moran, B., and Belytschko, T., 1998, "The Natural Element Method in Solid Mechanics," *Int. J. Numer. Methods Eng.*, **43**, pp. 839–887.
- [10] Dolbow, J., and Belytschko, T., 1999, "Numerical Integration of the Galerkin Weak Form in Meshfree Methods," *Comput. Mech.*, **23**, pp. 219–230.
- [11] Gonzalez, D., Cueto, E., Martinez, M. A., and Doblare, M., 2004, "Numerical integration in Natural Neighbour Galerkin methods," *Int. J. Numer. Methods Eng.*, **60**, pp. 2077–2104.
- [12] Laguardia, J. J., Cueto, E., and Doblare, M., 2005, "A Natural Neighbour Galerkin Method with Quadtree Structure," *Int. J. Numer. Methods Eng.*, **63**, pp. 789–812.
- [13] Krongauz, Y., and Belytschko, T., 1996, "Enforcement of Essential Boundary Conditions in Meshless Approximations Using Finite Elements," *Comput. Methods Appl. Mech. Eng.*, **131**, pp. 133–145.
- [14] Kumar, A. V., and Lee, J. H., 2006, "Step Function Representation of Solid Models and Application to Mesh Free Engineering Analysis," *ASME J. Mech. Des.*, **128**(1), pp. 46–56.
- [15] Kantorovich, L. W., and Krylov, W. I., 1956, "Näherungsmethoden der Höheren Analysis," VEB Deutscher Verlag der Wissenschaften, Berlin.
- [16] Shapiro, V., and Tsukanov, I., 1999, "Meshfree Simulation of Deforming Domains," *Comput.-Aided Des.*, **31**, pp. 459–471.
- [17] Rvachev, V. L., and Shieko, T. I., 1995, "R-functions in Boundary Value Problems in Mechanics," *Appl. Mech. Rev.*, **48**, pp. 151–188.
- [18] Belytschko, T., Parimi, C., Moes, N., Sukumar, N., and Usui, S., 2003, "Structured Extended Finite Element Methods for Solids Defined by Implicit Surfaces," *Int. J. Numer. Methods Eng.*, **56**, pp. 609–635.
- [19] Clark, B. W., and Anderson, D. C., 2002, "Finite Element Analysis in 3D Using Penalty Boundary Method," *Proceedings of Design Engineering Technical Conferences*, Montreal, Canada.
- [20] Osher, S., and Fedkiw, R., 2002, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, New York.
- [21] Mortenson, M. E., 1997, *Geometric Modeling*, Wiley, New York.
- [22] Rvachev, V. L., Sheiko, T. I., Shapiro, V., and Tsukanov, I., 2001, "Transfinite Interpolation Over Implicitly Defined Sets," *Comput. Aided Geom. Des.*, **18**, pp. 195–220.
- [23] Timoshenko, S. P., and Goodier, J. N., 1969, *Theory of Elasticity*, 3rd ed., McGraw-Hill, New York.
- [24] Gu, Y. T., and Liu, G. R., 2001, "A Local Point Interpretation Method for Static and Dynamic Analysis of Thin Beams," *Comput. Methods Appl. Mech. Eng.*, **190**, pp. 515–552.