THEORETICAL AND APPLIED MECHANICS
Article in Press (2026) DOI: https://doi.org/10.2298/TAM260601007H

# DUAL-MESH CONTROL-DOMAIN ANALYSIS OF NONLINEAR PROBLEMS IN MECHANICS

**Tanmaye Yashodan Heblekar, J. N. Reddy, and Arun R. Srinivasa**

Abstract. The dual mesh control domain method, introduced by Reddy [23], is a hybrid numerical technique that combines features of the finite element and finite volume methods. It uses integral balance statements of governing principles, as in the finite volume method, together with systematic finite element interpolation of the dependent unknown functions, as in the finite element method. This combination provides a convenient and locally conservative framework for the spatial discretization of boundary-value problems. In this article, we present dual mesh control domain formulations for nonlinear problems arising in applied mechanics. The focus is on two representative classes of problems: the finite deformation of hyperelastic solids and the flow of viscous incompressible fluids in domains with moving boundaries. For the hyperelasticity problem, the governing equations are written in the reference configuration and solved using Newton linearization. For the moving-boundary flow problem, an Arbitrary Lagrangian–Eulerian formulation is used to account for the motion of the computational mesh. Numerical examples are presented to illustrate the implementation of the method and to assess the proposed formulations.

## 1. Introduction

The finite element method (FEM) and finite volume method (FVM) are among the most widely used numerical techniques for solving boundary- and initial-value problems in applied mechanics. Traditionally, the FEM has been more widely used in solid and structural mechanics, whereas the FVM enjoyed popularity in the computational fluid dynamics community. Each method has its own strengths. The FEM provides a systematic framework for approximating field variables using interpolation functions over elements which allow arbitrary and non-structured meshes, while the FVM is based on satisfying integral balance statements over

2020 *Mathematics Subject Classification:* Primary 65N30; Secondary 74B20, 76D05, 74F10.
*Key words and phrases:* dual mesh control domain method; nonlinear continuum mechanics; finite-deformation hyperelasticity; viscous incompressible flow; Arbitrary Lagrangian–Eulerian formulation.

control volumes which, in most cases, are rectangular. The dual mesh control domain method (DMCDM), introduced by Reddy [23], combines the positive features of both approaches by satisfying integral balance laws over control domains while approximating the dependent variables using finite element interpolations.

The essential idea of the DMCDM is straightforward. The computational domain is first discretized into a *primal mesh* of finite elements, as shown in Fig. 1, over which the dependent variables are approximated using finite element approximation functions. A *dual mesh* is then constructed by associating each node of the primal mesh with a surrounding non-overlapping region, referred to as a *control domain*. The integral form of the governing equations is enforced over these control domains, thereby giving the method its locally conservative character, as in the finite volume method. At the same time, the use of finite element interpolation and the mapping of physical elements to the canonical master element provide a convenient way to perform numerical quadrature and to treat arbitrary non-rectangular geometries and curved boundaries. They also allow higher-order discretizations to be generated in a modular manner by simply increasing the polynomial order of the elements. This is in contrast to classical finite volume formulations, where increasing the order of approximation often requires a tedious re-derivation of the stencil equations and corresponding modifications to the solver implementation. Detailed descriptions of the method and its implementation may be found in [14, 24, 25].

Since its introduction in 2019, the DMCDM has been applied to a range of problems in applied mechanics. The seminal work by Reddy [23] demonstrated the method for one- and two-dimensional diffusion problems. Subsequent developments extended the method to steady-state convection–diffusion problems [28], nonlinear Poisson equations, and the Navier–Stokes equations for incompressible

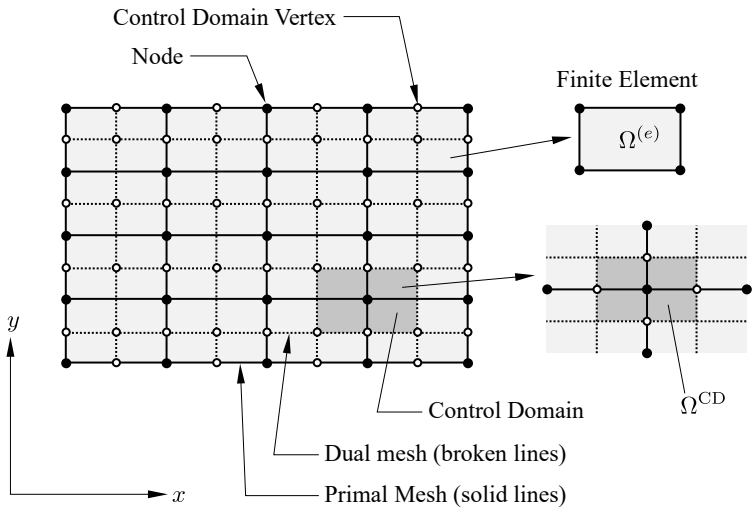


FIGURE 1. Primal and dual meshes used in the dual mesh control domain method.

fluid flow [27], including unsteady flows on arbitrary unstructured meshes [11]. In the context of solid and structural mechanics, the method has been used for linear plane elasticity on irregular domains [16], nonlinear scalar boundary-value problems [10], functionally graded beams [29, 31], functionally graded rectangular and circular plates [19, 30], static, free vibration, and buckling analyses of functionally graded plates [17], and functionally graded auxetic doubly curved shells [18]. More recently, Areias et al. [1] applied the DMCDM to finite-strain solid mechanics problems involving hyperelastic and $J_2$ plastic material models.

An important recent direction in the development of the DMCDM is the use of higher-order interpolation. The first systematic study of $p$-refinement in the DMCDM was presented by Heblekar et al. [12] in the context of a one-dimensional prototype problem. This study showed that the use of higher-order interpolation within the DMCDM framework can be carried out in a systematic manner, much like in the finite element method. The approach was subsequently extended to higher-order formulations for the bending analysis of functionally graded beams and orthotropic plates [14], and later to spectral/higher-order formulations for buoyancy-driven flow and conjugate heat-transfer problems [13]. These developments are important because higher-order elements can provide improved rates of convergence, increased accuracy, and better representation of curved geometries.

The present article builds on these developments and presents the DMCDM as a general framework for selected nonlinear problems in applied mechanics. The focus is on two classes of problems: the finite deformation of hyperelastic solids and the flow of viscous incompressible fluids in domains with moving boundaries. These problems are useful test cases because they involve different sources of nonlinearity and arise in different areas of applied mechanics. In the hyperelasticity problem, the nonlinearity arises from finite-deformation kinematics and nonlinear constitutive behavior. In the moving-boundary fluid-flow problem, the governing equations are written in an Arbitrary Lagrangian–Eulerian (ALE) framework to account for the motion of the computational mesh.

A central feature of the formulations presented here is that higher-order interpolation can be incorporated in a modular manner. Since the dependent variables are approximated using finite element shape functions, the transition from low-order to higher-order elements follows naturally from the interpolation structure. This is similar in spirit to higher-order finite element formulations, while the governing equations continue to be enforced in integral form over the dual control domains.

The remainder of this article is organized as follows. Section 2 reviews the notation and preliminary concepts from continuum mechanics used in the subsequent developments. Section 3 presents the general discretization procedure of the dual mesh control domain method for a representative system of initial-boundary-value problems. Section 4 develops the DMCDM formulation for finite-deformation hyperelasticity and presents two benchmark examples involving Cook's membrane and a curved beam. Section 5 presents the ALE-based DMCDM formulation for viscous incompressible flows in domains with moving boundaries, followed by a numerical example involving an oscillating cylinder in channel flow.

## 2. Preliminaries

In this section, a few definitions, notational conventions, and preliminary concepts from applied mechanics are revisited for completeness.

**2.1. Notational Conventions for Vectors and Tensors.** All vectors are denoted by italicized boldface symbols and are expressed in component form as column arrays. For example, the position vector of a point in space is denoted by $\boldsymbol{x}$ in invariant form and is written as $\{x_1, x_2, x_3\}^{\mathrm{T}}$ in component form.

The derivative of a vector function $\boldsymbol{f}(\boldsymbol{x})$ with respect to $\boldsymbol{x}$ is expressed component-wise as

$$\left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right]_{ij} = \frac{\partial f_i}{\partial x_j},$$

where $i$ denotes the row index and $j$ denotes the column index. Thus, the quantity $\partial f_i/\partial x_j$ occupies the $i$th row and $j$th column of the matrix representation of $\partial \boldsymbol{f}/\partial \boldsymbol{x}$.

Second-order tensors are denoted by upright boldface letters; for instance, a tensor is denoted by $\mathbf{T}$, and its components are expressed using lightface italic symbols as $T_{ij}$, where $i$ is the row index and $j$ is the column index. Tensor contraction between two tensors $\mathbf{A}$ and $\mathbf{B}$ is expressed as $\mathbf{A} \cdot \mathbf{B}$, where the dot is included explicitly to indicate tensor contraction. In component form,

$$[\mathbf{A} \cdot \mathbf{B}]_{ij} = \sum_{k=1}^{3} A_{ik} B_{kj} = A_{ik} B_{kj},$$

where, in going from the second expression to the third, we have used the Einstein summation convention; that is, summation is implied over indices that appear twice in a term.

**2.2. Concepts from Continuum Mechanics.** We briefly review certain key concepts from continuum mechanics that are used in this article. This discussion is intended to provide a self-contained recapitulation of the relevant concepts and formulae. Interested readers are referred to Reddy [20] for a comprehensive treatment of continuum mechanics.

2.2.1. *Kinematics.* Referring to Fig. 2, let $\mathcal{B}^0$ denote the reference configuration of a body, and let $\mathcal{B}$ denote its deformed configuration. Let $\boldsymbol{X}$ denote the position vector of a material point in the reference configuration; the components of $\boldsymbol{X}$ are referred to as the material coordinates. Similarly, let $\boldsymbol{x}$ denote the position vector of the same material point in the deformed configuration. The motion of the body is defined by the mapping

$$\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t).$$

Let $\mathrm{d}\boldsymbol{X}$ denote an infinitesimal material line element in $\mathcal{B}^0$, which is mapped to the corresponding infinitesimal spatial line element $\mathrm{d}\boldsymbol{x}$ in $\mathcal{B}$. The deformation gradient $\mathbf{F}$ is defined as

$$\mathbf{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \frac{\partial \boldsymbol{\varphi}}{\partial \boldsymbol{X}}.$$

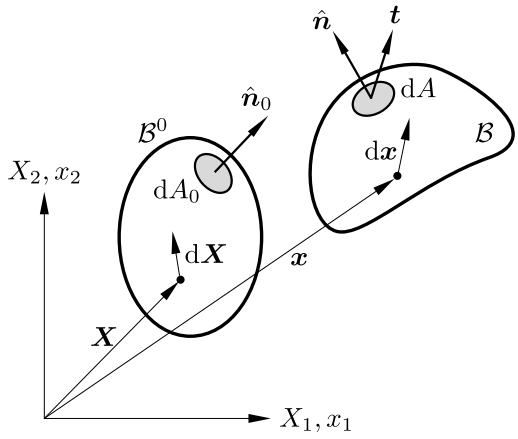


FIGURE 2. Reference and deformed configurations of a continuum, showing the mapping of material points and oriented area elements.

Thus, $\mathbf{F}$ acts as a linear transformation that maps infinitesimal line elements in the reference configuration to their corresponding line elements in the deformed configuration:

$$\mathrm{d}\boldsymbol{x} = \mathbf{F} \cdot \mathrm{d}\boldsymbol{X}.$$

From the deformation gradient $\mathbf{F}$, the right Cauchy-Green deformation tensor $\mathbf{C}$ is defined as

$$\mathbf{C} = \mathbf{F}^{\mathrm{T}} \cdot \mathbf{F}.$$

The tensor $\mathbf{C}$ is a fundamental measure of deformation because it removes the rotational part of the motion and is therefore invariant under superposed rigid body rotations. For a purely rigid body motion, $\mathbf{C} = \mathbf{I}$, where $\mathbf{I}$ is the identity tensor.

Next, we introduce the displacement vector

$$\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X},$$

where the reference and deformed position vectors are understood to be represented in a common basis. The displacement gradient $\mathbf{H}$ is defined with respect to the material coordinates $\boldsymbol{X}$ as

$$\mathbf{H} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}.$$

Using $\boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}$, the deformation gradient can also be written as

$$\mathbf{F} = \mathbf{I} + \mathbf{H}.$$

2.2.2. *Stress Measures.* We define three useful measures of stress that are used in this work (see Reddy [20] for details). As a preliminary step, let us introduce a few definitions. Let $\hat{\boldsymbol{n}}_0$ denote the unit normal vector to an oriented differential area element in the reference configuration, so that

$$\mathrm{d}\boldsymbol{A}_0 = \hat{\boldsymbol{n}}_0 \, \mathrm{d}A_0.$$

After deformation, this area element is mapped to

$$\mathrm{d}\boldsymbol{A} = \hat{\boldsymbol{n}}\,\mathrm{d}A,$$

where $\hat{\boldsymbol{n}}$ is the unit normal vector to the deformed area element. Let $\mathrm{d}\boldsymbol{f}$ denote the force acting on the deformed area element $\mathrm{d}\boldsymbol{A}$. The Cauchy traction vector $\boldsymbol{t}$ is then defined as

$$\boldsymbol{t} = \frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}A},$$

that is, as the force per unit deformed area.

The Cauchy stress tensor $\boldsymbol{\sigma}$ is defined through Cauchy's stress theorem as the linear transformation that maps the unit normal vector $\hat{\boldsymbol{n}}$ to the corresponding traction vector $\boldsymbol{t}$. Thus,

$$\boldsymbol{t} = \boldsymbol{\sigma} \cdot \hat{\boldsymbol{n}}.$$

Next, we consider the first Piola–Kirchhoff stress tensor $\mathbf{P}$. It is defined as the linear transformation that maps the reference unit normal vector $\hat{\boldsymbol{n}}_0$ to the nominal traction vector $\boldsymbol{t}_0$, which represents force per unit reference area. That is,

$$\boldsymbol{t}_0 = \mathbf{P} \cdot \hat{\boldsymbol{n}}_0.$$

The force $\mathrm{d}\boldsymbol{f}$ acting on the deformed area element can then be computed from $\boldsymbol{t}_0$ as

$$\mathrm{d}\boldsymbol{f} = \boldsymbol{t}_0\,\mathrm{d}A_0.$$

Lastly, we briefly mention the second Piola-Kirchhoff stress tensor $\mathbf{S}$ and state its relation to the stress measures discussed above. The second Piola–Kirchhoff stress is a more mathematically motivated stress measure and does not have as direct or straightforward a physical interpretation as the Cauchy stress. In this article, it arises only briefly in the discussion of one of the numerical examples for finite-deformation hyperelasticity. The second Piola–Kirchhoff stress is related to the Cauchy stress through

$$\mathbf{S} = J\mathbf{F}^{-1} \cdot \boldsymbol{\sigma} \cdot \mathbf{F}^{-\mathrm{T}}, \quad \text{where} \quad J = \det \mathbf{F},$$

and it is related to the first Piola–Kirchhoff stress as

$$\mathbf{P} = \mathbf{F} \cdot \mathbf{S}.$$

Before concluding this section, we state the integral form of the balance of linear momentum. The material form of the balance of linear momentum reads

$$\oint_{\partial\Omega^0} \mathbf{P} \cdot \hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 = \int_{\Omega^0} \rho_0(\breve{\boldsymbol{a}} - \breve{\boldsymbol{b}})\mathrm{d}V_0 \qquad \forall \Omega^0 \subset \mathcal{B}^0, \tag{2.1}$$

where $\rho_0$ is the reference density, $\breve{\boldsymbol{a}}$ is the acceleration, and $\breve{\boldsymbol{b}}$ is the body force per unit mass. The breve $(\breve{\cdot})$ over the acceleration and body force is used to indicate that these quantities are expressed as functions of the material coordinates $\boldsymbol{X}$, since the domain of integration lies in the reference configuration; see Tadmor et al. [33] for more details. The corresponding form of the momentum balance written over the deformed configuration reads

$$\oint_{\partial\Omega} \boldsymbol{\sigma} \cdot \hat{\boldsymbol{n}}\,\mathrm{d}A = \int_{\Omega} \rho(\boldsymbol{a} - \boldsymbol{b})\mathrm{d}V \qquad \forall \Omega \subset \mathcal{B}. \tag{2.2}$$

For the nonlinear solid mechanics application discussed in Section 4, we use the material form of the balance of linear momentum given in Eq. (2.1), whereas for the fluid mechanics application discussed in Section 5, we use the spatial form given in Eq. (2.2), albeit with modifications to account for mesh motion, as detailed therein.

In the interest of concision, we conclude this section on preliminaries. Additional considerations, including material constitutive behavior and internal constraints related to incompressibility, will be addressed as needed while discussing the specific applications in the ensuing sections.

## 3. General Procedure of the Dual Mesh Control Domain Method

In this section, we detail the discretization procedure followed in the dual mesh control domain method for solving an initial-boundary-value problem (IBVP). Consider the problem of determining the unknown functions $\phi^\alpha(\boldsymbol{x}, t)$, $\alpha = 1, \dots, \mathcal{P}$, that satisfy the following coupled system of partial differential equations:

$$\sum_{\beta=1}^{\mathcal{P}} \left[ H^{\alpha\beta} \frac{\partial \phi^\beta}{\partial t} + \frac{\partial}{\partial x_i} \left( A_i^{\alpha\beta} \phi^\beta + B_{ij}^{\alpha\beta} \frac{\partial \phi^\beta}{\partial x_j} \right) \right] + f^\alpha + \frac{\partial g_j^\alpha}{\partial x_j} = 0 \tag{3.1}$$

for $\alpha = 1, \dots, \mathcal{P}$. Here, summation is implied over the repeated indices $i, j = 1, 2, 3$. The coefficients $H^{\alpha\beta}$, $A_i^{\alpha\beta}$, and $B_{ij}^{\alpha\beta}$, together with the source terms $f^\alpha$ and $g_i^\alpha$, are prescribed functions of the position $\boldsymbol{x}$, time $t$, and/or the dependent variables $\phi^\alpha$. In the latter case, the system becomes nonlinear. The domain $\Omega$ denotes the region of interest over which the system of equations is to be solved.

To use the dual mesh control domain method for spatial discretization, the strong form of the system of equations in Eq. (3.1) is first converted into an integral form. Accordingly, integrating Eq. (3.1) over an arbitrary control domain $\Omega^{\mathrm{CD}} \subset \Omega$ and applying the Gauss theorem to the divergence term gives

$$\begin{aligned} \sum_{\beta=1}^{\mathcal{P}} \int_{\Omega^{\mathrm{CD}}} H^{\alpha\beta} \frac{\partial \phi^\beta}{\partial t} \mathrm{d}V + \sum_{\beta=1}^{\mathcal{P}} \oint_{\partial\Omega^{\mathrm{CD}}} \left[ (\boldsymbol{A}^{\alpha\beta})^{\mathrm{T}} \phi^\beta + \frac{\partial \phi^\beta}{\partial \boldsymbol{x}} \cdot (\mathbf{B}^{\alpha\beta})^{\mathrm{T}} \right] \cdot \hat{\boldsymbol{n}} \, \mathrm{d}A \\ + \int_{\Omega^{\mathrm{CD}}} f^\alpha \, \mathrm{d}V + \oint_{\partial\Omega^{\mathrm{CD}}} \boldsymbol{g}^\alpha \cdot \hat{\boldsymbol{n}} \, \mathrm{d}A = 0, \quad \alpha = 1, \dots, \mathcal{P}. \end{aligned} \tag{3.2}$$

The above representation serves as a convenient general template into which several problems from heat transfer, solid and structural mechanics, and fluid dynamics can be cast through a suitable choice of the coefficients and forcing functions. Minor case-specific modifications may be required for certain problems, such as the inclusion of additional convective transport terms, as discussed in Section 5 in the context of applying the DMCDM to the flow of a viscous incompressible fluid past moving rigid objects.

To use the dual mesh control domain method, we first partition the domain $\Omega$ into a set of finite elements, as shown in Fig. 1. Within an element $\Omega^{(e)}$, the

dependent unknowns are approximated using finite element interpolation as

$$\phi^{\alpha}(\boldsymbol{x},t) \approx \sum_{j=1}^{N} \psi_j(\boldsymbol{x})\phi_j^{\alpha}(t), \quad \boldsymbol{x} \in \Omega^{(e)}, \tag{3.3}$$

where $N$ is the number of nodes in the element, $\psi_j$, $j = 1,\dots,N$, are the Lagrange interpolation functions, and $\phi_j^{\alpha}(t)$, $j = 1,\dots,N$, are the nodal values of the dependent variable $\phi^{\alpha}$. These nodal values are a priori unknown and must be determined as part of the numerical solution.

To generate a system of equations for these nodal unknowns, a dual mesh is constructed over the primal finite element mesh, as shown in Fig. 1. The dual mesh consists of non-overlapping control domains associated with the nodes of the primal mesh. Each node of the primal mesh is assigned one control domain; therefore, the number of control domains in the dual mesh is equal to the number of nodes in the primal mesh. This construction is analogous in spirit to the use of control volumes in the finite volume method, except that the field variables are approximated using finite element interpolation functions over the primal mesh.

At the element level, the integral equations in Eq. (3.2) are enforced over the portions of the control domains associated with the nodes of the element. In other words, for each node $i$ of an element, the corresponding control domain $\Omega_i^{\mathrm{CD}}$ is used as the domain of integration. Substituting the finite element approximation in Eq. (3.3) into the integral form in Eq. (3.2), and choosing $\Omega^{\mathrm{CD}} = \Omega_i^{\mathrm{CD}}$, gives the following semi-discrete system of control-domain equations for the $i$th node:

$$\sum_{\beta=1}^{\mathcal{P}} \sum_{j=1}^{N} \left( M_{ij}^{\alpha\beta} \frac{\mathrm{d}\phi_j^{\beta}}{\mathrm{d}t} + K_{ij}^{\alpha\beta}\phi_j^{\beta} \right) = F_i^{\alpha}, \quad \alpha = 1,\dots,\mathcal{P}, \tag{3.4}$$

where

$$M_{ij}^{\alpha\beta} = \int_{\Omega_i^{\mathrm{CD}}} H^{\alpha\beta}\psi_j \,\mathrm{d}V, \qquad F_i^{\alpha} = -\int_{\Omega_i^{\mathrm{CD}}} f^{\alpha} \,\mathrm{d}V - \oint_{\Gamma_i^{\mathrm{CD}}} \boldsymbol{g}^{\alpha} \cdot \hat{\boldsymbol{n}} \,\mathrm{d}A,$$
$$K_{ij}^{\alpha\beta} = \oint_{\Gamma_i^{\mathrm{CD}}} \left[ (\boldsymbol{A}^{\alpha\beta})^{\mathrm{T}}\psi_j + \frac{\partial \psi_j}{\partial \boldsymbol{x}} \cdot (\mathbf{B}^{\alpha\beta})^{\mathrm{T}} \right] \cdot \hat{\boldsymbol{n}} \,\mathrm{d}A.$$

The above equations are written for all nodes of the element, that is, for $i = 1,\dots,N$, thereby producing the element-level system of equations. These equations are ordinary differential equations in time for the nodal unknowns $\phi_j^{\alpha}(t)$. Thus, the spatial discretization reduces the original system of partial differential equations to a system of ordinary differential equations in time.

The element-level equations (3.4) are then assembled into a global system by collecting the contributions from all elements sharing a common node, in the same manner as in the finite element method. Boundary conditions are imposed on the assembled system according to their type. Essential boundary conditions prescribe the values of the relevant nodal unknowns, whereas natural boundary conditions enter through the boundary integrals over the corresponding portions of the control-domain boundaries.

For time-dependent problems, the assembled system of ordinary differential equations is further discretized in time using a suitable time integration scheme, resulting in a system of algebraic equations at each time level. The reader is referred to [21] for a detailed discussion of different types of time integration schemes. For steady problems, the time-derivative term is omitted, and the spatial discretization directly yields a system of algebraic equations. If the coefficients or source terms depend on the unknown fields, the resulting algebraic system is nonlinear and must be solved using an iterative nonlinear solution procedure.

The reader is referred to [14] for a more comprehensive discussion of implementation details related to the numerical evaluation of control-domain integrals, assembly of element equations, imposition of boundary conditions, and solution of the resulting global nonlinear system of equations. These details are omitted here to avoid repetition and to keep the present discussion focused on the general discretization procedure.

It is important to note that the template in Eq. (3.2) applies not only to equations written in terms of the primary unknowns $\phi^\alpha$, but also to incremental equations obtained by linearizing nonlinear partial differential equations. In the latter case, the unknowns in the template are interpreted as increments, or variations, $\delta\phi^\alpha$. This viewpoint will be used in the next section, where the DMCDM is applied to the incremental equations arising in the large-deformation analysis of hyperelastic solids.

## 4. Finite-Deformation Hyperelasticity

**4.1. Formulation.** In this section, we develop dual mesh control domain formulations to simulate the large deformation of hyperelastic solids. By definition, hyperelastic solids are those for which there exists a strain energy density $\psi(\boldsymbol{F})$ from which the stresses are derivable as

$$\mathbf{P} = \frac{\partial \psi}{\partial \mathbf{F}}, \quad \mathbf{S} = 2\frac{\partial \psi}{\partial \mathbf{C}}.$$

As an illustration, let us consider the following strain energy function for a compressible Neo-Hookean solid [32]:

$$\psi = \frac{\mu}{2}(I_C - 3) + \frac{\lambda}{4}(J^2 - 1) - \left(\frac{\lambda}{2} + \mu\right)\ln J,$$

for which the second Piola–Kirchhoff stress is given by

$$\mathbf{S} = \frac{\lambda}{2}(J^2 - 1)\mathbf{C}^{-1} + \mu(\mathbf{I} - \mathbf{C}^{-1}). \tag{4.1}$$

Considering only the static case, i.e., $\boldsymbol{a} = \mathbf{0}$, the objective is to determine the displacement field $\boldsymbol{u}(\boldsymbol{X})$ that satisfies the material form of the equilibrium equations in Eq. (2.1) together with the constitutive relation defined in Eq. (4.1). To this end, we define the vector-valued residual functional associated with Eq. (2.1) as

$$\boldsymbol{\mathcal{R}}(\boldsymbol{u}) = \oint_{\partial\Omega^0} \mathbf{F}\cdot\mathbf{S}\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 + \int_{\Omega^0}\rho_0\breve{\boldsymbol{b}}\,\mathrm{d}V_0 \qquad \forall\Omega^0 \subset \mathcal{B}^0, \tag{4.2}$$

where we have used $\mathbf{P} = \mathbf{F} \cdot \mathbf{S}$ and set the acceleration to zero, with the second Piola–Kirchhoff stress $\mathbf{S}$ appearing in Eq. (4.2) understood to be computed from Eq. (4.1). The solution of the boundary-value problem is therefore obtained by finding a displacement field $\boldsymbol{u}$ such that

$$\boldsymbol{\mathcal{R}}(\boldsymbol{u}) = \mathbf{0}.$$

The system of equations defined by $\boldsymbol{\mathcal{R}}(\boldsymbol{u}) = \mathbf{0}$ is nonlinear in the displacement components $\{u_1, u_2, u_3\}$. This nonlinearity arises because the displacement field determines the deformation gradient, $\mathbf{F} = \mathbf{I} + \partial \boldsymbol{u}/\partial \boldsymbol{X}$. Consequently, the definitions of $\mathbf{C} = \mathbf{F}^{\mathrm{T}} \cdot \mathbf{F}$ and $J = \det \mathbf{F}$ involve products of the derivatives of the unknown displacement components. Since the stress tensor $\mathbf{S}$ depends on $\mathbf{C}$ and $J$, the residual depends nonlinearly on $\boldsymbol{u}$ through both the kinematic relations and the constitutive relation. An iterative solution strategy is therefore required to determine $\boldsymbol{u}$.

In this work, we use Newton's method to solve the nonlinear system of equations in Eq. (4.2). Let $\boldsymbol{u}$ denote the current approximation to the displacement field, and let $\delta \boldsymbol{u}$ denote an increment to this approximation. A first-order Taylor expansion of the residual about $\boldsymbol{u}$ gives

$$\boldsymbol{\mathcal{R}}(\boldsymbol{u} + \delta \boldsymbol{u}) = \boldsymbol{\mathcal{R}}(\boldsymbol{u}) + \delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}] + \cdots , \tag{4.3}$$

where $\delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}]$ is the directional derivative of the residual in the direction of the increment $\delta \boldsymbol{u}$, which we refer to herein as the functional differential, or first variation, of the residual. This functional differential is defined by

$$\delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}] \equiv \frac{\mathrm{d}}{\mathrm{d}\eta} \boldsymbol{\mathcal{R}}(\boldsymbol{u} + \eta\, \delta \boldsymbol{u})\Big|_{\eta=0}.$$

This quantity is linear in the increment $\delta \boldsymbol{u}$[1]. Returning to the expansion in Eq. (4.3) and requiring the residual to vanish to first order, we obtain the Newton correction $\delta \boldsymbol{u}$ from

$$\delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}] = -\boldsymbol{\mathcal{R}}(\boldsymbol{u}).$$

The above equation is linear in the unknown correction $\delta \boldsymbol{u}$.

We can now state the Newton iteration procedure. Starting from an initial guess $\boldsymbol{u}_0$, usually taken to be zero, we compute the correction $\delta \boldsymbol{u}_k$ at iteration $k$ by solving

$$\delta \boldsymbol{\mathcal{R}}(\boldsymbol{u}_k)[\delta \boldsymbol{u}_k] = -\boldsymbol{\mathcal{R}}(\boldsymbol{u}_k),$$

and then update the displacement field according to

$$\boldsymbol{u}_{k+1} \leftarrow \boldsymbol{u}_k + \delta \boldsymbol{u}_k.$$

This process is repeated until the residual is reduced below a prescribed tolerance.

We now derive an expression for the variation of the residual defined in Eq. (4.2). Standard properties of variational operators are used in this derivation and are not

[1]Linearity implies that for any $\delta \boldsymbol{u}_1$ and $\delta \boldsymbol{u}_2$,

$$\delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\alpha \delta \boldsymbol{u}_1 + \beta \delta \boldsymbol{u}_2] = \alpha \delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}_1] + \beta \delta \boldsymbol{\mathcal{R}}(\boldsymbol{u})[\delta \boldsymbol{u}_2] \qquad \forall \alpha, \beta \in \mathbb{R}$$

repeated here; interested readers are referred to Reddy [22] for a detailed treatment of the relevant principles of variational calculus.

Assuming that the prescribed body force is independent of the displacement field, the variation of the residual is given by

$$\delta\boldsymbol{\mathcal{R}} = \oint_{\partial\Omega^0} \delta(\mathbf{F}\cdot\mathbf{S})\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 = \oint_{\partial\Omega^0} (\delta\mathbf{F}\cdot\mathbf{S} + \mathbf{F}\cdot\delta\mathbf{S})\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0. \tag{4.4}$$

The variation $\delta\mathbf{S}$ of the second Piola–Kirchhoff stress may be computed from the constitutive equation in Eq. (4.1) as

$$\begin{aligned}\delta\mathbf{S} &= \delta\Big[\frac{\lambda}{2}(J^2-1)\mathbf{C}^{-1} + \mu(\mathbf{I}-\mathbf{C}^{-1})\Big] = \frac{\lambda}{2}(J^2-1)\delta(\mathbf{C}^{-1}) + \lambda J\mathbf{C}^{-1}\delta J - \mu\delta(\mathbf{C}^{-1}) \\ &= \Big[\frac{\lambda}{2}(1-J^2)+\mu\Big]\mathbf{C}^{-1}\cdot\delta\mathbf{C}\cdot\mathbf{C}^{-1} + \lambda J\mathbf{C}^{-1}\delta J,\end{aligned} \tag{4.5}$$

where we have used the identity

$$\delta(\mathbf{C}^{-1}) = -\mathbf{C}^{-1}\cdot\delta\mathbf{C}\cdot\mathbf{C}^{-1}.$$

The variations of $\mathbf{C}$ and $J$ are given by

$$\delta\mathbf{C} = \delta(\mathbf{F}^{\mathrm{T}}\cdot\mathbf{F}) = (\delta\mathbf{F})^{\mathrm{T}}\cdot\mathbf{F} + \mathbf{F}^{\mathrm{T}}\cdot\delta\mathbf{F}, \quad \delta J = J\,\mathrm{tr}(\mathbf{F}^{-1}\cdot\delta\mathbf{F}), \tag{4.6}$$

where the variation of the deformation gradient is simply

$$\delta\mathbf{F} = \frac{\partial\delta\boldsymbol{u}}{\partial\boldsymbol{X}}.$$

For a detailed treatment of such useful linearization results for kinematic quantities, the reader is referred to Bonet and Wood [4].

Substituting Eqs. (4.5) and (4.6) into Eq. (4.4), we obtain the following incremental equations in component form:

$$\begin{aligned}\delta\mathcal{R}^1[\delta\boldsymbol{u}] &= \oint_{\partial\Omega^0}\frac{\partial\delta u}{\partial\boldsymbol{X}}\cdot(\mathbf{T}^{11})^{\mathrm{T}}\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 + \oint_{\partial\Omega^0}\frac{\partial\delta v}{\partial\boldsymbol{X}}\cdot(\mathbf{T}^{12})^{\mathrm{T}}\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 = -\mathcal{R}^1, \\ \delta\mathcal{R}^2[\delta\boldsymbol{u}] &= \oint_{\partial\Omega^0}\frac{\partial\delta u}{\partial\boldsymbol{X}}\cdot(\mathbf{T}^{21})^{\mathrm{T}}\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 + \oint_{\partial\Omega^0}\frac{\partial\delta v}{\partial\boldsymbol{X}}\cdot(\mathbf{T}^{22})^{\mathrm{T}}\cdot\hat{\boldsymbol{n}}_0\,\mathrm{d}A_0 = -\mathcal{R}^2.\end{aligned} \tag{4.7}$$

A couple of important remarks are in order regarding the above system of equations. First, we have assumed plane strain conditions and therefore consider only two nonzero displacement components, $u_1 = u$ and $u_2 = v$. The $2\times 2$ matrices $\mathbf{T}^{\alpha\beta}$ are functions of the displacement field $\boldsymbol{u}$, and the expressions for their components may be derived using a symbolic algebra package. It is straightforward to see that these coefficients are given by

$$T_{ij}^{\alpha\beta} = \frac{\partial(\delta P_{\alpha i})}{\partial(\delta u_{\beta,j})} \quad \text{where} \quad \delta\mathbf{P} = \delta\mathbf{F}\cdot\mathbf{S} + \mathbf{F}\cdot\delta\mathbf{S}. \tag{4.8}$$

The form shown in Eq. (4.7) is more amenable to the DMCDM implementation because it fits the template in Eq. (3.2) with $\phi^1 = \delta u$ and $\phi^2 = \delta v$. Since only the static case is considered here, $H^{\alpha\beta} = 0$. The remaining coefficients are identified as

$$\mathbf{B}^{\alpha\beta} = \mathbf{T}^{\alpha\beta}, \quad A_i^{\alpha\beta} = 0, \quad f^\alpha = \rho_0\breve{b}_\alpha, \quad g_i^\alpha = [\mathbf{F}\cdot\mathbf{S}]_{\alpha i}.$$

Next, to discretize the system of equations in Eq. (4.7), we follow the procedure detailed in Section 3. Specifically, we replace the displacement increment functions $\delta u(\boldsymbol{X})$ and $\delta v(\boldsymbol{X})$ by their finite element approximations:

$$\delta u(\boldsymbol{X}) \approx \sum_{j=1}^{N} \delta u_j \psi_j(\boldsymbol{X}), \quad \delta v(\boldsymbol{X}) \approx \sum_{j=1}^{N} \delta v_j \psi_j(\boldsymbol{X}), \quad \boldsymbol{X} \in \Omega^{(e)}.$$

We then set $\Omega^0 \leftarrow \Omega_i^{\mathrm{CD}}$, where $\Omega_i^{\mathrm{CD}}$ denotes the control domain associated with the $i$th node, $i = 1, \ldots, N$, and obtain the following discrete element equations:

$$\begin{bmatrix} \mathbf{K}^{11} & \mathbf{K}^{12} \\ \mathbf{K}^{21} & \mathbf{K}^{22} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\Delta}^1 \\ \boldsymbol{\Delta}^2 \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{R}^1 \\ \boldsymbol{R}^2 \end{Bmatrix}.$$

The components of the coefficient submatrices and residual subvectors are given by

$$K_{ij}^{\alpha\beta} = \oint_{\Gamma_i^{\mathrm{CD}}} \frac{\partial \psi_j}{\partial \boldsymbol{X}} \cdot (\mathbf{T}^{\alpha\beta})^{\mathrm{T}} \cdot \hat{\boldsymbol{n}}_0 \, \mathrm{d}A_0, \quad R_i^{\alpha} = \oint_{\Gamma_i^{\mathrm{CD}}} F_{\alpha m} S_{mj} (\hat{n}_0)_j \, \mathrm{d}A_0 + \int_{\Omega_i^{\mathrm{CD}}} \rho_0 b_\alpha \, \mathrm{d}V_0.$$

Here, $\boldsymbol{\Delta}^1$ and $\boldsymbol{\Delta}^2$ are the vectors of unknown nodal displacement increments in the $X$- and $Y$-directions, respectively.

Once the element-level incremental equations are obtained, they are assembled into the global Newton system in the usual manner. The resulting linear algebraic system is solved for the nodal displacement increments, and the displacement field is updated accordingly. The procedure is repeated until convergence is achieved, as measured by the reduction of the residual below a prescribed tolerance. In the ensuing subsections, two numerical examples are presented to illustrate the application of the DMCDM to the large-deformation analysis of hyperelastic solids.

**4.2. Example 1: Cook's Membrane.** The Cook's membrane problem [32], involving a tapered cantilever beam loaded by a distributed shear force at the free end, combines bending and shear deformation and is a popular benchmark for validating elasticity formulations and computational implementations. The geometry, representative mesh, and boundary conditions are shown in Fig. 3. The analysis is performed using $\lambda = 432.099$ MPa, $\mu = 185.185$ MPa, and $\tau_0 = 20$ MPa. The representative baseline mesh shown in Fig. 3 consists of $5 \times 5$ Q4 bilinear quadrilateral elements. However, the problem is solved using multiple meshes obtained by both $h$- and $p$-refinement of this baseline mesh. In $h$-refinement, the mesh is refined by subdivision such that the coarser mesh is contained in the finer mesh, whereas in $p$-refinement, the polynomial order of the elements is increased.

For the Newton iteration procedure, a convergence tolerance of $\varepsilon_{\mathrm{TOL}} = 10^{-3}$ is used with the following termination criterion:

$$\sqrt{\frac{(\boldsymbol{\Delta}_k)^{\mathrm{T}} \cdot \boldsymbol{\Delta}_k}{(\boldsymbol{U}_k)^{\mathrm{T}} \cdot \boldsymbol{U}_k}} < \varepsilon_{\mathrm{TOL}},$$

where $\boldsymbol{\Delta}_k$ is the vector of displacement increments at the $k$th iteration and $\boldsymbol{U}_k$ is the displacement vector up to the $k$th iteration. The problem is solved using load continuation, where the total load is applied incrementally rather than all at once. For the present problem, the total load is divided into 10 uniform load increments.

TABLE 1. Caption for the table-like figure.

| Elem. Order | Discretization | Nodes | Elements | $u_2$ at Point A |
|---|---|---|---|---|
| $p = 1$ | 5×5 | 36 | 25 | 8.8685 |
| | 10×10 | 121 | 100 | 10.0201 |
| | 20×20 | 441 | 400 | 10.411 |
| | 40×40 | 1681 | 1600 | 10.5342 |
| | 80×80 | 6561 | 6400 | 10.5738 |
| | 160×160 | 25921 | 25600 | 10.5871 |
| $p = 2$ | 3×3 | 49 | 9 | 10.2274 |
| | 5×5 | 121 | 25 | 10.4282 |
| | 10×10 | 441 | 100 | 10.5341 |
| | 20×20 | 1681 | 400 | 10.5726 |
| | 40×40 | 6561 | 1600 | 10.5863 |
| | 80×80 | 25921 | 6400 | 10.5914 |
| $p = 4$ | 5×5 | 441 | 25 | 10.5559 |
| | 10×10 | 1681 | 100 | 10.5804 |
| | 20×20 | 6561 | 400 | 10.5888 |
| | 40×40 | 25921 | 1600 | 10.5921 |

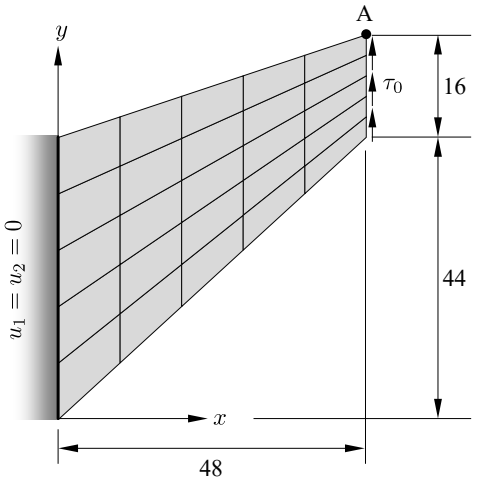


FIGURE 3. Geometry, boundary conditions, and representative baseline mesh for the Cook's membrane problem [32].

The results for the Cook's membrane problem are summarized in Table 1. The transverse displacement $u_2$ at point A, indicated in Fig. 3, is reported for different element orders and mesh sizes. The converged displacement value, $(u_2)_A = 10.5921$, obtained using the $p = 4$, $40 \times 40$ mesh, is in excellent agreement with the value 10.5952 reported in [32], where the analysis was performed using a graded mesh

of 13 quadrilateral elements with polynomial order $p = 14$. The results in Table 1 also show the benefit of higher-order interpolation. For a comparable number of nodes, and hence a comparable number of total degrees of freedom, the higher-order elements provide more accurate solutions than the corresponding lower-order elements. The deformed configuration of the membrane, plotted with a scale factor of 1, is shown in Fig. 4. The displacement magnitude contours and the $\sigma_{xx}$ contours are also shown on the deformed configuration.

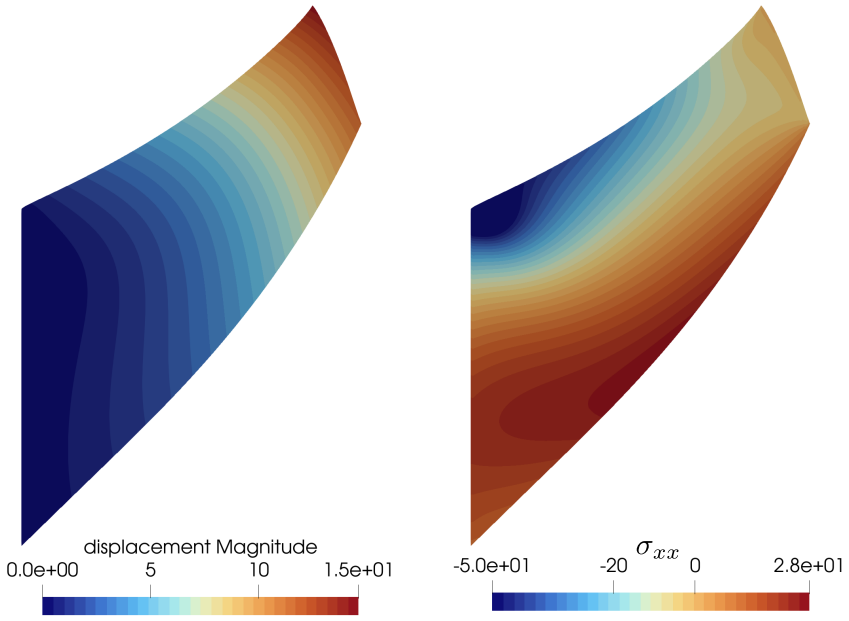


FIGURE 4. Deformed configuration of the Cook's membrane problem showing the displacement magnitude and normal stress $\sigma_{xx}$.

**4.3. Example: Curved Beam.** This second benchmark, taken from [9], involves the finite deformation of a curved arch. The geometry, boundary conditions, and mesh used for the analysis are shown in Fig. 5. The bottom end of the arch is fixed, while the other, free end is subjected to combined shear and compressive distributed loads. This problem is characterized by moderate strains but large rotations, and therefore serves as a useful test of the ability of the DMCDM to converge in problems where nonlinear effects are pronounced.

For this example, in order to match the benchmark in [9], the following material model is used:

$$\mathbf{P} = \mu(\mathbf{F} - \mathbf{F}^{-\mathrm{T}}) + \kappa(J^2 - J)\mathbf{F}^{-\mathrm{T}},$$

with $\mu = 80.194$ N/mm$^2$ and $\kappa = 120.291$ N/mm$^2$. The distributed load magnitude is $f = 0.5$ N/mm$^2$. As shown in Fig. 5, the mesh used for this problem consists of 10 elements along the circumferential direction and one element through the

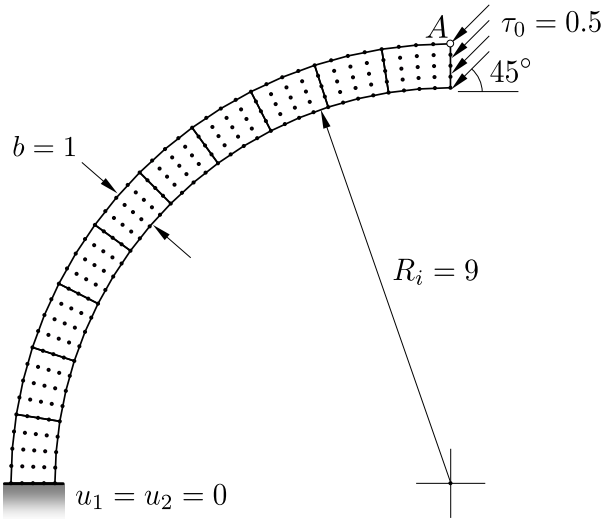


FIGURE 5. Geometry, boundary conditions, and mesh with $p = 4$ elements for the curved beam problem [9].

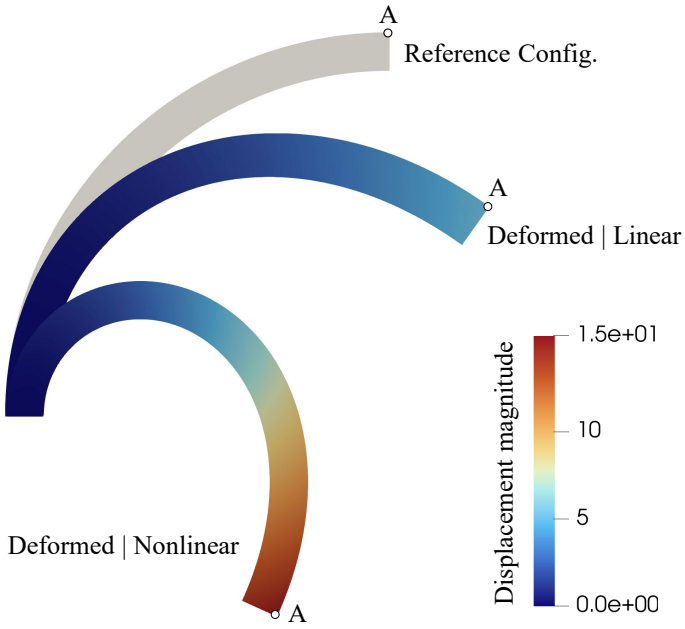


FIGURE 6. Reference and deformed configurations of the curved beam obtained from linear and nonlinear analyses.

radial thickness, with polynomial order $p = 4$. Thus, the benefit of higher-order interpolation is exploited in this example: a small number of higher-order elements can represent the curved geometry more accurately while also providing improved accuracy compared with lower-order discretizations. As in the preceding example, a nonlinear convergence tolerance of $\varepsilon_{\mathrm{TOL}} = 10^{-3}$ is used, and the problem is solved using load continuation with 20 uniform load increments.

The reference and deformed configurations of the beam are shown in Fig. 6 for both the linear and nonlinear analyses. The nonlinear solution clearly shows that the beam undergoes large rotations, which cannot be represented accurately within a linearized kinematic description. For the same applied load, the linear analysis underpredicts the displacement compared with the nonlinear analysis, indicating that the linearized model gives an effectively stiffer response for this problem. Figure 7 shows the corresponding load–deflection response. The response is initially close to the linear solution, but as the deformation increases, geometric and material nonlinearities become more pronounced and the nonlinear curve departs substantially from the linear prediction.

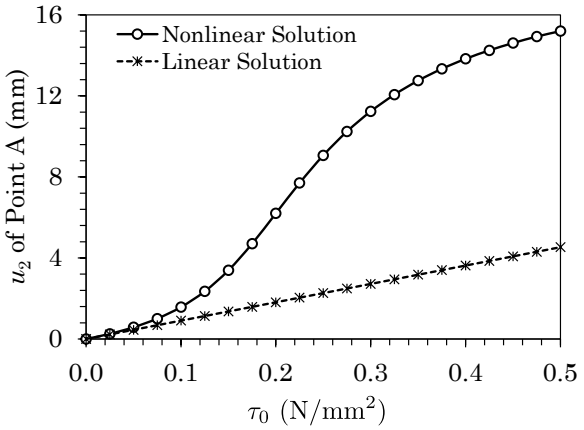


FIGURE 7. Load–deflection response for the curved hyperelastic beam problem.

## 5. Fluid Flow in Domains with Moving Boundaries

In this section, we present formulations for simulating the flow of viscous incompressible fluids in domains with nonstationary boundaries using the dual mesh control domain method. Moving-boundary problems are important in many practical applications and form the basis for modeling fluid–structure interaction (FSI). In its fully coupled form, FSI involves the interaction between a deformable structure and a flowing fluid. The fluid forces deform the structure, while the structural deformation, in turn, modifies the flow domain. This gives rise to a bidirectional, or two-way, coupling between the fluid motion and the structural deformation, which must be resolved iteratively to obtain a mutually consistent physical state.

In the present work, however, we do not consider fully coupled FSI problems. Instead, we consider the flow of a viscous incompressible fluid past a rigid body whose motion is prescribed. The objective is to develop and demonstrate the DM-CDM formulation for fluid-flow problems involving moving boundaries.

In this work, we employ the Arbitrary Lagrangian–Eulerian (ALE) formulation. A brief discussion of the ALE description is therefore in order. In a purely Lagrangian description, the quantities of interest are referred to the reference configuration and are expressed as functions of the material coordinates $X_i$. Accordingly, in a numerical discretization, the mesh is generated over the reference configuration. Each mesh node remains associated with a particular material point. Such a description is well suited for modeling solids, as seen in the previous section on hyperelasticity.

On the other hand, in a purely Eulerian description, the quantities of interest are referred to the spatial domain and are expressed as functions of the spatial coordinates $x_i$. In this case, the mesh is generated over the spatial domain, and each mesh node is associated with a fixed point in space rather than with a particular material particle. Thus, at different instants of time, different material particles may occupy the spatial location associated with a given mesh node. The Eulerian description is therefore well suited for modeling fluid flow in domains with stationary boundaries.

The Arbitrary Lagrangian–Eulerian description provides an intermediate framework between these two descriptions. In the ALE formulation, the quantities of interest are referred neither directly to material particles, as in the Lagrangian description, nor to fixed points in space, as in the Eulerian description. Instead, they are expressed as functions of the coordinates $\chi_i$ of a third domain, called the referential domain. This referential domain is mapped to the spatial, or current, domain through a bijective time-dependent mapping

$$\boldsymbol{x} = \hat{\boldsymbol{\varphi}}(\boldsymbol{\chi}, t).$$

The computational mesh is generated in the referential domain, and its image under the mapping $\bar{\boldsymbol{\varphi}}(\boldsymbol{\chi}, t)$ produces a deforming mesh in the spatial domain. Thus, the ALE mesh is neither fixed in space nor constrained to move with the material. Instead, it moves according to the chosen referential-to-spatial mapping. This freedom in prescribing the mesh motion is the reason for the term Arbitrary Lagrangian-Eulerian.

Next, we briefly review the derivation of the governing equations in the ALE framework. Detailed treatments of the ALE formulation may be found in works such as [7] and [15]. Nevertheless, we revisit the main steps of the derivation here to make the present development self-contained and to present the underlying ideas in a more direct and accessible manner.

**5.1. Preliminaries on the ALE Formulation.** Let $\mathcal{F}$ represent the value of a particular field variable; it may be a scalar quantity, such as temperature or pressure, or a component of a vector or tensor. Let $\breve{f}$, $f$, and $\hat{f}$ denote the functional dependence of this quantity on the material, spatial, and referential coordinates,

respectively. Thus,

$$\mathcal{F} = \breve{f}(\boldsymbol{X}, t) = f(\boldsymbol{x}, t) = \hat{f}(\boldsymbol{\chi}, t).$$

Since, as mentioned before, the finite element mesh in the ALE formulation is constructed in the referential domain, we approximate $\mathcal{F}$ using finite element interpolation in the referential domain as

$$\mathcal{F} = \sum_{j=1}^{N} \psi_j(\boldsymbol{\chi}) f^j(t) \implies \mathcal{F} = \hat{f}(\boldsymbol{\chi}, t), \quad \boldsymbol{\chi} \in \hat{\Omega}^{(e)}, \tag{5.1}$$

where $\hat{\Omega}^{(e)}$ is a typical element in the referential domain and $f^j(t)$ are the time-dependent nodal values of $\mathcal{F}$.

The time derivative that follows naturally from Eq. (5.1), and which can be computed directly from the finite element approximation, is given by

$$\mathcal{F}' = \frac{\partial \hat{f}(\boldsymbol{\chi}, t)}{\partial t}\Big|_{\boldsymbol{\chi}} = \sum_{j=1}^{N} \psi_j(\boldsymbol{\chi}) \frac{\mathrm{d} f^j(t)}{\mathrm{d}t}. \tag{5.2}$$

We use the notation $\mathcal{F}'$ to denote the rate of change of $\mathcal{F}$ at a fixed referential point. This derivative is therefore called the referential derivative.

Next, let the motion of the referential domain be given by

$$x_i = \sum_{j=1}^{N} \psi_j(\boldsymbol{\chi}) x_i^j(t) \implies x_i = \hat{\varphi}_i(\boldsymbol{\chi}, t), \tag{5.3}$$

where $\boldsymbol{x}^j(t)$, $j = 1, \ldots, N$, are the coordinates of the mesh nodes in the spatial domain. This choice for the mapping from the referential domain to the spatial domain is convenient because, in practice, the motion of the mesh nodes $\boldsymbol{x}^j(t)$ in the spatial domain is often known from the mesh-motion algorithm. The finite element approximation then provides a convenient way to bijectively map points from the referential domain to the spatial domain. For simplicity, the same shape functions used to approximate the field variable $\mathcal{F}$ are also used to approximate the geometry.

Differentiating Eq. (5.3) with respect to time at fixed $\boldsymbol{\chi}$, the mesh velocity is obtained as

$$w_i = \frac{\partial \varphi_i}{\partial t}\Big|_{\boldsymbol{\chi}} = \sum_{j=1}^{N} \psi_j(\boldsymbol{\chi}) \frac{\mathrm{d} x_i^j}{\mathrm{d}t} = \sum_{j=1}^{N} \psi_j(\boldsymbol{\chi}) w_i^j. \tag{5.4}$$

Once again, note that the derivatives in Eqs. (5.2) and (5.4) are computed at a fixed point in the referential domain; the notation $(\,\cdot\,)|_{\boldsymbol{\chi}}$ is used to indicate this explicitly. Using the chain rule, it can be shown that

$$\dot{\mathcal{F}} = \frac{\partial \hat{f}}{\partial t}\Big|_{\boldsymbol{\chi}} + (v_i - w_i) \frac{\partial f}{\partial x_i}, \quad \text{where} \quad \dot{\mathcal{F}} = \frac{\partial \breve{f}(\boldsymbol{X}, t)}{\partial t}\Big|_{\boldsymbol{X}} \tag{5.5}$$

is the material derivative. This equation is often referred to in the literature as the fundamental ALE equation. It relates the material time derivative $\dot{\mathcal{F}}$ to the referential time derivative $\mathcal{F}'$.

To understand why this relation is necessary, it is useful to ask two questions: what derivative do we need, and what derivative can we compute? In conservation laws, the required time derivative is fundamentally the material derivative. For example, when Newton's law of motion is applied to a material particle, the rate of change of momentum is computed by following that particle; hence, the relevant derivative is the material derivative. However, in the ALE finite element discretization, the quantity that is computed directly is the referential derivative in Eq. (5.2), namely, the rate of change of a quantity at a fixed referential point. The fundamental ALE equation (5.5) therefore provides the connection between the derivative required by the conservation laws and the derivative naturally obtained from the finite element discretization in the referential domain.

One final point remains to be addressed. The ALE relation written above is in local form, whereas the DMCDM requires an integral form of the relation between the material and referential derivatives. To this end, consider

$$I(t) = \int_{\Omega(t)} \mathcal{F} \mathrm{d}V = \int_{\Omega(t)} f(\boldsymbol{x}, t) \mathrm{d}V,$$

where $\Omega(t)$ is attached to the mesh and is the spatial image of the referential volume $\hat{\Omega}$. It can be shown that the integral form of the ALE relation is given by

$$\dot{I} = \int_{\Omega(t)} \Big[ \mathcal{F}' + (v_i - w_i) \frac{\partial f}{\partial x_i} + (\operatorname{div} \boldsymbol{v}) \mathcal{F} \Big] \mathrm{d}V. \tag{5.6}$$

**5.2. Governing Equations in ALE Form.** Based on the preceding discussion, the governing equations for the flow of viscous incompressible fluids must be suitably modified to account for the arbitrarily moving mesh. This is done by replacing the material time derivative appearing in the conservation laws by the corresponding ALE expression given in Eq. (5.6). The resulting integral forms of the conservation of mass, linear momentum, and energy equations are stated below. The conservation of mass under the assumption of incompressibility:

$$\int_{\Omega(t)} \operatorname{div} \boldsymbol{v} \, \mathrm{d}V = 0 \implies \int_{\Omega(t)} \frac{\partial v_i}{\partial x_i} \, \mathrm{d}V = 0. \tag{5.7}$$

The conservation of linear momentum:

$$\int_{\Omega(t)} \rho \Big[ v_i' + (v_j - w_j) \frac{\partial v_i}{\partial x_j} \Big] \mathrm{d}V = \oint_{\Gamma(t)} \sigma_{ij} n_j \mathrm{d}\Gamma + \int_{\Omega(t)} \rho b_i \, \mathrm{d}V. \tag{5.8}$$

The conservation of energy:

$$\int_{\Omega(t)} \rho c \Big[ \theta' + (v_j - w_j) \frac{\partial \theta}{\partial x_j} \Big] \mathrm{d}V + \oint_{\Gamma(t)} q_j n_j \, \mathrm{d}\Gamma = 0, \tag{5.9}$$

where $\theta$ is the temperature, $c$ is the specific heat capacity, and $q_i$ are the components of the heat flux vector.

For simplicity, the fluid is assumed to be linear Newtonian. Accordingly, the Cauchy stress components are expressed as

$$\sigma_{ij} = -P\delta_{ij} + \mu \Big( \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \Big), \tag{5.10}$$

where $\mu$ is the viscosity, and $P$ is the pressure determined as part of the solution of the boundary-value problem.

We also assume that Fourier's law of heat conduction is applicable, so that the heat flux components are given by

$$q_i = -k\frac{\partial \theta}{\partial x_i}. \tag{5.11}$$

where $k$ is the thermal conductivity.

**5.3. Dual Mesh Control Domain Discretization.** A comprehensive discussion on the application of the DMCDM to the flow of viscous incompressible fluids can be found in [11, 13, 27]. Two approaches have been used in the DMCDM literature for imposing the incompressibility constraint. The first and earlier approach is to impose incompressibility through a penalty formulation (see Reddy [21]). Application of the penalty function method amounts to replacing the pressure by

$$P = -\gamma \operatorname{div} \boldsymbol{v}, \tag{5.12}$$

where $\gamma$ is a large number, called the penalty parameter.

This approach has two main drawbacks. First, the terms involving the penalty parameter must be under-integrated for the approach to yield meaningful results; see [21] for more details. In particular, the penalty contribution to the coefficient matrix must be rendered singular through reduced integration. Otherwise, the penalty terms dominate the global system and can lead to a trivial zero-velocity solution, which satisfies the incompressibility constraint but is not the desired physical solution. Second, the pressure is not obtained directly as an unknown. Instead, it must be post-computed from the velocity solution. The pressure field obtained in this manner is generally less accurate, since it is computed from derivatives of the velocity field, and is also not directly available at the nodes. This can be a disadvantage in FSI-type problems, where nodal pressure values are useful for computing the forces exerted by the fluid on an immersed structure.

The second approach, introduced for the DMCDM in [13], is the $\boldsymbol{v}$–$P$ formulation, also referred to as the velocity–pressure formulation, in which the incompressibility equation is retained as an additional governing equation along with the momentum equations. In two dimensions, this leads to three unknowns, $(v_1, v_2, P)$, and three equations, namely the two momentum equations and the incompressibility constraint. However, as is well known, using equal-order interpolation for velocity and pressure in such mixed formulations may violate the Ladyzhenskaya–Babuška–Brezzi (LBB) stability condition; see [2, 26]. A common way to avoid this difficulty is to use interpolation functions for pressure that are one order lower than those used for velocity, as in Taylor–Hood elements [5, 34]. In this article, we adopt this approach: the velocity components, and also the temperature, are interpolated using shape functions of order $p$, whereas the pressure is interpolated using shape functions of order $p-1$.

We do not discuss the implementation of this approach in detail here, since it has already been presented in great depth in [13]. It suffices to state that

the discrete dual mesh control domain equations are obtained by substituting the finite element approximations for the velocity components into the mass conservation equation in Eq. (5.7), and for the velocity components, their referential time derivatives, and the pressure into the momentum equations in Eq. (5.8). These approximations are given by

$$v_i(\boldsymbol{\chi},t) \leftarrow \sum_{j=1}^{N_v} v_i^j(t)\psi_j(\boldsymbol{\chi}), \quad v_i'(\boldsymbol{\chi},t) \leftarrow \sum_{j=1}^{N_v} \frac{\mathrm{d}v_i^j(t)}{\mathrm{d}t}\psi_j(\boldsymbol{\chi}), \qquad i=1,2,$$
$$P(\boldsymbol{\chi},t) \leftarrow \sum_{j=1}^{N_P} P^j(t)\phi_j(\boldsymbol{\chi}). \tag{5.13}$$

Similarly, the temperature and its referential time derivative in the energy equation in Eq. (5.9) are approximated as

$$T(\boldsymbol{\chi},t) \leftarrow \sum_{j=1}^{N_v} T^j(t)\psi_j(\boldsymbol{\chi}), \quad T'(\boldsymbol{\chi},t) \leftarrow \sum_{j=1}^{N_v} \frac{\mathrm{d}T^j(t)}{\mathrm{d}t}\psi_j(\boldsymbol{\chi}). \tag{5.14}$$

In writing the above approximations, $\psi_j$, $j=1,\ldots,N_v$, denote the shape functions used for the velocity components and temperature, while $\phi_j$, $j=1,\ldots,N_P$, denote the pressure shape functions. Once again, we emphasize that the Lagrange interpolation functions $\{\phi_j\}$ are of polynomial order one less than that of $\{\psi_j\}$. For a quadrilateral element with velocity interpolation of order $p$, the number of velocity nodes is $N_v=(p+1)^2$, while the pressure is interpolated using an element of order $p-1$, with $N_P=p^2$ pressure nodes.

In this sense, two interpolation elements are used: one for the velocity components and temperature, and another for the pressure. These two elements are coincident in the spatial domain, but their nodal distributions need not be identical. Except for the corner nodes, the remaining pressure and velocity nodes are not necessarily coincident. The $Q_2Q_1$ element, corresponding to biquadratic velocity interpolation and bilinear pressure interpolation, is used in the present work.

**5.4. Mesh Motion Algorithm.** An important aspect of the ALE framework is the prescription of the mesh motion. The main purpose of using an ALE description is to allow the mesh to have a motion of its own, independent of the material motion. In fluid-flow problems involving moving boundaries, the mesh motion must be prescribed in such a way that the motion of the boundary nodes is propagated smoothly into the interior of the domain. Otherwise, elements in the immediate vicinity of the moving boundary may become excessively distorted.

The mesh-motion problem can be stated as follows: given the prescribed motion of the boundary nodes, determine the motion of the interior nodes of the mesh in the spatial domain. It is not necessary that the entire boundary be moving; the portion of the boundary on which motion is prescribed may be only a subset of the total boundary. On the stationary portion of the boundary, zero mesh velocity must be prescribed so that the mesh-update procedure does not move those nodes.

In this work, we use a simple, effective, and computationally efficient Laplacian mesh-update procedure. At a given instant of time, let $\boldsymbol{w}_\Gamma(t)$ denote the prescribed velocity of the moving boundary $\Gamma(t)$ of the domain $\Omega(t)$. To determine the mesh velocity $\boldsymbol{w}$ in the interior of the domain, we solve the following Dirichlet boundary-value problem:

$$\nabla^2 \boldsymbol{w} = \boldsymbol{0} \ \text{ in } \Omega(t), \quad \boldsymbol{w} = \boldsymbol{w}_\Gamma \ \text{ on } \Gamma(t). \tag{5.15}$$

This corresponds to a system of two uncoupled Laplace equations for the respective components of the mesh velocity, which may be solved using any suitable numerical method, such as the DMCDM or the FEM. The resulting solution smoothly propagates the prescribed boundary velocity into the interior of the domain.

For convenience, the mesh-update problem defined in Eq. (5.15) is solved using the same mesh that is used to interpolate the velocity components. Once the nodal mesh velocities are determined, they are used in the ALE form of the governing equations in Eqs. (5.8) and (5.9), where the mesh velocities appear in the convective transport terms. The resulting system is then solved to determine the unknown flow variables, namely, the velocity, pressure, and temperature.

**5.5. Example: Oscillating Cylinder in Channel Flow.** We consider the problem of flow past an oscillating circular cylinder placed inside a rectangular channel carrying a viscous incompressible fluid, as shown in Fig. 8. This problem is a popular benchmark and has been studied extensively; notable examples include [3, 6, 8].

The cylinder has diameter $D$, and its center is initially located on the horizontal midline of the channel. Starting at time $t = 0$, the cylinder is prescribed a vertical simple harmonic motion given by

$$x_2 = a_0 \sin(\omega t) \implies \boldsymbol{w}_{\text{cyl}} = a_0 \omega \cos(\omega t) \hat{\boldsymbol{e}}_2.$$

For the present analysis, we take $a_0 = 0.4D$, where $a_0$ is the amplitude of cylinder oscillation, and $\omega = 2.1803$ rad/s, based on the data given in [6].

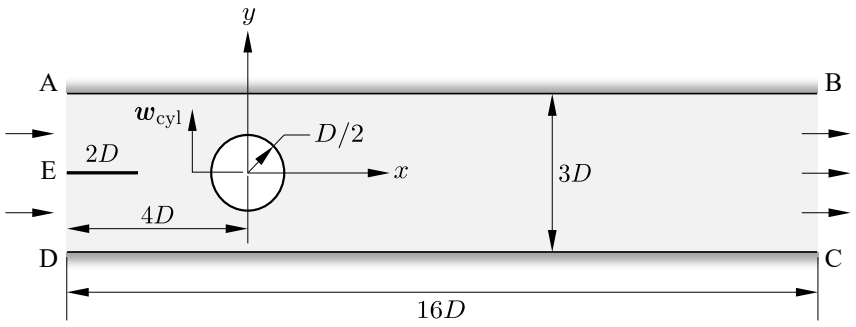


FIGURE 8. Geometry and boundary conditions for the oscillating cylinder in a rectangular channel [6].

At the entrance of the channel, which is located $4D$ upstream of the center of the cylinder, a splitter plate extends $2D$ downstream. The vertical spacing between the channel walls is $3D$. At the inlet, parabolic velocity profiles corresponding to fully developed flow are prescribed separately over the two portions on either side of the splitter. Thus, the inlet boundary conditions on the velocity components are

$$v_1(x=-4D,y)=\begin{cases}\frac{3}{2}v_{\text{avg}}\left[1-(y+0.75D)^2/0.75D^2\right] & -1.5D\leqslant y<0,\\ \frac{3}{2}v_{\text{avg}}\left[1-(y-0.75D)^2/0.75D^2\right] & 0\leqslant y\leqslant 1.5D,\end{cases}$$

$$v_2(x=-4D,y)=0,\quad -1.5D\leqslant y\leqslant 1.5D.$$

The average velocity is taken to be $v_{\text{avg}}=1$. On the top and bottom walls, the no-slip condition is imposed by setting $v_1=v_2=0$. The no-slip condition is enforced on the cylinder surface by setting $\boldsymbol{v}=\boldsymbol{w}_{\text{cyl}}$. On the outflow boundary segment BC, the outflow boundary condition is imposed.

The analysis is performed at a Reynolds number of $\text{Re}=100$ by taking the density as $\rho=1$ and the viscosity as $\mu=0.01$.

At the inlet, $x=-4D$, the temperature is specified as $\theta=1.0$ for $0<y\leqslant 1.5D$ and $\theta=0.0$ for $-1.5D\leqslant y<0$. This represents the mixing of two streams

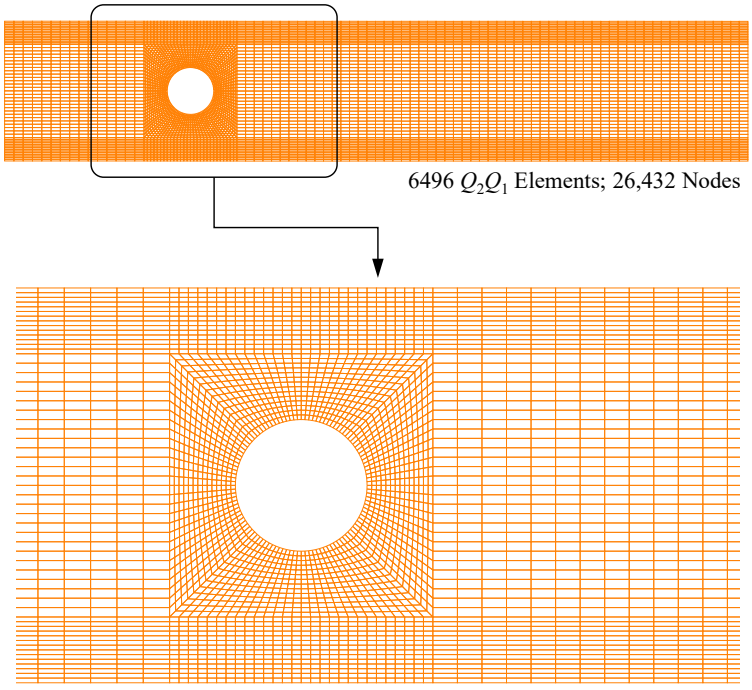


FIGURE 9. Computational mesh used for the oscillating-cylinder problem, consisting of 6496 $Q_2Q_1$ elements and 26,432 nodes. The bottom panel shows a magnified view of the mesh near the cylinder.

at different temperatures, with mixing promoted by the oscillating cylinder. The Peclet number is taken to be Pe = 200, which is achieved by choosing $c = 1.0$ and $k = 0.005$ in the energy transport equation (5.9).

The mesh used for this analysis is shown in Fig. 9; it comprises 6496 $Q_2Q_1$ elements with 26,432 nodes. For time integration, the backward difference integrator is used with a time step of $\Delta t = 0.05$. For the nonlinear iterations, a convergence tolerance of $\varepsilon_{\text{TOL}} = 10^{-3}$ is used.

The results of the analysis are presented in Figs. 10 and 11. Figure 10 shows the velocity-magnitude contours and streamline patterns at three representative instants during the cylinder oscillation. The top and bottom panels correspond to the upper and lower extreme positions of the cylinder, respectively, where the cylinder is momentarily stationary. The middle panel corresponds to the instant at which the cylinder passes through the center of its oscillation while moving downward, where its prescribed velocity is maximum. The streamline patterns show the distortion of the flow caused by the cylinder motion and the resulting unsteady wake development downstream of the cylinder. Figure 11 shows the corresponding

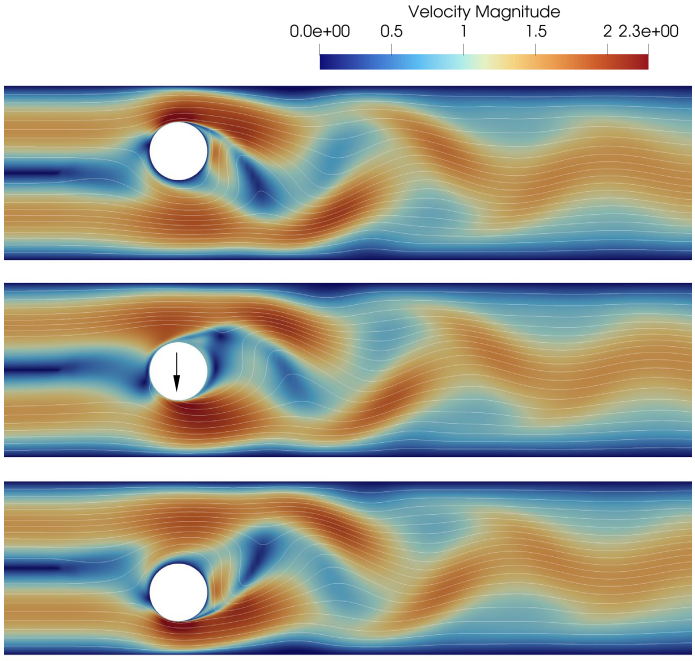


FIGURE 10. Velocity magnitude contours and streamlines for flow past the oscillating cylinder at three representative instants. The top panel shows the cylinder at its upper extreme position, the middle panel shows the cylinder at the center of its oscillation while moving downward, and the bottom panel shows the cylinder at its lower extreme position.

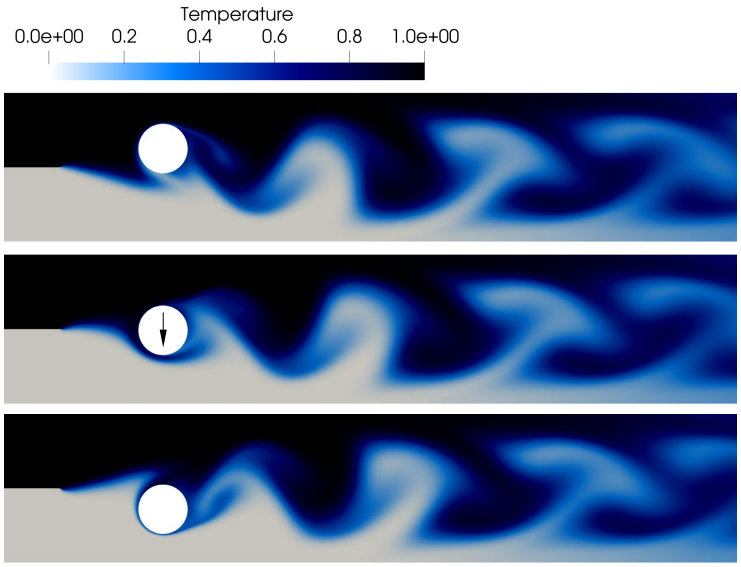


FIGURE 11. Temperature contours for flow past the oscillating cylinder at three representative instants during the oscillation cycle. The top and bottom panels correspond to the extreme positions of the cylinder, while the middle panel corresponds to the cylinder passing through the center of its oscillation while moving downward.

temperature contours at the same three instants. The imposed temperature difference between the two inlet streams is transported downstream, and the oscillatory motion of the cylinder enhances the mixing between the two streams. The characteristic mushroom-like thermal features formed downstream of the cylinder are clearly captured. These results demonstrate the ability of the ALE-based DMCDM formulation to effectively model moving-boundary flow problems with coupled thermal transport.

## 6. Conclusions

In this article, dual mesh control domain formulations were presented for selected nonlinear problems in applied mechanics. A general DMCDM discretization procedure was first reviewed, followed by its application to finite-deformation hyperelasticity and viscous incompressible flow in domains with moving boundaries. For the hyperelasticity problems, the governing equations were written in the reference configuration and solved using a Newton linearization procedure. Numerical examples involving Cook's membrane and a curved beam demonstrated the ability of the method to handle large-deformations and nonlinear material behavior. For the moving-boundary fluid-flow problem, an ALE formulation was used to account for the motion of the computational mesh, and the oscillating-cylinder example

illustrated the application of the method to flow and thermal transport in a deforming domain.

The results indicate that the DMCDM provides a consistent framework for treating nonlinear solid and fluid mechanics problems while retaining the locally conservative property of integral control-domain formulations and the flexibility and modularity of systematic finite element interpolation. Future work will focus on extending the present moving-boundary formulation to fully coupled bidirectional fluid–structure interaction problems, in which the structural deformation and fluid motion are solved in a coupled manner.

**Acknowledgments.** The first and second authors gratefully acknowledge support from the O'Donnell Foundation Chair IV, while the last author acknowledges support from the J. N. Reddy Chair in Applied Mechanics.

## References

1. P. Areias, A. R. Srinivasa, F. Moleiro, J. N. Reddy, *Finite strain analysis with the dual mesh control domain method*, Int. J. Numer. Methods Eng. **126**(1) (2025), e7654.
2. M. Bercovier, O. Pironneau, *Error estimates for finite element method solution of the Stokes problem in the primitive variables*, Numer. Math. **33**(2) (1979), 211–224.
3. A. Beskok, T. C. Warburton, *An unstructured hp finite-element scheme for fluid flow and heat transfer in moving domains*, J. Comput. Phys. **174**(2) (2001), 492–509.
4. J. Bonet, R. D. Wood, *Nonlinear Continuum Mechanics for Finite Element Analysis*, Cambridge University Press, 1997.
5. F. Brezzi, R. S. Falk, *Stability of higher-order Hood–Taylor methods*, SIAM J. Numer. Anal. **28**(3) (1991), 581–590.
6. B. Celik, U. Akdag, S. Gunes, A. Beskok, *Flow past an oscillating circular cylinder in a channel with an upstream splitter plate*, Phys. Fluids **20**(10) (2008), 103603.
7. J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferran, *Arbitrary Lagrangian–Eulerian methods*, in: E. Stein, R. de Borst, T. J. R. Hughes (eds.), *Encyclopedia of Computational Mechanics* **1**, Ch. 14, John Wiley & Sons, Ltd., 2004.
8. S. Ghanbari, M. Sefid, and M. A. Sabbaghi, *Numerical investigating the impact of fluid properties and oscillation frequency on micro mixer efficiency*, Heliyon **11**(1) (2025), e41290.
9. R. Hassani, R. Ansari, H. Rouhi, *Large deformation analysis of 2D hyperelastic bodies based on the compressible nonlinear elasticity: A numerical variational method*, Int. J. Non-Linear Mech. **116** (2019), 39–54.
10. T. Y. Heblekar, J. N. Reddy, A. R. Srinivasa, *Analysis of nonlinear problems using the dual mesh control domain method with arbitrary meshes*, Comput. Methods Appl. Mech. Eng. **427** (2024), 117044.
11. T. Y. Heblekar J. N. Reddy, *An improved dual mesh control domain formulation for the unsteady flow of viscous incompressible fluids*, Phys. Fluids **37**(3) (2025), 033119.
12. T. Y. Heblekar, J. N. Reddy, A. R. Srinivasa, *p-refinement in the dual mesh control domain method*, Comput. Sci. Eng. **28**(1) (2026), 64–73.
13. T. Y. Heblekar, J. N. Reddy, A. R. Srinivasa, *A spectral dual mesh control domain framework for nonlinear fluid flow and coupled multi-field problems*, Int. J. Non-Linear Mech. (2026), 105350.
14. T. Y. Heblekar, J. N. Reddy, A. R. Srinivasa, *The dual mesh control domain method: Theory and applications to functionally-graded beams and orthotropic plates*, in: H. Altenbach, N. Fantuzzi, A. J. M. Ferreira, (eds.), *Compos. Struct. and Technology: In Memoriam to Prof. Cristóvão Manuel Mota Soares*, Springer, 2026, 35–104.
15. T. J. R. Hughes, W. K. Liu, T. K. Zimmermann, *Lagrangian–Eulerian finite element formulation for incompressible viscous flows*, Comput. Methods Appl. Mech. Eng. **29**(3) (1981), 329–349.

16. Z. Jiao, T. Heblekar, G. Wang, R. Xu, W. Chen, J. N. Reddy, *Analysis of plane elasticity problems using the dual mesh control domain method*, Comput. Methods Appl. Mech. Eng. **416** (2023), 116342.
17. Z. Jiao, T. Heblekar, G. Wang, R. Xu, J. N. Reddy, *Static, free vibration, and buckling analysis of functionally graded plates using the dual mesh control domain method*, Comput. Struct. **305** (2024), 107575.
18. Z. Jiao, S. Zhao, G. Wang, R. Xu, J. Yang, *Bending and vibration analyses of functionally graded auxetic doubly curved shells via dual mesh control domain model*, Int. J. Solids Struct. **320** (2025), 113526.
19. P. Nampally, E. Ruocco, J. N. Reddy, *Bending analysis of functionally graded rectangular plates using the dual mesh control domain method*, Int. J. Comput. Methods Eng. Sci. Mech. **22**(5) (2021), 425–437.
20. J. N. Reddy, *An Introduction to Continuum Mechanics*, Cambridge University Press, 2013.
21. J. N. Reddy, *An Introduction to Nonlinear Finite Element Analysis: With Applications to Heat Transfer, Fluid Mechanics, and Solid Mechanics*, Oxford University Press, 2015.
22. J. N. Reddy, *Energy Principles and Variational Methods in Applied Mechanics*, John Wiley & Sons, 2017.
23. J. N. Reddy, *A dual mesh finite domain method for the numerical solution of differential equations*, Int. J. Comput. Methods Eng. Sci. Mech. **20**(3) (2019), 212–228.
24. J. N. Reddy, *Computational Methods in Engineering: Finite Difference, Finite Volume, Finite Element, and Dual Mesh Control Domain Methods*, CRC Press, 2024.
25. J. N. Reddy, N. K. Anand, P. Roy *The Finite Element and Finite Volume Methods for Heat Transfer and Fluid Dynamics*, Cambridge University Press, New York, NY, 2023.
26. J. N. Reddy, D. K. Gartling, *The Finite Element Method in Heat Transfer and Fluid Dynamics*, CRC Press, 2010.
27. J. N. Reddy, N. Kim, M. Martinez, *A dual mesh control domain method for the solution of nonlinear Poisson's equation and the Navier–Stokes equations for incompressible fluids*, Phys. Fluids **32**(9) (2020), 093608.
28. J. N. Reddy, M. Martinez, *A dual mesh finite domain method for steady-state convection–diffusion problems*, Comput. Fluids **214** (2021), 104760.
29. J. N. Reddy, P. Nampally, *A dual mesh finite domain method for the analysis of functionally graded beams*, Compos. Struct. **251** (2020), 112648.
30. J. N. Reddy, P. Nampally, N. Phan, *Dual mesh control domain analysis of functionally graded circular plates accounting for moderate rotations*, Compos. Struct. **257** (2021), 113153.
31. J. N. Reddy, P. Nampally, A. R. Srinivasa, *Nonlinear analysis of functionally graded beams using the dual mesh finite domain method and the finite element method*, Int. J. Non-Linear Mech. **127** (2020), 103575.
32. J. Schröder, T. Wick, S. Reese, P. Wriggers, R. Müller, S. Kollmannsberger, M. Kästner, A. Schwarz, M. Igelbüscher, N. Viebahn, *A selection of benchmark problems in solid mechanics and applied mathematics: J. Schröder et al.*, Arch. Comput. Methods Eng. **28**(2) (2021), 713–751.
33. E. B. Tadmor, R. E. Miller, R. S. Elliott, *Continuum Mechanics and Thermodynamics: From Fundamental Concepts to Governing Equations*, Cambridge University Press, 2012.
34. C. Taylor, P. Hood, *A numerical solution of the Navier–Stokes equations using the finite element technique*, Comput. Fluids **1**(1) (1973), 73–100.

## АНАЛИЗА КОНТРОЛЕ ДОМЕНА СА ДВОСТРУКОМ МРЕЖОМ НЕЛИНЕАРНИХ ПРОБЛЕМА У МЕХАНИЦИ

Резиме. Метод контроле домена са двоструком мрежом, који је представио Реди [23], је хибридна нумеричка техника која комбинује карактеристике методе коначних елемената и методе коначних запремина. Користи тврђења интегралног баланса, као у методи коначних запремина, заједно са интерполацијом зависних непознатих функција коначним елементима, као у методи коначних елемената. Ова комбинација пружа погодан и локално конзервативан оквир за просторну дискретизацију граничних проблема. У овом чланку представљамо формулације домена управљања са двоструком мрежом за нелинеарне проблеме који се јављају у примењеној механици. Фокус је на две репрезентативне класе проблема: коначна деформација хипереластичних чврстих тела и ток вискозних нестишљивих флуида у доменима са покретним границама. За проблем хипереластичности, једначине су написане у референтној конфигурацији и решене коришћењем Њутнове линеаризације. За проблем тока са покретним границама, користи се произвољна Лагранж-Ојлерова (ALE) формулација да би се објаснило кретање рачунске мреже. Представљени су нумерички примери да би се илустровала имплементација методе и проценила предложена формулација.

(Received 01.06.2026)
(Available online 28.07.2026)

J. Mike Walker '66 Department of Mechanical Engineering
Texas A&M University
College Station
USA
tanmayeyashodan.h@tamu.edu
0009-0009-6400-889X

J. Mike Walker '66 Department of Mechanical Engineering
Texas A&M University
College Station
USA
jnreddy@tamu.edu
0000-0002-9739-1639

J. Mike Walker '66 Department of Mechanical Engineering
Texas A&M University
College Station
USA
arun-r-srinivasa@tamu.edu
0000-0002-9825-7482