An Effective Overlapping Finite Element Method: The Method of Finite Spheres for Three-Dimensional Linear Elasticity Problems

by

Benjamin Lai

B.A., Bard College at Simon's Rock, 2009
B.S., Columbia University, 2009
M.S., Stanford University, 2011

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2016

Author . . . . . .
Benjamin Lai
Department of Mechanical Engineering
January 8, 2016

Certified by . . . . . .
Klaus-Jürgen Bathe
Professor of Mechanical Engineering
Thesis Supervisor

Accepted by . . . . . .
Rohan Abeyaratne
Professor of Mechanical Engineering
Chairman, Departmental Committee on Graduate Students

An Effective Overlapping Finite Element Method:
The Method of Finite Spheres for Three-Dimensional Linear Elasticity Problems

by

Benjamin Lai

Submitted to the Department of Mechanical Engineering
on January 8, 2016 in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

Abstract

The method of finite spheres is an effective overlapping finite element method developed to overcome challenges in mesh-based numerical methods. Commonly recognized challenges include mesh generation for geometrically complex domains, severe element distortion in nonlinear analysis with large strain effects, and modeling problems involving discontinuities and singularities which require mesh alignment and refinement. Substantial research efforts have been focused on addressing these issues, resulting in the introduction of numerous meshless methods. The ultimate purpose of the method of finite spheres is to be distinguished as a reliable and efficient meshless computational technique for the solution of boundary value problems on complex domains, to supplement the capabilities of the standard finite element method.

The reliability of the method of finite spheres was previously verified for one- and two-dimensional linear static analysis of solids and fluids. The objective of this thesis is to demonstrate the reliability and effectiveness of the method of finite spheres for the solution of practical three-dimensional linear elasticity problems. An effective local approximation space, which is multiplied by the Shepard partition of unity function, is presented for the construction of three-dimensional interpolation functions. The piecewise Gauss-Legendre quadrature rule, a simple and efficient scheme for the integration of complicated nonpolynomial basis functions, is introduced for three-dimensional spherical domains. The three-dimensional formulation of the method of finite spheres is implemented in a commercial finite element analysis program with the user-element subroutine, in order to perform a computational efficiency comparison with the standard finite element method.

A series of increasingly complex three-dimensional problems is studied (1) to establish the reliability of the method of finite spheres for practical three-dimensional linear elastic static problems and (2) to assess the effectiveness of the method of finite spheres as compared to the standard finite element method. Solution times indicate that the method of finite spheres is approximately an order of magnitude slower than the standard finite element method during the computation phase. However, this is still a promising result since there are significant time savings for the method of finite spheres during the pre-processing phase, particularly in the discretization of complicated three-dimensional geometries.

Thesis Supervisor: Klaus-Jürgen Bathe
Title: Professor of Mechanical Engineering

Acknowledgements

My experience at MIT has been an extremely rewarding and influential chapter of my life. I am sincerely grateful to my thesis advisor, Professor Klaus-Jürgen Bathe, for his guidance and support throughout my doctoral program. It has been a great privilege to have the mentorship of a truly world-class academic and a passionate innovator in the field of finite element analysis.

I would also like to express genuine gratitude to my thesis committee members. Professor Tomasz Wierzbicki has been extremely generous and supportive of my research, and I truly appreciate the opportunity to work with him. I would also like to thank Professor Nicholas Patrikalakis for his valued expertise and insights on my research topic.

A special thanks to the employees at ADINA R&D, Inc. for their assistance with ADINA. In particular, I am grateful to Jian Dong for his depth of knowledge and support with the implementation of the method of finite spheres in ADINA.

Finally, I would like to thank the MIT community, my colleagues in the Finite Element Research Group and Impact and Crashworthiness Laboratory, and all my friends and family for their encouragement and unconditional support throughout my academic career.

Contents

List of Figures

List of Tables

Chapter 1

Introduction to the overlapping finite element method

The finite element method is used to solve physical problems in virtually every field of engineering analysis due to its effectiveness and general applicability. However, in the analysis of some practical engineering problems, inaccurate finite element solutions may arise due to a poor finite element discretization. The objective of this thesis is to present an effective and reliable overlapping finite element method, known as the method of finite spheres, which circumvents the requirement of constructing a mesh.

1.1 Challenges in finite element analysis

The finite element method requires the discretization of a problem domain into a set of non-overlapping elements. Difficulties in discretization typically arise in complex three-dimensional domains. Not only is this process of mesh generation time-consuming, but oftentimes automatic mesh generators are unable to obtain a good quality mesh and

special attention is required to remove distorted elements. For example, elements need to satisfy certain aspect ratio and included angle criteria in order to ensure there is not a possible loss of predictive capability. Otherwise, the effects of element distortion are that the same order of polynomials are not represented as without the distortion, and so nonpolynomial functions must be integrated leading to inaccuracies in numerical integration. Consequences of a poorly constructed finite element discretization are an increase in computational time due to individual small elements or distorted elements which decrease the stable time step, as well as a significant loss in solution accuracy [1,2].

While the procedure of mesh generation poses one challenge in the finite element method, the modeling and analysis of certain types of problems such as nonlinear large deformation analysis or problems involving discontinuities and singularities (e.g., crack propagation) leads to an additional challenge. For these problems, frequent remeshing and/or refinement of the domain may be required due to severe mesh distortion or sensitivity to mesh alignment.

Challenges in finite element analysis have led to the development of meshless methods where boundary value problems can be solved without a predefined mesh. Numerous meshless techniques have been proposed and will be discussed in Section 1.2.

1.2 Summary of meshless methods

The objective of recently developed meshless methods is to eliminate the challenges encountered in problems that are not well suited to conventional computational methods such as the finite element method [3,4]. While there are a variety of techniques proposed for various applications, the most common meshless methods include smoothed particle hydrodynamics (SPH), diffuse element method (DEM), element-free Galerkin

method (EFG), and meshless local Petrov-Galerkin (MLPG) method. These methods will be summarized to illustrate the common features of meshless methods and their relative properties.

The SPH method was the first meshless method to be introduced and was initially developed to model astrophysical phenomena. It is a Lagrangian particle method in which the computational domain is represented by discrete points. The method represents the field quantity in an integral form based on kernel approximation functions. Common difficulties that arise include tensile instability and spurious boundary effects. Tensile instability refers to an unstable solution when tensile stresses are present. Boundary deficiency is a consequence of not satisfying zeroth-order consistency near or on the boundary of the problem domain. Several approaches have been proposed to address these difficulties, including utilizing stress points, which is increasingly complex in multi-dimensional problems. While improvements to the standard SPH method are being researched, in general the scheme requires a large number of particles, particularly in three-dimensional problems, to obtain reasonably accurate solutions. Furthermore, the use of adjustable solution factors makes the method inadequate for certain types of problems such as simple linear elastic static problems [5,6].

The DEM approximation is defined by a local weighted least squares polynomial fitting based on a variable number of nodes surrounding a particular point. The purpose of DEM is to provide smooth approximations using only a set of discretization points. While advantages over FEM include not relying on a mesh and providing more accurate evaluation of the derivatives of the unknown functions, there are a number of oversimplifications which affect the adequacy of the method. In particular, the derivative

of the approximation functions is inaccurate, a very low quadrature rule for numerical integration is applied, and the Dirichlet boundary conditions are not accurately enforced. Consequently, DEM does not pass the patch test and fails consistency requirements. Furthermore, the method involves integrals over the global domain and boundary implying it is based on the global weak form. Therefore, a background mesh is required for numerical integration suggesting the method is only meshless with regard to constructing interpolation functions [7].

The EFG method uses moving least squares (MLS) interpolants to construct the trial and test functions. In contrast to DEM, the method accurately computes the derivatives of the approximation functions. While the method also uses a background cell structure similar to EFG, a much larger number of integration points are used. Since the interpolation functions do not satisfy the Kronecker delta property, the imposition of the Dirichlet boundary conditions requires utilizing Lagrange multipliers or the penalty method. However, the former leads to a larger system of algebraic equations and a loss of positive definiteness reducing computational efficiency, while the latter requires appropriately selecting the penalty factors which is not straightforward as it will depend on the problem being considered. While MLS provides reasonable accuracy, it requires an expensive matrix inversion at every evaluation point and adds the constraint that at every integration point there is a minimum number of spheres that must have nonzero support. While an intrinsic basis approximation avoids adding additional degrees of freedom to the problem, the complications concerning matrix inversion and a minimum overlap criterion significantly reduce the computational efficiency of the method [8,9].

The MLPG method is a concept that can adopt trial and test functions from different approximation spaces, resulting in various formulations which offer flexibility to deal with different boundary value problems. Unlike DEM and EFG, the MLPG method works with a local weak form instead of a global weak form, which means that numerical integration is performed over local subdomains rather than using a background mesh or cell structure. Therefore, it is a truly meshless method since a mesh is not required for either interpolation or integration. When the approximation functions are based on the MLS functions, the method suffers from the same complications as DEM and EFG. Ultimately, the method has been successfully applied to a wide range of problems but is still less computationally efficient than the finite element method [10,11].

Although a variety of meshless techniques have been developed, the currently available methods are generally much more expensive than the finite element method and come with various complications that impact their overall effectiveness towards solving a large class of problems. In the next section, we will discuss the motivation for overlapping elements which takes the advantages of both finite elements and meshless methods and focuses on being a reliable and computationally efficient method.

1.3 Motivation for overlapping finite elements

The method of finite spheres is a truly meshless overlapping finite element method. The most important issues to be considered in developing the method are reliability and computational efficiency. In order to truly assess reliability and computational efficiency, it is important to solve practical three-dimensional problems, where the traditional finite element method suffers from costly mesh generation procedures and errors resulting from element distortion. We will further motivate overlapping finite elements by discussing the

basic procedures of the method of finite spheres in comparison to the finite element method.

In the method of finite spheres, the discretization is performed using interpolation functions that are compactly supported over spherical domains, and since the elements are allowed to overlap, discretization is simplified and element distortion is avoided as compared to the finite element method. Spherical domains are chosen as the element type because there is no directional bias and their positions, as well as the regions of sphere overlap, are completely determined by their center coordinates and radii. Being truly meshless is the primary motivation for the method of finite spheres, and efficient selection of interpolation functions and numerical integration procedures will distinguish the method from other meshless methods to be computationally competitive compared to the finite element method [12-14].

The interpolation functions for the method of finite spheres are based on using the Shepard partition of unity functions. Since these are rational, nonpolynomial functions with only zeroth-order consistency, a local approximation space is defined at each node to generate interpolation functions with higher-order consistency. Although having an extrinsic basis increases the number of degrees of freedom per node and hence the overall system of equations, it does not add the difficulties of matrix inversion that are characteristic of meshless methods based on an intrinsic basis, such as when using MLS functions. Furthermore, the flexibility of selecting the functions in the local basis is an advantage since the solution space can be optimized based on the type of problem being solved and different local approximation spaces can be defined in different regions of the domain allowing for local enrichment. In the traditional finite element method, the

interpolation functions are piecewise continuous polynomials, where the mesh is expected to be compatible and the elements are undistorted. The functions satisfy the Kronecker delta property which ensures that the Dirichlet boundary conditions can be easily enforced. For the method of finite spheres, the interpolation functions in general do not satisfy the Kronecker delta property at the nodes, unless a special arrangement of spheres is employed. Therefore, efficient imposition of the Dirichlet boundary conditions is more complicated, except in the special case of uniformly arranged spheres along the boundary.

For the method of finite spheres, numerical integration is a focal point of development towards becoming computationally efficient compared to other meshless methods, as well as the finite element method. The method requires integration of nonpolynomial functions over complicated integration domains, such as spheres, truncated spheres, and general lens-shaped regions for the overlap of spheres. Since exact integration is not possible, the goal is to obtain a solution of sufficient accuracy using a minimum number of integration points. In the finite element method, numerical integration is efficient since the interpolation functions are polynomial and Gauss-Legendre quadrature ensures polynomials are exactly integrated with minimal cost.

The motivation for overlapping finite elements is ultimately in the ability to reliably and efficiently solve complex boundary value problems without the complication of constructing a good quality mesh, which is a requirement for an accurate finite element solution. The main advantage is obtained by simply discretizing the domain with an arrangement of overlapping spheres. To achieve the same level of accuracy as a finite element solution, typically a fewer number of spheres than elements is required, although each node in the method of finite spheres will have more degrees of freedom than each

node in the finite element method. Furthermore, using the method of finite spheres allows for selecting interpolation functions which are most effective for solving certain types of problems. However, these advantages come at the cost of more expensive numerical integration due to nonpolynomial interpolation functions. With advances in computational performance, it is anticipated that the advantages of the method of finite spheres will be worth the additional costs of computations, making it a powerful numerical procedure.

1.4 Research objectives

The overall purpose of developing the method of finite spheres is to demonstrate its reliability and efficiency for practical commercial and research applications. The reliability of the method was previously verified for one- and two-dimensional linear static analysis of solids and fluids [12]. While efficiency was also studied, the most dependable assessment of efficiency requires analysis of three-dimensional problems.

The objective of this thesis is to demonstrate the reliability and computational efficiency of the method of finite spheres for the solution of practical three-dimensional linear elasticity problems. First, we develop the theory and formulation of the method of finite spheres, an effective overlapping finite element method, for three-dimensional static linear elasticity problems to overcome the challenges encountered in mesh-based numerical methods. To make this method competitive compared to other meshless methods and the finite element method, we next propose an effective numerical integration scheme, known as the piecewise Gauss-Legendre quadrature rule. Lastly, we implement the method of finite spheres in the commercial program ADINA, to assess the efficiency of the method by comparing solution times with the finite element method for about the same level of accuracy.

1.5 Thesis outline

In this section, we will briefly outline the content of the remaining chapters of this thesis, focusing on the contributions of this research on the method of finite spheres for three-dimensional linear elasticity problems.

In Chapter 2, we present the theory and formulation of the method of finite spheres. This includes a discussion of the general requirements for discretization, construction of the interpolation functions based on the Shepard partition of unity and an approximation space for three-dimensional elliptic problems, and the general formulation of the displacement-based method of finite spheres from the governing differential equations to the discrete equilibrium equations.

In Chapter 3, we start with an overview of numerical integration for the finite element method and meshless methods. Since the method of finite spheres aims to be computationally efficient and has not been previously developed for three-dimensional problems, we discuss the various sphere integration domains encountered and present the piecewise Gauss-Legendre quadrature rule for each of these domains.

In Chapter 4, implementation of the method of finite spheres in a commercial finite element software package is discussed. The three-dimensional linear sphere element, to be used for our three-dimensional linear elasticity numerical examples, is introduced and compared with the three-dimensional linear brick element. In particular, we will compare the nodal degrees of freedom, local element degrees of freedom, and global structure degrees of freedom for finite elements and finite spheres. Then we will present the flowchart for implementing the method of finite spheres in the commercial program

ADINA and the user-element subroutine used to calculate the element stiffness matrix and element load vector.

In Chapter 5, we study three-dimensional static linear elasticity problems since these problems are best for assessing the computational efficiency of the method of finite spheres, where the solution is expensive with regard to numerical integration and bandwidth of the system of equations. In particular, we analyze three increasingly complex three-dimensional models, each considering two load cases. For each numerical example, we establish reliability of the method of finite spheres by showing the accuracy of the solution compared to the finite element solution, in terms of overall strain energy and displacements and stresses along model lines. To assess computational efficiency of the method of finite spheres, the computational time for each numerical example is compared with the finite element solution time, for a comparable level of accuracy determined by the strain energy error.

In Chapter 6, we summarize the major contributions of this thesis and limitations of our current work. Based on the encouraging results of this work, we discuss broader implications and possible future work for the method of finite spheres, namely the strengths of a finite element and finite sphere coupled formulation for three-dimensional problems and the significant benefits of using distributed memory parallel processing. Ultimately, we hope that continuous research efforts will lead to widespread commercial applicability of the method of finite spheres.

Chapter 2

Theory and formulation of the method of finite spheres

This chapter will present a discussion of the theory and formulation of the method of finite spheres. The four basic procedures for the method of finite spheres are discretization, interpolation, integration, and equilibrium equations. For discretization, we will discuss the requirements for a valid sphere arrangement. The interpolation functions are based on Shepard partition of unity functions, which are nonpolynomial, as compared to the finite element method which uses polynomial interpolation functions. Consequently, numerical integration is much more costly for the method of finite spheres, so we introduce the piecewise Gauss-Legendre quadrature rule, a reasonably simple and efficient integration technique. Lastly, the equilibrium equations for linear elasticity problems are presented, which is similar to the finite element method, since the method is also based on the Galerkin weak form.

2.1 Sphere discretization

To solve a three-dimensional boundary value problem, we must come up with a valid sphere arrangement for the discretization of a three-dimensional domain. Figure 2.1 illustrates a general three-dimensional domain V discretized using a set of nodes. Each node I corresponds to a sphere $B(\underline{x}_I, r_I)$, where $\underline{x}_I$ is the coordinates of the center of the sphere and r_I is the radius of the sphere. Spheres can be classified as either interior spheres, which lie completely inside the domain, or boundary spheres, which intersect the boundary of the domain $S = S_u \bigcup S_f$, where S_u is the Dirichlet boundary and S_f is the Neumann boundary. The unit normal to the domain boundary, $\underline{n}$, is positive in the outward direction.

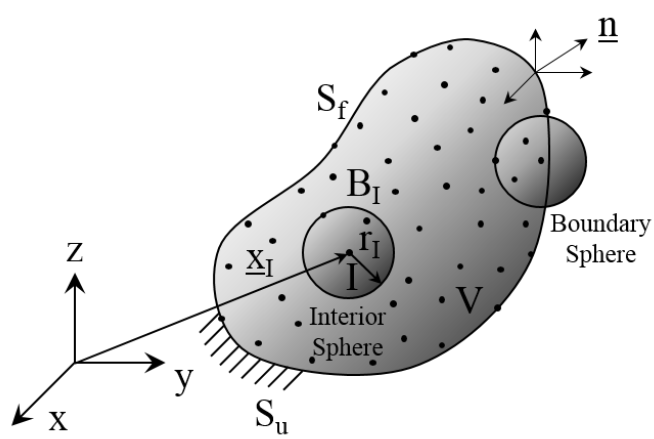


Figure 2.1 General three-dimensional problem domain V with domain boundary $S = S_u \bigcup S_f$ discretized using a set of nodes, each corresponding to either an interior or boundary sphere

The discretization of a general three-dimensional domain is defined by the coordinates and radius of each sphere, where N is the total number of spheres. There are a few general requirements to a valid sphere discretization, namely, that all nodal points must be within the domain,

$$\underline{x}_I \subset V,\ I = 1,\ldots,N \tag{2.1}$$

the domain must be completely covered by the union of all spheres,

$$V \subset \bigcup_{I=1}^{N} B\left(\underline{x}_I, r_I\right) \tag{2.2}$$

and that no sphere is completely included in any other sphere,

$$B\left(\underline{x}_I, r_I\right) \cap B\left(\underline{x}_J, r_J\right) < \min\left(B\left(\underline{x}_I, r_I\right), B\left(\underline{x}_J, r_J\right)\right),\ I \neq J \tag{2.3}$$

As compared to the finite element method, discretization is simplified since a conforming mesh is no longer a requirement and elements are now allowed to be overlapping. Furthermore, in the implementation of these methods, the method of finite spheres offers reduced data storage in defining the discretization, since spheres are only defined by the position vector of the centroid and its radius. There are several node generation schemes (e.g., hierarchical tree-based discretization, biting method, Monte Carlo probabilistic techniques), but we will use a uniform arrangement of spheres in solving the three-dimensional linear elasticity numerical examples [15,16].

2.2 Interpolation scheme

The interpolation scheme for the method of finite spheres is based on the partition of unity paradigm. The Shepard partition of unity functions and local approximation space are chosen with computational efficiency and solution accuracy as considerations. In this section, we will discuss the conditions for an effective approximation scheme, the construction of the Shepard partition of unity functions, and the choice of local approximation space for linear elasticity problems.

Several requirements should be considered when constructing an interpolation scheme, the most important of which are consistency and continuity, but also desirable are local approximability and localization by compact support [12]. The consistency condition,

as in the finite element method, depends on the degree of the governing equations. For example, when solving linear elasticity problems using the displacement-based formulation, the interpolation functions should not only be able to reproduce constant functions, i.e., rigid body modes, but also linear functions, i.e., constant strain states, so the functions should at least satisfy first-order consistency. The formulation of the discrete equilibrium equations requires taking derivatives, and so the interpolation functions should satisfy certain minimal continuity conditions. Since the Shepard partition of unity functions only satisfy zeroth-order consistency, a local approximation space is defined for higher-order consistency. Unlike the finite element method, the method of spheres offers a choice in defining the local approximation space, which allows incorporating specific functions in the global approximation space in order to more efficiently solve certain types of problems where the nature of the solution is known. Localization by compact support is important because it allows for localization of the approximation, the stiffness matrix is banded since only a few spheres overlap at any given point of the domain, and h- and p-refinement is easily employed.

2.2.1 Shepard partition of unity functions

The partition of unity paradigm has been adopted to generate compactly supported interpolation functions for the method of finite spheres [17-20]. The first step in the partition of unity method is to generate the partition of unity functions.

Let V be an open bounded three-dimensional domain and let S be the domain boundary, with $S = S_u \cup S_f$ and $S_u \cap S_f = 0$. Then let $\left\{ B\left(\underline{x}_I, r_I\right); I = 1,\ldots,N \right\}$ be a set of spheres which form a covering for V, i.e., $V \subset \bigcup_{I=1}^{N} B\left(\underline{x}_I, r_I\right)$. A system of functions $\left\{\varphi_I\right\}_{I=1}^{N}$

is defined as a partition of unity subordinate to the open cover $\{B(\underline{x}_I, r_I)\}$ if the following are satisfied:

1. $$\sum_{I=1}^{N} \varphi_I(\underline{x}) = 1 \ \forall \underline{x} \in V \tag{2.4}$$

2. $$supp(\varphi_I(\underline{x})) \subset B(\underline{x}_I, r_I) \tag{2.5}$$

3. $$\varphi_I(\underline{x}) \in C_0^s; \ s \geq 0 \tag{2.6}$$

Let $W_I(\underline{x})$ denote a positive radial weighting function with the following properties:

1. $$W_I(\underline{x}) \in C_0^s; \ s \geq 0 \tag{2.7}$$

2. $$supp(W_I) \subset B(\underline{x}_I, r_I) \tag{2.8}$$

The Shepard partition of unity function is then given by

$$\varphi_I^0(\underline{x}) = \frac{W_I}{\sum_{J=1}^{N} W_J}, \ I = 1, \ldots, N \tag{2.9}$$

These functions are rational, nonpolynomial functions satisfying zeroth-order consistency, ensuring that rigid body modes can be reproduced exactly. The choice of weighting function should consider the continuity class and the ease of differentiation and integration so that low cost partitions of unity are obtained. We use the quartic spline weighting function defined as

$$W_I(s) = \begin{cases} 1 - 6s^2 + 8s^3 - 3s^4 & , \ 0 \leq s \leq 1 \\ 0 & , \ s > 1 \end{cases} \tag{2.10}$$

where $s = (\|\underline{x} - \underline{x}_I\|)/r_I$.

2.2.2 Approximation space

Since the Shepard partition of unity functions only satisfy zeroth-order consistency, a local approximation space $V_I^h = \text{span}_{m \in \mathcal{J}} \{p_m(\underline{x})\}$ is defined at each node I to generate

approximation spaces of higher-order consistency, where h is a measure of the sphere size, $\mathcal{J}$ is an index set, and $p_m\left(\underline{x}\right)$ is a member of the local basis. The global approximation space V_h is generated by multiplying the Shepard partition of unity function at each node I with the functions from the local basis

$$V_h = \sum_{I=1}^{N} \varphi_I^0 V_I^h \tag{2.11}$$

Hence any function v_h in the solution space V_h can be written as

$$v_h\left(\underline{x}\right) = \sum_{I=1}^{N} \sum_{m\in\mathcal{J}} h_{Im}\left(\underline{x}\right)\alpha_{Im} \tag{2.12}$$

where the interpolation functions h_{Im} are defined as

$$h_{Im}\left(\underline{x}\right) = \varphi_I^0\left(\underline{x}\right) p_m\left(\underline{x}\right) \tag{2.13}$$

and are associated with α_{Im}, the mth degree of freedom at node I. Therefore, the number of degrees of freedom at each node depends on the number of functions included in the local basis at each node. Proper selection of the local basis functions will depend on the degree of the governing differential equations and the nature of the solutions to be predicted. For example, trigonometric interpolation functions are suitable for hyperbolic problems, which has been previously demonstrated for the dynamic analysis of wave propagations [21]. For the solution of elliptic problems, such as for three-dimensional linear elasticity problems, a suitable local approximation space is

$$V_I^h = \text{span}\left\{1, \overline{x}, \overline{y}, \overline{z}, \overline{xy}, \overline{yz}, \overline{zx}\right\} \tag{2.14}$$

containing terms of a complete first-order polynomial, where $\overline{x} = \left(x - x_I\right)/r_I$, $\overline{y} = \left(y - y_I\right)/r_I$, and $\overline{z} = \left(z - z_I\right)/r_I$.

Regarding continuity of the global approximation, the displacement field is continuous provided the weighting function W_I and the terms in the local approximation

space $p_m(\underline{x})$ are continuous. The stress field is continuous provided the derivatives of the weighting function with respect to the spatial coordinates are continuous and the functions $p_m(\underline{x})$ and their derivatives are sufficiently smooth. The derivatives of the weight function are continuous provided that W_I has zero slope at the center of the sphere and on the surface of the sphere, which is satisfied for the chosen quartic spline weighting function.

2.3 Displacement-based method of finite spheres

In this section, we present the formulation of the displacement-based method of finite spheres for three-dimensional linear elasticity problems. We first present the governing differential equations for the elasticity problem to be solved, followed by a discussion of the variational formulation and nodal interpolations which are used to derive the discrete equilibrium equations.

2.3.1 Governing differential equations

The governing differential equations for a linear elastic continuum $V \in R^3$ with domain boundary S are

$$\underline{\partial}_{\varepsilon}^{T}\underline{\tau} + \underline{f}^{B} = \underline{0} \text{ in } V \tag{2.15}$$

with Neumann boundary conditions

$$\underline{N}\underline{\tau} = \underline{f}^{S} \text{ on } S_f \tag{2.16}$$

and Dirichlet boundary conditions

$$\underline{u} = \underline{u}^{S} \text{ on } S_u \tag{2.17}$$

The strain-displacement relation is given by

$$\underline{\varepsilon} = \underline{\partial}_{\varepsilon}\underline{u} \tag{2.18}$$

and the linear elastic constitutive relation is given by

$$\underline{\tau} = \underline{C}\underline{\varepsilon} \tag{2.19}$$

In Eqs. (2.15-2.19), $\underline{u}$ is the displacement vector, $\underline{\varepsilon}$ is the strain vector, $\underline{\tau}$ is the stress vector, $\underline{f}^B$ is the body force vector, $\underline{f}^S$ is the prescribed traction vector on the Neumann boundary S_f, $\underline{u}^S$ is the prescribed displacement vector on the Dirichlet boundary S_u, $\underline{\partial}_\varepsilon$ is a linear gradient operator, $\underline{N}$ is the direction cosine matrix for the unit normal to the domain boundary (positive outwards), and $\underline{C}$ is the elasticity matrix.

2.3.2 Variational formulation

For the linear elastic domain $V \in R^3$, the variational indicator is

$$\prod(\underline{u}) = \frac{1}{2}\int_V \underline{\varepsilon}^T(\underline{u})\underline{C}\underline{\varepsilon}(\underline{u})dV - \Re \tag{2.20}$$

where the term $\Re$ accounts for the externally applied body forces, surface tractions, and prescribed displacements, given by

$$\Re = \int_V \underline{u}^T \underline{f}^B dV + \int_{S_f} \underline{u}^T \underline{f}^S dS + \int_{S_u} \underline{f}^{uT}\left(\underline{u} - \underline{u}^S\right)dS \tag{2.21}$$

The traction vector $\underline{f}^u$ on the Dirichlet boundary may be expressed as

$$\underline{f}^u = \underline{N}\underline{C}\underline{\varepsilon}(\underline{u}) \tag{2.22}$$

By invoking the stationarity of the variational indicator $\prod$ in Eq. (2.20), we obtain the following weak form:

Find $\underline{u} \in H^1(V)$ such that

$$\begin{aligned}&\int_V \underline{\varepsilon}^T(\underline{v})\underline{C}\underline{\varepsilon}(\underline{u})dV - \int_{S_u}\left[\underline{\varepsilon}^T(\underline{v})\underline{C}\underline{N}^T\underline{u} + \underline{v}^T\underline{N}\underline{C}\underline{\varepsilon}(\underline{u})\right]dS \\ &\quad = \int_V \underline{v}^T\underline{f}^B dV + \int_{S_f}\underline{v}^T\underline{f}^S dS - \int_{S_u}\underline{\varepsilon}^T(\underline{v})\underline{C}\underline{N}^T\underline{u}^S dS \;\; \forall \underline{v} \in H^1(V)\end{aligned} \tag{2.23}$$

where $H^1(V)$ is the first-order Hilbert space.

When the variational formulation in Eq. (2.23) is satisfied with stresses obtained through the linear elastic constitutive relation in Eq. (2.19) from a continuous displacement field that satisfies the displacement boundary conditions on the Dirichlet boundary, then all three fundamental requirements of mechanics are fulfilled, namely equilibrium, compatibility, and the stress-strain law are all satisfied.

2.3.3 Nodal interpolations

For three-dimensional analysis, the displacement field approximation is

$$\underline{u}(x,y,z)=\begin{Bmatrix} u \\ v \\ w \end{Bmatrix}=\sum_{J=1}^{N}\sum_{n\in\mathcal{J}}\underline{H}_{Jn}(x,y,z)\underline{\alpha}_{Jn}=\underline{H}(x,y,z)\underline{U} \tag{2.24}$$

The corresponding strain field is

$$\underline{\varepsilon}(x,y,z)=\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix}=\sum_{J=1}^{N}\sum_{n\in\mathcal{J}}\underline{B}_{Jn}(x,y,z)\underline{\alpha}_{Jn}=\underline{B}(x,y,z)\underline{U} \tag{2.25}$$

and the corresponding stress field is

$$\underline{\tau}(x,y,z)=\begin{Bmatrix} \tau_{xx} \\ \tau_{yy} \\ \tau_{zz} \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix}=\sum_{J=1}^{N}\sum_{n\in\mathcal{J}}\underline{C}\underline{B}_{Jn}(x,y,z)\underline{\alpha}_{Jn}=\underline{C}\underline{B}(x,y,z)\underline{U} \tag{2.26}$$

where $\underline{U}=\begin{bmatrix}\underline{\alpha}_{11} & \underline{\alpha}_{12} & \cdots & \underline{\alpha}_{Jn} & \cdots\end{bmatrix}^T$ is the vector of nodal unknowns for the method of finite spheres, $\underline{\alpha}_{Jn}=\begin{bmatrix}u^{Jn} & v^{Jn} & w^{Jn}\end{bmatrix}$ is the vector of nodal unknowns at node J

corresponding to the nth degree of freedom, and u^{Jn}, v^{Jn}, and w^{Jn} are the x-, y-, and z-displacements, respectively, at node J corresponding to the nth degree of freedom.

The displacement interpolation matrix is

$$\underline{H}_{Jn}(x,y,z)=\begin{bmatrix} h_{Jn} & 0 & 0 \\ 0 & h_{Jn} & 0 \\ 0 & 0 & h_{Jn} \end{bmatrix} \tag{2.27}$$

and the strain-displacement matrix is

$$\underline{B}_{Jn}(x,y,z)=\underline{\partial}_{\varepsilon}\underline{H}_{Jn}(x,y,z)=\begin{bmatrix} \frac{\partial h_{Jn}}{\partial x} & 0 & 0 \\ 0 & \frac{\partial h_{Jn}}{\partial y} & 0 \\ 0 & 0 & \frac{\partial h_{Jn}}{\partial z} \\ \frac{\partial h_{Jn}}{\partial y} & \frac{\partial h_{Jn}}{\partial x} & 0 \\ 0 & \frac{\partial h_{Jn}}{\partial z} & \frac{\partial h_{Jn}}{\partial y} \\ \frac{\partial h_{Jn}}{\partial z} & 0 & \frac{\partial h_{Jn}}{\partial x} \end{bmatrix} \tag{2.28}$$

The elasticity matrix is given by

$$\underline{C}=\frac{E(1-\nu)}{(1+\nu)(1-2\nu)}\begin{bmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-\nu} & 1 & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix} \tag{2.29}$$

with E and ν being Young's modulus and Poisson's ratio of the material, respectively.

2.3.4 Discrete equilibrium equations

In this section, we will derive the method of finite spheres equilibrium equations for the response of a general three-dimensional domain. By substituting Eqs. (2.24-2.26) into Eq. (2.23), we obtain the discretized system of algebraic equations corresponding to node I and degree of freedom m

$$\sum_{J=1}^{N}\sum_{n\in\mathcal{J}} \underline{K}_{ImJn}\underline{\alpha}_{Jn} = \underline{f}_{Im} + \hat{\underline{f}}_{Im} \tag{2.30}$$

where the stiffness matrix is

$$\underline{K}_{ImJn} = \int_{V_I} \underline{B}_{Im}^{T}\underline{C}\underline{B}_{Jn}dV \tag{2.31}$$

and the body force load vector is

$$\underline{f}_{Im} = \int_{V_I} \underline{H}_{Im}\underline{f}^{B}dV \tag{2.32}$$

with $V_I = V \bigcap B\left(\underline{x}_I, r_I\right)$.

The traction force vector corresponding to node I and degree of freedom m is

$$\hat{\underline{f}}_{Im} = \begin{cases} \underline{0} & \text{, for an interior sphere} \\ \int_{S_{f_I}} \underline{H}_{Im}\underline{f}^{S}dS & \text{, for a Neumann boundary sphere} \\ \sum_{J=1}^{N}\sum_{n\in\mathcal{J}} \underline{KU}_{ImJn}\underline{\alpha}_{Jn} - \underline{fU}_{Im} & \text{, for a Dirichlet boundary sphere} \end{cases} \tag{2.33}$$

where

$$\underline{KU}_{ImJn} = \int_{S_{u_I}} \underline{H}_{Im}\underline{N}\underline{C}\underline{B}_{Jn}dS + \int_{S_{u_I}} \underline{B}_{Im}^{T}\underline{C}\underline{N}^{T}\underline{H}_{Jn}dS \tag{2.34}$$

and

$$\underline{fU}_{Im} = \int_{S_{u_I}} \underline{B}_{Im}^{T}\underline{C}\underline{N}^{T}\underline{u}^{S}dS \tag{2.35}$$

with $S_f = \bigcup_{I \in N_f} S_{f_I}$ where N_f is the index set of nodes with nonzero intercept on the Neumann boundary and $S_u = \bigcup_{I \in N_u} S_{u_I}$ where N_u is the index set of nodes with nonzero intercept on the Dirichlet boundary.

2.4 Boundary conditions

In the displacement-based finite element method, the two types of boundary conditions that are encountered are Dirichlet (also called essential or displacement) boundary conditions and Neumann (also called natural or force) boundary conditions. In the method of finite spheres, and in meshless methods in general, the imposition of boundary conditions is more complicated than in the finite element method, particularly for the Dirichlet boundary conditions. In this section, we will discuss how the boundary conditions in the method of finite spheres can be imposed efficiently.

2.4.1 Neumann boundary conditions

In the finite element method, the Neumann boundary conditions are imposed by evaluating the externally applied nodal point force vector which is nonzero only for the nodes on the boundary that are subjected to the applied force boundary conditions. However, in the method of finite spheres, the traction force vector is nonzero for the Neumann boundary spheres, which is not just the spheres on the Neumann boundary, but any sphere with nonzero intersection with the Neumann boundary, i.e., $S_{f_I} = B\left(\underline{x}_I, r_I\right) \bigcap S_f$. Any sphere with nonzero intercept with the boundary contributes to the boundary integral for the method of finite spheres because the interpolation functions do not satisfy the Kronecker delta property.

For a Neumann boundary sphere,

$$\underline{\hat{f}}_{Im} = \int_{S_{f_I}} \underline{H}_{Im} \underline{f}^S dS \tag{2.36}$$

where S_{f_I} is the intersection of $B(\underline{x}_I, r_I)$ and Neumann boundary S_f. Also, $S_f = \bigcup_{I \in N_f} S_{f_I}$, where N_f is the index set of nodes with nonzero intercept on the Neumann boundary.

2.4.2 Dirichlet boundary conditions

In the finite element method, only the nodes on the Dirichlet boundary are considered when imposing the displacement boundary conditions. Since the interpolation functions satisfy the Kronecker delta property at the nodes, homogeneous (zero) Dirichlet boundary conditions are exactly satisfied while nonhomogeneous boundary conditions converge to the prescribed displacement boundary conditions in a weak sense. With the lack of the Kronecker delta property, for the method of finite spheres, imposing the Dirichlet boundary conditions is more difficult than imposing the Neumann boundary conditions. Some meshless methods impose the Dirichlet boundary conditions using techniques involving Lagrange multipliers, penalty factors, or the use of finite elements, which each introduce additional difficulties [22-24]. However, there is a technique for efficiently imposing the Dirichlet boundary conditions for the method of finite spheres, based on a specific arrangement of the spheres on the Dirichlet boundary, which allows the interpolation functions along the Dirichlet boundary to possess characteristics of the Kronecker delta property.

In general, any sphere with nonzero intercept with the Dirichlet boundary, referred to as a Dirichlet boundary sphere, i.e., $S_{u_I} = B(\underline{x}_I, r_I) \bigcap S_u$, contributes to enforcing the displacement boundary condition through Eqs. (2.33-2.35). We note that $S_u = \bigcup_{I \in N_u} S_{u_I}$,

where N_u is the index set of nodes with nonzero intercept on the Dirichlet boundary. Therefore, by substituting Eq. (2.33) for a Dirichlet boundary sphere into Eq. (2.30), we obtain

$$\sum_{J=1}^{N}\sum_{n\in\mathcal{J}}\left(\underline{K}_{ImJn}-\underline{KU}_{ImJn}\right)\underline{\alpha}_{Jn}=\underline{f}_{Im}-\underline{fU}_{Im} \tag{2.37}$$

where the term $\underline{KU}_{ImJn}$ is a symmetric stiffness matrix and $\underline{fU}_{Im}$ is a known forcing vector which is nonzero only when there is a prescribed nonzero displacement.

To efficiently impose the Dirichlet boundary conditions in the method of finite spheres, we employ a special arrangement of spheres, where the nodes are placed uniformly on the boundary such that the distance between two successive nodes is the radius of the spheres and that no other spheres intercept the Dirichlet boundary other than those that lie on the boundary. Then the interpolation functions at any such boundary node I are such that

$$h_{Im}\left(\underline{x}_I\right)=\begin{cases}\varphi_I^0\left(\underline{x}_I\right) & ,\ m=1\\ 0 & ,\ m\neq 1\end{cases} \tag{2.38}$$

and by definition of the partition of unity function,

$$\varphi_I^0\left(\underline{x}_I\right)=\frac{W_I}{\sum_{J=1}^{N}W_J}=1 \tag{2.39}$$

Therefore, the interpolation function h_{I1} satisfies the Kronecker delta property

$$h_{I1}\left(\underline{x}_J\right)=\begin{cases}1 & ,\ I=J\\ 0 & ,\ I\neq J\end{cases} \tag{2.40}$$

while the higher-order interpolation functions exhibit the property

$$h_{Im}\left(\underline{x}_J\right)=\begin{cases}0 & ,\ I=J\\ 0 & ,\ I\neq J\end{cases}\ \text{for } m\neq 1 \tag{2.41}$$

Hence, for any such boundary node I, Eq. (2.12) becomes

$$v_h\left(\underline{x}_I\right) = \underline{\alpha}_{I1} \tag{2.42}$$

With this uniform arrangement of spheres along the Dirichlet boundary, the displacement boundary conditions can be efficiently imposed, since for homogeneous (zero) Dirichlet boundary conditions, we simply remove the rows and columns of the stiffness matrix corresponding to the degrees of freedom for which the associated interpolation functions satisfy the Kronecker delta property, i.e., $\underline{\alpha}_{Im}$ where node I corresponds to a Dirichlet boundary sphere and $m = 1$ corresponds to the interpolation function defined as the Shepard partition of unity function. For nonhomogeneous Dirichlet boundary conditions, we also remove the same rows and columns but need to account for the nonzero prescribed displacements by introducing an additional known forcing term. Then with this special arrangement of spheres, for a Dirichlet boundary sphere, Eq. (2.37) becomes

$$\sum_{J=1}^{N} \sum_{\substack{n \in \mathcal{J} \\ \text{if } J \in N_u \\ n \neq 0}} \left(\underline{K}_{ImJn} - \underline{KU}_{ImJn}\right)\underline{\alpha}_{Jn} = \underline{f}_{Im} - \underline{fU}_{Im} - \overline{\underline{fU}}_{Im} \tag{2.43}$$

where

$$\underline{fU}_{Im} = \sum_{J \in N_u} \left(K_{ImJ1} - \underline{KU}_{ImJ1}\right)\underline{u}^S\left(\underline{x}_J\right) \tag{2.44}$$

and $\overline{\underline{fU}}_{Im} = \underline{0}$ when homogeneous (zero) Dirichlet boundary conditions are prescribed.

When imposing the Dirichlet boundary conditions, the method of finite spheres differs from the finite element method due to the lack of the Kronecker delta property. Furthermore, the method of finite spheres employs the interpolation functions h_{Im} at each node, whereas in the traditional finite element method, multiple interpolation functions are not defined at each node. From Eqs. (2.40-2.41) based on the special arrangement of the

boundary spheres, the interpolation function h_{I1} satisfies the Kronecker delta property, whereas the higher-order interpolation functions h_{Im} where $m \neq 1$ do not satisfy this property, so the special arrangement leads to a form that only closely resembles the Kronecker delta property. The implications for the method of finite spheres is that for either homogeneous or nonhomogeneous Dirichlet boundary conditions, the solution is exact only at the boundary nodes and converges in a weak sense between nodes along the boundary, which contrasts from the finite element method where homogeneous Dirichlet boundary conditions are exactly satisfied over the entire boundary.

Chapter 3

Numerical integration procedure

Numerical integration is a focal point of development for the method of finite spheres. The method of finite spheres requires integration of nonpolynomial functions over complicated integration domains, namely spheres, truncated spheres, and general lens-shaped regions for the overlap of spheres. Specialized integration schemes have been developed for the method in one- and two-dimensions [25]. Since the integrands in the method of finite spheres are nonpolynomial, exact integration is not possible because integration rules can at best guarantee polynomial accuracy. Therefore, the aim for the method of finite spheres is to obtain a numerical integration procedure that achieves a reasonable level of accuracy using a minimal number of integration points. In contrast, for the finite element method, the functions to be integrated are polynomials and the elements are non-overlapping. Consequently, numerical integration is efficient since Gauss-Legendre product rules can achieve high accuracy with relatively few integration points.

In this chapter we will provide an overview of some numerical integration procedures commonly used for numerical methods. However, having established that numerical integration is much more complicated for the method of finite spheres, especially in three-dimensions, we will discuss the specific integration domains encountered, namely interior spheres, boundary spheres, and the overlap region of spheres, and present a simple and efficient procedure known as the piecewise Gauss-Legendre quadrature rule.

3.1 Overview of numerical integration

Numerical integration has been studied and analyzed extensively for the finite element method, resulting in accurate and efficient integration rules for one-dimensional integrals, which can be easily extended to multi-dimensional integrals by applying the integration rules successively in each direction. Well known one-dimensional integration rules include Newton-Cotes quadrature and Gauss-Legendre quadrature.

In Newton-Cotes quadrature, when using $(n+1)$ equally spaced integration points, a polynomial of order up to n can be integrated exactly. To achieve greater accuracy in the integration, either a higher-order formula or a lower-order formula in a repeated manner, referred to as a composite formula, can be used. In practice, the latter is preferred since convergence is ensured as the sampling interval decreases. Furthermore, sampling intervals can vary depending on the function to be integrated, which is particularly advantageous when there are discontinuities in the function. It has also been shown that a higher-order Newton-Cotes formula may suffer from instability, indicating greater accuracy is not necessarily achieved [1].

In Gauss-Legendre quadrature, the weights and positions of n unequally spaced integration points are used to integrate a polynomial of order up to $(2n-1)$ exactly. In the

calculation of the finite element matrices, the integration points can be anywhere in the element with no additional difficulties from having unequally spaced points. Therefore, Gauss-Legendre quadrature is typically favored over Newton-Cotes formulas since accuracy of the integration can be improved by optimizing the positions and weights of the integration points, as proven by the superior polynomial order of accuracy of Gaussian quadrature [1].

Numerical integration for meshless methods is not as well established as for the finite element method. Since there are numerous meshless methods, different integration rules are being developed for the various classifications of meshless formulations [26-30]. We will focus on discussing integration schemes proposed for meshless methods based on the Galerkin weak form since this is the basis of the method of finite spheres and the finite element method. Numerical integration rules typically only differ with regard to the locations and weights of the integration points. Common integration techniques include Monte-Carlo integration, which is seen as an interpretation of integration in collocation meshless methods, the trapezoidal rule, which is one of the Newton-Cotes formulas, and Gaussian quadrature, which is the most frequently used integration scheme for meshless methods. Another way to characterize numerical integration schemes for meshless methods is whether it is based on direct nodal integration, integration with a background mesh or cell structure, or integration over supports and intersection of supports.

Direct nodal integration refers to evaluating integrals only at nodal points rather than introducing integration points. While this approach clearly reduces computational cost compared to full integration, it leads to relatively large integration errors as well as stability issues, similar to those discussed for collocation methods like SPH. While there are

stabilized nodal integration techniques for Galerkin meshless methods, the accuracy and convergence rate are significantly lower compared to full integration schemes [31].

For the earlier proposed methods such as DEM and EFG, the domain is divided into integration domains using a background mesh or cell structure and Gaussian quadrature is performed in each of these subdomains. These meshless methods are often considered pseudo-meshless since only the construction of the approximation functions is truly meshfree and the integration scheme still requires a mesh. With a background mesh, nodes and integration domain vertices generally coincide as in a traditional finite element mesh, although the background mesh does not have to be conforming. With a cell structure, nodes and integration cell vertices generally do not coincide at all. A significant source of integration error arises from the misalignment between the supports of the interpolation functions and the integration domains, which leads to decreased accuracy and convergence which cannot be improved simply by using higher-order integration rules. A bounding box technique to construct integration cells that align with the supports of the interpolation functions significantly reduces the integration error, but requires more integration cells and can only be used with parallelepiped supports and not spherical supports. There have also been adaptive numerical integration schemes which adaptively refine the background mesh or cell structure [7-9,26].

Integration over supports and intersection of supports results in a scheme that is truly meshless. This is a natural choice for methods that are based on the local weak form like MLPG and the method of finite spheres. The integration domains are simply the support of each node or the intersection of the support with either the boundary or other supports. The results for these integration schemes are more accurate than when using a

background mesh or cell structure simply for the fact that there is no misalignment between integration domains and interpolation supports [10,11]. Specialized Gaussian quadrature rules and isoparametric mappings can be used to perform integration for spheres intersecting with the boundary of the domain. Even specialized integration rules have been developed for the method of finite spheres for one- and two-dimensional problems, such as the piecewise midpoint quadrature rule [25]. Ultimately, we aim to use a numerical integration scheme that is simple and truly meshless like those discussed here, but that will be quite efficient even for three-dimensional problems involving complex integration domains, namely spheres, truncated spheres, and general lens-shaped regions for the overlap of spheres.

3.2 Sphere integration domains

In the traditional finite element method, isoparametric mapping is used to transform between the global coordinate system and the natural coordinate system. Numerical integration is performed in the natural coordinate system using a Gauss-Legendre quadrature rule. Therefore, if a domain was discretized using linear brick elements, in their natural coordinate system, all of these elements would use the same quadrature rule with identical sampling points and weights. The Jacobian operator, relating the natural coordinate derivatives to the global coordinate derivatives, is used to revert back to the global coordinate system. The finite element matrices, defined in the global coordinate system, are assembled by repeating this procedure over each element [1].

In the method of finite spheres, the global system of equations is not only comprised of the contribution of each sphere, but also from the overlap region of spheres. Furthermore, spheres are allowed to intersect the boundary, introducing additional

integration domains to be considered. Consequently, it is important to classify the types of spheres encountered in the method of finite spheres. Since the interpolation functions defined within these different types of integration domains are rational, nonpolynomial functions, transforming between the global coordinate system and a natural coordinate system is much more complicated and computationally expensive than in the finite element method. To avoid additional computations of working in a natural coordinate system and calculating the Jacobian operator, the calculation of the system of equations for the method of finite spheres is performed only in the global coordinate system.

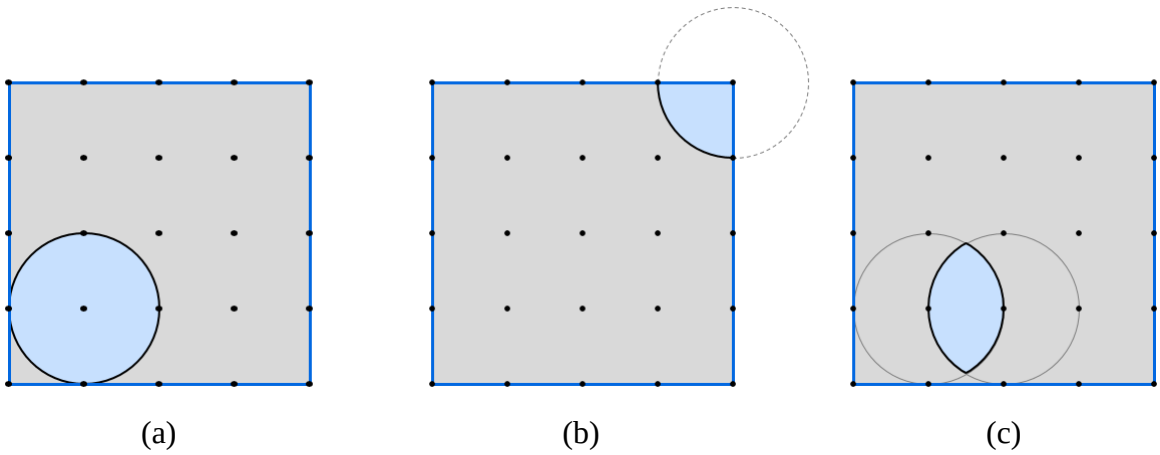


Figure 3.1 Sphere domains for the method of finite spheres: (a) interior sphere, (b) boundary sphere, and (c) sphere overlap region

In the method of finite spheres, the types of sphere domains encountered are classified as interior spheres, boundary spheres (both Neumann boundary spheres and Dirichlet boundary spheres), and overlap regions of spheres. In Figure 3.1, we show these classifications, but for simplicity the images are of two-dimensional “spheres” which are circles, though it should be inferred that three-dimensional spheres are implied. As will be discussed further in Chapter 4, we note that the global structure stiffness matrix is a block matrix with each block along the diagonal representing the stiffness for either an interior

sphere or boundary sphere. Additional block matrices that are not on the diagonal are the contribution to the global structure stiffness matrix from the overlap of spheres. Therefore, each position in the global stiffness matrix is uniquely reserved for either an interior sphere, boundary sphere, or sphere overlap.

3.3 Piecewise Gauss-Legendre quadrature rule

A computationally efficient numerical integration procedure is the key to the success of the method of finite spheres. Specialized integration rules such as the piecewise midpoint quadrature rule have been previously proposed for the method of finite spheres in one- and two-dimensional analysis. A challenge for this numerical integration scheme is reliability since the accuracy is highly dependent on the appropriate selection of the number of integration points used as well as the balance of integration points among different integration domains. For example, the integration rule for the sphere overlap region does not coincide with the integration rule for the interior sphere or boundary sphere, and so the overall accuracy when solving the system of equations is compromised when the accuracy for these integration domains are not of the same order.

The piecewise Gauss-Legendre quadrature rule was developed to be a simple and efficient quadrature rule that took the integration domain and divided it along the coordinate axes, creating four quadrants in two-dimensional analysis and eight subdomains in three-dimensional analysis, and applied the same Gauss-Legendre quadrature rule in each subdomain, accounting for only the integration points that lie in the intersection of the sphere and domain. Since the local approximation space contains polynomial terms defined from the center of the sphere, subdividing the integration domain of a sphere into eight parts results in less complicated interpolation functions in each subdomain. The

overall accuracy is improved when using more integration domains with lower integration order versus using fewer integration domains with higher integration order.

The objective of this numerical integration scheme is to achieve a reasonably accurate solution using a minimal number of integration points. An advantage over the piecewise midpoint quadrature rule is that the contribution of the overlap regions to the system of equations is calculated using the same set of integration points as for the interior sphere or boundary sphere. Therefore, the unique properties for this numerical integration scheme are a uniform density of integration points throughout the support of the interpolation functions, and the sphere overlap regions are integrated directly without need for a different integration rule for these integration domains. Furthermore, the scheme does not make use of isoparametric mapping and all calculations are performed in the global coordinate system since transformations using the Jacobian operator would be more cumbersome due to the presence of complicated, nonpolynomial interpolation functions. We will briefly summarize the piecewise Gauss-Legendre quadrature rule for the different sphere domains, and as before, the two-dimensional "sphere" illustrations that will be depicted are meant to represent three-dimensional spheres.

3.3.1 Interior sphere

For an interior sphere, the piecewise Gauss-Legendre rule subdivides the sphere into eight equal parts, in which the usual Gauss-Legendre quadrature rule is used in each subdomain. Only the integration points in the intersection of the sphere and the domain are considered, and since this is an interior sphere, all points within the sphere contribute to the integration.

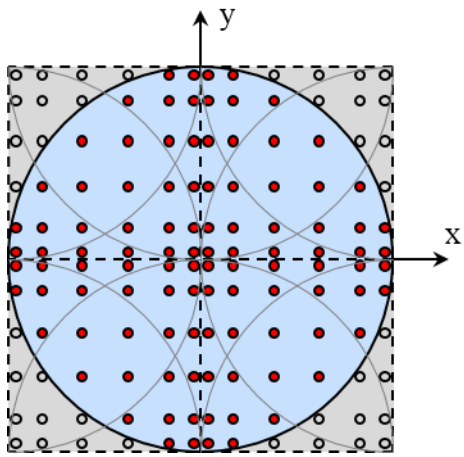


Figure 3.2 Piecewise Gauss-Legendre quadrature rule for an interior sphere

An interior sphere divided by the coordinate axes is illustrated in Figure 3.2. Within each subdomain, the same Gauss-Legendre quadrature rule is employed. Since this is an interior sphere, all points within the sphere contribute to the integration and are shown as red points, while all points outside the sphere do not contribute to the integration. By subdividing the domain, the complexity of the integrand is reduced and accuracy is improved because of the additional integration domains. Furthermore, it is clear that there is a uniform density of integration points throughout the integration domain.

3.3.2 Boundary sphere

For a boundary sphere, the integration domain is again subdivided into eight equal parts according to the coordinate axes. However, since this is a boundary sphere, by definition the sphere intersects the domain boundary so portions of the sphere that lie outside the domain should not contribute to the integration.

In Figure 3.3, a boundary sphere is depicted where the domain is shaded in grey and the portion of the sphere which lies in the domain is shaded in blue. A Gauss-Legendre

quadrature rule is considered in each of the eight equal subdomains, but only the integration points that lie in the intersection of the sphere and the domain are used in the integration calculations, and are shown as red points.

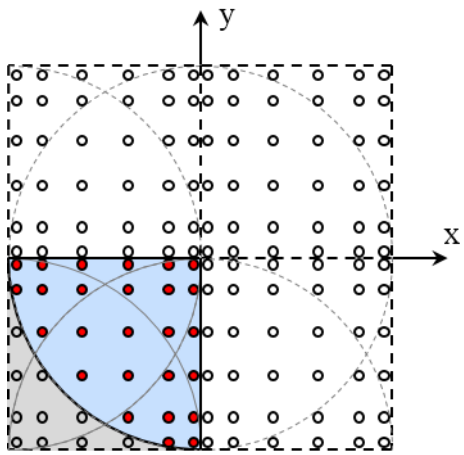


Figure 3.3 Piecewise Gauss-Legendre quadrature rule for a boundary sphere

3.3.3 Sphere overlap region

For the sphere overlap region, no special numerical integration scheme is used, but rather the same integration points as for the interior sphere or boundary sphere are used. This is in contrast to the previously proposed piecewise midpoint quadrature rule, where the overlap regions cannot be integrated directly and required determining a new set of integration positions and weights for these regions of overlap.

To demonstrate employing the piecewise Gauss-Legendre quadrature rule for a sphere overlap region, we consider the sphere overlap illustrated in Figure 3.1c. The coordinate system shown in Figure 3.4 is centered on the left sphere, which is an interior sphere. The domain is subdivided into eight parts based on the coordinate axes, and a Gauss-Legendre quadrature rule is considered in each subdomain, as is usually done for an interior sphere. However, now that we are considering the sphere overlap with the sphere

to the right, we simply consider the same set of integration points, and integrate the overlap region using a subset of these points, which are shown as red points in Figure 3.4. Of course, there are multiple sphere overlaps for the sphere which is being considered, and each of these sphere overlaps would consider a subset of the interior sphere integration points and yield a unique nonzero sphere stiffness matrix. For a uniform arrangement of spheres in all directions, an interior sphere would have 26 overlap regions.

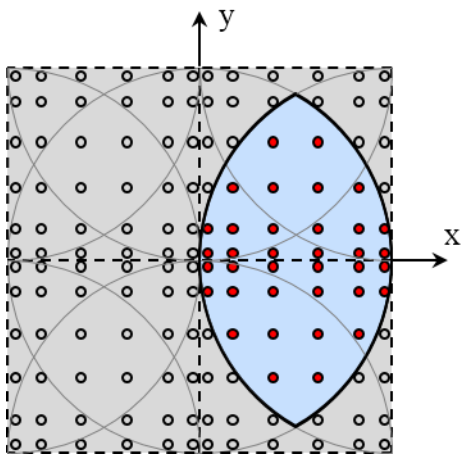


Figure 3.4 Piecewise Gauss-Legendre quadrature rule for a sphere overlap region

Ultimately, the piecewise Gauss-Legendre quadrature rule is a reasonably simple and efficient truly meshless numerical integration scheme developed for the method of finite spheres. The calculations are performed in the global coordinate system in contrast to other truly meshless numerical integration schemes like those used in MLPG, where isoparametric mapping is used to perform the integrations. Furthermore, this scheme differs from the usual Gauss product rules used in the traditional finite element method because the integration domains are first subdivided by the coordinate axes to simplify the interpolation functions to be integrated.

Chapter 4

Implementation of the method of finite spheres

The theory and formulation and the numerical integration procedure have been presented for the analysis of three-dimensional problems using the method of finite spheres. In this chapter, we focus on the computational aspects for the implementation of the method of finite spheres within a commercial finite element program. The motivation is to reduce the manual pre-processing costs associated with the discretization of complicated domains in the finite element method. This comes at the expense of longer computational times, which is justified given the prospect of improved computational performance.

The focus will be in comparing and improving the relative performance of the method of finite spheres in the solution phase. A study of the manual pre-processing time will not be included, except to note that there is a significant time savings for the method of finite spheres. The first step towards comparing solution times will be to implement the

method of finite spheres in a commercial finite element program, in order to make a comparison on the same platform. Since our aim is to study practical three-dimensional problems, we have chosen to use the commercial program ADINA because it is robust and has an efficient equation solver. In order to implement the method of finite spheres in ADINA, we will modify the main code and include a user-element subroutine written in Fortran. Since the method of finite spheres is also based on the Galerkin weak form, only certain modules of the code need to be modified. The method was intentionally based on a similar theory and formulation, where the outcome is not to replace the finite element method, but rather to supplement and address certain difficulties, with the eventual aim of a coupled finite element and finite sphere formulation. To further illustrate the implementation process, we will compare the procedures of the finite element method and the method of finite spheres in terms of discretization, interpolation, integration, and equilibrium equations.

Discretization is the first step in implementation, and as a major cost in the finite element method, we discuss how the method of finite spheres is more automated, requiring less manual effort. For the finite element method, discretization requires defining a set of nodes and a connectivity array associating local element degrees of freedom to global structure degrees of freedom. In order to achieve an accurate solution, the mesh should not have distorted elements, in addition to the prerequisite of a conforming mesh with no overlapping elements. Representation of the domain boundary requires that the nodes are on the boundary. For the method of finite spheres, discretization is more efficient, since as a truly meshless method, only requires defining a set of spheres, distinctly defined by the coordinates of the sphere center and radius. In contrast to the finite element method, there

is no need for a connectivity array because sphere overlaps can automatically be determined from the discretization. Sphere discretization requires that all nodal points must be within the domain, the domain must be completely covered by the spheres, and that no sphere is completely included in any other sphere. Representation of the domain boundary does not require that nodes are on the boundary. Ultimately, discretization in the method of finite spheres is more efficient due to the ability to easily mesh complicated domains without the complication of distorted elements.

Interpolation is the next step in the implementation process, and while the finite element method employs efficient interpolation functions, the method of finite spheres uses more costly interpolation functions due to the nature of the discretization. For the finite element method, interpolation is based on polynomial functions typically defined in the natural coordinate system. For the method of finite spheres, the interpolation functions are based on the Shepard partition of unity functions, which only satisfy zeroth-order consistency. To achieve higher-order consistency, a local approximation space is defined, and the interpolation functions are then defined as the product of the Shepard partition of unity function and the functions from the local basis. These functions are defined in the global coordinate system to avoid calculating the Jacobian operator. In terms of ease of implementation, we simply define a user-element subroutine in ADINA to substitute the finite element polynomial interpolation functions for the method of finite spheres nonpolynomial interpolation functions.

Integration is a major computational component of the solution phase. For the finite element method, the standard Gauss-Legendre quadrature rules can exactly integrate the polynomial interpolation functions very efficiently. For the method of finite spheres, due

to the nonpolynomial nature of the interpolation functions, we utilize the piecewise Gauss-Legendre quadrature rule, where many more integration points are required. Consequently, the solution time is likely to be increased in the method of finite spheres, by a factor which is to be determined in the study of three-dimensional numerical examples in Chapter 5. In terms of implementation, the user-element subroutine is used to define the positions and weights of the integration points for this integration scheme.

The equilibrium equations are the matrix system of equations to be solved, determined from the numerical integration procedure. For the finite element method, the nodal degrees of freedom in a three-dimensional analysis correspond to the x-, y-, and z-displacements. For the method of finite spheres, the nodal degrees of freedom do not directly correspond to the displacements, as there are additional degrees of freedom per node. Although there are more unknowns per node, the method of finite spheres can achieve comparable accuracy to the finite element solution with a fewer number of nodes, meaning the total number of unknowns is still comparable. In terms of implementation, we modify the main code to adjust the size of the element matrices and to define the procedure to assemble the global structure matrices. Once the system of equations has been assembled, the equations are solved in the same manner, but there is additional post-processing to obtain the nodal displacements in the method of finite spheres.

The objective of this chapter will be to expand on the implementation of the method of finite spheres in the commercial finite element program ADINA. In particular, we will introduce the three-dimensional linear sphere element, which will be used in the three-dimensional numerical examples in Chapter 5. We discuss the nodal degrees of freedom, as well as the local element and global structure degrees of freedom for the respective three-

dimensional elements, namely the linear brick element and the linear sphere element. We explain the advantages of using ADINA, detailing where modifications must be made in order to implement the method of finite spheres. The user-element subroutine is further detailed within the overall context of a linear static analysis flowchart.

4.1 Three-dimensional linear sphere element

In this section, we discuss the implementation of the three-dimensional linear sphere element for the solution of three-dimensional linear elasticity problems. In particular, we describe the choice of the local approximation space for the linear sphere element and compare with the linear brick element used in the finite element method.

For the finite element method, the 8-node linear brick element is a general purpose finite element which has three translational degrees of freedom per node. The interpolation functions contain polynomial terms given by a first-degree complete polynomial. Therefore, the element can be fully integrated using $2 \times 2 \times 2$ Gauss quadrature and has an order of convergence of one. For the method of finite spheres, the linear sphere element has $3m$ degrees of freedom per node, where m is the number of terms in the local approximation space. A suitable local approximation space for three-dimensional linear elasticity problems has been given in Eq. (2.14), where there are $m = 7$ terms in the local basis, so the linear sphere element has 21 degrees of freedom per node. As previously discussed, the interpolation functions are the product of the Shepard partition of unity function and the terms in the local approximation space, and since these functions are nonpolynomial, a large number of integration points using the piecewise Gauss-Legendre quadrature rule is required to obtain reasonable accuracy.

4.1.1 Nodal degrees of freedom

In general, there are six degrees of freedom per node in the finite element method, corresponding to three translational and three rotational degrees of freedom. For the displacement-based formulation, the rotational degrees of freedom are excluded from the analysis, so only three degrees of freedom per node are considered, representing the x-, y-, and z-displacements. Of course, there may be less than three degrees of freedom per node if any displacements are prescribed. The possible finite element nodal degrees of freedom are shown in Figure 4.1a.

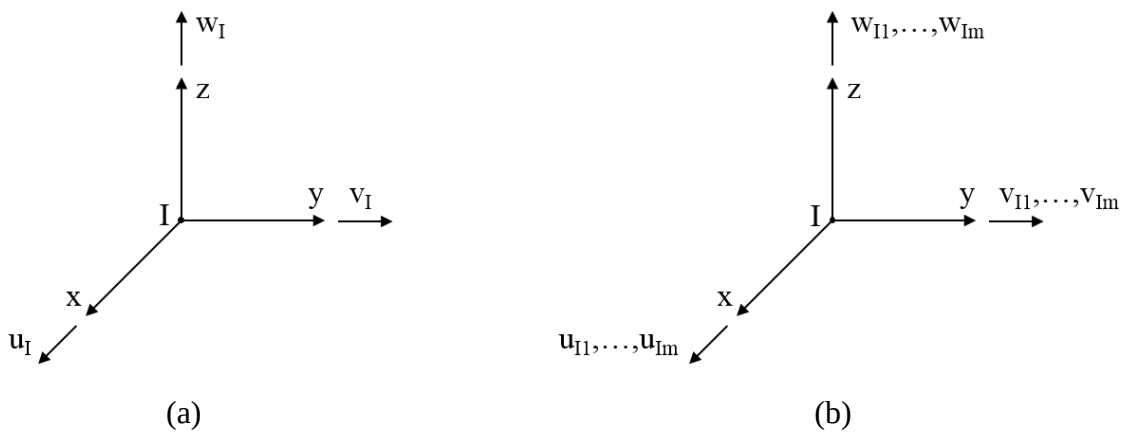


Figure 4.1 (a) Finite element nodal degrees of freedom and (b) finite sphere nodal degrees of freedom

For the method of finite spheres, there are $3m$ degrees of freedom per node, as shown in Figure 4.1b, where m distinct degrees of freedom represent each of the x-, y-, and z-displacements, and m is the number of terms in the local basis. For the linear sphere element, the selected local approximation space in Eq. (2.14), which is suitable for linear elasticity problems, includes one constant term, three linear terms, and three quadratic terms from the Pascal pyramid, illustrated in Figure 4.2. Therefore, given that there are

$m = 7$ terms in the local approximation space, the linear sphere element has up to 21 degrees of freedom per node.

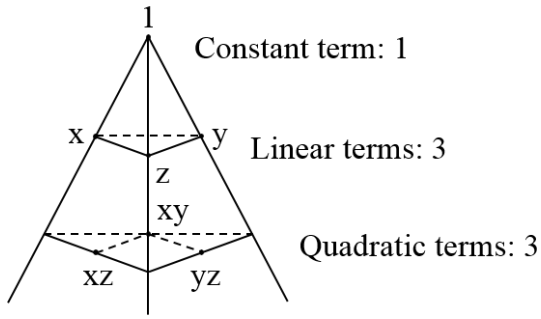


Figure 4.2 Pascal pyramid representing the terms in the local approximation space for the linear sphere element

To implement the linear sphere element in ADINA, we need to modify the allowed number of degrees of freedom per node to 21 degrees of freedom, from the currently allowed 6 degrees of freedom for the finite element method. This modification was made to the main code of ADINA so that all subsequent arrays for the linear sphere element defined in the user-element subroutine would account for the larger number of degrees of freedom per node.

4.1.2 Local element degrees of freedom

For the three-dimensional linear elasticity numerical examples in Chapter 5, we will compare solution times between the finite element method and the method of finite spheres. For the finite element solution, we use the 8-node linear brick element, and for the finite spheres solution, we use the linear sphere element, both of which are illustrated in Figure 4.3. With three degrees of freedom per node in the finite element method, the linear brick element has a total of 24 degrees of freedom. In the method of finite spheres, the

linear sphere element has a total of 21 degrees of freedom since the nodal degrees of freedom are the local element degrees of freedom.

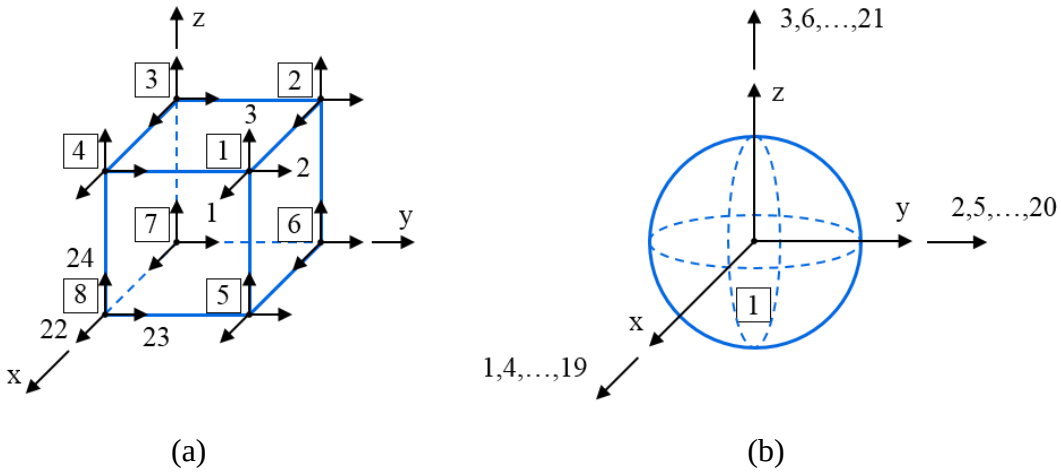


Figure 4.3 (a) Linear brick element and (b) linear sphere element

In terms of implementation, the finite element method requires defining a connectivity array to relate the local element degrees of freedom to the global structure degrees of freedom. Finite elements are composed of several nodes, which can be shared among other elements. Only the solution at the nodes that comprise the finite element are needed to describe the field variable everywhere within the element. To implement the method of finite spheres in ADINA, instead of defining a connectivity array, sphere overlaps need to be determined based on the sphere discretization. While the linear sphere element is composed of a single node, to describe the field variable everywhere within the sphere element, not only are the values of the field variables at that single node needed, but also the nodal values for all other spheres that have an overlap with the sphere of interest are needed.

4.1.3 Global structure degrees of freedom

Before solving the discrete equilibrium equations, the global structure matrices must be assembled, which for a static analysis are the global structure stiffness matrix and load vector. The size of these arrays, which is also the number of equilibrium equations to be solved, is equal to the number of global structure degrees of freedom, which is the total number of nodes in the discretization multiplied by the number of degrees of freedom per node, excluding non-active degrees of freedom, which are those that are prescribed through the Dirichlet boundary conditions.

In terms of implementation, for the finite element method, the global stiffness matrix is assembled by adding the contribution of each element stiffness matrix, which is known as the direct stiffness method. The element stiffness matrix is a square matrix with dimensions equal to the number of local element degrees of freedom, i.e., a 24×24 matrix for the linear brick element. When global structure degrees of freedom are shared among different elements, the stiffness contribution from each of the included elements are added together. For the method of finite spheres, the global structure stiffness matrix is assembled from not only the sphere element itself, either an interior sphere or a boundary sphere, but also from every sphere overlap region. Each of these sphere domains contributes to the global structure stiffness matrix by the addition of a square matrix with dimensions equal to the number of local element degrees of freedom, i.e., a 21×21 matrix for the linear sphere element. The global structure stiffness matrix is a banded matrix with a unique location for each of the square matrices corresponding to each of the sphere domains. A typical layout of the global structure stiffness matrix for the method of finite spheres is illustrated in Figure 4.4, where the square matrices along the main diagonal correspond to the sphere

element itself, either an interior sphere or a boundary sphere, e.g., $\underline{K}_{II}$, and the square matrices not along the main diagonal correspond to sphere overlap regions, e.g., $\underline{K}_{IJ}$, which are nonzero only when the spheres corresponding to nodes I and J overlap.

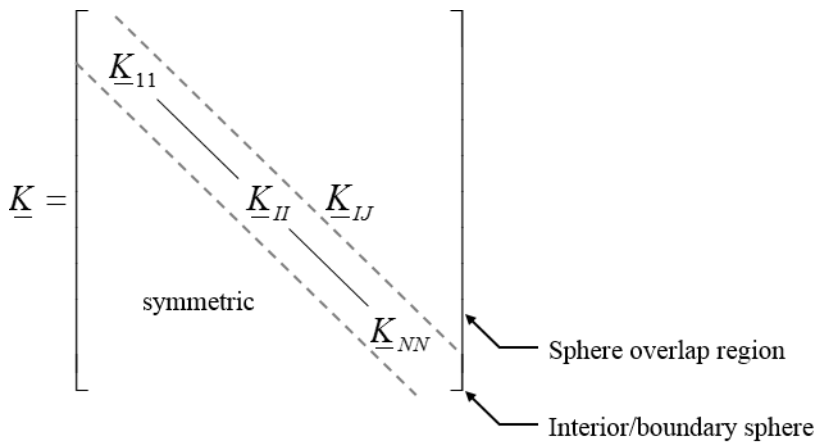


Figure 4.4 Typical layout of structure stiffness matrix for the method of finite spheres

4.2 Commercial finite element software

ADINA is a commercial finite element software for linear and nonlinear analysis of solids and structures, heat transfer, CFD, and electromagnetics. In this section, we discuss the implementation of the displacement-based method of finite spheres in ADINA for three-dimensional linear static analysis. It should be noted that the method is also applicable for analyses with almost or fully incompressible media using mixed formulations [14]. Furthermore, the method of finite spheres has previously been employed for the dynamic analysis of two-dimensional wave propagation problems [21].

The effectiveness of a program depends on the use of efficient programming methods and efficient numerical techniques. The advantage of using a commercial finite element program like ADINA is that we are able to solve practical three-dimensional problems making use of efficient programming methods and equation solvers.

Furthermore, when comparing solution times between the finite element method and the method of finite spheres using the same program, computational efficiency comparisons will truly reflect the effectiveness of the methods. The main computational costs of a program are the calculation of system matrices and the solution of equilibrium equations. For the method of finite spheres, the calculation of system matrices is the major computational cost because many integration points are needed to accurately evaluate these matrices. The purpose of this section is to describe the implementation of the method of finite spheres in ADINA and to discuss the important aspects of the user-element subroutine, which calculates the system matrices, namely the element stiffness matrix and element load vector.

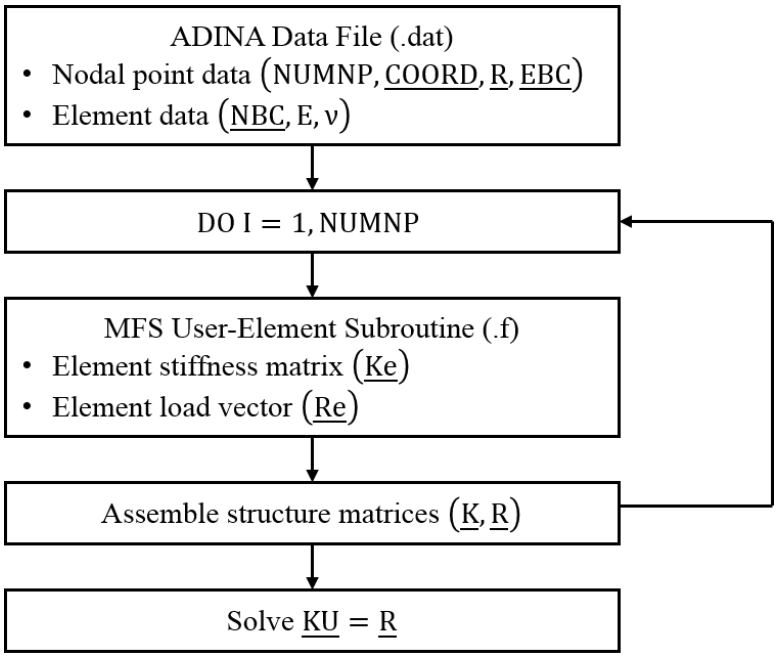


Figure 4.5 Flowchart of method of finite spheres linear static analysis in ADINA

A flowchart of the method of finite spheres for a linear static analysis in ADINA is shown in Figure 4.5. The first step is to read the nodal point data (number of nodal points, coordinates, radius, and essential boundary conditions) and element data (natural boundary

conditions and material properties) from an ADINA data file. This data file is very similar to the one which is used for a finite element analysis, except that we no longer require a connectivity array relating the assemblage degrees of freedom to the element local degrees of freedom, but we do now require an additional element data input, which is the sphere radius. From the data for the coordinates and radius of all spheres, we are able to directly determine the overlaps for all spheres, which is needed to calculate the element stiffness contributions of the sphere overlap domains. Once the ADINA data file has been read, the program then loops over every sphere to calculate the element matrices, implemented through a user-element subroutine which is further discussed in Section 4.2.1. The element matrices are then assembled into the global structure matrices, making use of an efficient storage scheme. The nodal unknowns can then be determined by solving the system of equations using the ADINA equation solver, and post-processing of these results obtains displacements, strains, and stresses throughout the problem domain.

4.2.1 ADINA user-element subroutine

A user-element subroutine, written in Fortran, was used to implement the method of finite spheres in ADINA. The main purpose of the subroutine was to calculate the element matrices for each sphere, which would then be output to the main code for assembly into the global structure matrices. For a given sphere, which can be either an interior or boundary sphere, we first determine the location and weights of the integration points according to the piecewise Gauss-Legendre quadrature rule. Then an element stiffness matrix can be computed for the sphere itself and for each sphere overlap region for that given sphere, by summing the evaluation of the stiffness integrand at every integration point. The element load vector calculation for each sphere does not depend on

the overlap of spheres, and is therefore much less computationally intensive than the element stiffness matrix calculation. Ultimately, the major computational cost for the implementation of the method of finite spheres in ADINA is in the user-element subroutine for the calculation of the element matrices, in particular, the element stiffness matrices, which is due to the fact that many integration points are needed to accurately integrate the nonpolynomial interpolation functions.

Chapter 5

Numerical results for three-dimensional linear elasticity problems

The objective of this chapter is to demonstrate the reliability and computational efficiency of the method of finite spheres, an effective overlapping finite element method, through the examination of several three-dimensional linear elasticity problems. Aside from being a dominant class of problems in practical engineering analysis, three-dimensional static linear elasticity is best for assessing the computational efficiency of the method of finite spheres because the solution is expensive with respect to numerical integration and bandwidth of the system of equations compared to the finite element method. Furthermore, when confronted with an analysis involving a complicated three-dimensional geometry, the advantages of a meshless method are apparent, but there has only been limited research on meshless methods for solid mechanics, especially for three-dimensional problems [32-36].

The formulation and implementation procedures discussed in Chapters 2-4 are used in this chapter to perform the modeling of three increasingly complex three-dimensional models, each considering two load cases, using the method of finite spheres. The commercial finite element software program ADINA is used to obtain the method of finite spheres solution and the finite element solution, so that accuracy and solution times can be compared on the same platform with a robust and efficient solver.

The selected numerical examples are relatively simple, but representative of three-dimensional linear elasticity problems [37]. It is important to recognize that the claims in the discussion of numerical results are only based on the selected problems but provide a baseline for determining the competitiveness of the method of finite spheres. For about the same level of accuracy, if the method of finite spheres solution time is within a reasonable time multiplier of the finite element solution time, the method of finite spheres becomes competitive. This is because it provides the benefit of circumventing the mesh generation process and avoiding errors due to element distortion. Furthermore, with continuous research efforts and increasing processing speeds, the method of finite spheres will become even more competitive as compared to the finite element method, since solution times will continue to decrease, and the main differentiation will be in the time and effort spent in the pre-processing phase.

5.1 Model 1: Short cantilever beam with square solid section

The model we would like to study first is the short cantilever beam with a square solid section. The geometry, boundary conditions, and material data are illustrated in Figure 5.1.

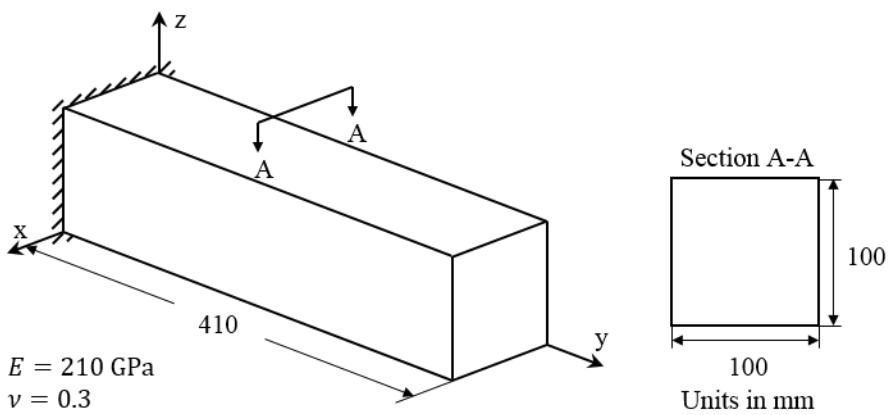
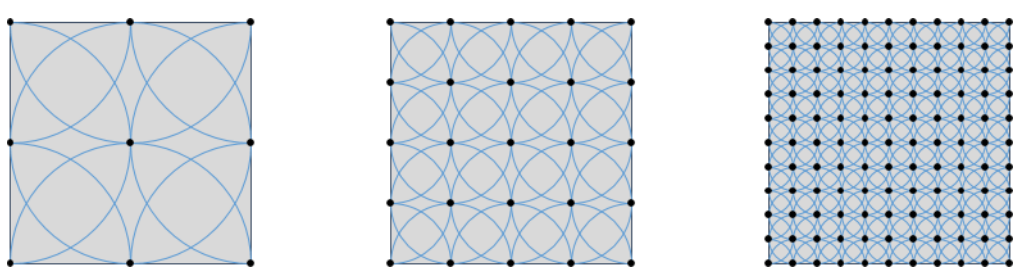


Figure 5.1 Geometry, boundary conditions, and material data for short cantilever beam with square solid section

The problem will be solved using three discretizations for both the method of finite spheres and the finite element method. For the method of finite spheres, the sphere arrangement is uniform with equal radius size for each discretization, as shown in Figure 5.2. We use the linear sphere element described in Section 4.1. For the finite element method, we use a sequence of compatible meshes consisting of 8-node brick elements. The mesh refinement involves subdividing each brick element into eight brick elements so that the coarser mesh will be embedded in the finer mesh, or mathematically, the new space of finite element interpolation functions will contain the previously used space, so we can expect monotonic convergence [1]. We use the finest finite element discretization as the reference solution to assess the accuracy of the remaining discretizations.



(a) MFS1: r = 50 mm (b) MFS2: r = 25 mm (c) MFS3: r = 10 mm

Figure 5.2 MFS discretizations at Section A-A for short cantilever beam with square solid section

The following sections illustrate the results of the short cantilever beam with square solid section when subjected to a bending load and torsional load, respectively, at the free end. To assess overall solution accuracy, the strain energy over the entire model will be considered. Displacements and stresses along model lines will be studied to further demonstrate the reliability of the method of finite spheres.

5.1.1 Load 1: Bending load at free end

In Figure 5.3, we illustrate the short cantilever beam with square solid section with a bending load at the free end, referred to as Model 1 Load 1 (M1L1).

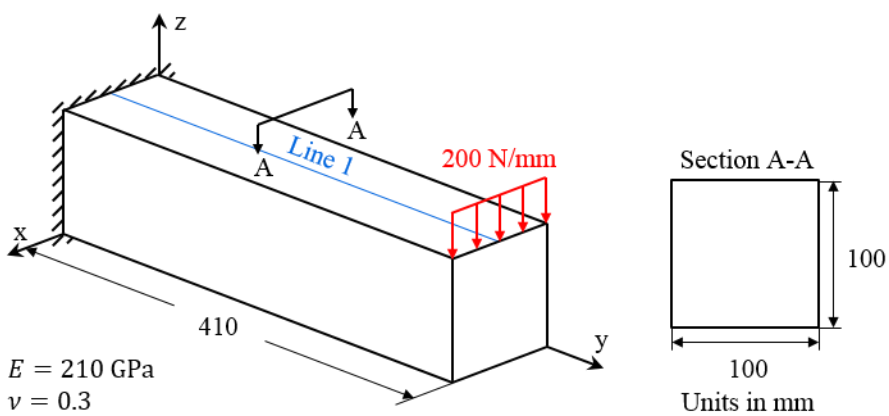


Figure 5.3 M1L1: Short cantilever beam with square solid section with bending load at free end

The results for MFS and FEM discretizations are given in Table 5.1, showing strain energy errors and time multipliers with respect to FEM3, the reference solution. All three MFS discretizations provide results with strain energy errors within 9%. A lower number of nodes can be used in the method of finite spheres to obtain comparable accuracy with the finite element method, due to the fact that continuous approximations based on the Shepard partition of unity functions are used. The time multiplier for MFS3 is 11.31, corresponding to a strain energy error of 0.48%, with respect to the reference solution.

Table 5.1 M1L1 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	90	8.19	0.08
MFS2	450	2.83	0.37
MFS3	5082	0.48	11.31
FEM1	756	3.55	0.01
FEM2	4961	1.06	0.04
FEM3	35721	*	*

*FEM3: Strain energy (N·mm) = 2753.9; Time (s) = 25.97

The convergence of strain energy for the method of finite spheres and the finite element method is shown in Figure 5.4. For equal radius and element size in the method of finite spheres and finite element method, respectively, the method of finite spheres provides better accuracy based on the strain energy error norm with respect to the reference solution. Another important observation is that the method of finite spheres exhibits a similar rate of convergence as the classical finite element method.

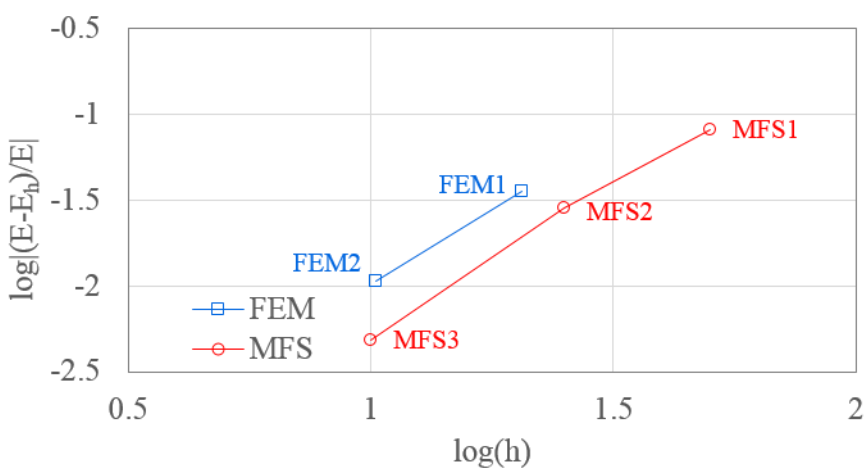


Figure 5.4 M1L1 convergence of strain energy for the method of finite spheres and the finite element method

The following results illustrate the qualitative behavior of the short cantilever beam with square solid section for the finest method of finite spheres discretization MFS3, which has a total of 5082 spheres. To demonstrate the reliability of the method of finite spheres, we consider the results for transverse displacement and longitudinal normal stress. The transverse displacement contour plot is shown in Figure 5.5, and qualitatively, seems reasonable given the geometry, boundary conditions, material data, and loading conditions. The uniform arrangement of spheres is illustrated by the red nodes which are spaced evenly apart by a distance equal to the sphere radius $r = 10$ mm.

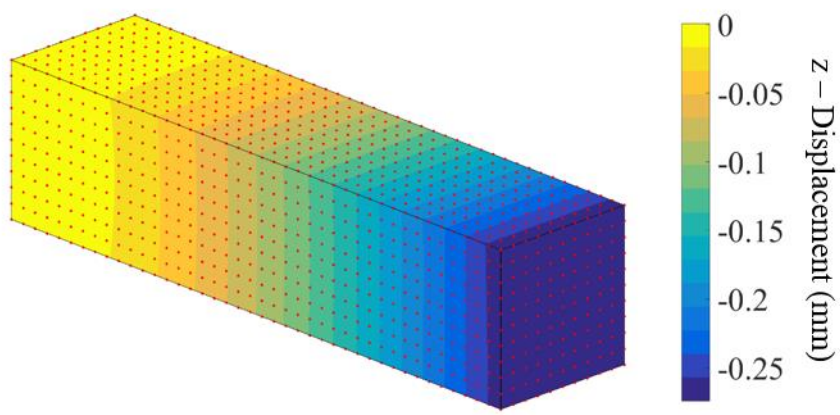


Figure 5.5 M1L1 transverse displacement contour plot for MFS3

From Figure 5.3, we select Line 1 to more closely examine the transverse displacement and longitudinal normal stress between MFS3 and FEM3, the reference solution. We observe in Figure 5.6 that the z-displacement results along Line 1 are in agreement. The longitudinal normal stresses along Line 1 are shown in Figure 5.7. This figure reveals that the method of finite spheres is also reliable in capturing stresses. Based on these qualitative results, the method of finite spheres seems to be reliable for solving the short cantilever beam with square solid section with a bending load at the free end.

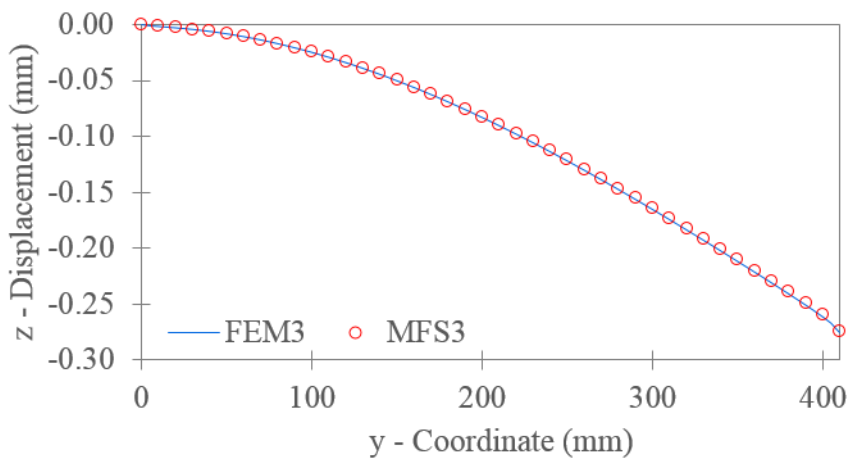


Figure 5.6 M1L1 transverse displacement along Line 1

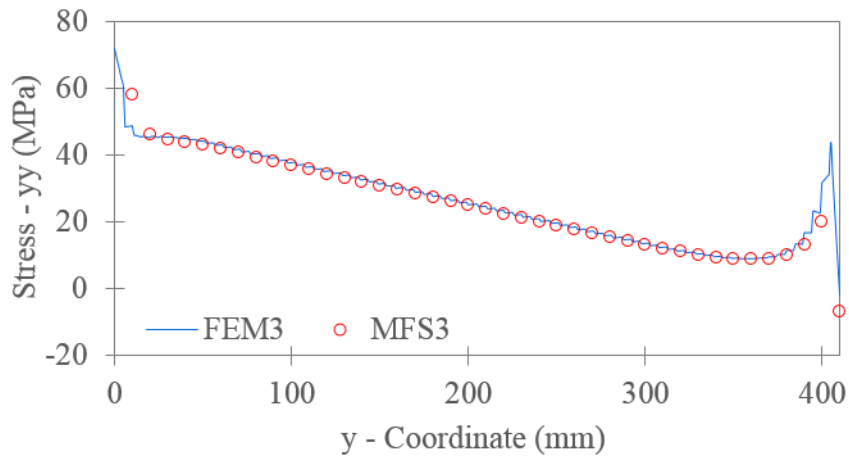


Figure 5.7 M1L1 longitudinal normal stress along Line 1

M1L1 has provided some initial insights into the reliability and efficiency of the method of finite spheres. The results for strain energy over the entire model indicate that the method of finite spheres is capable of obtaining comparable accuracy as the finite element method, with computational times that are roughly one order of magnitude longer than the solution times for the finite element method. Furthermore, the method of finite spheres is more accurate than the finite element method when comparing solutions of equal sphere size and element size. Also, a comparable convergence rate was observed for the two methods, which is expected since the linear brick element and linear sphere element

both have interpolation functions with degree of polynomial completeness of one. Lastly, we observed that displacement and stress results along a model line indicate that qualitatively, the method of finite spheres provides reliably accurate results compared to the finite element reference solution. We will look to see whether these trends are consistent with different loading conditions in the next section.

5.1.2 Load 2: Torsional load at free end

In Figure 5.8, we illustrate the short cantilever beam with square solid section with a torsional load at the free end, referred to as Model 1 Load 2 (M1L2).

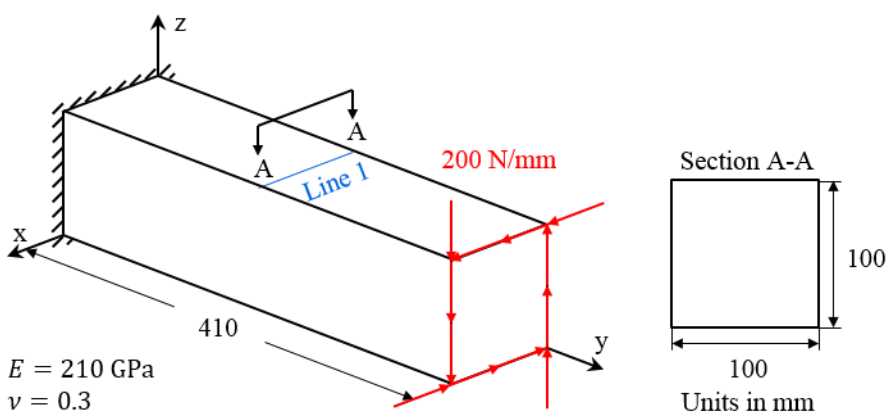


Figure 5.8 M1L2: Short cantilever beam with square solid section with torsional load at free end

The results for the MFS and FEM discretizations are summarized in Table 5.2, showing the strain energy errors and time multipliers with respect to FEM3, the reference solution. The discretizations are the same as those used in Section 5.1.1, and similar to the strain energy results in that section, all three MFS discretizations provide results with strain energy errors within 8%. The time multiplier for MFS3 is 12.61, corresponding to a strain energy error of 0.30%, with respect to the reference solution.

Table 5.2 M1L2 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	90	7.04	0.08
MFS2	450	3.77	0.38
MFS3	5082	0.30	12.61
FEM1	756	6.77	0.01
FEM2	4961	2.43	0.03
FEM3	35721	*	*

*FEM3: Strain energy (N·mm) = 3068.1; Time (s) = 24.91

The convergence of strain energy for the method of finite spheres and the finite element method is shown in Figure 5.9. For equal sphere radius versus element size, the method of finite spheres still provides better accuracy than the finite element method. Furthermore, the rate of convergence is comparable for the two methods, due to the fact that the element types, namely the linear brick element and the linear sphere element, have the same degree of polynomial completeness.

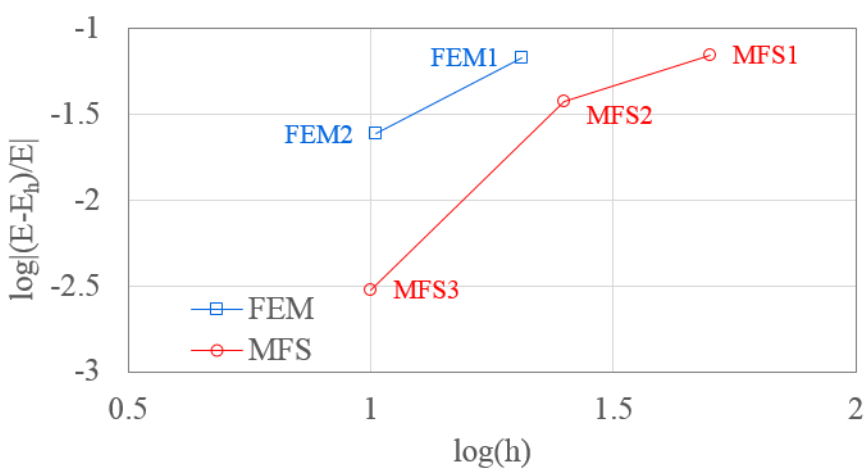


Figure 5.9 M1L2 convergence of strain energy for the method of finite spheres and the finite element method

The qualitative results for M1L2 are now examined for the finest method of finite spheres discretization MFS3. In Figure 5.10, the displacement magnitude contour plot is shown. Discretization of the problem domain is based on a uniform sphere arrangement with a total of 5082 spheres, and as before, is denoted by the red nodes. Qualitatively, the results are in agreement with the expected displacements for a torsional load at the free end.

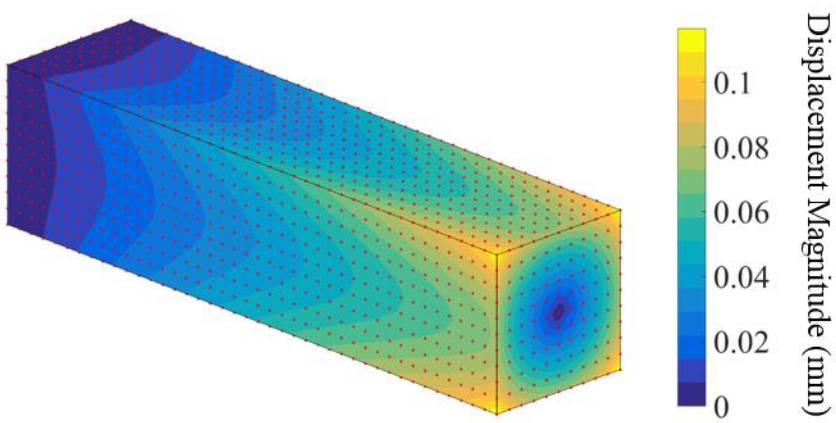


Figure 5.10 M1L2 displacement magnitude contour plot for MFS3

We can more closely compare the results between MFS3 and the reference solution by considering the results along Line 1 in Figure 5.8. The displacement magnitude results are shown in Figure 5.11 and the effective stress results are shown in Figure 5.12. The displacement magnitude results are in agreement while the effective stress results are similar but variations are observed. Specifically, the stress is discontinuous in the finite element solution since we are using a linear brick element, but the 27-node brick element would produce a continuous stress, while for the method of finite spheres the stress is in fact continuous but has oscillations due to the nature of the interpolations functions based on the Shepard partition of unity.

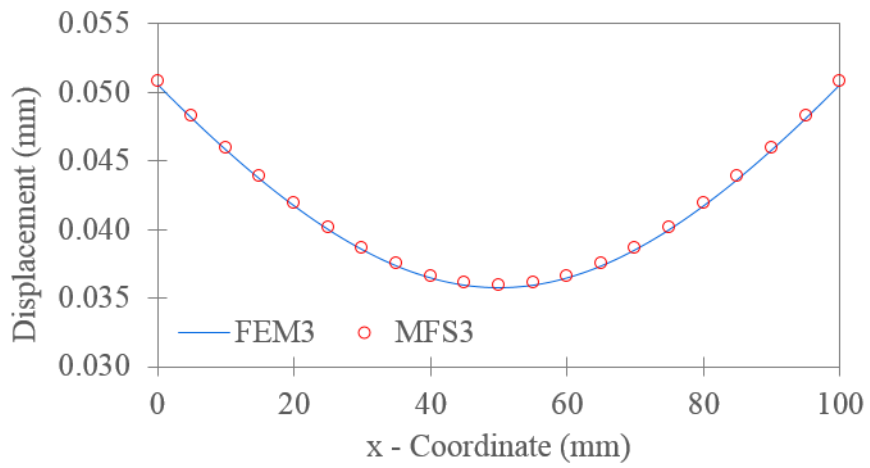


Figure 5.11 M1L2 displacement magnitude along Line 1

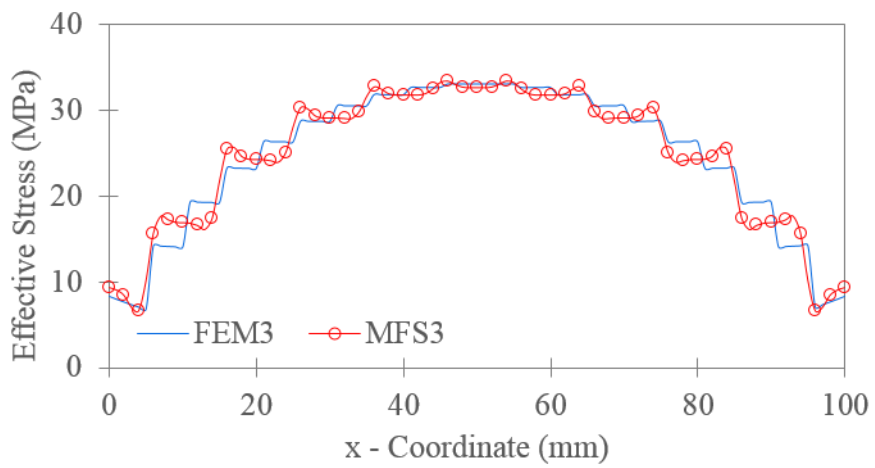


Figure 5.12 M1L2 effective stress along Line 1

After considering two different load cases for the same three-dimensional model, namely the short cantilever beam with square solid section, collective insights regarding the reliability and efficiency of the method of finite spheres are observed. In summary, the method of finite spheres is capable of obtaining comparable accuracy to the finite element method in terms of strain energy, with time multipliers around roughly one order of magnitude. Results along specific model lines show that qualitatively, the method of finite spheres provides the same results as the finite element method. These numerical examples can of course be solved easily and efficiently with the finite element method, but they have

provided some insights into the reliability and efficiency of the method of finite spheres. In the next section, we will consider a slightly more complex three-dimensional model.

5.2 Model 2: Short cantilever beam with square hollow section

The next model to be considered is the short cantilever beam with a square hollow section. The geometry, boundary conditions, and material data are given in Figure 5.13.

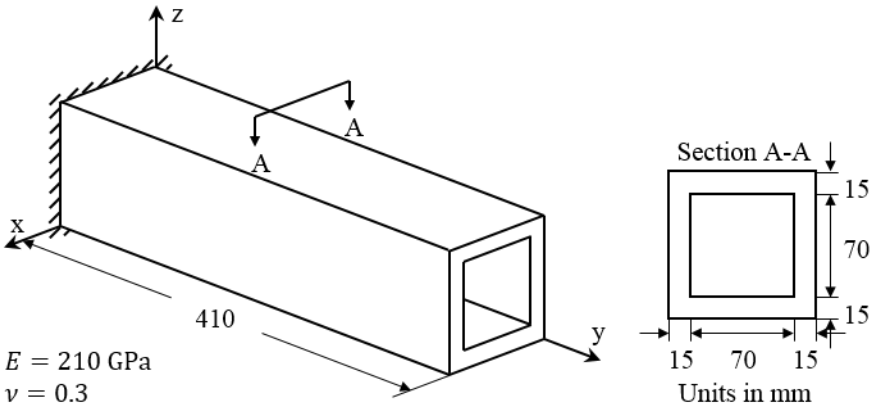
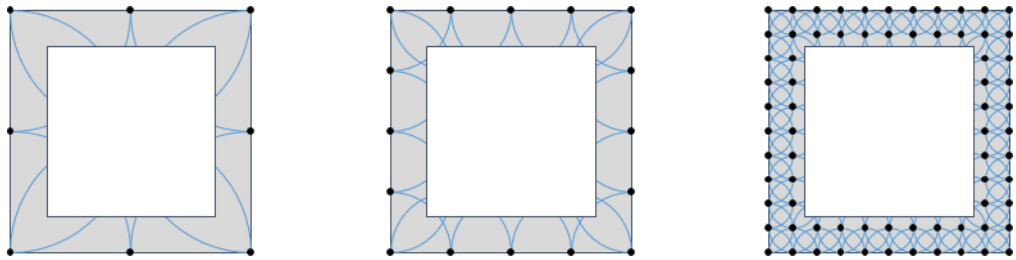


Figure 5.13 Geometry, boundary conditions, and material data for short cantilever beam with square hollow section

The problem will be solved using three discretizations for both the method of finite spheres and the finite element method. For the method of finite spheres, the sphere arrangement is uniform with equal radius size for each discretization, as shown in Figure 5.14. We will again be using the linear sphere element described in Section 4.1. Furthermore, it is worth noting that to arrive at the discretization for the square hollow section from the square solid section, we simply remove the spheres that have center coordinates that no longer lie within the domain of this model. Also, from all three MFS discretizations it is evident that sphere nodes do not need to coincide with the geometry boundaries, supported by the absence of sphere nodes on the internal surface of the square

hollow section. Mesh refinement for the finite element method involves subdividing the 8-node brick elements into eight smaller brick elements, and the finest finite element discretization will be used as the reference solution.



(a) MFS1: r = 50 mm (b) MFS2: r = 25 mm (c) MFS3: r = 10 mm

Figure 5.14 MFS discretizations at Section A-A for short cantilever beam with square hollow section

The following sections illustrate the results of the short cantilever beam with square hollow section when subjected to a bending load and torsional load, respectively, at the free end.

5.2.1 Load 1: Bending load at free end

In Figure 5.15, we illustrate the short cantilever beam with square hollow section with a bending load at the free end, referred to as Model 2 Load 1 (M2L1).

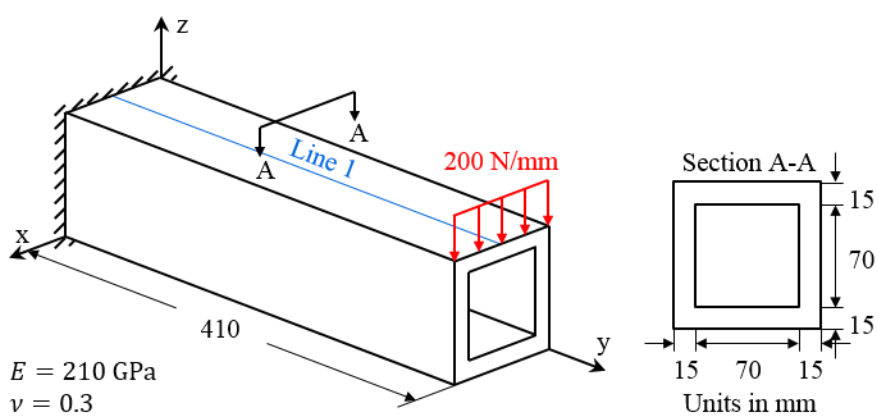


Figure 5.15 M2L1: Short cantilever beam with square hollow section with bending load at free end

The results for the MFS and FEM discretizations are summarized in Table 5.3, showing the strain energy errors and time multipliers with respect to FEM3, the reference solution. All three MFS discretizations provide results with strain energy errors within 7%. The time multiplier for MFS3 is 6.54, corresponding to a strain energy error of 1.04%, with respect to the reference solution. This is an encouraging result since for this slightly more complex three-dimensional model, the time multiplier indicates that the method of finite spheres is less than one order of magnitude slower than the finite element method for a comparable level of accuracy.

Table 5.3 M2L1 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	80	6.87	0.68
MFS2	288	4.04	2.38
MFS3	3024	1.04	6.54
FEM1	840	4.81	0.01
FEM2	4920	1.52	0.05
FEM3	32400	*	*

*FEM3: Strain energy (N·mm) = 4096.9; Time (s) = 11.64

Strain energy convergence curves for the method of finite spheres and the finite element method are shown in Figure 5.16. Based on the chosen element types, namely the linear brick element and the linear sphere element, the method of finite spheres will exhibit better accuracy compared to the finite element method when equal finite sphere and finite element sizes are considered. Furthermore, the method of finite spheres continues to exhibit a similar rate of convergence as the finite element method, since the degree of polynomial completeness is the same for both elements.

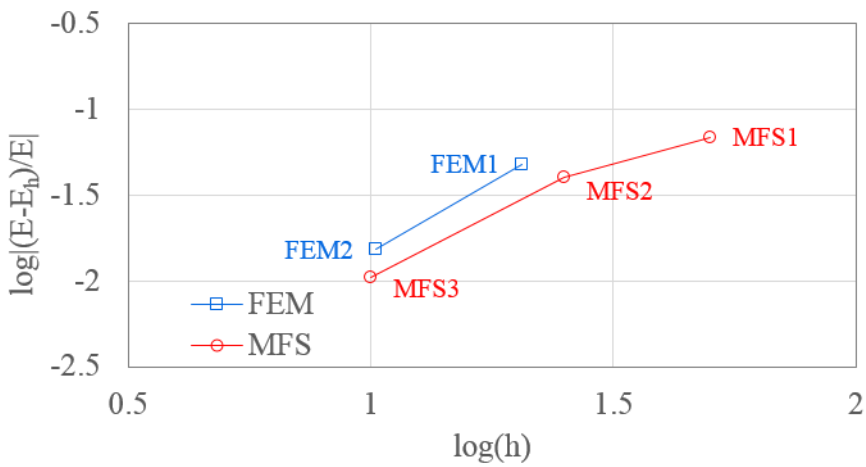


Figure 5.16 M2L1 convergence of strain energy for the method of finite spheres and the finite element method

The following results illustrate the qualitative behavior of the short cantilever beam with square hollow section for MFS3, the finest method of finite spheres discretization with a total of 3024 spheres. The variables to be predicted are transverse displacement and longitudinal normal stress. The transverse displacement contour plot is shown in Figure 5.17, where the centers of the spheres with radius $r = 10$ mm are denoted by the red nodes. Qualitatively, the transverse displacement contour plot appears to be accurate.

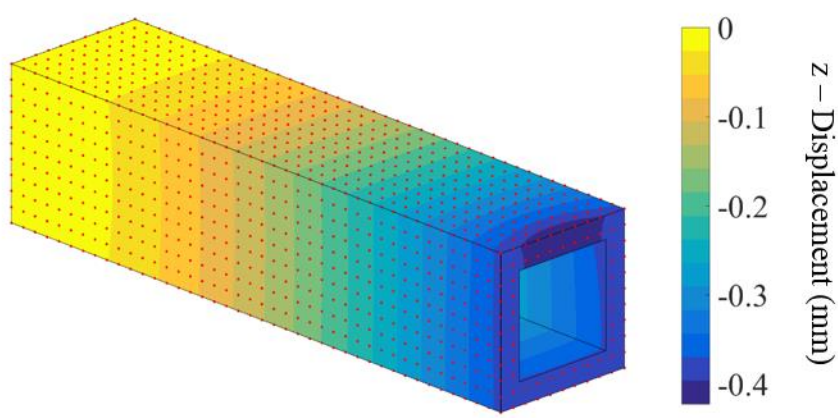


Figure 5.17 M2L1 transverse displacement contour plot for MFS3

From Figure 5.15, we choose Line 1 to compare the transverse displacement and longitudinal normal stress between MFS3 and FEM3, the reference solution. Since we are considering a short cantilever beam with a hollow structural section, opposed to a solid section as in Section 5.1, we expect the transverse displacements along Line 1 to be larger than those in Figure 5.6. Indeed the z-displacement results along Line 1 are larger, and the results between the two methods are in agreement, as shown in Figure 5.18. The longitudinal normal stresses along Line 1 are shown in Figure 5.19, and again we see that the method of finite spheres provides reliable results as compared to the reference solution.

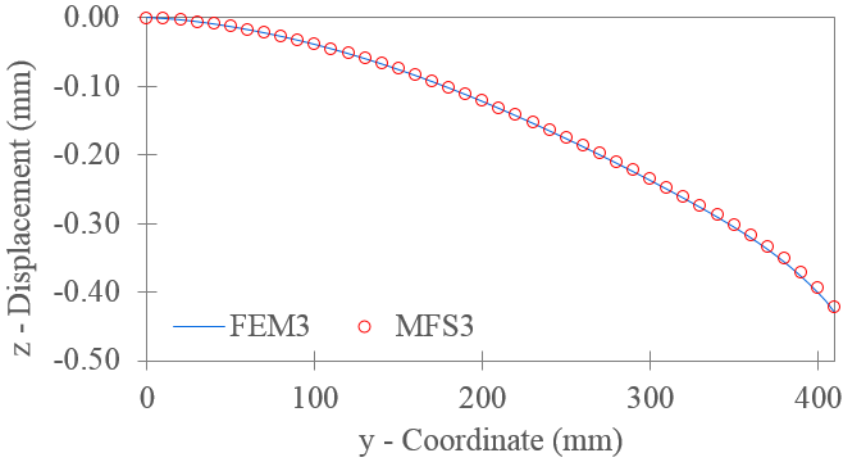


Figure 5.18 M2L1 transverse displacement along Line 1

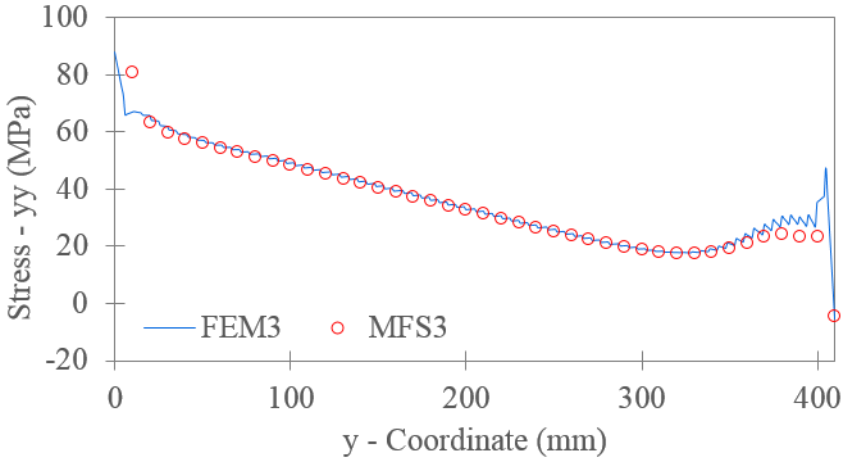


Figure 5.19 M2L1 longitudinal normal stress along Line 1

M2L1 has provided some additional insights into the method of finite spheres. The strain energy results are within a small error and indicate that the overall displacement response is accurate compared to the reference solution. Displacement and stress results along model lines further substantiate the claim that the method of finite spheres is a reliable method for solving three-dimensional linear elasticity problems, and the fact that the solution times are within an order of magnitude of the finite element method indicate that it remains competitive in terms of computational efficiency.

5.2.2 Load 2: Torsional load at free end

In Figure 5.20, we illustrate the short cantilever beam with square hollow section with a torsional load at the free end, referred to as Model 2 Load 2 (M2L2).

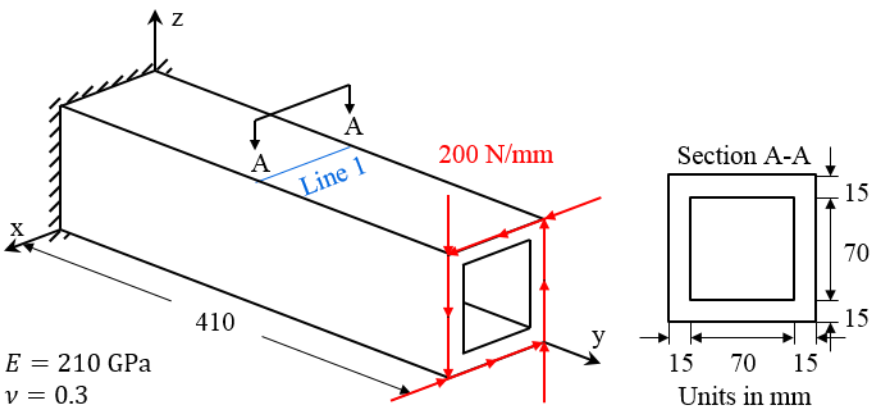


Figure 5.20 M2L2: Short cantilever beam with square hollow section with torsional load at free end

The results for the MFS and FEM discretizations are summarized in Table 5.4, showing the strain energy errors and time multipliers with respect to FEM3, the reference solution. These are the same discretizations used in Section 5.2.1. All three MFS discretizations provide results with strain energy errors within 8%. The time multiplier for

MFS3 is 6.91, corresponding to a strain energy error of 0.50%, with respect to the reference solution, which is a promising result for the effectiveness of the method of finite spheres.

Table 5.4 M2L2 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	80	7.31	0.71
MFS2	288	3.56	2.54
MFS3	3024	0.50	6.91
FEM1	840	5.64	0.01
FEM2	4920	2.03	0.05
FEM3	32400	*	*

*FEM3: Strain energy (N·mm) = 4145.4; Time (s) = 10.87

The convergence of strain energy for the method of finite spheres and the finite element method is shown in Figure 5.21. The same observations can be made that the method of finite spheres has superior accuracy compared to the finite element method for the same element size, and the rate of convergence between the two methods is similar.

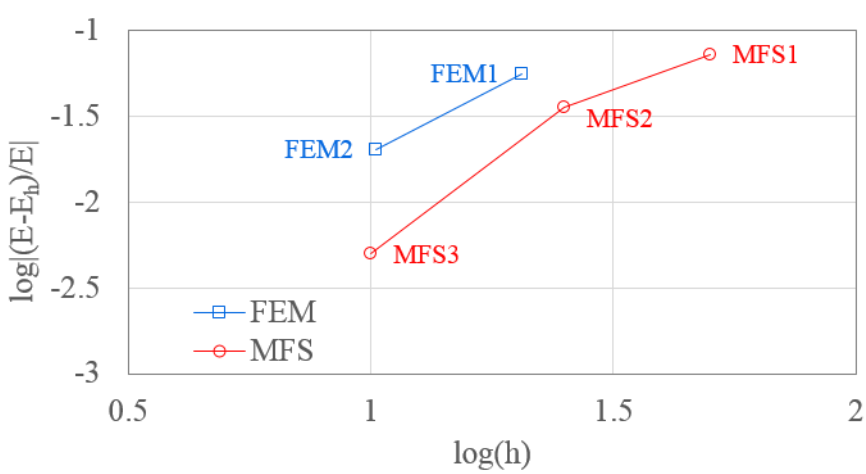


Figure 5.21 M2L2 convergence of strain energy for the method of finite spheres and the finite element method

The following will consider the qualitative results for MFS3, which has a total of 3024 spheres in the discretization. In Figure 5.22, the displacement magnitude contour plot for the short cantilever beam with square hollow section with a torsional load at the free end is shown, where the red nodes represent the sphere centers. As previously mentioned, sphere centers do not need to coincide with the domain boundaries, which is the case for the internal surface of the short cantilever beam with square hollow section. Qualitatively, the displacement magnitude contour plot appears to be reasonable for the given geometry, boundary conditions, material data, and loading conditions.

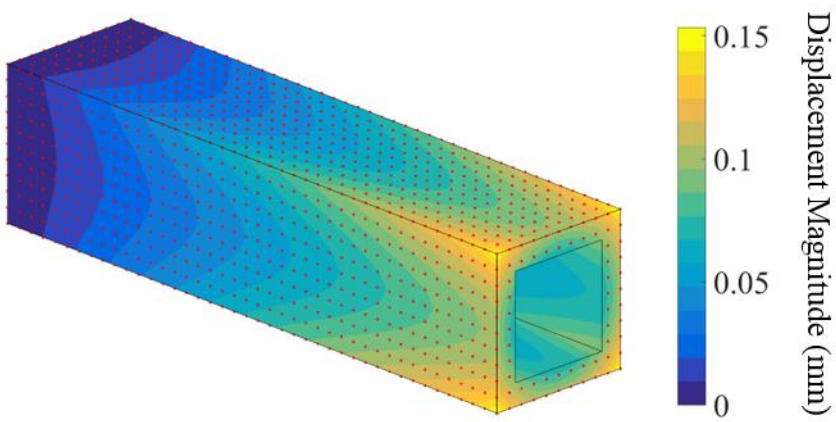


Figure 5.22 M2L2 displacement magnitude contour plot for MFS3

We consider Line 1 in Figure 5.20 to compare the results between MFS3 and FEM3, the finite element reference solution. In Figure 5.23, we see that the displacement magnitude results are in agreement. Qualitatively, the effective stress results in Figure 5.24 indicate that the method of finite spheres provides reasonably accurate results, but as previously mentioned, the slight variations are due to the differences in the continuity of the interpolation functions between the finite element method and the method of finite spheres.

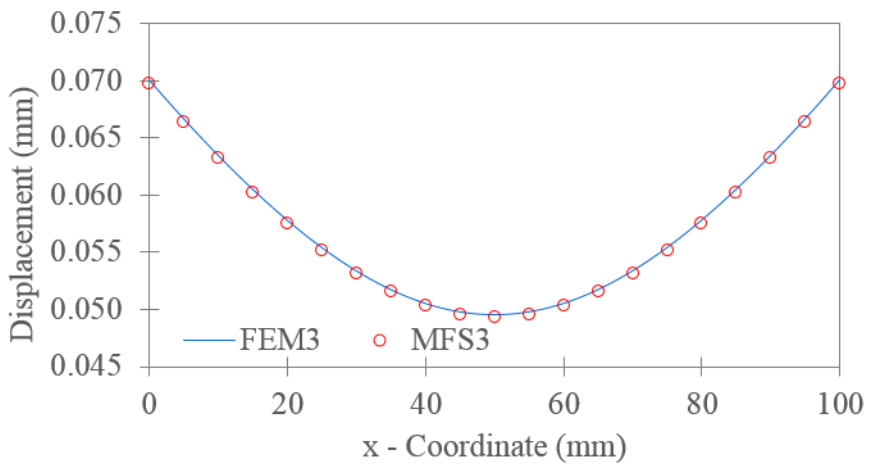


Figure 5.23 M2L2 displacement magnitude along Line 1

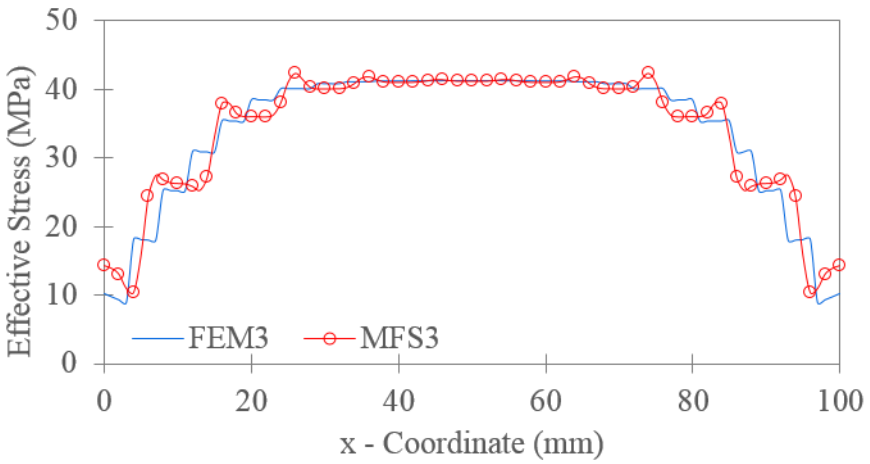


Figure 5.24 M2L2 effective stress along Line 1

We have now studied the short cantilever beam with either a solid or hollow square section, considering either a bending load or torsional load at the free end. The results for M2L2 further substantiate the trends observed in previous cases. The most important result thus far is that for comparable accuracy, the solution time for the method of finite spheres is within an order of magnitude time multiplier compared to the finite element method. For example, considering M2L2, for FEM3, the finite element reference solution, the solution time was 10.87s, and for MFS3, the method of finite spheres solution that has a 0.50% strain energy error, the solution time was 75.12s. These computational times are promising

results for the method of finite spheres since there is still a substantial time savings in the discretization of these three-dimensional problems that is not accounted for in the reported solution times. In the next section, we will consider the machine tool jig, a slightly more complex three-dimensional model with curved boundaries, representing geometries where the use of the method of finite spheres is certainly more justified.

5.3 Model 3: Machine tool jig

The final model we would like to investigate is the machine tool jig, which introduces a cutout to Model 2, the short cantilever beam with square hollow section. The geometry, boundary conditions, and material data are given in Figure 5.25. Due to the curved boundaries, discretization of the domain using finite elements is no longer as straightforward as the previous two models. Of course, this problem is still easily and efficiently solved using the finite element method, but we use this representative three-dimensional linear elasticity problem to motivate the use of the method of finite spheres where geometries are truly very complicated and when meshing in the traditional finite element method is cumbersome and may introduce distorted elements.

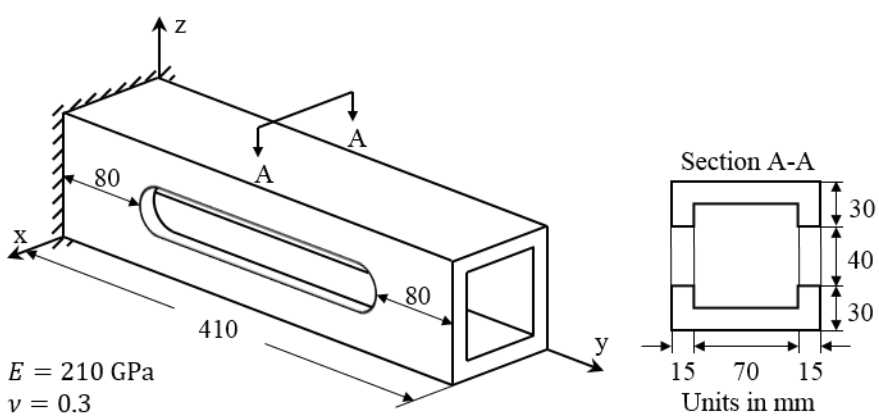
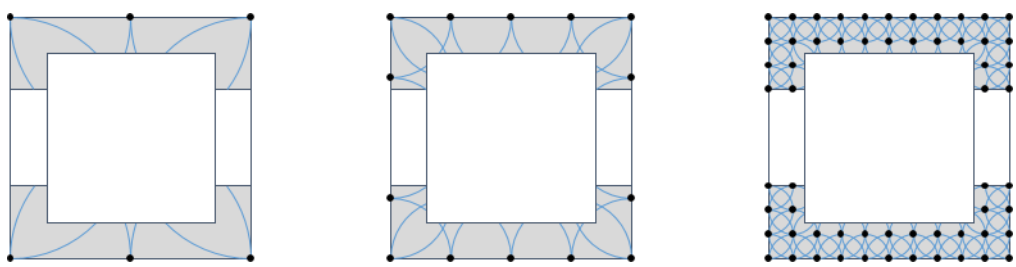


Figure 5.25 Geometry, boundary conditions, and material data for machine tool jig

The problem will be solved using three discretizations for both the method of finite spheres and the finite element method. For the method of finite spheres, the sphere arrangement is uniform with equal radius size for each discretization, as shown in Figure 5.26. The linear sphere element described in Section 4.1 is used. To arrive at the discretization for the machine tool jig from the short cantilever beam with square hollow section, we simply remove the spheres that have center coordinates in the volume of the machine tool jig cutout. The three finite element discretizations use the 8-node linear brick element, and each mesh refinement involves subdividing the brick element into eight smaller brick elements, with the final discretization being used as the reference solution.



(a) MFS1: r = 50 mm (b) MFS2: r = 25 mm (c) MFS3: r = 10 mm

Figure 5.26 MFS discretizations at Section A-A for machine tool jig

The following sections illustrate the results of the machine tool jig when subjected to a bending load and torsional load, respectively, at the free end.

5.3.1 Load 1: Bending load at free end

In Figure 5.27, we illustrate the machine tool jig with a bending load at the free end, referred to as Model 3 Load 1 (M3L1). This specific case has been studied in depth using different hierarchical models from a beam model, to a shell model, to the three-dimensional elasticity model [37].

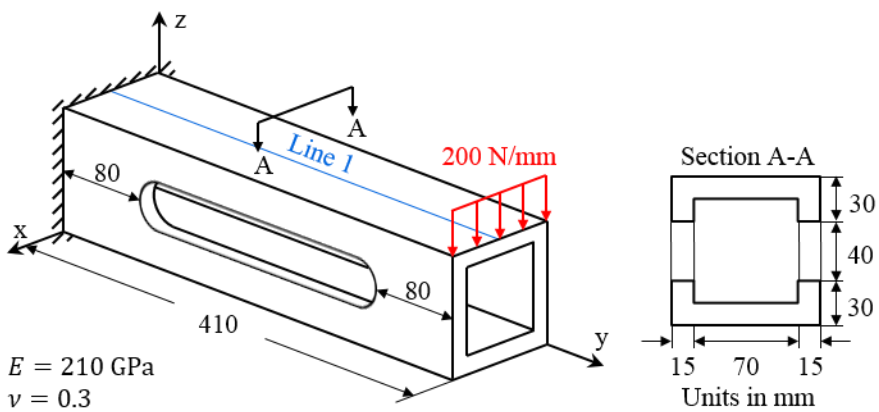


Figure 5.27 M3L1: Machine tool jig with bending load at free end

The results for the MFS and FEM discretizations are summarized in Table 5.5, showing the strain energy errors and time multipliers with respect to FEM3, the reference solution. For this case, we see that the strain energy errors are within 30%, much larger than in the previous numerical examples. This suggests that the coarse MFS discretizations are not enough to accurately capture the stiffness of the machine tool jig which has a cutout. However, MFS3 is reasonably accurate with a strain energy error of 4.27% corresponding to a time multiplier of 7.59, with respect to the finite element reference solution.

Table 5.5 M3L1 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	68	29.92	1.22
MFS2	268	17.24	4.53
MFS3	2736	4.27	7.59
FEM1	1060	13.03	0.01
FEM2	6018	3.57	0.09
FEM3	38950	*	*

*FEM3: Strain energy (N·mm) = 10185.1; Time (s) = 8.99

The convergence in strain energy for the method of finite spheres and the finite element method are shown in Figure 5.28. Even though the strain energy errors are larger for the coarse MFS discretizations than in previous models, the same is true of the coarse FEM discretizations. Consequently, it is still true that the method of finite spheres is more accurate than the finite element method for equal sphere radius versus element size, and they both still exhibit similar convergence rates. These trends seem to remain valid regardless of the geometry or loading conditions since accuracy and convergence considerations are mostly dependent on the accuracy of the element, where the degree of polynomial completeness is generally considered to be the most important measure of the accuracy of an element.

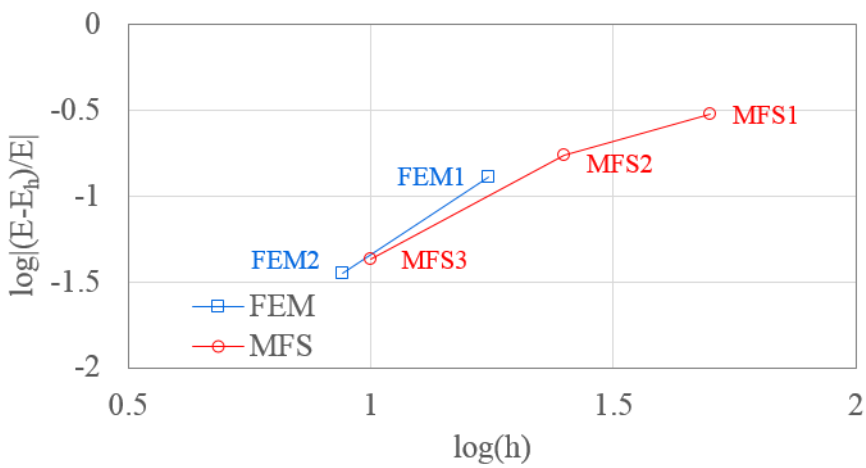


Figure 5.28 M3L1 convergence of strain energy for the method of finite spheres and the finite element method

The following results illustrate the qualitative behavior of the machine tool jig for MFS3, the finest method of finite spheres discretization with a total of 2736 spheres. The variables to be predicted are transverse displacement and longitudinal normal stress. The transverse displacement contour plot, which is qualitatively correct, is shown in Figure 5.29, with sphere centers denoted by the red nodes.

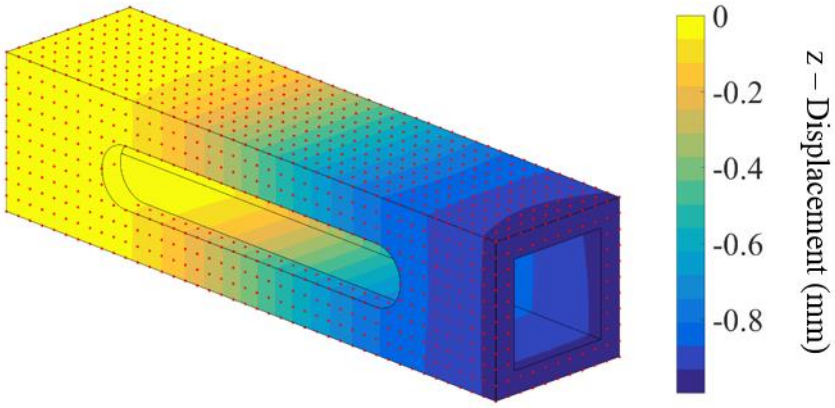


Figure 5.29 M3L1 transverse displacement contour plot for MFS3

From Figure 5.27, we select Line 1 to closely examine and compare the transverse displacement and longitudinal normal stress between MFS3 and FEM3, the finite element reference solution. In Figure 5.30, we observe that the method of finite spheres provides reasonably accurate z-displacement results as compared to the finite element reference solution, but are not as precise as in previous numerical examples. Of course, with further refinement of the sphere discretization, the method of finite spheres numerical solution would converge to the reference solution. However, this discrepancy in transverse displacement, which is more noticeably observed at the free end of the machine tool jig, explains the slightly larger strain energy error of 4.27%. Furthermore, comparing the results to those in Figure 5.18 for the short cantilever beam with square hollow section, we observe that the transverse displacement at the free end is more than double, illustrating the effect of the cutout of the machine tool jig. The longitudinal normal stress results along Line 1 are shown in Figure 5.31, and are shown to be in agreement between the two methods. For this relatively complex three-dimensional model with curved boundaries, the method of finite spheres is still capable of providing reasonably accurate solutions with regard to strain energy and displacements and stresses along model lines.

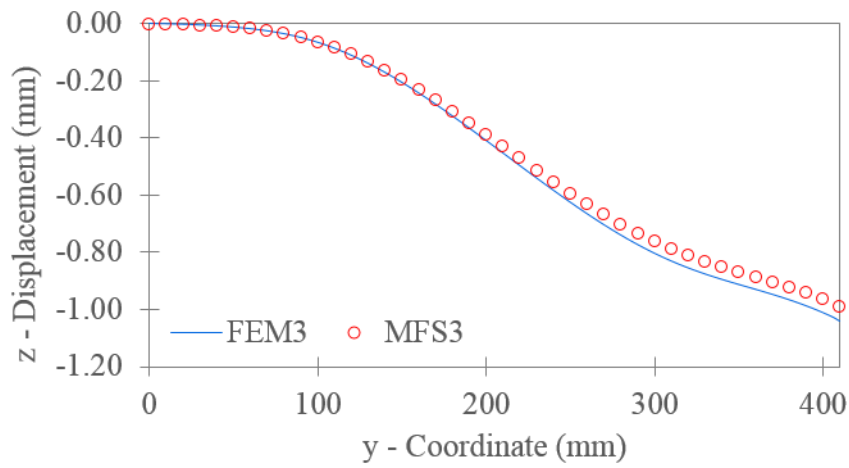


Figure 5.30 M3L1 transverse displacement along Line 1

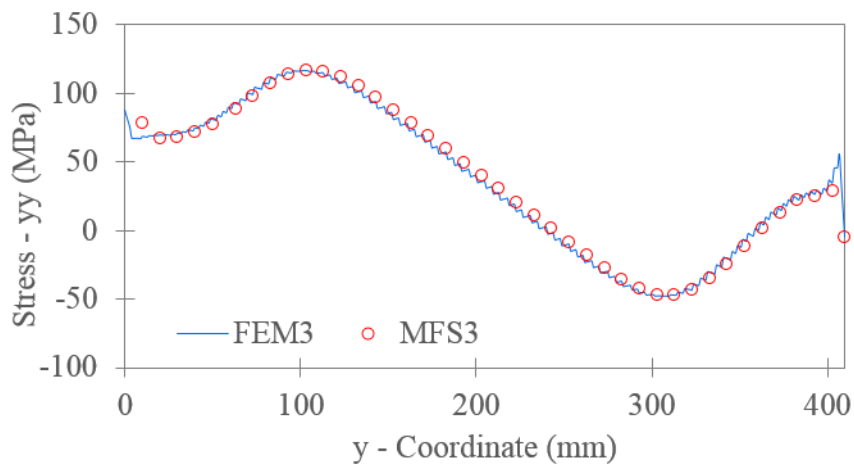


Figure 5.31 M3L1 longitudinal normal stress along Line 1

M3L1 has provided some additional interesting insights into the method of finite spheres. From the coarse MFS discretizations, we observed that the strain energy errors are quite significant, explained by the fact that the distribution of spheres is not fine enough to capture the three-dimensional geometry, particularly in the vicinity of the cutout region. However, with the finest method of finite spheres discretization, we observed that the strain energy error is fairly reasonable and can of course be improved with further refinement. Also, it is worth nothing that the number of nodes in MFS3 is roughly two orders of magnitude smaller than the number of nodes in FEM3, the finite element reference

solution, and so strain energy errors within roughly 5% are fairly reasonable. Differences in the displacement results along model lines were observed, which explains the slightly larger strain energy error as compared to previous numerical examples. Overall, despite the small discrepancies, the method of finite spheres still provides reasonably accurate results, with solution times within an order of magnitude of the finite element method, even for this more complex three-dimensional model, where the advantages in discretization during the pre-processing phase are more substantial.

For the machine tool jig model, we would also like to show a comparison between the method of finite spheres and the finite element method using the 27-node element instead of the 8-node linear brick element. The problem will be solved using three finite element discretizations as before, except with the 8-node elements replaced by 27-node elements, and the finest discretization will be used as the reference solution. At the cost of additional computational time, this reference solution will be closer to the exact solution. In Table 5.6, we illustrate the strain energy errors and time multipliers for the MFS and FEM discretizations using the 27-node finite element, with respect to FEM3, the reference solution. When comparing these results to those in Table 5.5 where the 8-node finite element is used, it is evident that strain energy errors for the method of finite spheres are slightly larger but with drastically reduced time multipliers, since the solution time for the 27-node finite element reference solution is significantly longer, by roughly 150 times, than for the 8-node finite element reference solution. Ultimately, MFS3 provides a solution with a strain energy error of 5.82% corresponding to a time multiplier of 0.05 with respect to the reference solution, which demonstrates that the linear sphere element is more computationally efficient than the 27-node finite element, given this level of error.

Table 5.6 M3L1 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution) with 27-node finite element

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	68	31.05	0.01
MFS2	268	18.58	0.03
MFS3	2736	5.82	0.05
FEM1	6018	1.13	<0.01
FEM2	38950	0.27	0.02
FEM3	276174	*	*

*FEM3: Strain energy (N·mm) = 10352.9; Time (s) = 1371.97

The convergence in strain energy for the method of finite spheres and the finite element method using the 27-node element are shown in Figure 5.32. The results indicate that for equal radius size in the method of finite spheres to element size in the finite element method, the finite element results are more accurate. This is expected since the 27-node element has polynomial completeness of degree two while the linear sphere element has polynomial completeness of degree one. Furthermore, a slightly higher rate of convergence is observed for the 27-node finite element than for the linear sphere element.

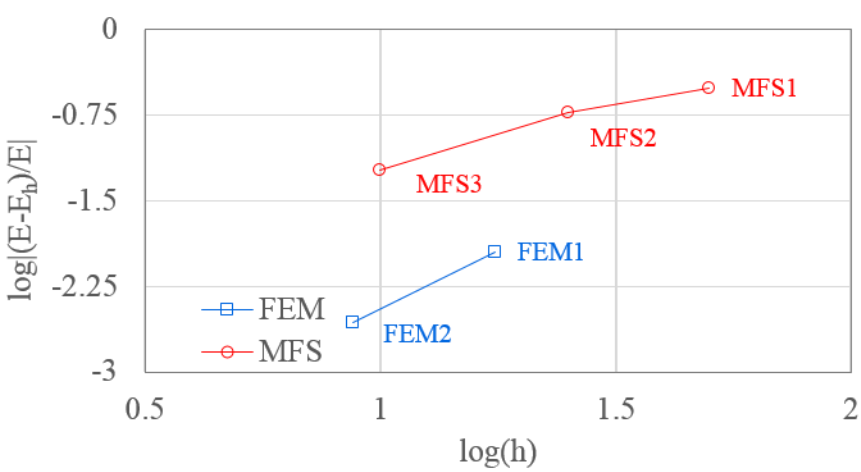


Figure 5.32 M3L1 convergence of strain energy for the method of finite spheres and the finite element method with 27-node finite element

The following results revisit the machine tool jig qualitative behavior along Line 1 from Figure 5.27, but now using the 27-node finite element reference solution FEM3. The transverse displacement and longitudinal normal stress results are shown in Figure 5.33 and Figure 5.34, respectively. Similar to the results in Figure 5.30 with the 8-node finite element, the MFS3 results for transverse displacement differ from the reference solution, justifying the slightly larger strain energy error of 5.82%. The longitudinal normal stress results along Line 1 are still in agreement, but unlike the results with the 8-node finite element in Figure 5.31, there are no discontinuities for the 27-node finite element solution.

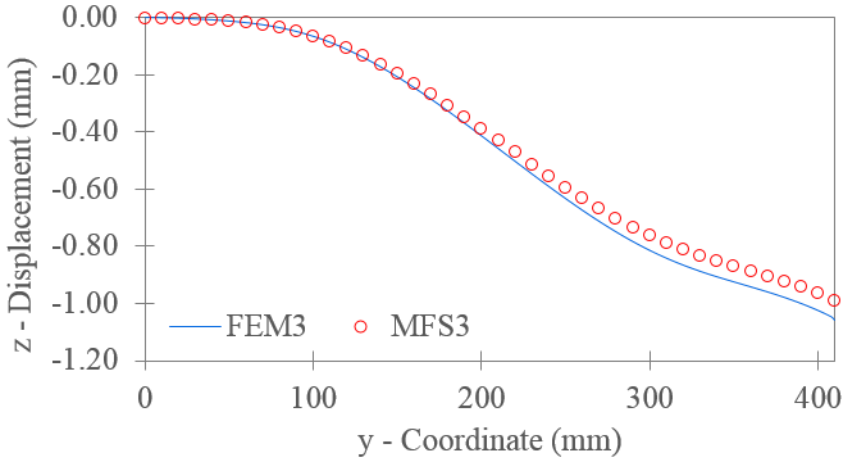


Figure 5.33 M3L1 transverse displacement along Line 1 with 27-node finite element

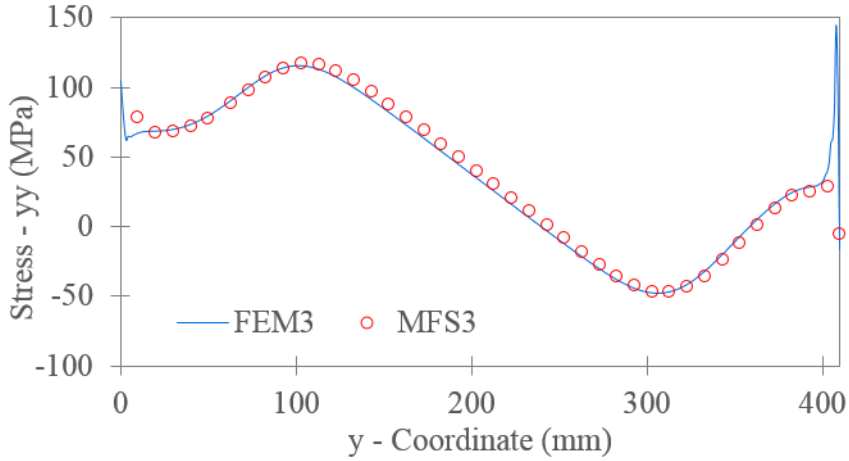


Figure 5.34 M3L1 longitudinal normal stress along Line 1 with 27-node finite element

By including a comparison with the 27-node finite element for M3L1, we observed that the method of finite spheres is quite computationally efficient. The strain energy error for the finest sphere discretization MFS3 was roughly 6% corresponding to a time multiplier of 0.05, indicating that the solution time was roughly 20 times faster than for the finite element reference solution. With a finer sphere discretization which would reduce the strain energy error, it is likely that the linear sphere element would still be computationally efficient compared to the 27-node finite element. The method of finite spheres provided reasonable results along model lines with slight variations in transverse displacement. As expected, the higher-order 27-node finite element resulted in smooth stresses whereas the 8-node finite element solution contained minor discontinuities.

5.3.2 Load 2: Torsional load at free end

In Figure 5.35, we illustrate the machine tool jig with a torsional load at the free end, referred to as Model 3 Load 2 (M3L2).

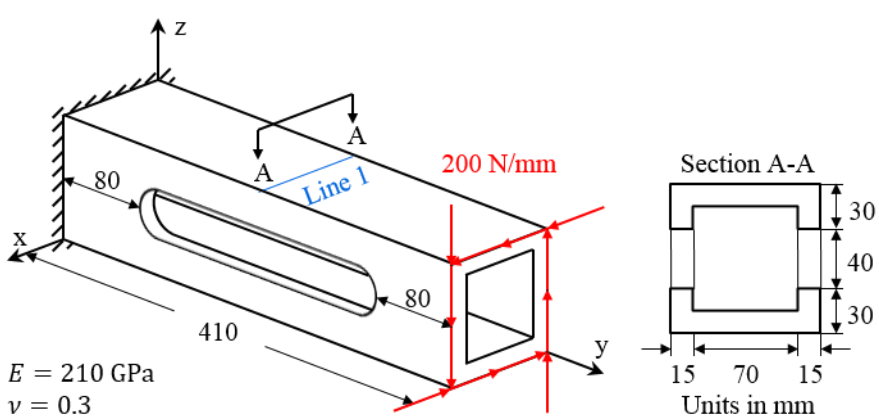


Figure 5.35 M3L2: Machine tool jig with torsional load at free end

The results for the MFS and FEM discretizations are summarized in Table 5.7, showing the strain energy errors and time multipliers with respect to FEM3, the reference

solution. The discretizations are the same as those used in Section 5.3.1. MFS3 provides reasonable accuracy with a 2.37% strain energy error, corresponding to a time multiplier of 7.83, with respect to the reference solution.

Table 5.7 M3L2 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution)

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	68	16.50	1.23
MFS2	268	9.70	4.51
MFS3	2736	2.37	7.83
FEM1	1060	9.86	0.01
FEM2	6018	3.20	0.06
FEM3	38950	*	*

*FEM3: Strain energy (N·mm) = 16369.5; Time (s) = 8.91

Strain energy convergence curves for the method of finite spheres and the finite element method are shown in Figure 5.36. The method of finite spheres provides better accuracy than the finite element method for equal radius versus element size and the rate of convergence for the two methods are very similar.

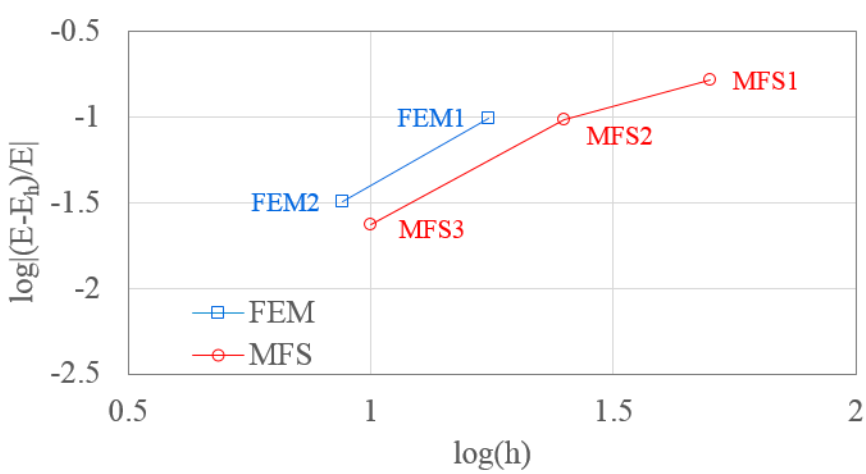


Figure 5.36 M3L2 convergence of strain energy for the method of finite spheres and the finite element method

The following results will investigate the qualitative behavior of the machine tool jig with a torsional load at the free end for the MFS3 discretization, which has a total of 2736 spheres. The displacement magnitude contour plot is shown in Figure 5.37.

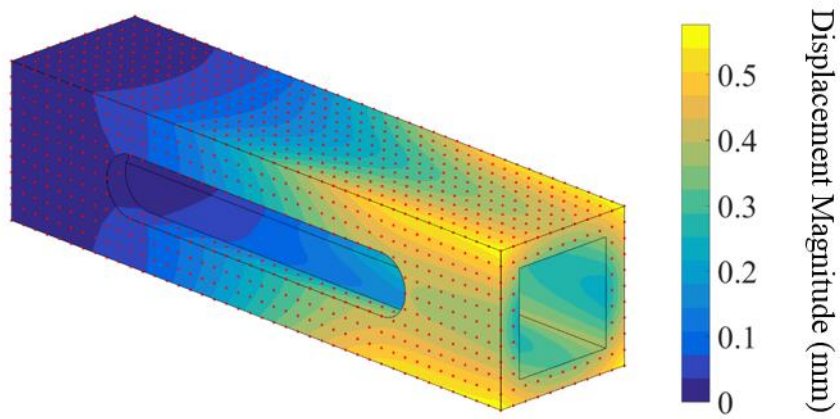


Figure 5.37 M3L2 displacement magnitude contour plot for MFS3

We will once again compare the results between MFS3 and FEM3 by considering the results along Line 1 in Figure 5.35. The results of interest are displacement magnitude and effective stress, shown in Figure 5.38 and Figure 5.39, respectively. The displacement magnitude results are in agreement while the effective stress results, as before, show some slight variations due to differences in the interpolation functions used in either method.

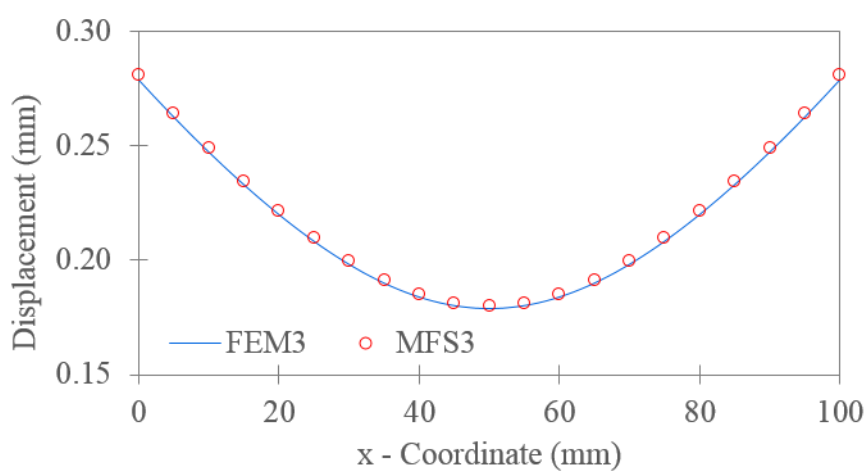


Figure 5.38 M3L2 displacement magnitude along Line 1

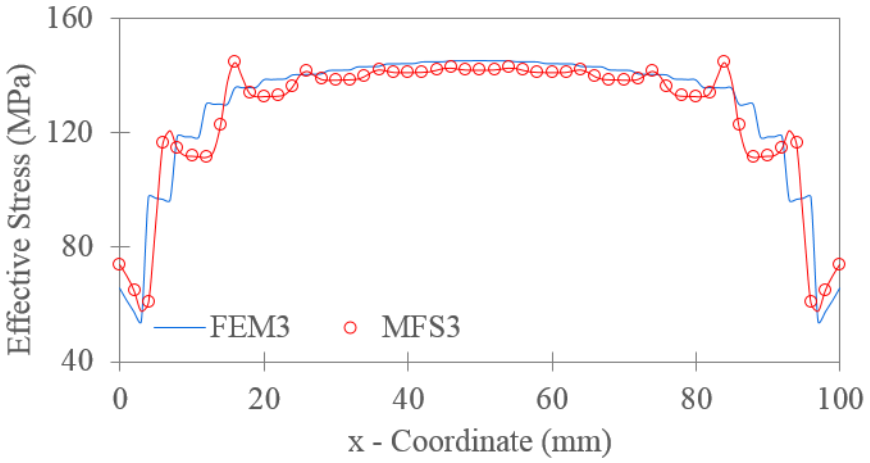


Figure 5.39 M3L2 effective stress along Line 1

M3L2 is the final numerical example, and has provided similar insights into the reliability and effectiveness of the method of finite spheres as compared to previous models and load cases. In particular, solution times are within an order of magnitude of the solution times for the finite element method, when considering comparable solution accuracy based on the strain energy error norm. Convergence rates for the two methods are very similar which is due to the fact that the elements have the same degree of polynomial completeness. Lastly, the displacement magnitude and effective stress results along the model line are in agreement.

For this torsional load case, we would again like to show a comparison between the method of finite spheres and the finite element method with the 27-node finite element. In Table 5.8, we illustrate the strain energy errors and time multipliers for the MFS and FEM discretizations using the 27-node finite element, with respect to the reference solution FEM3. For the method of finite spheres, strain energy errors are slightly larger but with drastically reduced time multipliers. The solution time for MFS3 is roughly 20 times faster than for the reference solution, but corresponds to a 4.12% error in strain energy.

Table 5.8 M3L2 strain energy errors and computational time multipliers for MFS and FEM discretizations (as compared to FEM3 reference solution) with 27-node finite element

	Number of nodes	Strain energy error (%)	Time multiplier
MFS1	68	18.00	0.01
MFS2	268	11.31	0.03
MFS3	2736	4.12	0.05
FEM1	6018	2.22	<0.01
FEM2	38950	0.55	0.02
FEM3	276174	*	*

*FEM3: Strain energy (N·mm) = 16668.2; Time (s) = 1359.04

The convergence in strain energy for the method of finite spheres and the finite element method using the 27-node element are shown in Figure 5.40. Since the 27-node finite element has a higher degree of polynomial completeness than the linear sphere element, the finite element method has a higher convergence rate than the method of finite spheres, and is also more accurate when considering equal element versus sphere radius size, as observed by the greater slope and lower position of the finite element curve.

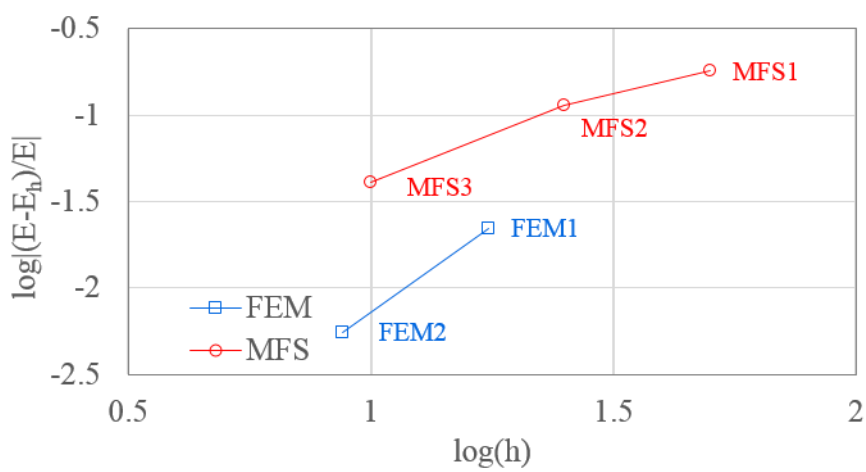


Figure 5.40 M3L2 convergence of strain energy for the method of finite spheres and the finite element method with 27-node finite element

We consider Line 1 in Figure 5.35 to again compare the results between MFS3 and the reference solution FEM3 using the 27-node element. In Figure 5.41, there is a noticeable difference in displacement magnitude between the two methods, although the general response is similar and sphere refinement would reduce this discrepancy. Using the 27-node finite element, the effective stress results are continuous, as shown in Figure 5.42, and while the method of finite spheres provides a fairly reasonable comparison, there are nonphysical oscillations which will be further discussed in Section 5.4.

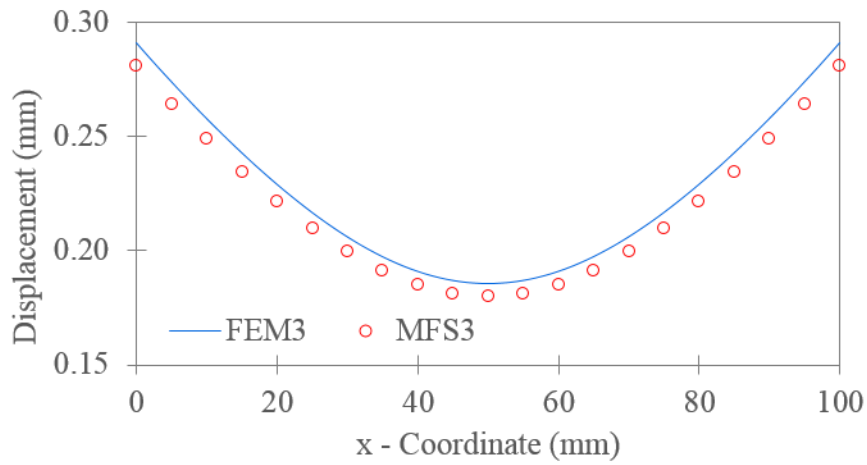


Figure 5.41 M3L2 displacement magnitude along Line 1 with 27-node finite element

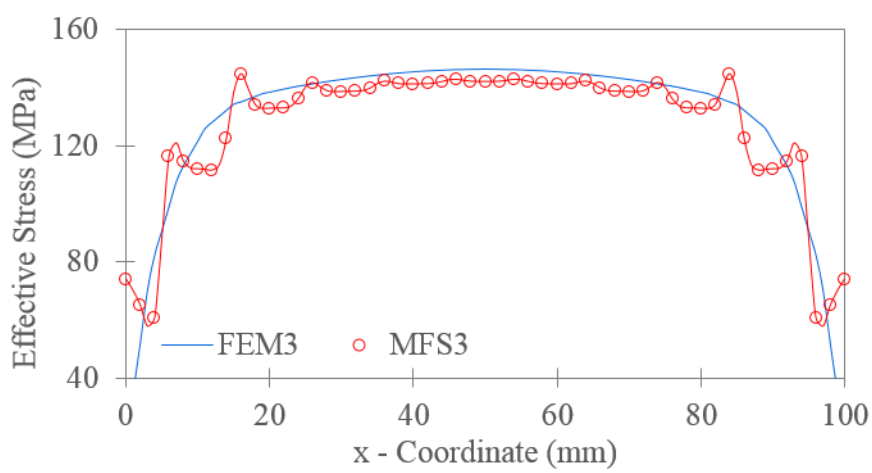


Figure 5.42 M3L2 effective stress along Line 1 with 27-node finite element

By solving the machine tool jig problem with the method of finite spheres using the linear sphere element, and the finite element method using both the 8-node and 27-node element, we believe the method of finite spheres is computationally efficient for three-dimensional static linear elasticity problems, especially when compared to the 27-node finite element. For strain energy errors within approximately 5%, computational times for the method of spheres are roughly an order of magnitude slower compared to the 8-node finite element, but are an order of magnitude faster compared to the 27-node finite element. These computational times are promising results for the effectiveness of the method of finite spheres, particularly with the additional benefits of the method, such as ease of discretization and the elimination of element distortion. In the next section, we will summarize and discuss the key observations from the numerical examples that have been presented.

5.4 Discussion of numerical results

The discussion of numerical results will highlight the key observations from the analysis of three-dimensional linear elasticity problems. As previously mentioned, these key insights cannot be generalized to all three-dimensional static linear elasticity problems, but it does provide a reasonable foundation for determining the reliability and effectiveness of the method of finite spheres for these types of problems.

Our study of three-dimensional linear elasticity problems consisted of three increasingly complex models each with two load cases, a bending load and a torsional load, respectively. To establish reliability of the method of finite spheres, we evaluated the overall strain energy as well as displacements and stresses along model lines.

Strain energy errors were based on the finite element method reference solution, and for both the finite element method and the method of finite spheres, convergence in strain energy was observed from below, as expected. The numerical results for strain energy indicated that the method of finite spheres has a comparable convergence rate as the finite element method using the 8-node linear brick element. This observation is also expected because the two methods should have a comparable order of convergence since the elements used, namely the linear brick element and linear sphere element, employ complete polynomials of degree one. Furthermore, for equal average element size in the finite element method to radius size in the method of finite spheres, the latter is more accurate indicating fewer nodes can be used to obtain comparable accuracy due to the nature of the interpolation functions based on the Shepard partition of unity.

The displacement and stress results along model lines further established the reliability of the method of finite spheres. However, from the stress results, particularly the effective stress results for the torsional load cases, oscillations were observed. These nonphysical oscillations have also been observed in other meshless methods, especially for regions of high stress gradients. The oscillation error is typically attributed to the high frequency content of the derivatives of the interpolation functions. This is the case for the method of finite spheres which employs the quartic spline weighting function in the definition of the Shepard partition of unity functions. These errors can be addressed by changing the domain of influence, i.e., the radius of the spheres, or the interpolation functions, i.e., the weighting function or the terms in the local basis. For the finite element method, stress results were discontinuous for the 8-node element but as shown for the machine tool jig model, the 27-node element provided smooth stress results.

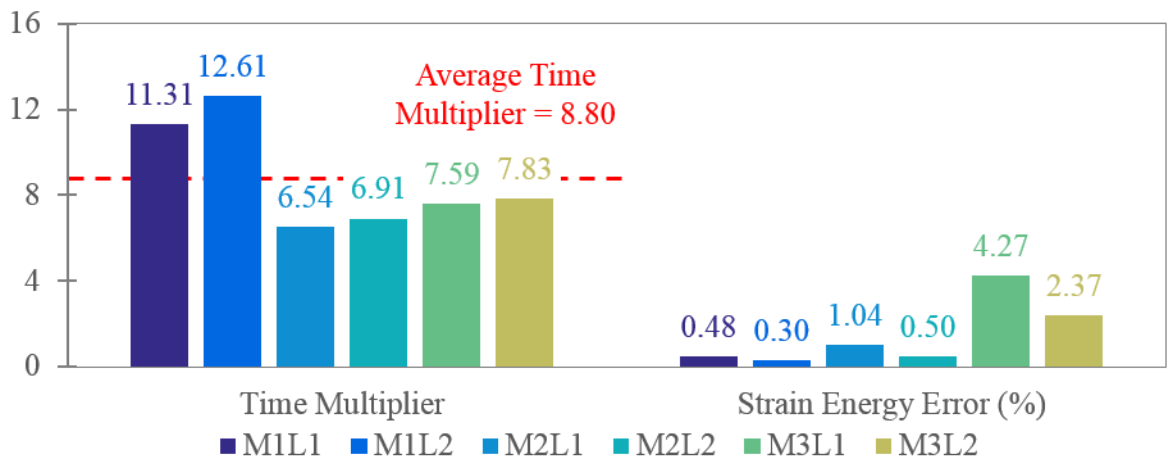


Figure 5.43 Time multiplier and strain energy error for the method of finite spheres (MFS3) as compared to the finite element method (FEM3) reference solution, for the different models and load cases

In Figure 5.43, we show the time multipliers and corresponding strain energy errors for the method of finite spheres (MFS3) solution as compared to the finite element method (FEM3) reference solution using the 8-node element. The average time multiplier for the six numerical examples is 8.80 for strain energy errors that are all within 5%. However, in some cases strain energy errors of this magnitude correspond to significant discrepancies in peak stress results, so ideally we would like to compare solution times for the same level of accuracy.

For the method of finite spheres, strain energy errors can be further reduced by refinement of sphere size and/or including higher-order polynomial terms in the local basis, but with the tradeoff of longer computational times. For Model 1 and Model 2, the solution time comparison is for about the same level of accuracy, since strain energy errors are all within roughly 1%, so we can say that the method of finite spheres is roughly one order of magnitude slower than the finite element method. However, for Model 3, the machine tool jig, in order to reduce strain energy errors to the same level of accuracy as the reference

solution, it is likely that the time multipliers would increase by an order of magnitude, and so now the method of finite spheres is between one and two orders of magnitude slower than the finite element method. Based on our study of Model 3 using the 27-node finite element, the method of finite spheres is an order of magnitude faster than the finite element method, when considering strain energy errors of roughly 5%. However, by extrapolation, we anticipate that for the same level of accuracy as the 27-node finite element reference solution, the method of finite spheres would again be around one order of magnitude slower than the finite element method. Overall, we believe that these results indicate that the method of finite spheres is approximately an order of magnitude slower than the finite element method.

The broader implications of this study are that for three-dimensional static linear elasticity problems, computational times for the method of finite spheres are roughly an order of magnitude longer than for the finite element method. For the method of finite spheres, the majority of the computational time is spent in the calculation of element matrices due to expensive numerical integration, whereas for the finite element method, the majority of time is spent in the triangularization of the stiffness matrix and the solution of equations. Continuous research in numerical integration efficiency and advances in computational speed will continue to make the method of finite spheres even more competitive. Furthermore, there is a time savings in the discretization of the domain for the method of finite spheres that is not accounted for in this time comparison, not to mention the fact that the method also eliminates the risk of distorted elements.

Chapter 6

Concluding remarks

In this chapter, we summarize the contributions of our research towards the objective of assessing the reliability and computational efficiency of the method of finite spheres for three-dimensional linear elasticity problems. We discuss the limitations of our work and propose recommendations for the future direction of the method of finite spheres in order to achieve the broader impact of distinguishing the method as a reliable and efficient meshless computational technique to supplement the capabilities of the standard finite element method.

6.1 Summary of contributions

Early research on the method of finite spheres focused on demonstrating the reliability of the method for one- and two-dimensional steady state and static problems [12,13]. Further research has focused on additional capabilities such as a mixed displacement/pressure formulation, improved numerical integration, finite element

coupling, enrichment strategies, automatic discretization, genetic algorithm numerical integration, and dynamic analysis of wave propagations [14-16,21,25,38-40]. The specific contributions identified from the research in this thesis include (1) presenting the three-dimensional interpolation functions for the linear sphere element, (2) proposing the piecewise Gauss-Legendre quadrature rule for three-dimensional spherical domains, (3) implementing the method of finite spheres in ADINA, a commercial finite element software program, using the user-element subroutine, and (4) establishing the reliability of the method of finite spheres for three-dimensional static linear elasticity problems and concluding that the computational times are roughly an order of magnitude longer than the finite element method computational times.

In the method of finite spheres, the construction of the interpolation functions is based on the terms in the local approximation space. The ability to select different local approximation spaces is advantageous because it allows for effectively solving specific problems as well as allowing for local enrichment. For example, interpolation functions with trigonometric terms in the local approximation space have been shown to be effective for solving hyperbolic problems, such as wave propagations [21]. For the solution of three-dimensional linear elasticity problems, we presented the linear sphere element in Section 4.1, which utilizes polynomial terms in the local approximation space. This element has polynomial completeness of degree one which is similar to the 8-node linear brick element in the standard finite element method.

Numerical integration has been identified as a major issue in the method of finite spheres, and in general for most meshless methods. Therefore, the key to developing a computationally efficient meshless method is to develop an efficient numerical integration

scheme, particularly in three-dimensions. In the finite element method, numerical integration is performed using Gauss-Legendre product rules which provide high accuracy with relatively low computational cost, since the functions to be integrated are polynomials and the elements do not overlap. For the method of finite spheres, while specialized integration schemes have been developed, we presented a simple and quite efficient procedure known as the piecewise Gauss-Legendre quadrature rule in Section 3.3. The integration domain is subdivided and a Gauss-Legendre quadrature rule is used in each subdomain. Advantages of this approach are that a uniform density of integration points is used and the overlap regions are integrated directly instead of considering a different quadrature rule for these regions. While many integration points are still required in comparison to the standard finite element method, this quadrature rule is relatively simple and also differs from other meshless methods in that isoparametric mapping is not performed so the Jacobian operator is not calculated.

The method of finite spheres is implemented in ADINA, a commercial finite element software program, using the user-element subroutine. Since the method of finite spheres is based on the Galerkin weak form, implementation procedures are similar to the finite element method. The user-element subroutine is used to calculate the element stiffness matrix and element load vector for each sphere and each sphere overlap region. The main program will call the user-element subroutine and assemble the element matrices into the structure stiffness and load matrices, similar to the finite element method. Since the procedures are similar to the standard finite element method, it is natural to consider the method of finite spheres as an overlapping finite element method. In Section 6.3.1, we will also discuss the advantages of coupling the finite element method and the method of

finite spheres since there are obvious benefits of using finite elements and finite spheres in different regions of the problem domain.

While the method of finite spheres has provided reliable results for a variety of one- and two-dimensional problems, until now the method had not been analyzed for three-dimensional problems. Furthermore, the issue of computational efficiency can only be adequately addressed by analyzing practical three-dimensional engineering problems, where circumventing the mesh generation stage provides a significant time savings. We choose to focus on three-dimensional static linear elasticity, a dominant class of problems in engineering analysis that is generally underrepresented by meshless methods. The major computational cost for the method of finite spheres is in the calculation of the element matrices in the user-element subroutine. We established the reliability of the method of finite spheres for three-dimensional linear elasticity problems by showing the accuracy of the solution compared to the finite element solution, in terms of overall strain energy and displacements and stresses along model lines. From the three-dimensional static linear elasticity numerical examples studied, the computational times for the method of finite spheres are roughly an order of magnitude longer than for the finite element method.

6.2 Limitations of current work

The current work presented in this thesis for the method of finite spheres is developed for linear elasticity problems. Since the discretization of the domain for the numerical examples studied used a regular arrangement of spheres, and since the material properties were constant throughout the entire domain, the element stiffness for certain spheres were identical. Therefore, we used this attribute to our advantage by only calculating the element stiffness for unique spheres once, and replicating the element

stiffness of similar spheres in the assembly of the structure stiffness matrix. However, it is important to note that this is only possible because of the limited scope of this research and if we were to consider nonlinear analysis, where the method of finite spheres is expected to be advantageous in dealing with large deformations, the ability to replicate sphere element stiffness matrices would not be possible because the spheres would no longer be uniformly arranged in the deformed configuration. Of course the method of finite spheres would still be applicable, but the solution times would be longer.

6.3 Broader implications

In this section, the broader implications of the method of finite spheres will be discussed, with the aim of widespread availability of an overlapping finite element method. In particular, having reviewed the advantages and disadvantages of mesh-based methods and meshless methods, it is apparent that a coupled formulation would overcome the disadvantages by taking advantage of their respective benefits. Additionally, since the sphere element stiffness calculation is computationally expensive, distributed memory parallel processing is especially valuable because the sphere element stiffness matrices are independent and their calculations can be allocated to different processors, thus enabling greater computational efficiency.

6.3.1 Finite element and finite sphere coupled formulation

The finite element method is a powerful engineering analysis and design tool that is effective and reliable for the analysis of increasingly complex structures and phenomena. While the method is considered to be reliable and efficient, mesh generation remains an expensive process which may lead to distorted elements and subsequently poor quality

finite element solutions, particularly for complicated three-dimensional geometries. Meshless methods have been developed to provide efficient means of discretizing complicated geometries, but result in longer solution times due to expensive numerical integration of complicated nonpolynomial functions. A coupled formulation can take advantage of the strengths of both the finite element method and meshless methods. While other meshless methods have been coupled with the finite element method, the method of finite spheres is a particularly attractive meshless method for this purpose because it is based on a similar formulation as the finite element method, and the coupled formulation has been shown to be reliable and effective for two-dimensional problems [38,41,42].

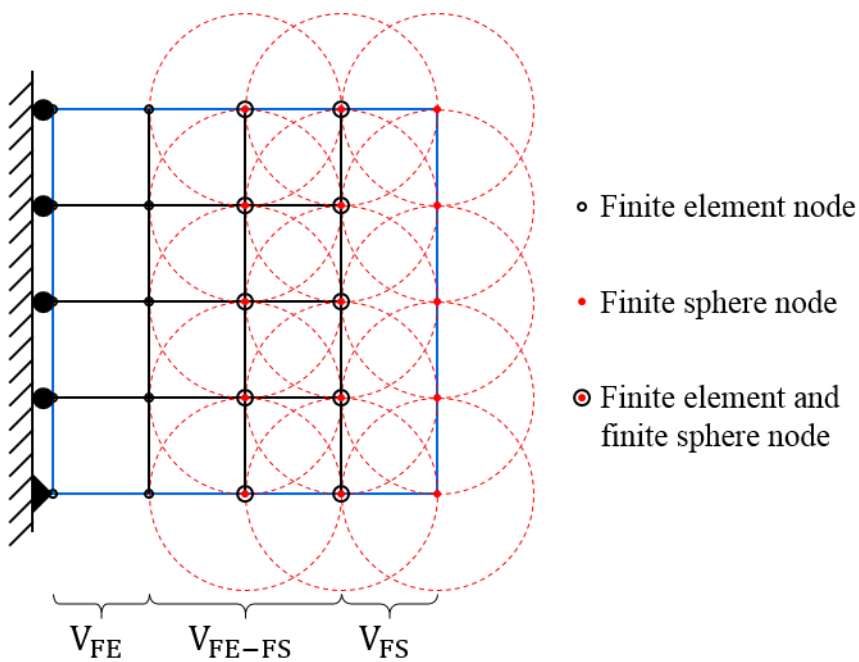


Figure 6.1 Coupled finite element and finite sphere domain

In a coupled finite element and finite sphere formulation, as shown in Figure 6.1, the finite elements are employed for regions of the domain which are easily discretized and do not require remeshing, while the finite spheres are utilized for regions of the domain which are complicated to discretize and may require remeshing. The idea is that the finite

spheres can be considered another element type within a standard finite element analysis, essentially an overlapping finite element, which can be employed in regions that are difficult to mesh or that may undergo large strain effects which typically would result in distorted elements when using standard non-overlapping finite elements. A coupled finite element and finite sphere formulation has been previously presented which overcomes the main difficulty in any coupled formulation, of enforcing continuity of the displacement, strain, and stress fields [38].

Another coupling scheme that was introduced, and that is particularly useful for certain applications such as crack propagation, is enriching the finite element interpolations with the local approximation space of finite spheres. Advantages of this approach is the ability to adaptively insert and remove spheres throughout an analysis, and to locally enrich the approximation space in areas of the domain that require specific solutions, such as discontinuities or singularities.

6.3.2 Distributed memory parallel processing

High-performance computing has made tremendous advances in terms of the computational speed available for numerical simulations. One area that has greatly enabled the analysis of more complex finite element models is parallel computing. In particular, distributed memory parallel processing has become increasingly popular over shared memory parallel processing. Instead of separate processors sharing data on the same memory space, with distributed memory parallel processing, processors have independent memory space. This allows the finite element model to be decomposed into different domains which are solved on different cores, ultimately allowing more computations to be

performed in parallel leading to faster solution times as well as more efficient memory usage.

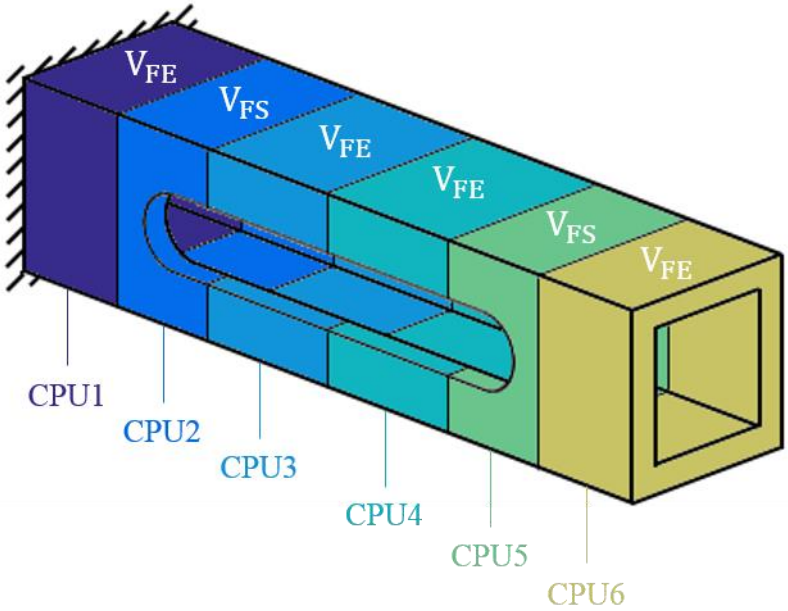


Figure 6.2 Distributed memory parallel processing for a coupled finite element and finite sphere domain

For the method of finite spheres, because the major computational cost is in the calculation of the element matrices as opposed to the solution of the equations, distributed memory parallel processing is even more advantageous. Since the element stiffness matrix and element load vector calculations can be performed independently, it can be used effectively for the method of finite spheres by allocating spheres to multiple cores to be calculated simultaneously. This can be further applied to the coupled FEM-MFS formulation, by using multiple cores for both the finite element and finite sphere regions, respectively, as shown in Figure 6.2, for example. It is important to note that, of course, reproducible and consistent results are achieved irrespective of the number of cores used. Ultimately, distributed memory parallel processing holds potential to solve very complex

problems more efficiently and to greatly reduce the overall computational cost for the method of finite spheres.

6.4 Conclusion

The objective of this thesis is to demonstrate the reliability and effectiveness of the method of finite spheres for the solution of practical three-dimensional linear elasticity problems. This goal is motivated by the challenges of mesh-based numerical methods, such as mesh generation for complex three-dimensional domains, severe element distortion in nonlinear analysis with large strain effects, and complications in modeling problems involving discontinuities and singularities which require mesh alignment and refinement. Advantages of the method of finite spheres as compared to other meshless methods include the fact that it is truly meshless and does not require a background mesh for numerical integration, that the formulation is based on the symmetric weak form resulting in a symmetric system of equations, and that it uses an extrinsic basis which means there is no matrix inversion or minimum overlap criterion in the construction of the interpolation functions.

In the three-dimensional formulation of the method of finite spheres, the domain is discretized using a set of nodal points and the approximation functions are compactly supported on spherical domains centered at these nodes. Similar to the finite element method, a weighted residual scheme is used to obtain the system of linear algebraic equations. Numerical integration has been identified as a focal point of development for the method of finite spheres and has been addressed by introducing the piecewise Gauss-Legendre quadrature rule for three-dimensional integration domains. The method of finite spheres has been implemented in a commercial finite element program to solve practical

three-dimensional linear elasticity problems. The results are compared with the finite element method to ensure reliability and to assess computational efficiency by determining the time multiplier for roughly the same level of accuracy based on strain energy comparisons.

The primary conclusion of the work in this thesis is that the method of finite spheres is reliable for three-dimensional linear elasticity problems, and based on the numerical examples studied, the method of finite spheres is approximately an order of magnitude slower than the standard finite element method. This result is very promising for the fact that there are substantial advantages that the method of finite spheres provides in terms of discretization of geometrically complex domains. Ultimately, we hope that with continuous research efforts in areas such as three-dimensional coupled FEM-MFS formulation, distributed memory parallel processing, as well as fundamental considerations such as different overlapping domains and numerical integration improvements, the method of finite spheres will be more widely used with eventual commercial viability as an effective overlapping finite element method.

Bibliography

[1] Bathe KJ. Finite element procedures. Prentice Hall; 1996.

[2] Lee NS, Bathe KJ. Effects of element distortions on the performance of isoparametric elements. *Int J Numer Meth Eng* 1993;36:3553-76.

[3] Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Comput Meth Appl M* 1996;139:3-47.

[4] Liu GR. Meshfree methods: moving beyond the finite element method. CRC Press; 2003.

[5] Monaghan JJ. An introduction to SPH. *Comput Phys Commun* 1988;48:89-96.

[6] Liu MB, Liu GR. Smoothed particle hydrodynamics (SPH): an overview and recent developments. *Arch Comput Methods Eng* 2010;17:25-76.

[7] Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. *Comput Mech* 1992;10:307-18.

[8] Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. *Int J Numer Meth Eng* 1994;37:229-56.

[9] Lu YY, Belytschko T, Gu L. A new implementation of the element free Galerkin method. *Comput Method Appl M* 1994;113:397-414.

[10] Atluri SN, Zhu T. A new meshless local Petrov-Galerkin approach in computational mechanics. *Comput Mech* 1998;22:117-27.

[11] Atluri SN, Kim HG, Cho JY. A critical assessment of the truly meshless local Petrov-Galerkin (MLPG), and local boundary integral equation (LBIE) methods. *Comput Mech* 1999;24:348-72.

[12] De S, Bathe KJ. The method of finite spheres. *Comput Mech* 2000;25:329-45.

[13] De S, Bathe KJ. Towards an efficient meshless computational technique: the method of finite spheres. *Eng Computation* 2001;18:170-92.

[14] De S, Bathe KJ. Displacement/pressure mixed interpolation in the method of finite spheres. *Int J Numer Meth Eng* 2001;51:275-92.

[15] Macri M, De S, Shephard MS. Hierarchical tree-based discretization for the method of finite spheres. *Comput Struct* 2003;81:789-803.

[16] Macri M, De S. Towards an automatic discretization scheme for the method of finite spheres and its coupling with the finite element method. *Comput Struct* 2005;83:1429-47.

[17] Melenk JM, Babuška I. The partition of unity finite element method: basic theory and applications. *Comput Method Appl M* 1996;139:289-314.

[18] Babuška I, Melenk JM. The partition of unity method. *Int J Numer Meth Eng* 1997;40:727-58.

[19] Duarte CA, Oden JT. H-p clouds – an h-p meshless method. *Numer Meth Part D E* 1996;12:673-705.

[20] Duarte CA, Oden JT. An h-p adaptive method using clouds. *Comput Method Appl M* 1996;139:237-62.

[21] Ham S, Lai B, Bathe KJ. The method of finite spheres for wave propagation problems. *Comput Struct* 2014;142:1-14.

[22] Krongauz Y, Belytschko T. Enforcement of essential boundary conditions in meshless approximations using finite elements. *Comput Method Appl M* 1996;131:133-45.

[23] Zhu T, Atluri SN. A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free Galerkin method. *Comput Mech* 1998;21:211-22.

[24] Wagner GJ, Liu WK. Application of essential boundary conditions in mesh-free methods: a corrected collocation method. *Int J Numer Meth Eng* 2000;47:1367-79.

[25] De S, Bathe KJ. The method of finite spheres with improved numerical integration. *Comput Struct* 2001;79:2183-96.

[26] Dolbow J, Belytschko T. Numerical integration of the Galerkin weak form in meshfree methods. *Comput Mech* 1999;23:219-30.

[27] Carpinteri A, Ferro G, Ventura G. The partition of unity quadrature in meshless methods. *Int J Numer Meth Eng* 2002;54:987-1006.

[28] Mazzia A, Ferronato M, Pini G. A comparison of numerical integration rules for the meshless local Petrov-Galerkin method. *Numer Algorithms* 2007;45:61-74.

[29] Babuška I, Banerjee U, Osborn JE, Li Q. Quadrature for meshless methods. *Int J Numer Meth Eng* 2008;76:1434-70.

[30] Babuška I, Banerjee U, Osborn JE, Zhang Q. Effect of numerical integration on meshless methods. *Comput Mech Appl M* 2009;198:2886-97.

[31] Beissel S, Belytschko T. Nodal integration of the element-free Galerkin method. *Comput Method Appl M* 1996;139:49-74.

[32] Chen Y, Lee JD, Eskandarian A. Meshless methods in solid mechanics. Springer; 2006.

[33] Belytschko T, Krysl P, Krongauz Y. A three-dimensional explicit element-free Galerkin method. *Int J Numer Meth Fl* 1997;24:1253-70.

[34] Barry W, Saigal S. A three-dimensional element-free Galerkin elastic and elastoplastic formulation. *Int J Numer Meth Eng* 1999;46:671-93.

[35] Atluri SN, Cho JY, Kim HG. Analysis of thin beams, using the meshless local Petrov-Galerkin method, with generalized moving least square interpolations. *Comput Mech* 1999;24:334-47.

[36] Han ZD, Atluri SN. Meshless local Petrov-Galerkin (MLPG) approaches for solving 3D problems in elasto-statics. *CMES-Comp Model Eng* 2004;6:169-88.

[37] Bucalem ML, Bathe KJ. The mechanics of solids and structures – hierarchical modeling and the finite element solution. Springer; 2011.

[38] Hong JW, Bathe KJ. Coupling and enrichment schemes for finite element and finite sphere discretizations. *Comput Struct* 2005;83:1386-95.

[39] Macri M, De S. Enrichment of the method of finite spheres using geometry-independent localized scalable bubbles. *Int J Numer Meth Eng* 2007;69:1-32.

[40] Banihani S, De S. Development of a genetic algorithm-based lookup table approach for efficient numerical integration in the method of finite spheres with application to the solution of thin beam and plate problems. *Int J Numer Meth Eng* 2006;67:1700-29.

[41] Belytschko T, Organ D, Krongauz Y. A coupled finite element-element-free Galerkin method. *Comput Mech* 1995;17:186-95.

[42] Liu GR, Gu YT. Meshless local Petrov-Galerkin (MLPG) method in combination with finite element and boundary element approaches. *Comput Mech* 2000;26:536-46.