The Enriched Subspace Iteration Method and Wave Propagation Dynamics with Overlapping Finite Elements

by

Ki-Tae Kim

B.S., Pusan National University (2009)
M.S., Korea Advanced Institute of Science and Technology (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

June 2018

© Massachusetts Institute of Technology 2018. All rights reserved.

Signature redacted

Author ....................

Department of Mechanical Engineering May 4, 2018

Signature redacted

Certified by ............

Klaus-Jürgen Bathe

Professor of Mechanical Engineering Thesis Supervisor

Signature redacted

Accepted by ............

Rohan Abeyaratne

Chairman, Department Committee on Graduate Theses
The Enriched Subspace Iteration Method and Wave Propagation Dynamics with Overlapping Finite Elements

by

Ki-Tae Kim

Submitted to the Department of Mechanical Engineering on May 4, 2018, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Mechanical Engineering

Abstract

In structural dynamic problems, the mode superposition method is the most widely used solution approach. The largest computational effort (about 90% of the total solution time) in the mode superposition method is spent on calculating the required eigenpairs and it is of critical importance to develop effective eigensolvers. We present in this thesis a novel solution scheme for the generalized eigenvalue problem. The scheme is an extension of the Bathe subspace iteration method and a significant reduction in computational time is achieved.

For the solution of wave propagation problems, the finite element method with direct time integration has been extensively employed. However, using the traditional finite element solution approach, accurate solutions can only be obtained of rather simple one-dimensional wave propagation problems. In this thesis, we investigate the solution characteristics of a solution scheme using 'overlapping finite elements', disks and novel elements, enriched with harmonic functions and the Bathe implicit time integration method to solve transient wave propagation problems. The proposed solution scheme shows two important properties: monotonic convergence of calculated solutions with decreasing time step size and a solution accuracy almost independent of the direction of wave travel through uniform, or distorted, meshes. These properties make the scheme promising to solve general wave propagation problems in complex geometries involving multiple waves.

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

I would like to express my sincere gratitude to my thesis supervisor, Professor Klaus-Jürgen Bathe, for his support, guidance, encouragement and trust throughout my doctoral research. His excellent insight, enthusiasm and infinite patience have been crucial for me to complete this work.

I would also like to thank my committee members, Professor Rohan Abeyaratne and Professor Nicholas Patrikalakis, for their valuable comments and encouragements throughout my studies.

I am deeply thankful to Professor Phill-Seung Lee who was my advisor during my master’s degree studies. He always encouraged and helped me to pursue my passion and to build my future career path. Also, I would like to express my gratitude to Professor Sun Hong Kwon and Professor K. C. Park for their invaluable advices.

I cannot imagine going through the long years of PhD study without a lot of incredible friends I have met at MIT. I would like to thank my colleagues, Soyoung You, Lingbo Zhang, Jun-Bin Huang and Bryce Wilkins, in the Finite Element Research group at MIT for their help and friendship. As another part of my life at MIT, special thanks should be given to my great friends: Kyung-Seok Han, Taekeun Kim, Hyundo Lee, Hyunryul Ryu, Keunwhan Park, Sung Kwan Lee and Hyunho Kim. I also appreciate help and friendship of Korea Graduate Student Association at Mechanical Engineering at MIT (sorry, I cannot list them all here).

Last, I am deeply grateful to my lovely family for their endless love and understanding. Without their unconditional support, I could not finish this work. My special thanks go to my mother for her unlimited encouragements.
# Contents

1 Introduction ................................................. 21

2 The enriched subspace iteration method ....................... 25
   2.1 The subspace iteration enriched by turning vectors ......... 26
      2.1.1 The algorithm ........................................ 27
      2.1.2 Additional aspects .................................... 34
   2.2 A simplified convergence analysis ........................ 35
   2.3 Illustrative example solutions ............................. 38
      2.3.1 Solution of beam problem ............................. 44
      2.3.2 Solution of wall problem .............................. 44
      2.3.3 Solution of ring problem ............................ 47
   2.4 Concluding remarks ....................................... 51

3 Wave propagation dynamics with the method of finite spheres 53
   3.1 Spatial approximation scheme ............................... 55
   3.2 Dispersion analysis ....................................... 56
      3.2.1 Spatial discretization error .......................... 57
      3.2.2 Additional effects introduced by the Bathe time integration .... 63
   3.3 Numerical examples ....................................... 67
      3.3.1 One-dimensional impact of an elastic bar .............. 68
      3.3.2 Two-dimensional scalar wave propagation .............. 68
      3.3.3 Two-dimensional elastic wave propagation ............ 71
   3.4 Concluding remarks ....................................... 77
4 Overlapping finite elements enriched for wave propagation problems 79
4.1 Spatial approximation scheme 81
4.2 Dispersion analysis 82
4.2.1 Spatial discretization error 82
4.2.2 Additional effects introduced by the Bathe time integration 86
4.3 Numerical examples 89
4.3.1 One-dimensional wave propagation 90
4.3.2 Two-dimensional scalar wave propagation 93
4.3.3 Two-dimensional elastic wave propagation 95
4.3.4 Two-dimensional wave propagation in a pre-stressed membrane with circular holes 100
4.3.5 Two-dimensional scalar wave propagation in a pre-stressed bi-material membrane 105
4.3.6 Two-dimensional elastic wave propagation in a semi-infinite medium with a crack 108
4.3.7 Two-dimensional elastic wave propagation in an anisotropic medium 109
4.3.8 Solution times 123
4.4 Concluding remarks 123

5 Conclusions 127

A Calculation of the amount of vector turning and the turning vectors 129
List of Figures

1-1 A simple illustration of the use of turning vectors in the Bathe subspace iteration method; a single eigenvector $\phi_1$ is sought, the red arrow; the blue arrow is the iteration vector and the gray arrow is the turning vector. ......................................................... 23

1-2 Displacement distributions of a 2D scalar wave propagation problem when using the method of finite spheres enriched for wave propagation problems with the Bathe time integration method; $\text{CFL} = c\Delta t/h$ where $c$ is the wave propagation velocity, $\Delta t$ is the time step size and $h$ is the distance between spheres. ......................................................... 24

2-1 Geometrical illustration of the subspaces $E_k$ and $E_{k+1}$ when $M = I$ and only two iteration vectors are considered: (a) in the basic method and (b) in the enriched method; the red (blue) vectors span the red (blue) plane. ......................................................... 33

2-2 Relative errors measured in the solution of the simple example problem when using the basic method and the enriched method: (a) first, (b) second and (c) third eigenvalues; the dashed lines are the theoretical convergence rates. ......................................................... 39

2-3 Clamped-clamped beam problem. ......................................................... 40

2-4 Wall problem. ................................................................. 40

2-5 Ring problem. ......................................................... 41
2-6 CPU time for calculating the smallest p eigenvalues of the beam problem when using the basic method and the enriched method; meshes used are (a) B-MESH1 \((n = 534,357\) and \(m = 243\)), (b) B-MESH2 \((n = 1,016,037\) and \(m = 363\)) and (c) B-MESH3 \((n = 1,520,493\) and \(m = 507\)).

2-7 CPU time for calculating the smallest p eigenvalues of the wall problem when using the basic method and the enriched method; meshes used are (a) W-MESH1 \((n = 107,484\) and \(m = 2067\)), (b) W-MESH2 \((n = 204,984\) and \(m = 2847\)) and (c) W-MESH3 \((n = 291,798\) and \(m = 3393\)).

2-8 CPU time for calculating the smallest p eigenvalues of the ring problem when using the basic method and the enriched method; meshes used are (a) R-MESH1 \((n = 216,000\) and \(m = 675\)), (b) R-MESH2 \((n = 389,880\) and \(m = 1083\)) and (c) R-MESH3 \((n = 603,060\) and \(m = 1587\)).

3-1 Uniformly distributed nodes used in the method of finite spheres: (a) one-dimensional case and (b) two-dimensional case.

3-2 Dispersion properties of the method of finite spheres in one-dimensional case: MFS-LIN, MFS-TRI1, MFS-TRI2 and MFS-TRI3 discretizations.

3-3 Wavenumber spectrums and stopping bands of MFS-TRI1 and MFS-TRI2 discretizations.

3-4 Propagation angle of a sinusoidal plane wave.

3-5 Dispersion properties of the method of finite spheres in two-dimensional case: (a) MFS-LIN, (b) MFS-TRI1, (c) MFS-TRI2 and (d) MFS-TRI3 discretizations.

3-6 Dispersion (left) and dissipation (right) properties of MFS-TRI3 scheme in one-dimensional case.

3-7 Dispersion (left) and dissipation (right) properties of MFS-TRI3 scheme in two-dimensional case.

3-8 1D impact of an elastic bar problem: problem description and spatial discretization by finite spheres.
3-9 Velocity distributions of the bar at \( t = 1 \) s when decreasing the CFL number: using (a) MFS-LIN, (b) MFS-TRI1, (c) MFS-TRI2 and (d) MFS-TRI3 schemes, with 51 spheres. ................................. 69

3-10 2D scalar wave propagation in a pre-stressed membrane: problem description and spatial discretization by finite sphere. ................................. 70

3-11 Displacement distributions of the membrane along horizontal axis at \( t = 0.8 \) s when decreasing the CFL number: using (a) MFS-LIN scheme with \( 31 \times 31 \) spheres, (b) MFS-TRI1 scheme with \( 31 \times 31 \) spheres, (c) MFS-TRI2 scheme with \( 16 \times 16 \) spheres and (d) MFS-TRI3 scheme with \( 11 \times 11 \) spheres. ................................. 71

3-12 Displacement distributions of the membrane along various directions \((\theta = 0^\circ, 22.5^\circ, 45^\circ)\) at \( t = 0.8 \) s: when using (a) MFS-TRI1 scheme with \( 31 \times 31 \) spheres, (b) MFS-TRI2 scheme with \( 16 \times 16 \) spheres and (c) MFS-TRI3 scheme with \( 11 \times 11 \) spheres; CFL \( \approx 0.016 \). ................................. 72

3-13 Snapshots of displacement distributions of the membrane at \( t = 0.8 \) s when decreasing the CFL number: using (a) MFS-LIN scheme with \( 31 \times 31 \) spheres, (b) MFS-TRI1 scheme with \( 31 \times 31 \) spheres, (c) MFS-TRI2 scheme with \( 16 \times 16 \) spheres and (d) MFS-TRI3 scheme with \( 11 \times 11 \) spheres. ................................. 73

3-14 Displacement distributions of the membrane at \( t = 0.9 \) s when using MFS-TRI3 scheme with \( 55 \times 55 \) spheres: (a) displacement distributions along horizontal axis with decreasing the CFL number, (b) displacement distributions along various directions \((\theta = 0^\circ, 22.5^\circ, 45^\circ)\) when CFL \( \approx 0.016 \) and (c) snapshot of displacement distributions when CFL \( \approx 0.016 \). ................................. 74

3-15 The Lamb's problem [1] - 2D wave propagation in a plane strain elastic half-space: problem description and spatial discretization by finite spheres. ................................. 75
3-16 Horizontal and vertical displacements of the elastic medium when using MFS-TRI3 scheme with $36 \times 36$ spheres: (a) displacement histories at $\mathbf{x} = (640, 0)$ m with decreasing the CFL number, (b) displacement histories at $\mathbf{x} = (1280, 0)$ m with decreasing the CFL number and (c) snapshots of displacement distributions at $t = 1$ s when CFL $\approx 0.016$.

4-1 Uniform mesh and propagation angle of a sinusoidal plane wave.

4-2 Dispersion properties of (a) OFE-TRI1 and (b) FE-TRI1 discretizations for various propagation angles.

4-3 Uniform and distorted meshes used for the simple problem for which the exact solution is $u = \sin(2\pi x); \ h = 1/N$ where $N$ is the number of elements along each side.

4-4 The $L^2$ relative error norms of the solutions to the simple problem when using FE-TRI1 and OFE-TRI1 discretizations.

4-5 Dispersion (left) and dissipation (right) properties of OFE-TRI1 scheme for various propagation angles: when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.

4-6 Dispersion (left) and dissipation (right) properties of FE-TRI1 scheme for various propagation angles: when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.

4-7 1D impact of an elastic bar problem: problem description and mesh used.

4-8 Velocity distributions of the bar at $t = 0.001$ s calculated using FE scheme with the uniform and non-uniform meshes when CFL = 1; $h$ denotes the element size used for the CFL number.

4-9 Velocity distributions of the bar at $t = 0.001$ s calculated using (a) FE-TRI2 scheme and (b) OFE-TRI2 scheme with the uniform and non-uniform meshes; $h$ denotes the element size used for the CFL number.

4-10 Problem description of 2D scalar wave propagation in a pre-stressed membrane.
4-11 Structured and unstructured meshes used for the 2D scalar wave propagation problem; $N$ is the number of elements along each side.

4-12 Contour plots of displacement distributions of the membrane at $t = 0.95$ s calculated using the FE-TRI1 scheme (left) and the OFE-TRI1 scheme (right), both with the structured mesh ($h = 0.03125$ m): when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.

4-13 Displacement distributions of the membrane along various directions ($\theta = 0^\circ, 22.5^\circ, 45^\circ$) at $t = 0.95$ s calculated using FE-TRI1 scheme (left) and OFE-TRI1 scheme (right), both with the structured mesh ($h = 0.03125$ m): when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.

4-14 Displacement distributions of the membrane at $t = 0.95$ s calculated using OFE-TRI1 with the unstructured mesh ($h = 0.03125$ m): (a) displacement distributions along horizontal axis with decreasing the CFL number and (b) displacement distributions along various directions when CFL = 0.125.

4-15 Displacement distributions of the membrane at $t = 0.9$ s calculated using OFE-TRI1 scheme with the structured mesh (left) and with the unstructured mesh (right); in both cases $h = 0.015625$ m: (a) displacement distributions along horizontal axis with decreasing the CFL number, (b) displacement distributions along various directions when CFL = 0.125 and (c) contour plots of displacement distributions when CFL = 0.125.

4-16 Problem description of the Lamb’s problem [1] - 2D wave propagation in a plane strain elastic half-space.

4-17 Horizontal displacements (left) and vertical displacements (right) of the elastic medium calculated using OFE-TRI1 scheme with the structured mesh ($h = 50$ m): (a) displacement histories at $x = (640, 0)$ m with decreasing CFL number and (b) displacement histories at $x = (1280, 0)$ m with decreasing CFL number.
4-18 Horizontal displacements (left) and vertical displacements (right) of the elastic medium calculated using OFE-TRI1 scheme with the unstructured mesh \( h = 50 \text{ m} \): (a) displacement histories at \( x = (640, 0) \) m with decreasing CFL number and (b) displacement histories at \( x = (1280, 0) \) m with decreasing CFL number.

4-19 Snapshots of von Mises stress distributions of the elastic medium at \( t = 1 \text{ s} \) calculated using OFE-TRI1 with the structured mesh (top) and with the unstructured mesh (bottom); in both cases \( h = 50 \text{ m} \) and CFL = 0.125. Brighter color represents larger stress.

4-20 2D scalar wave propagation in a pre-stressed membrane: problem description and the mesh used; the line colored in red is the direction of \( \theta = 22.5^\circ \) and the line colored in green is the direction of \( \theta = 67.5^\circ \).

4-21 Displacement distributions of the membrane with circular holes at \( t = 1 \text{ s} \) calculated using OFE-TRI1 scheme: (a) displacement distributions along the direction of \( \theta = 22.5^\circ \) with decreasing the CFL number and (b) displacement distributions along the two directions \( (\theta = 22.5^\circ \text{ and } \theta = 67.5^\circ) \) when CFL = 0.125.

4-22 Snapshots of displacement distributions of the membrane with circular holes at various observation times calculated using OFE-TRI1 scheme; CFL = 0.125.

4-23 Problem description of 2D scalar wave propagation in a pre-stressed bi-material membrane; \( c_1 = 1 \text{ m/s} \) and \( c_2 = 4 \text{ m/s} \).

4-24 Structured and unstructured meshes used for the pre-stressed bi-material membrane problem; \( N_1 \) and \( N_2 \) are the number of elements along each side of the inner square and the outer square, respectively.

4-25 Displacement distributions of the bi-material membrane along the direction of \( \theta = 0^\circ \) calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) \( t = 0.5 \text{ s} \), (b) \( t = 0.65 \text{ s} \) and (c) \( t = 0.8 \text{ s} \).
4-26 Displacement distributions of the bi-material membrane along the direction of $\theta = 22.5^\circ$ calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.

4-27 Displacement distributions of the bi-material membrane along the direction of $\theta = 45^\circ$ calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.

4-28 Contour plots of displacement distributions of the bi-material membrane calculated using OFE-TRI1 scheme ($\text{CFL} = 0.125$) with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.

4-29 Time history of displacements of the elastic medium with a horizontal crack at $x = (640, 0)$ m calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right): (a) horizontal displacement and (b) vertical displacement.

4-30 Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.3$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $\text{CFL} = 0.125$. Brighter color represents larger stress.

4-31 Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.5$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $\text{CFL} = 0.125$. Brighter color represents larger stress.

4-32 Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.7$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $\text{CFL} = 0.125$. Brighter color represents larger stress.
4-33 Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at \( t = 0.9 \) s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases CFL = 0.125. Brighter color represents larger stress.

4-34 Computational domain for the anisotropy elastic wave propagation problem and the positions of two receivers.

4-35 Time history of horizontal displacements (left) and vertical displacements (right) of the anisotropy elastic medium calculated using OFE-TRI1 scheme (CFL = 0.125) with the structured mesh and the unstructured mesh: (a) at the receiver 1 and (b) at the receiver 2.

4-36 Snapshots of horizontal displacement distributions of the anisotropy elastic medium at various observation times calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right); in both cases CFL = 0.125.

4-37 Snapshots of vertical displacement distributions of the anisotropy elastic medium at various observation times calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right); in both cases CFL = 0.125.
## List of Tables

2.1 Comparison of operation counts (number of multiplications); the results of the basic method are taken from References [2, 3]; for the enriched method, we also consider the algorithm summarized in Appendix A; here $m$ denotes the mean half-bandwidth of $\mathbf{K}$ and $\mathbf{M}$. ... 32

2.2 Finite element meshes used to solve the beam problem. ............. 41

2.3 Finite element meshes used to solve the wall problem. ............. 41

2.4 Finite element meshes used to solve the ring problem. ............. 42

2.5 CPU time for calculating the smallest $p = 100$ eigenvalues of the beam problem when using the associated eigenvalue problem; mesh used is B-MESH1 ($n = 534, 357$ and $m = 243$); if the associated eigenvalue problem is not used, 15 iterations are performed and 1540 s are used. 42

2.6 CPU time for calculating the smallest $p = 100$ eigenvalues of the wall problem when using the associated eigenvalue problem; mesh used is W-MESH1 ($n = 107,484$ and $m = 2067$); if the associated eigenvalue problem is not used, 12 iterations are performed and 1282 s are used. 42

2.7 CPU time for calculating the smallest $p = 100$ eigenvalues of the ring problem when using the associated eigenvalue problem; mesh used is R-MESH1 ($n = 216,000$ and $m = 675$); if the associated eigenvalue problem is not used, 11 iterations are performed and 977 s are used. 43

2.8 Speed-up of the enriched method when compared to the basic method for the beam problem. .......................... 44
2.9 CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the beam problem when using the basic method and the enriched method; mesh used is B-MESH3 ($n = 1,520,493$ and $m = 507$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

2.10 Speed-up of the enriched method when compared to the basic method for the wall problem.

2.11 CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the wall problem when using the basic method and the enriched method; mesh used is W-MESH3 ($n = 291,798$ and $m = 3393$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

2.12 Speed-up of the enriched method when compared to the basic method for the ring problem.

2.13 CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the ring problem when using the basic method and the enriched method; mesh used is R-MESH3 ($n = 603,060$ and $m = 1587$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

3.1 Sets of local basis functions used in the method of finite spheres; $\bar{x} = (x - x_I)/r_I$ and $\bar{y} = (y - y_I)/r_I$ where $x_I$ and $y_I$ are the horizontal and vertical center positions of sphere $I$, respectively; $p = 1, 2, 3$. 

4.1 CPU times required to solve the two-dimensional example problems when using OFE-TRI1 scheme with CFL = 0.125... 124
Chapter 1

Introduction

Dynamic analysis falls into structural dynamics and wave propagation dynamics depending on the number of modes being excited by the externally applied load. In structural dynamic problems, the lowest modes of a physical system are excited and an analyst is particularly interested in the steady-state response of the system. On the other hand, in wave propagation dynamic problems, a larger number of modes participates in the response and transient solutions are important.

In structural dynamic problems, since long-term effects need to be considered and only the first few modes are actually excited, the mode superposition method is more effective than direct time integration methods [3]. Here, most of the computational effort is expended in solving the eigenvalue problem, taking up about 90% of total solution time. Hence, it is very valuable to develop a very efficient and robust solution scheme for the eigenvalue problem.

In wave propagation dynamic problems, accurate solutions are difficult to achieve because much higher modes need to be accurately represented by a numerical model. The finite element method with direct time integration has been widely employed, but the scheme still suffers from the dispersion and dissipation errors caused by the spatial and temporal discretizations. Reasonably accurate solutions can be obtained only when using a uniform mesh and optimal time step size. If very accurate solutions are sought, the traditional finite element method with direct time integration can, in fact, only be applied to rather simple problems, like one-dimensional problems with
a single wave traveling through the domain.

We present in this thesis novel schemes to address the difficulties mentioned above. For structural dynamic analyses, we focus on developing a significant improvement of eigensolver. For the solution of wave propagation dynamic problems, a considerable part of this thesis is devoted to investigating and developing very effective overlapping finite elements.

The material given in this thesis has largely been published already in References [4] [5] [6], and we use much material directly from these papers.

In chapter 2, a novel extension of the Bathe subspace iteration method is presented for the solution of the generalized eigenvalue problem in structural dynamics. The key idea in accelerating the iteration is to enrich the subspace by using some turning vectors to get faster to the required eigenvectors, see e.g. Figure 1-1. A considerable improvement in computational efficiency is achieved and various example solutions are given to show the effectiveness of the method.

In chapter 3, we study the use of the method of finite spheres, an overlapping finite element method, for wave propagation problems. In the previous work [7], using the method of finite spheres enriched for wave propagation problems with the Bethe time integration method, we observed important properties that have been difficult to achieve in the numerical solution of wave propagation problems: a monotonic convergence of the calculated solutions with decreasing time step size and a solution accuracy almost independent of the direction of wave propagation through the mesh, see e.g. Figure 1-2. We give a mathematical study and numerical solutions of some wave propagation problems to demonstrate these attributes.

In chapter 4, we present novel overlapping finite elements to solve wave propagation problems. The scheme also shows the two important properties mentioned above and can be efficiently used with irregular meshes. A dispersion analysis is given and various example problems are solved to illustrate the performance of the solution scheme.

Finally we conclude with some remarks and possible future works.
Figure 1-1: A simple illustration of the use of turning vectors in the Bathe subspace iteration method; a single eigenvector $\phi_1$ is sought, the red arrow; the blue arrow is the iteration vector and the gray arrow is the turning vector.
Figure 1-2: Displacement distributions of a 2D scalar wave propagation problem when using the method of finite spheres enriched for wave propagation problems with the Bathe time integration method; $\text{CFL} = \frac{c\Delta t}{h}$ where $c$ is the wave propagation velocity, $\Delta t$ is the time step size and $h$ is the distance between spheres.
Chapter 2

The enriched subspace iteration method

An essential step in structural dynamics is to evaluate some frequencies and mode shapes of the structure considered, and in today’s analyses very large finite element systems are solved. Hence extensive efforts have been made to establish effective eigensolution techniques. The Bathe subspace iteration method [2,3,8,9] is a successful scheme and has been abundantly employed in research and industry because of its robustness and effectiveness.

An attractive ingredient of the Bathe subspace iteration method is that it is particularly amenable to parallel implementations. As discussed in Reference [10], by partitioning the iteration vectors, most computations in the method can be programmed in shared and distributed memory processing and a linear increase in solution time with the number of eigenpairs sought is achieved.

Another widely used method is the Lanczos method [3,11]. Although the method initially suffered from numerical instabilities resulting from the loss of orthogonality, techniques have been proposed to overcome this deficiency and at present variations of the Lanczos method enjoy much success, see, for example, References [12–14]. Using this method, the computational cost increases almost linearly proportional to the number of eigenpairs sought, which makes the method also efficient when many eigenpairs are needed. However, since the algorithmic steps in the solution are
sequential, the technique is not directly suitable to parallel processing.

Since the Bathe subspace iteration method is widely used and particularly suited to parallel processing, it is of great interest to speed up the iterations. This speed-up should be achieved even without parallel processing and should be present in particular when seeking many eigenpairs (typically more than 100 pairs) of very large eigenvalue problems. In fact, single processor solutions provide a good test for basic increases in efficiency, and any significant decrease in solution times is very valuable.

In this chapter, we present a novel algorithm to accelerate the Bathe subspace iteration method. In the basic method, the subspace iteration vectors turn in each iteration a certain amount towards the required subspace vectors [3, 10]. The fundamental idea for accelerating the iterations is to use the direction of turning of the subspace in the iterations. This is achieved by establishing new iteration vectors denoted as “forward turning vectors” to replace iteration vectors that are much less useful. This enrichment of the iterations is a simple addition to the basic Bathe subspace iteration method, and yields a considerable reduction in computational cost.

In the following sections, we first describe the algorithm, referred to as the “enriched subspace iteration method”. Then we provide a simplified convergence analysis to give some insight into how the new vectors accelerate the iterations. Finally, we give the results of some example solutions to demonstrate the performance of the scheme.

2.1 The subspace iteration enriched by turning vectors

We consider the generalized symmetric eigenvalue problem

\[ \mathbf{K}\phi = \lambda\mathbf{M}\phi \]  \hspace{1cm} (2.1)

where \( \mathbf{K} \) and \( \mathbf{M} \) are the stiffness matrix and the consistent mass matrix of a finite element system with \( n \) degrees of freedom. The matrices \( \mathbf{K} \) and \( \mathbf{M} \) are assumed (with-
out loss of generality [3]) to be positive definite. We seek the smallest \( p \) eigenvalues and corresponding eigenvectors \((\lambda_i, \phi_i); i = 1, \ldots, p\), with the ordering

\[
0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_p,
\]

which satisfy

\[
K \phi_i = \lambda_i M \phi_i
\]

and

\[
\phi_i^T M \phi_j = \delta_{ij},
\]

\[
\phi_i^T K \phi_j = \lambda_i \delta_{ij},
\]

where \( \delta_{ij} \) is the Kronecker delta.

In the following we describe the algorithm used to solve for these eigenvalues and eigenvectors.

2.1.1 The algorithm

We start the enriched subspace iteration method by using first a simple scheme given in References [3, 10] to construct \( q \) linearly independent vectors in \( MX_0 \), with \( q > p \), in which the diagonal entries of \( M \), unit vectors that excite the degrees of freedom of the maximum values \( m_{ii}/k_{ii} \) and a random vector are used. Usually \( q = \max\{2p, p + 8\} \).

Using the linear independent vectors in \( MX_0 \), we perform a single iteration using the basic subspace iteration method to obtain the \( M \)-orthonormal starting iteration vectors in \( X_1 \), which span the \( q \)-dimensional subspace \( E_1 \), and makes it possible to directly use the enrichment algorithm.

- The 10 steps in each iteration

For each iteration \( k = 1, 2, \ldots \), the following steps 1–10 are performed until convergence is established to the \( p \) required eigenvalues. Steps 1–5 include the enrichment by establishing and using the ‘forward turning vectors’ and steps 6–10 are as in the basic subspace iteration method.
1. For $k = 1, 2, \ldots$, partition the iteration vectors which span $E_k$

$$X_k = [\Phi_k, X^a_k, X^b_k]$$  \hspace{1cm} (2.6)

where $\Phi_k$ stores the $p_k$ vectors which have converged to the required tolerance in the previous iterations, with $p_1 = 0$, and the rest of the iteration vectors are equally partitioned into $X^a_k$ and $X^b_k$, which are both of order $n \times r_k$. Hence we use $r_k = (q - p_k)/2$.

2. Evaluate $X^a_{k+1}$

$$KX^a_{k+1} = MX^a_k.$$  \hspace{1cm} (2.7)

3. Construct $Y_k$ of order $n \times r_k$ using the following steps in part (a) and thereafter part (b):

(a) Calculate in reverse order the amount of turning of the iteration vectors in $X^a_{k+1}$ and choose the vectors for which the measure used is larger than the tolerance $tolt$, i.e., for $i = r_k, r_k - 1, \ldots, 2, 1$, with $t_k = 0$, evaluate in this part (a):

$$\hat{x}_i = \bar{x}^{a(k+1)}_i - X_k \left(X_k^T M \bar{x}^{a(k+1)}_i \right) - \sum_{j=1}^{t_k} u_j \left(u_j^T M \bar{x}^{a(k+1)}_i \right)$$  \hspace{1cm} (2.8)

where $\bar{x}^{a(k+1)}_i$ is the $i$th column vector in $X^a_{k+1}$ and the last term is only included if $t_k \geq 1$. Then evaluate

$$\alpha_i = \frac{\hat{x}_i^T M \hat{x}_i}{\left(\bar{x}^{a(k+1)}_i \right)^T M \bar{x}^{a(k+1)}_i}.$$  \hspace{1cm} (2.9)

If $\alpha_i \leq tolt$ go to the next value of $i$. If $\alpha_i > tolt$ we proceed as follows

$$t_k = t_k + 1,$$  \hspace{1cm} (2.10)

$$u_{i_k} = \frac{\hat{x}_i}{\sqrt{\hat{x}_i^T M \hat{x}_i}}.$$  \hspace{1cm} (2.11)
\[ v_{tk} = x_i^{a(k+1)} \]  

(2.12)

and now go to the next value of \( i \).

(b) Next, let \( t_k \) be the last value reached in the above loop, part (a). We now construct \( Y_k = [x_1^{b(k)}, \ldots, x_{r_k-t_k}^{b(k)}, v_1, \ldots, v_{t_k}] \) where \( x_i^{b(k)}, i = 1, \ldots, r_k-t_k \), are the first \( r_k-t_k \) column vectors in \( X_k^b \). We denote the \( i \)th column vector in \( Y_k \) by \( y_i^{(k)} \). Calculate for \( i = 1, \ldots, t_k \):

\[
\tilde{x}_i = v_i - X_k^a \left( (X_k^a)^T M v_i \right) - \Phi_k \left( (\Phi_k^T M v_i) - \sum_{j=1}^{r_k-t_k+i-1} y_j^{(k)} \left( (y_j^{(k)})^T M v_i \right) \right),
\]

(2.13)

\[
y_{r_k-t_k+i}^{(k)} = \frac{\tilde{x}_i}{\sqrt{\tilde{x}_i^T M \tilde{x}_i}}.
\]

(2.14)

Here, we call \( \tilde{x}_i, i = 1, \ldots, t_k \), and their normalizations by Equation (2.14) the 'turning vectors'.

4. Evaluate \( Y_{k+1} \) from

\[ K Y_{k+1} = M Y_k \]  

(2.15)

where we now have in \( Y_{k+1} \) the 'forward turning vectors' which are the key ingredient to obtain a faster convergence.

5. Construct \( X_{k+1} \) where the column vectors span \( E_{k+1} \)

\[ X_{k+1} = [\Phi_k, X_{k+1}^a, Y_{k+1}]. \]  

(2.16)

6. Project the matrices \( K \) and \( M \) onto the subspace \( E_{k+1} \)

\[ K_{k+1} = X_{k+1}^T K X_{k+1}, \]  

(2.17)

\[ M_{k+1} = X_{k+1}^T M X_{k+1}. \]  

(2.18)
7. Solve for the eigensystem of the projected matrices

\[ K_{k+1}Q_{k+1} = K_{k+1}Q_{k+1}A_{k+1}. \]  

(2.19)

8. Calculate an improved approximation to the eigenvectors

\[ X_{k+1} = \overline{X}_{k+1}Q_{k+1}. \]

(2.20)

9. Measure which of the eigenvalue approximations \( \lambda_i^{(k+1)} \) have converged [3], that is, satisfy

\[
\left[ 1 - \frac{\left( \lambda_i^{(k+1)} \right)^2}{\left( q_i^{rr(k+1)} \right)^T q_i^{rr(k+1)}} \right]^{1/2} \leq \text{tolc}; \quad i = p_k + 1, \ldots, p \]

(2.21)

where \( q_i^{rr(k+1)} \) is the \((i-p_k)\)th column vector in \( Q_{k+1}^{rr} \) with \( Q_{k+1}^{rr} \) being a \( 2r_k \times 2r_k \) submatrix obtained by partitioning \( Q_{k+1} \) into

\[ Q_{k+1} = \begin{bmatrix} Q_{k+1}^{cc} & Q_{k+1}^{cr} \\ Q_{k+1}^{rc} & Q_{k+1}^{rr} \end{bmatrix}. \]

(2.22)

where the submatrices \( Q_{k+1}^{cc}, Q_{k+1}^{cr} \) and \( Q_{k+1}^{rc} \) are of order \( p_k \times p_k, p_k \times 2r_k \) and \( 2r_k \times p_k \), respectively.

10. Update the number of converged iteration vectors to \( p_{k+1} \) and increase \( k \) if \( p_{k+1} < p \).

- End of 10 steps in each iteration

In the above iteration, it is effective to order the iteration vectors such that the corresponding approximate eigenvalues increase in magnitude. Then, provided that the starting iteration vectors in \( X_1 \) are not \( M \)-orthogonal to one of the eigenvectors
sought we have, for \( i = 1, \ldots, p, \)

\[
\lambda_i^{(k+1)} \to \lambda_i \quad \text{and} \quad x_i^{(k+1)} \to \phi_i \quad \text{as} \ k \to \infty
\]  

(2.23)

where \( x_i^{(k+1)} \) is the \( i \)th column vector in \( X_{k+1} \), see Section 2.2.

Note that in step 3(a) we consider the vectors \( \tilde{x}_i, i = r_k, r_k - 1, \ldots, 2, 1 \), which are \( M \)-orthogonal to all previous iteration vectors in \( X_k \) and choose only the corresponding vectors \( x_i^{(k+1)} \) that are providing a stable solution. With the ordering of the iteration vectors given above, the amount of turning is negligible for the first few iteration vectors, and near convergence very small. Hence we consider the vectors \( x_i^{(k+1)} \) in reverse order in Equation (2.8). In step 3(b) then, the vectors \( \tilde{x}_i, i = 1, \ldots, t_k \), are \( M \)-orthogonalized to the vectors in \( \Phi_k \) and \( X^c_k \) and the first \( r_k - t_k \) column vectors in \( X^v_k \).

In the actual implementation, we do not evaluate Equation (2.8), and instead proceed to calculate the \( \alpha_i \) values using the efficient algorithm given Appendix A. Also, as shown in the appendix some results obtained in step 3 can be reused in Equations (2.17) and (2.18).

The turning vectors in \( Y_k \) are \( M \)-orthonormal and improved by solving Equation (2.15). The entries in \( Y_{k+1} \) corresponding to the turning vectors are the 'forward turning vectors'. For example, consider a subspace \( E_k \) spanned by \( \{x_1^{(k)}, x_2^{(k)}\} \). We also assume that the turning vector of \( x_1^{(k)} \) is used, so that \( y_1^{(k)} = \tilde{x}_i / \sqrt{\tilde{x}_i^TM\tilde{x}_i} \). Since \( y_1^{(k)} \) can be expressed as a linear combination of \( x_1^{(k)} \) and \( x_1^{(k+1)} \), \( \bar{y}_1^{(k+1)} \) is equal to a linear combination of \( x_1^{(k+1)} \) and \( \bar{x}_1^{(k+1)} \) where \( \bar{x}_1^{(k+1)} = K^{-1}(M\tilde{x}_1^{(k+1)}) \), and thus \( \bar{y}_1^{(k+1)} \) can be interpreted as a forward turning vector. Figure 2-1 shows geometrically the subspace \( E_{k+1} \) used in the basic subspace iteration method and that used in the enriched subspace iteration method, when \( M = I \).

Table 2.1 summarizes the complete procedure of the enriched subspace iteration method and gives the number of operations required with a comparison to the basic subspace iteration method. Here, one operation means one multiplication which almost always is followed by one addition.
Table 2.1: Comparison of operation counts (number of multiplications); the results of the basic method are taken from References [2, 3]; for the enriched method, we also consider the algorithm summarized in Appendix A; here $m$ denotes the mean half-bandwidth of $K$ and $M$.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Calculation</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic subspace iteration method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factorization of $K$</td>
<td>$K = LDL^T$</td>
<td>$\frac{1}{2}mn^2 + \frac{3}{2}nm$</td>
</tr>
<tr>
<td>Iteration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_{k+1} = X_{k+1}^T R_k$</td>
<td>$\frac{1}{2}nq(q+1)$</td>
<td></td>
</tr>
<tr>
<td>$R_{k+1} = MX_{k+1}$</td>
<td>$nq(2m+1)$</td>
<td></td>
</tr>
<tr>
<td>$M_{k+1} = X_{k+1}^T R_k$</td>
<td>$\frac{1}{2}nq(q+1)$</td>
<td></td>
</tr>
<tr>
<td>$K_{k+1}Q_{k+1} = M_{k+1}Q_{k+1}A_{k+1}$</td>
<td>$O(q^3)$</td>
<td></td>
</tr>
<tr>
<td>$R_{k+1} = R_{k+1}Q_{k+1}$</td>
<td>$nq^2$</td>
<td></td>
</tr>
<tr>
<td>Total in single iteration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\bar{K} = K - \rho M$</td>
<td>$nq(4m + 2q + 3) + \mathcal{O}(q^3)$</td>
<td></td>
</tr>
<tr>
<td>$\bar{K} = LDL^T$</td>
<td>$\frac{1}{2}mn^2 + \frac{3}{2}nm$</td>
<td></td>
</tr>
<tr>
<td>Sturm sequence check</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enriched subspace iteration method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factorization of $K$</td>
<td>$K = LDL^T$</td>
<td>$\frac{1}{2}mn^2 + \frac{3}{2}nm$</td>
</tr>
<tr>
<td>Iteration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$R_k = \left[ \begin{array}{ccc} \Phi_k &amp; R_k &amp; R_k^T \end{array} \right]$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_{k+1} = H_k$</td>
<td>$nr_k(2m + 1)$</td>
<td></td>
</tr>
<tr>
<td>$[D_{k+1}^T, A_{k+1}^T, B_{k+1}^T] = (X_{k+1}^T)^T R_k$</td>
<td>$nr_k(p_k + \frac{4}{3}n_k + r_k)$</td>
<td></td>
</tr>
<tr>
<td>$R_{k+1} = MX_{k+1}$</td>
<td>$nr_k(2m + 1)$</td>
<td></td>
</tr>
<tr>
<td>$C_{k+1} = (X_{k+1}^T)^T R_{k+1}$</td>
<td>$\frac{1}{2}nr_k(r_k + 1)$</td>
<td></td>
</tr>
<tr>
<td>Check the amount of turning (see step (a) in Appendix A)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate $S_k := MY_k$ (see step (b) in Appendix A)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_{k+1} = S_k$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_{k+1} = \left[ \begin{array}{ccc} A_k &amp; 0 &amp; \nu_k^* \end{array} \right]$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$R_{k+1} = MY_{k+1}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_{k+1} = \left[ \begin{array}{ccc} \tilde{D}<em>{k+1} &amp; \tilde{C}</em>{k+1} \ \nu_k^* &amp; \nu_k^* &amp; \nu_k^* \end{array} \right]$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_{k+1}Q_{k+1} = M_{k+1}Q_{k+1}A_{k+1}$</td>
<td>$O(q^3)$</td>
<td></td>
</tr>
<tr>
<td>$R_{k+1} = \left[ \begin{array}{ccc} \Phi_k &amp; R_k &amp; R_k^T \end{array} \right]$</td>
<td>$nq^2$</td>
<td></td>
</tr>
<tr>
<td>Total in single iteration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\bar{K} = K - \rho M$</td>
<td>$2nr_k(4m + q + 3) + nq^2 + nr_k(q + r_k + 1) + \mathcal{O}(q^3)$</td>
<td></td>
</tr>
<tr>
<td>$\bar{K} = LDL^T$</td>
<td>$n(m + 1)$</td>
<td></td>
</tr>
<tr>
<td>Sturm sequence check</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 2-1: Geometrical illustration of the subspaces $E_k$ and $E_{k+1}$ when $M = I$ and only two iteration vectors are considered: (a) in the basic method and (b) in the enriched method; the red (blue) vectors span the red (blue) plane.

An important aspect of enriching the subspace by forward turning vectors is that for the vector for which the turning is used, a single iteration in the enriched subspace iteration method has the effect of two iterations in the basic subspace iteration method. Furthermore, the computational effort related to the enrichment is not expensive. In a single iteration the number of operations (neglecting the $O(q^3)$ operations) of the enriched method is $2nr_k(4m+q+3)+nq^2+nt_k(q+r_k+1)$ and the basic method is $nq(4m+2q+3)$ where $m$ is the half-bandwidth of $K$ and $M$, see Table. Of course, when $r_k = q/2$ and $t_k = 0$, the computational effort in both methods is the same. As an example when the enrichment is used, consider in three-dimensional analysis, $m = q$ and $r_k = t_k = q/2$ (although in practice $r_k$ actually decreases as the iterations are performed). The ratio of the number of operations per iteration is then
about 1.125, giving a 12.5% increase in the computational effort per iteration using the enrichment scheme. In many practical large scale analyses the increase in the cost is probably smaller because the half-bandwidth of the system is large, and indeed the increase may be practically negligible. Hence, considering not only the acceleration of the iteration by the use of turning vectors but also the inexpensive computational cost, the enrichment leads to a significant speed-up in solving the eigenvalue problem.

It is also important to note that the use of the turning vectors preserves the important characteristic of the basic subspace iteration method that the method is amenable to parallel processing, as discussed in Reference [10]. Indeed, the computations in steps 2, 4, 6 and 8, in which most of the computational effort in the iterations is expended, can be parallelized as presented in Reference [10]. The operations in step 3 can be also partly parallelized. Only some operations in the last summation of Equation (2.13) are not amenable to parallel processing because in the loop the vector \( v_i \) is used with the vectors \( y_j^{(k)}, j = r_k - t_k + 1, \ldots, r_k - t_k + i - 1 \), which were previously calculated in the same loop. However, the computational effort for this sequential algorithm is relatively small compared to the other computations which can be performed in parallel processing.

Several different ways could be developed to exploit the idea of using turning vectors, like dividing the iteration vectors in step 1 such that the number of vectors in \( X_k^p \) is equal to \( p - p_k \). However, we focus in this study on the simple dividing scheme described above.

2.1.2 Additional aspects

In the enriched subspace iteration method, the reduction in solution time is achieved by, firstly, the enrichment using forward turning vectors, which is most significant, and, secondly, by not performing the iteration for the converged iteration vectors. The second aspect is not important when only few eigenpairs are required, as it was the case in the early use of the method.

Considering that in the method the multiplication of the consistent mass matrix \( M \) by the iteration vectors might be computationally expensive and that the mode
shapes are in many cases not very different from those of the same discretization using alumped mass matrix, a further reduction in computations may be achieved by performing first a few iterations using a diagonal mass matrix. We construct this diagonal mass matrix using the diagonal elements of the consistent mass matrix $M$ and scaling these elements so that all mass of the finite elements is applied. We call the eigenproblem using this lumped mass matrix the “associated eigenvalue problem” to the generalized eigenvalue problem in Equation (2.1).

The solution procedure is then as follows:

(i) Establish the $q$ starting iteration vectors, $q > p$, as discussed in Section 2.1.1.

(ii) Perform the enriched subspace iterations to solve the “associated eigenvalue problem” until a stopping criterion is satisfied (this step might not be included, see example solutions).

(iii) Using the solutions obtained from step (ii) continue the enriched subspace iterations to solve the generalized eigenvalue problem (Equation (2.1)).

(iv) Carry out the Sturm sequence check to verify that the required eigenvalues and eigenvectors have indeed been calculated [3,8].

Simple options are possible for the stopping criterion in step (ii): for example, we may stop when the error bounds for all the eigenvalues sought are below the tolerance $tols$ or we may stop when the number of iterations reaches a prescribed number of iterations. We experimented with both options, see Section 2.3.

2.2 A simplified convergence analysis

In the previous section we discussed the new scheme and merely stated that if the subspace iteration converges, the required eigenpairs have been obtained. Here we examine in a simple way the iteration properties to obtain insight into how the forward turning vectors accelerate the convergence of the iterations.
The first (conceptual) step is to consider the iterations in the basis of eigenvectors $[3, 15, 16]$

$$X_k = \Phi Z_k$$  \hspace{1cm} (2.24)

where $\Phi = [\phi_1, \ldots, \phi_n]$ and since $X_k^T M X_k = I$, it follows that $Z_k^T Z_k = I$. Introducing this change of basis into Equation (2.7) leads to

$$\Lambda Z_{k+1}^a = Z_k^a$$  \hspace{1cm} (2.25)

where $Z_k^a$ of order $n \times r_k$ corresponds to the first $r_k$ vectors in $Z_k$. We assume here, for simplicity, that $q = 2p, p_k = 0$, hence $r_k = p$, and that all the turning vectors are employed for use of corresponding forward turning vectors. The subspace $E_{k+1}$ is then the subspace spanned by $\{Z_{k+1}^a, \overline{Z}_{k+1}^a\}$ with $\overline{Z}_{k+1}^a = \Lambda^{-1} Z_{k+1}^a$.

For the analysis, let us consider a matrix $\Xi_k$ defined by $[3, 16]$

$$\Xi_k = \begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1 \\
\xi_{q+1,1}^{(k)} & \xi_{q+1,2}^{(k)} & \cdots & \xi_{q+1,q}^{(k)} \\
\xi_{q+2,1}^{(k)} & \xi_{q+2,2}^{(k)} & \cdots & \xi_{q+2,q}^{(k)} \\
\vdots & \vdots & \ddots & \vdots \\
\xi_{n,1}^{(k)} & \xi_{n,2}^{(k)} & \cdots & \xi_{n,q}^{(k)}
\end{bmatrix}$$ \hspace{1cm} (2.26)

Assuming that the iteration vectors in $Z_k$ are not deficient in the vectors $\hat{e}_i, i = 1, \ldots, q$, which are the eigenvectors corresponding to the lowest $q$ eigenvalues of $\Lambda$, the vectors $z_i^{a(k)}, i = 1, \ldots, p$, in $Z_k^a$ can be expressed as

$$z_i^{a(k)} = \gamma_i^{(k)} \xi_i^{(k)} + \delta_i^{(k)} r_k^{(k)}$$ \hspace{1cm} (2.27)

with the $i$th column vector $\xi_i^{(k)}$ in $\Xi_k$, some coefficients $\gamma_i^{(k)}$ and $\delta_i^{(k)}$, and a residual vector $r_k^{(k)}$. We also note that since the vectors in $\Xi_k$ in the basic subspace iteration
with \( p \) iteration vectors converge to the smallest \( p \) eigenvectors sought [16], we have that, for \( i = 1, \ldots, p \),

\[
\xi^{(k)}_i \to \hat{e}_i, \quad \gamma^{(k)}_i \to 1 \quad \text{and} \quad \delta^{(k)}_i \to 0 \quad \text{as} \quad k \to \infty. \tag{2.28}
\]

Let us now assume that \( \delta^{(k)}_i \) is small, equal to \( \epsilon^{(k)}_i \ll 1 \). The corresponding vectors \( \overline{z}^{(k+1)}_i, i = 1, \ldots, p \), in \( \overline{Z}_{k+1}^n \) are then

\[
\overline{z}^{(k+1)}_i = \gamma^{(k)}_i \Lambda^{-2} \xi^{(k)}_i + \epsilon^{(k)}_i \Lambda^{-2} r^{(k)}_i \tag{2.29}
\]

and we have

\[
\frac{\|\lambda^2 \overline{z}^{(k+1)}_i - \hat{e}_i\|_2}{\|\overline{z}^{(k)}_i - \hat{e}_i\|_2} = \frac{\|\lambda^2 \xi^{(k)}_i + O(\epsilon^{(k)}_i) \hat{e}_i\|_2}{\|\xi^{(k)}_i + O(\epsilon^{(k)}_i) \hat{e}_i\|_2} = \frac{\|\lambda^2 \xi^{(k)}_i - \hat{e}_i\|_2}{\|\xi^{(k)}_i - \hat{e}_i\|_2} + O(\epsilon^{(k)}_i)
\]

\[
= \left( \sum_{j=q+1}^{n} \left( \frac{\lambda_i}{\lambda_j} \right)^4 \left( \frac{\xi^{(k)}_j}{\xi^{(k)}_i} \right)^2 \right)^{1/2} + O(\epsilon^{(k)}_i)
\]

\[
\leq \left( \frac{\lambda_i}{\lambda_{q+1}} \right)^2 + O(\epsilon^{(k)}_i) \tag{2.30}
\]

where we used that \( \gamma^{(k)}_i = 1 + O(\epsilon^{(k)}_i) \).

Since \( z_i^{(k+1)} \) is the best approximation to the eigenvector \( \hat{e}_i \) in the subspace \( E_{k+1} \), we finally obtain

\[
\frac{\|z_i^{(k+1)} - \hat{e}_i\|_2}{\|z_i^{(k)} - \hat{e}_i\|_2} \leq \left( \frac{\lambda_i}{\lambda_{q+1}} \right)^2 + O(\epsilon^{(k)}_i); \quad i = 1, \ldots, p. \tag{2.31}
\]

We can therefore conclude that using \( q = 2p \) and \( p_k = 0 \), provided the iteration vectors in \( X_k \) are ordered appropriately, are not deficient in the eigenvectors, and in each iteration all turning vectors are used, the iteration vectors \( x_i^{(k)}, i = 1, \ldots, p \), converge with the rates \( (\lambda_i/\lambda_{q+1})^2 \). The rates of convergence of the corresponding eigenvalues are \( (\lambda_i/\lambda_{q+1})^4 \), \( i = 1, \ldots, p \), because these are calculated from the Rayleigh
quotient.

We illustrate the convergence of the new scheme in a simple example using the diagonal matrix \( K = \text{diag}(1, 2, \ldots, 12) \) and \( M = I \), and seeking the smallest \( p = 3 \) eigenvalues. For the solution, we set \( q = 2p \) and \( p_k = 0 \), and all turning vectors are used in the iterations. We also compare the results with those from the basic subspace iteration method with the number of iteration vectors \( q = 2p \). In both cases, six random vectors are used as the starting vectors.

Figure 2-2 shows the relative errors of the calculated eigenvalue approximations, denoted as \( e_r = (\lambda_i^{(k)} - \lambda_i)/\lambda_i \) when using the subspace iteration method with and without the enrichment. In the figure, the dashed lines are the theoretical rates of convergence. We see that the computed rates (slopes in Figure 2-2) follow quite closely the theoretically predicted rates and the rate of convergence from the solution using the enriched subspace iteration method is about two times the rate of convergence of the basic subspace iteration method.

### 2.3 Illustrative example solutions

The objective in this section is to demonstrate the performance of the enrichment scheme through some example solutions. The observed computational cost of the enriched method is compared to the observed cost of the basic method, using each time \( q = 2p \) iteration vectors. In both cases, the computational expense compared is the CPU time required to calculate all \( p \) eigenvalues with \( tolc = 10^{-6} \) for convergence, with the time used for the factorization of the stiffness matrix \( K \) and the Sturm sequence check not included. In the enriched subspace iterations we use \( tol = 10^{-8} \), see Section 2.1.1.

The solutions are obtained using a laptop with a single core Intel 2.40 GHz CPU and 24 GB RAM. In each case, we use the simple scheme to establish the starting iteration vectors described in Section 2.1.1.

We consider three problems: a clamped-clamped beam structure with mass density \( \rho = 7800 \text{ kg/m}^3 \), Young's modulus \( E = 2.11 \times 10^{11} \text{ N/m}^2 \) and Poisson's ratio \( \nu = 0 \),
Figure 2-2: Relative errors measured in the solution of the simple example problem when using the basic method and the enriched method: (a) first, (b) second and (c) third eigenvalues; the dashed lines are the theoretical convergence rates.
a wall structure with $\rho = 3000 \text{ kg/m}^3$, $E = 7.0 \times 10^{10} \text{ N/m}^2$ and $\nu = 0.3$ and a ring structure with $\rho = 7800 \text{ kg/m}^3$, $E = 2.11 \times 10^{11} \text{ N/m}^2$ and $\nu = 0.3$, as shown in Figures 2-3, 2-4 and 2-5, respectively. The beam problem is taken from Reference [10]. The structures are modeled using three-dimensional 8-node brick elements, and in all
$\rho = 7800 \text{ kg/m}^3$

$E = 2.11 \times 10^{11} \text{ N/m}^2$

$\nu = 0.3$

Figure 2-5: Ring problem.

Table 2.2: Finite element meshes used to solve the beam problem.

<table>
<thead>
<tr>
<th>Name</th>
<th>Number of elements (cross section x length)</th>
<th>Degrees of freedom</th>
<th>Half-bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-MESH1</td>
<td>8 x 8 x 2200</td>
<td>534,357</td>
<td>243</td>
</tr>
<tr>
<td>B-MESH2</td>
<td>10 x 10 x 2800</td>
<td>1,016,037</td>
<td>363</td>
</tr>
<tr>
<td>B-MESH3</td>
<td>12 x 12 x 3000</td>
<td>1,520,493</td>
<td>507</td>
</tr>
</tbody>
</table>

Table 2.3: Finite element meshes used to solve the wall problem.

<table>
<thead>
<tr>
<th>Name</th>
<th>Number of elements (width x length x height)</th>
<th>Degrees of freedom</th>
<th>Half-bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>W-MESH1</td>
<td>12 x 52 x 52</td>
<td>107,484</td>
<td>2067</td>
</tr>
<tr>
<td>W-MESH2</td>
<td>12 x 72 x 72</td>
<td>204,984</td>
<td>2847</td>
</tr>
<tr>
<td>W-MESH3</td>
<td>12 x 86 x 86</td>
<td>291,798</td>
<td>3393</td>
</tr>
</tbody>
</table>

cases using the consistent mass matrix. We calculate the solutions with three different mesh sizes for each problem, see Tables 2.2, 2.3 and 2.4, to investigate whether the
Table 2.4: Finite element meshes used to solve the ring problem.

<table>
<thead>
<tr>
<th>Name</th>
<th>Number of elements (radius x axis x circumference)</th>
<th>Degrees of freedom</th>
<th>Half-bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-MESH1</td>
<td>14 x 14 x 320</td>
<td>216,000</td>
<td>675</td>
</tr>
<tr>
<td>R-MESH2</td>
<td>18 x 18 x 360</td>
<td>389,880</td>
<td>1083</td>
</tr>
<tr>
<td>R-MESH3</td>
<td>22 x 22 x 380</td>
<td>603,060</td>
<td>1587</td>
</tr>
</tbody>
</table>

Table 2.5: CPU time for calculating the smallest $p = 100$ eigenvalues of the beam problem when using the associated eigenvalue problem; mesh used is B-MESH1 ($n = 534,357$ and $m = 243$); if the associated eigenvalue problem is not used, 15 iterations are performed and 1540 s are used.

<table>
<thead>
<tr>
<th>$tols$</th>
<th>Number of iterations performed or allowed in the associated eigenvalue problem</th>
<th>Total number of iterations performed</th>
<th>Total CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>9</td>
<td>17</td>
<td>1583</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>10</td>
<td>18</td>
<td>1639</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>11</td>
<td>18</td>
<td>1630</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>12</td>
<td>19</td>
<td>1687</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>13</td>
<td>20</td>
<td>1735</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>15</td>
<td>22</td>
<td>1826</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>15</td>
<td>1542</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>15</td>
<td>1480</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>15</td>
<td>1471</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>16</td>
<td>1539</td>
</tr>
</tbody>
</table>

Table 2.6: CPU time for calculating the smallest $p = 100$ eigenvalues of the wall problem when using the associated eigenvalue problem; mesh used is W-MESH1 ($n = 107,484$ and $m = 2067$); if the associated eigenvalue problem is not used, 12 iterations are performed and 1282 s are used.

<table>
<thead>
<tr>
<th>$tols$</th>
<th>Number of iterations performed or allowed in the associated eigenvalue problem</th>
<th>Total number of iterations performed</th>
<th>Total CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>4</td>
<td>12</td>
<td>1140</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>5</td>
<td>13</td>
<td>1219</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>6</td>
<td>14</td>
<td>1272</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>7</td>
<td>15</td>
<td>1318</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>10</td>
<td>18</td>
<td>1452</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>13</td>
<td>21</td>
<td>1531</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>12</td>
<td>1156</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>12</td>
<td>1140</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>13</td>
<td>1219</td>
</tr>
<tr>
<td>6</td>
<td>14</td>
<td>14</td>
<td>1272</td>
</tr>
</tbody>
</table>

solution time increases linearly with the number of the required eigenpairs. Here, the half-bandwidth means the mean half-band width of the stiffness matrix $K$ after factorization [3].

To establish how many iterations might best be performed using the associated
Table 2.7: CPU time for calculating the smallest $p = 100$ eigenvalues of the ring problem when using the associated eigenvalue problem; mesh used is R-MESH1 ($n = 216,000$ and $m = 675$); if the associated eigenvalue problem is not used, 11 iterations are performed and 977 s are used.

<table>
<thead>
<tr>
<th>$tols$</th>
<th>Number of iterations performed or allowed in the associated eigenvalue problem</th>
<th>Total number of iterations performed</th>
<th>Total CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>4</td>
<td>12</td>
<td>944</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>6</td>
<td>13</td>
<td>992</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>7</td>
<td>14</td>
<td>1038</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>8</td>
<td>15</td>
<td>1080</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>9</td>
<td>16</td>
<td>1056</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>12</td>
<td>19</td>
<td>1213</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>12</td>
<td>956</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>12</td>
<td>944</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>12</td>
<td>949</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>13</td>
<td>1012</td>
</tr>
</tbody>
</table>

eigenvalue problem (using the associated lumped mass matrix) before switching to the solution using the consistent mass matrix, we first solve the beam, the wall and the ring problems using the finite element meshes B-MESH1, W-MESH1 and R-MESH1, respectively. Tables 2.5, 2.6 and 2.7 show the results obtained when $p = 100$. When solving the associated eigenvalue problem, we applied the two different stopping criteria mentioned in Section 2.1.2.

As shown in Table 2.5, for the beam problem, the most efficient solution is reached when the maximum number of iterations allowed in the associated eigenvalue problem is equal to 5. For the wall and ring problems, however, it is most efficient to use the tolerance stopping criterion with $tols = 10^{-1}$, see Tables 2.6 and 2.7. From these numerical experiments, we can conclude that it is probably reasonable to stop solving the associated eigenvalue problem when the error bounds for all required eigenvalue approximations are below $tols = 10^{-1}$ or a maximum of 5 iterations has been reached. This is also reasonable because then about onethird of the total number of iterations might be performed using the associated eigenvalue problem. We use this stopping criterion in the following example solutions. However, we also note that in each case the solution time saved by considering in the first iterations the associated eigenvalue problem is not large. Hence in practice, we may actually not first consider this problem but directly solve the generalized eigenvalue problem in all iterations.
Table 2.8: Speed-up of the enriched method when compared to the basic method for the beam problem.

<table>
<thead>
<tr>
<th>Numerical model</th>
<th>p</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-MESH1 ((n = 534,357 \text{ and } m = 243))</td>
<td>50</td>
<td>4.95</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.62</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>2.93</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>5.41</td>
</tr>
<tr>
<td>B-MESH2 ((n = 1,016,037 \text{ and } m = 363))</td>
<td>100</td>
<td>3.23</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>2.75</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>4.54</td>
</tr>
<tr>
<td>B-MESH3 ((n = 1,520,493 \text{ and } m = 507))</td>
<td>100</td>
<td>3.18</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>2.73</td>
</tr>
</tbody>
</table>

2.3.1 Solution of beam problem

Table 2.8 shows the speed-up obtained using the enriched subspace iteration method, defined as the CPU time used in the basic method divided by the CPU time used in the enriched method. The enriched method is about 3–5 times faster than the basic method, depending on the number of degrees of freedom used and the number of eigenpairs sought. For the case B-MESH3 with \(p = 100\), Table 2.9 gives the computational cost per iteration with the number of forward turning vectors actually used and the convergence history. We note that the average computational time used per iteration is higher in the basic method because the converged vectors are still included in the iterations.

Figure 2-6 gives the CPU times used in the basic method and the enriched method. We see that the CPU time increases almost linearly with the number of frequencies and mode shapes required and that the slopes of the curves are smaller for the enriched method.

2.3.2 Solution of wall problem

In this problem solution the finite element matrices have a large half-bandwidth. The speed-up is also significant, an improvement of a factor larger than 3 is obtained, see Table 2.10. For the case W-MESH3 with \(p = 100\), Table 2.11 gives the computational
Table 2.9: CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the beam problem when using the basic method and the enriched method; mesh used is B-MESH3 ($n = 1,520,493$ and $m = 507$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

<table>
<thead>
<tr>
<th>Basic subspace iteration method</th>
<th>Enriched subspace iteration method</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total number of iterations performed</strong></td>
<td><strong>Iteration number</strong></td>
</tr>
<tr>
<td>44</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>00</td>
</tr>
<tr>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>17</td>
</tr>
</tbody>
</table>
Figure 2-6: CPU time for calculating the smallest $p$ eigenvalues of the beam problem when using the basic method and the enriched method; meshes used are (a) B-MESH1 ($n = 534,357$ and $m = 243$), (b) B-MESH2 ($n = 1,016,037$ and $m = 363$) and (c) B-MESH3 ($n = 1,520,493$ and $m = 507$).
Table 2.10: Speed-up of the enriched method when compared to the basic method for the wall problem.

<table>
<thead>
<tr>
<th>Numerical model</th>
<th>p</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>W-MESH1 ((n = 107,484 \text{ and } m = 2067))</td>
<td>50</td>
<td>3.25</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.82</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.69</td>
</tr>
<tr>
<td>W-MESH2 ((n = 204,984 \text{ and } m = 2847))</td>
<td>50</td>
<td>3.15</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.45</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.53</td>
</tr>
<tr>
<td>W-MESH3 ((n = 291,798 \text{ and } m = 3393))</td>
<td>50</td>
<td>3.23</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.59</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.50</td>
</tr>
</tbody>
</table>

cost per iteration with the number of forward turning vectors actually used and the convergence history. We again note that the average computational time used per iteration is higher in the basic method.

Figure 2-7 shows the CPU times used in the basic method and the enriched method. As in the previous example solutions, we observe that regardless of \(n\), the computational cost increases almost in linear proportion to the required eigenpairs, and that the slopes of the curves are smaller for the enriched method.

### 2.3.3 Solution of ring problem

Table 2.12 shows the speed-up achieved by the enriched subspace iteration method, and we see that the speed-up factor is larger than 3. For the case R-MESH3 with \(p = 100\), Table 2.13 gives the CPU time required per iteration with the number of forward turning vectors actually used and the convergence history. We again observe that in the enriched method, the computational cost per iteration gradually decreases and the total number of iterations performed is smaller.

Here, we also see approximately a linear increase in the CPU time and the slopes of the curves are smaller in the enriched method, see Figure 2-8.
Table 2.11: CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the wall problem when using the basic method and the enriched method; mesh used is W-MESH3 ($n = 291,798$ and $m = 3393$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

<table>
<thead>
<tr>
<th>Iteration number</th>
<th>Basic subspace iteration method</th>
<th>Enriched subspace iteration method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total number of iterations performed</td>
<td>Average CPU time (s) in each iteration/Total CPU time (s)</td>
</tr>
<tr>
<td>0</td>
<td>35</td>
<td>593/20757</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>81</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>65</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>44</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>78</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>22</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>22</td>
</tr>
</tbody>
</table>
Figure 2-7: CPU time for calculating the smallest $p$ eigenvalues of the wall problem when using the basic method and the enriched method; meshes used are (a) W-MESH1 ($n = 107,484$ and $m = 2067$), (b) W-MESH2 ($n = 204,984$ and $m = 2847$) and (c) W-MESH3 ($n = 291,798$ and $m = 3393$).
Figure 2-8: CPU time for calculating the smallest \( p \) eigenvalues of the ring problem when using the basic method and the enriched method; meshes used are (a) R-MESH1 \((n = 216,000 \text{ and } m = 675)\), (b) R-MESH2 \((n = 389,880 \text{ and } m = 1083)\) and (c) R-MESH3 \((n = 603,060 \text{ and } m = 1587)\).
Table 2.12: Speed-up of the enriched method when compared to the basic method for the ring problem.

<table>
<thead>
<tr>
<th>Numerical model</th>
<th>p</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-MESH1 ((n = 216,000 \text{ and } m = 675))</td>
<td>50</td>
<td>3.37</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.26</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.44</td>
</tr>
<tr>
<td>R-MESH2 ((n = 389,880 \text{ and } m = 1083))</td>
<td>50</td>
<td>3.41</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.20</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.44</td>
</tr>
<tr>
<td>R-MESH3 ((n = 603,060 \text{ and } m = 1587))</td>
<td>50</td>
<td>3.17</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.01</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>3.07</td>
</tr>
</tbody>
</table>

2.4 Concluding remarks

The objective in this chapter was to present a novel effective scheme to accelerate the Bathe subspace iteration method, especially for obtaining many eigenpairs of very large eigenvalue problems. The basic subspace iteration method was developed when only a relatively small number of frequencies and mode shapes were used for dynamic analyses, however, the demands in analyses have increased very much. In today's finite element practice, one hundred or more frequencies and mode shapes may be used.

We showed that the use of forward turning vectors in the subspace iterations is inexpensive and significantly speeds up the convergence of the iteration vectors. We measured speed-ups of 3–5 in some example solutions using a single core laptop machine. This speed-up is clearly significant, but the idea of using the “turning of iteration vectors” in the Bathe subspace iteration method might be further explored to possibly reach an even greater efficiency.

While we focused in this chapter on the case of using the consistent mass matrix (because of its wide use), the improvements in the subspace iteration method can also directly be employed in the case of a lumped mass matrix, and we expect then too a significant speed-up.
Table 2.13: CPU time used in each iteration for calculating the smallest $p = 100$ eigenvalues of the ring problem when using the basic method and the enriched method; mesh used is R-MESH3 ($n = 603,060$ and $m = 1587$); 0 denotes establishing the starting iteration vectors and performing a single iteration in the basic method, as described in Section 2.1.1, for the associated eigenvalue problem; 00 denotes performing the M-orthonormalization of the iteration vectors.

<table>
<thead>
<tr>
<th>Basic subspace iteration method</th>
<th>Iteration number</th>
<th>Number of turning vectors used in iteration</th>
<th>Enriched subspace iteration method</th>
<th>Cumulative number of converged vectors after iteration</th>
<th>CPU time (s) used in iteration/iterations performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of iterations performed</td>
<td>Average CPU time (s) in each iteration</td>
<td>Total CPU time (s)</td>
<td>Iteration number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>596/17278</td>
<td></td>
<td>0</td>
<td>100</td>
<td>323/323</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>100</td>
<td>342/555</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
<td>80</td>
<td>338/1002</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>3</td>
<td>74</td>
<td>326/1329</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
<td>62</td>
<td>310/1639</td>
</tr>
<tr>
<td>00</td>
<td></td>
<td></td>
<td>5</td>
<td>79</td>
<td>305/1494</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>6</td>
<td>33</td>
<td>618/2563</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>7</td>
<td>33</td>
<td>595/3158</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td>8</td>
<td>16</td>
<td>547/3705</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td>9</td>
<td>14</td>
<td>500/4205</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td>10</td>
<td>13</td>
<td>461/4666</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td>11</td>
<td>13</td>
<td>407/5074</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td>12</td>
<td>21</td>
<td>348/5422</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td>20</td>
<td>300/5731</td>
</tr>
</tbody>
</table>
Chapter 3

Wave propagation dynamics with the method of finite spheres

In transient wave propagation problems, spatial and temporal discretizations introduce dispersion not present in analytical solutions, and hence this error is referred to as 'numerical dispersion'. The accuracy of numerical solutions can be bad due to this dispersion. When applying numerical methods to wave propagation problems, spatial and temporal discretizations should, therefore, be chosen according to the dispersion properties of the methods used and it is difficult to obtain accurate solutions to complex problems.

The finite element method with direct time integrations has been widely used to solve wave propagation problems. Considerable research efforts have been focused on the dispersion analysis of finite element solutions to the wave or Helmholtz equation [17–24]. The finite element discretization with the consistent mass matrix in general results in a faster phase velocity than the exact propagation velocity, while a lumped mass approximation leads to slower phase velocity [17–20,25]. In addition, the finite element solutions using uniform meshes show numerical anisotropy, i.e., the solution error depends on the direction considered although the exact wave propagation is the same in all directions [19–21,23–25]. The effect of temporal discretizations using various time integration methods on solutions has also been studied [17,18,20,24] with the error giving period elongations and amplitude decays [3,26].

53
Numerous methods have been proposed to reduce the dispersion error of finite element solutions [20, 25, 27–33]. However, in general two- and three-dimensional problems, most of these methods suffer from numerical anisotropy and/or are complicated to use. The higher-order finite element method such as the spectral element method [34] can reduce the dispersion error and the numerical anisotropy [35], but it can be difficult to solve two- and three-dimensional problems in practical analyses.

Another approach to improve the finite element solutions of wave propagation problems is based on trying to cancel out opposing effects. The finite element discretization with the consistent mass matrix gives an overestimated phase velocity while the use of the trapezoidal rule results in an underestimated phase velocity, and it is found that the combined effect leads to a decrease in dispersion error [18].

The Bathe method [36, 37], an implicit time integration method, was shown to be very effective when used with bi-linear finite elements because the dissipation property of the Bathe method attenuates undesired high frequency waves. At optimal spatial and temporal discretizations for single types of waves, the solutions are almost non-dispersive but show numerical anisotropy [24]. Since the solutions of practical problems contain multiple types of waves, e.g., longitudinal, transverse and surface waves at different wave speeds, the optimality for all wave predictions is lost, and all waves cannot be accurately calculated at the same time.

Recently, the method of finite spheres [38–41], a meshless method, was applied to transient wave propagation problems using the Bathe method, and it was found that the method of finite spheres enriched by trigonometric polynomials is very effective for such problems [7]. The computational time was similar to the time used for bi-linear finite element solutions. An interesting observation was that the solution accuracy using the method of finite spheres increases as the time step size decreases. This result is very important in practice because accurate solutions for multiple types of waves can thus be obtained. Another attractive finding is that, in uniform spatial discretizations, the numerical anisotropy was negligible in the method of finite spheres solutions. These findings in numerical solutions suggest that the method of finite spheres enriched by trigonometric polynomials may be more effective than the
traditional finite element method for transient wave propagation problems. However, these results were only observed in numerical examples and were not derived from a systematical dispersion analysis.

The objective of this chapter is to study the method of finite spheres used with the Bathe time integration method for the solution of transient wave propagation problems. We use the Bathe method for the temporal discretization since it has favorable dispersion and dissipation properties [26]. In the following, we first describe the approximation scheme of the method of finite spheres for transient wave propagation problems. We then theoretically derive the dispersion properties of the proposed scheme in one- and two-dimensional analyses, and apply the method to some wave propagation problems to illustrate the important results.

3.1 Spatial approximation scheme

In the method of finite spheres, the solution variable $u$ is approximated as, see References [38–41] for the detailed formulation,

$$ u(x) \approx u_h(x) = \sum_{I=1}^{N} \varphi_I(x) \sum_{m \in \mathcal{J}_I} p_m(x) a_{Im} $$

(3.1)

where $x$ is a position vector measured from the origin of the coordinate system (here, the Cartesian coordinates $x, y$ are used), $N$ is the number of spheres used for spatial discretization, $\mathcal{I}_I$ is an index set for the sphere $I$, and $\varphi_I$, $p_m$ and $a_{Im}$ are the partition of unity function, the local basis function and the corresponding coefficients, respectively, which are subordinate to the sphere $I$. The partition of unity function is constructed by the Shepard function [42] 

$$ \varphi_I(x) = \frac{W_I(x)}{\sum_{j=1}^{N} W_j(x)} $$

(3.2)
Table 3.1: Sets of local basis functions used in the method of finite spheres; $\bar{x} = (x - x_I)/r_I$ and $\bar{y} = (y - y_I)/r_I$ where $x_I$ and $y_I$ are the horizontal and vertical center positions of sphere $I$, respectively; $p = 1, 2, 3$.

<table>
<thead>
<tr>
<th>Discretization</th>
<th>Local basis functions in 1D</th>
<th>Local basis functions in 2D</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFS-LIN</td>
<td>${1, \bar{x}}$</td>
<td>${1, \bar{x}, \bar{y}, \bar{x}\bar{y}}$</td>
</tr>
<tr>
<td>MFS-TRIp</td>
<td>${1, \bar{x}, \cos(\pi \bar{x}), \sin(\pi \bar{x}), \ldots, \cos(p\pi \bar{x}), \sin(p\pi \bar{x})}$</td>
<td>${1, \bar{x}, \bar{y}, \bar{x}\bar{y}}$</td>
</tr>
</tbody>
</table>

where we use the quartic spline function

$$W_I(x) = W(s_I),$$

$$W(s_I) = \begin{cases} 
1 - 6s_I^2 + 8s_I^3 - 3s_I^4 & \text{for } 0 \leq s_I < 1, \\
0 & \text{for } s_I \geq 1 
\end{cases} \tag{3.3}$$

with $s_I = ||x - x_I||/r_I$ where $x_I$ and $r_I$ refer to the center position vector and the radius of sphere $I$, respectively.

The performance of the method of finite spheres, of course, depends on the local basis function used. In this wave propagation analysis, we study different sets of local basis functions, i.e., the method of finite spheres with the bi-linear polynomials (MFS-LIN) and the method of finite spheres with the bi-linear polynomials and the trigonometric polynomials up to order $p$ (MFS-TRIp, $p = 1, 2, 3$), as listed in Table 3.1.

### 3.2 Dispersion analysis

In this section, we investigate the dispersion properties of solutions to the standard wave equation when using the method of finite spheres with the Bathe time integration. We first consider the dispersion errors caused by spatial discretization, the
method of finite spheres, and analyze additional effects resulting from temporal discretization, the Bathe time integration. We consider one- and two-dimensional cases, but the same approach can be extended to the three-dimensional case.

The wave equation is defined by

$$\nabla^2 u - \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} = 0$$  \hspace{1cm} (3.4)

where \( t \) is time, \( \nabla^2 \) is the Laplace operator and \( c \) is the wave propagation velocity. In a fixed Cartesian coordinate system, a basic sinusoidal plane wave solution to Equation (3.4) is given by

$$u = Ae^{i(k\mathbf{n} \cdot \mathbf{x} - \omega t)}$$  \hspace{1cm} (3.5)

where \( A \) is the amplitude of the wave, \( k \) is the wavenumber, \( \mathbf{n} \) is a unit vector in the direction of wave propagation, \( \omega \) is the angular frequency and \( i = \sqrt{-1} \). If we substitute from Equation (3.5) into Equation (3.4), we obtain

$$c^2 = \left( \frac{\omega}{k} \right)^2$$  \hspace{1cm} (3.6)

and the analytical solution is non-dispersive.

### 3.2.1 Spatial discretization error

Here, we consider a time independent form of the wave equation (3.4), the Helmholtz equation given as

$$\nabla^2 u + k^2 u = 0$$  \hspace{1cm} (3.7)

with the corresponding plane wave solution

$$u = Ae^{ik\mathbf{n} \cdot \mathbf{x}}.$$  \hspace{1cm} (3.8)

The method of finite spheres discretization of a weak form of Equation (3.7) yields,
without considering boundary conditions,

\[ K\mathbf{a} - (kh)^2 \mathbf{M}\mathbf{a} = 0 \]  \hspace{1cm} (3.9)

where \( K, M \) and \( \mathbf{a} \) are the associated dimensionless stiffness and mass matrices and the vector of unknown coefficients, respectively, and \( h \) is the typical radius of sphere.

For the dispersion analysis, we uniformly distribute nodes spaced a distance \( h \) apart and use \( r_I = h \) for each sphere \( I \), as shown in Figure 3-1. Also, the same \( n_p \) local basis functions are used in each sphere \( I \). With this translational invariant
distribution of nodes, the coefficient vectors $a_I$ associated with nodes $I, I = 1, 2, \ldots$, which correspond to the numerical plane wave solution, can be assumed of the form

$$a_I = \hat{a}e^{ik_hn \cdot x_I}$$  \hspace{1cm} (3.10)

where $\hat{a}, k_h$ and $x_I$ are the amplitude vector of order $n_p$, the numerical wavenumber and the nodal position vector, respectively. Substituting from Equation (3.10) into Equation (3.9) gives, after canceling the common factor, the same equation for all nodes, which can be written as

$$(D_{\text{stiff}} - (k h)^2 D_{\text{mass}}) \hat{a} = 0$$  \hspace{1cm} (3.11)

where $D_{\text{stiff}}$ and $D_{\text{mass}}$ are the corresponding resultant matrices, which are both Hermitian $n_p \times n_p$ matrices. Now we have a condition for the existence of nontrivial solution to Equation (3.11). The solution to this equation provides a relation between the values of $k h$ and $k_h h$.

In one-dimensional case, the matrices $D_{\text{stiff}}$ and $D_{\text{mass}}$ are

$$D_{\text{stiff}} = K_{n,n-1}e^{-ik_hh} + K_{n,n} + K_{n,n+1}e^{ik_hh}$$  \hspace{1cm} (3.12)

and

$$D_{\text{mass}} = M_{n,n-1}e^{-ik_hh} + M_{n,n} + M_{n,n+1}e^{ik_hh}.$$  \hspace{1cm} (3.13)

In the above equations, the stiffness matrix $K_{n,n+k}$ and the mass matrix $M_{n,n+k}$ are sub-matrices in the total stiffness and mass matrices defined in Equation (3.9), respectively, and the subscript $n+k$ ($k = -1, 0, 1$) means the associated sphere $n+k$, see Figure 3-1.

Figure 3-2 shows the dispersion error induced by the spatial discretizations only, MFS-LIN and MFS-TRIp (here, $p = 1, 2, 3$), as a function of the normalized numerical wavenumber $k_h h / \pi$ ($= h / (\lambda_h/2)$ where $\lambda_h$ is the numerical wavelength). Two valuable properties of the method of finite spheres solutions are observed in Figures 3-2
Figure 3-2: Dispersion properties of the method of finite spheres in one-dimensional case: MFS-LIN, MFS-TRI1, MFS-TRI2 and MFS-TRI3 discretizations.

Figure 3-3: Wavenumber spectrums and stopping bands of MFS-TRI1 and MFS-TRI2 discretizations

and 3-3. First, there are stopping bands, jumps in the values of the normalized numerical phase velocity, plotted as dotted vertical lines in Figures 3-2 and 3-3. In these bands, the corresponding numerical wavenumber is a complex number, which means that the corresponding wave modes decay. For the MFS-TRIP discretizations, the magnitude of stopping bands in lower wavenumber regions decreases as the degree
of trigonometric polynomials $p$ increases. For example, the magnitude of a stopping band corresponding to $k_h h/\pi = 2$ drops to almost zero as the degree $p$ increases from 1 to 2, see Figure 3-3. Second, most importantly, the MFS-TRIp discretizations have almost no dispersion for $0 \leq k_h h/\pi \leq p$. In other words, the solutions to the Helmholtz equation are very accurately calculated by the MFS-TRIn discretizations up to the wave mode with $k_h h/\pi = p$. This property is consistent with the fact that the method of finite spheres can exactly reproduce the function used in the basis [38].

In two-dimensional case, we obtain

$$ D_{\text{stiff}} = K_{n,n} + K_{n,n-1} e^{-ik_h h \cos \theta} + K_{n,n+1} e^{ik_h h \cos \theta} $$

$$ + K_{n,n-2} e^{ik_h h (\cos \theta - \sin \theta)} + K_{n,n+2} e^{ik_h h (-\cos \theta + \sin \theta)} + K_{n,n-3} e^{-ik_h h \sin \theta} $$

$$ + K_{n,n+3} e^{ik_h h \sin \theta} + K_{n,n-4} e^{-ik_h h (\cos \theta + \sin \theta)} + K_{n,n+4} e^{ik_h h (\cos \theta + \sin \theta)} \quad (3.14) $$
Figure 3-5: Dispersion properties of the method of finite spheres in two-dimensional case: (a) MFS-LIN, (b) MFS-TRI1, (c) MFS-TRI2 and (d) MFS-TRI3 discretizations.

and

\[ D_{\text{mass}} = M_{n,n} + M_{n,n-1}e^{-ik_h \cos \theta} + M_{n,n+1}e^{ik_h \cos \theta} \\
+ M_{n,n-2}e^{ik_h(\cos \theta - \sin \theta)} + M_{n,n+2}e^{ik_h(-\cos \theta + \sin \theta)} + M_{n,n-3}e^{-ik_h \sin \theta} \\
+ M_{n,n+3}e^{ik_h \sin \theta} + M_{n,n-4}e^{-ik_h(\cos \theta + \sin \theta)} + M_{n,n+4}e^{ik_h(\cos \theta + \sin \theta)} \] (3.15)

where \( \theta \) is the angle between the direction of wave propagation and the horizontal axis of the Cartesian coordinate system, as shown in Figure 3-4. As one-dimensional case, the subscript \( n + k \) (\( k = -4, -3, \ldots, 3, 4 \)) in the above equations denotes the associated sphere \( n + k \), see Figure 3-1.

The dispersion properties of the method of finite spheres due to the spatial discretizations only in the two-dimensional case are very similar to those in the one-dimensional case, i.e., for the MFS-TRIp discretizations (here, \( p = 1, 2, 3 \)), wave
modes with $k_h h/\pi$ are almost non-dispersive, as shown in Figure 3-5. Another important point to note is that the numerical anisotropy is negligible in these wave modes. This property is clearly seen in Figure 3-5 where the results of a MFS-TRIp discretizations for different angles $\theta$ are approximately the same.

### 3.2.2 Additional effects introduced by the Bathe time integration

Since the MFS-TRIp discretizations introduce negligible dispersion error for sinusoidal waves with $k_h h/\pi \in [0, \pi]$ the dispersion error of these wave modes is entirely related to the time integration method used, which is in this study the Bathe time integration method [36, 37]. In general, temporal discretizations using various time integration methods result in period elongations and amplitude decays [3, 26]. Period elongations lead to an underestimated phase velocity and amplitude decays attenuate the wave modes. In this section, we analyze these additional effects when using the Bathe time integration.

We consider the method of finite spheres wave equation

$$M \ddot{a} + \left(\frac{c}{h}\right)^2 K a = 0$$  \hspace{1cm} (3.16)

where an overdot denotes time derivative. Using the translational invariant uniform distribution of nodes shown in Figure 3-1 with Equation (3.10), the above equation becomes

$$D_{\text{mass}} \ddot{\mathbf{a}} + \left(\frac{c}{h}\right)^2 D_{\text{stiff}} \mathbf{a} = 0.$$ \hspace{1cm} (3.17)

Note that the amplitude vector $\mathbf{a}$ is now a function of time. For the analysis, we perform a change of basis from the amplitude vector to the basis of eigenvectors of the following eigenproblem:

$$D_{\text{stiff}} \phi = \lambda^2 D_{\text{mass}} \phi.$$ \hspace{1cm} (3.18)

Using

$$\mathbf{a}(t) = \Phi \mathbf{z}(t)$$ \hspace{1cm} (3.19)
where the columns in $\Phi$ are the $D_{\text{mass}}$-orthonormalized eigenvectors $\phi_1, \ldots, \phi_{n_p}$ and substituting for $\dot{a}$ into Equation (3.17), we have

$$\ddot{z} + \left(\frac{c}{h}\right)^2 \Lambda^2 z = 0$$

(3.20)

where $\Lambda^2$ is a diagonal matrix listing the eigenvalues $\lambda_1, \ldots, \lambda_{n_p}$. To ascertain the dispersion and amplitude decay properties, we focus on this uncoupled equation instead of considering Equation (3.17), and here only the integration of one typical row in this equation in required, which may be written

$$\ddot{z} + \gamma^2 z = 0$$

(3.21)

where $\gamma = c\lambda/h$.

Using the Bathe time integration, we obtain the following relationship [3]:

$$\begin{bmatrix} t+\Delta t \ddot{z} \\ t+\Delta t \dot{z} \\ t+\Delta t \dot{z} \end{bmatrix} = \mathbf{A} \begin{bmatrix} t \ddot{z} \\ t \dot{z} \\ t \dot{z} \end{bmatrix}$$

(3.22)

where $\Delta t$ is the time step size and

$$\mathbf{A} = \frac{1}{\beta_1 \beta_2} \begin{bmatrix} -28\gamma^2 \Delta t^2 & \gamma \left( -144\gamma \Delta t + 5\gamma^3 \Delta t^3 \right) & -\gamma^2 \left( 144 - 19\gamma^2 \Delta t^2 \right) \\ -4\Delta t \left( -12 + \gamma^2 \Delta t^2 \right) & 144 - 47\gamma^2 \Delta t^2 & \gamma^2 \Delta t \left( -96 + \gamma^2 \Delta t^2 \right) \\ 28\Delta t^2 & \Delta t \left( 144 - 5\gamma^2 \Delta t^2 \right) & 144 - 19\gamma^2 \Delta t^2 \end{bmatrix}$$

(3.23)

with $\beta_1 = 16 + \gamma^2 \Delta t^2$ and $\beta_2 = 9 + \gamma^2 \Delta t^2$. Note that characteristic polynomial of the matrix $\mathbf{A}$ is

$$p_A(\eta) = \eta^3 - \frac{288 - 94\gamma^2 \Delta t^2}{144 + 25\gamma^2 \Delta t^2 + \gamma^4 \Delta t^4} \eta^2 + \frac{144 + 25\gamma^2 \Delta t^2}{144 + 25\gamma^2 \Delta t^2 + \gamma^4 \Delta t^4} \eta$$

(3.24)

and it can be found from $p_A(\eta_i) = 0$; $i = 1, 2, 3$ that there are two complex conjugate eigenvalues, $\eta_1, \eta_2$, which satisfy $|\eta_1|, |\eta_2| \leq 1$, and a zero eigenvalue, $\eta_3 = 0$, of $\mathbf{A}$. 

64
The recursive use of Equation (3.22) with the equilibrium equation (3.21) at time $t$ can eliminate the velocities and accelerations to have

$$
t^{+2\Delta t} - \frac{288 - 94\gamma^2 \Delta t^2}{144 + 25\gamma^2 \Delta t^2 + \gamma^4 \Delta t^4} t^{+\Delta t} + \frac{144 + 25\gamma^2 \Delta t^2}{144 + 25\gamma^2 \Delta t^2 + \gamma^4 \Delta t^4} t = 0. \quad (3.25)
$$

By comparing this with $p_A(\eta) = 0$, we note that the discrete solution to the above equation is of the form

$$
z_h = c_1 e^{(-\xi_h + i)\omega_h t_h} + c_2 e^{(-\xi_h - i)\omega_h t_h}, \quad (3.26)
$$
where $c_1$ and $c_2$ are undetermined coefficients, $\xi_h$ is the numerical damping ratio, $\omega_h$ is the numerical angular frequency and $t_h$ is discretized time. From this solution, we can obtain the relation between the numerical wavenumber $k_h$ and the numerical angular frequency $\omega_h$ depending on the CFL number defined as $\text{CFL} = c\Delta t/h$. Percentage amplitude decay is calculated using

$$
\text{Percentage amplitude decay} = 1 - e^{-2\pi \xi_h} \times 100. \quad (3.27)
$$

Figures 3-6 and 3-7 show the dispersion errors and the percentage amplitude decays of wave modes with $k_h h / \pi \in [0, 3]$ when using the MFS-TRI3 discretization and the Bathe method (denoted as MFS-TRI3 scheme), for one- and two-dimensional
cases, respectively. We observe that a decrease in the CFL number results in a decrease in both the dispersion error and the amplitude decay, i.e., more accurate solutions, and their accuracies are almost independent on the propagation directions considered. These properties imply that for wave modes with \( \frac{k_h h}{\pi} \in [0, p] \), the MFS-TRIp schemes do not require an optimal CFL number which gives the best solution, i.e., more accurate solutions are obtained monotonically by decreasing the CFL number, regardless of the propagation direction.

The wave modes with \( \frac{k_h h}{\pi} > p \) are damped out in practical analyses since in practice we do not use a very small CFL number for these modes. Note that the amplitude decay depends on the CFL number and the normalized numerical wavenumber and an increase in the CFL number or in the normalized numerical wavenumber leads to an increase in the amplitude decay, see Figure 3-6 and 3-7.

From the above results, it can be concluded that if \( h \) is chosen such that the response is well approximated by wave modes with \( \frac{k_h h}{\pi} \in [0, p] \) and the contributions of wave modes with \( \frac{k_h h}{\pi} > p \) can be neglected, the MFS-TRIp schemes give more accurate solutions with negligible numerical anisotropy as the CFL number decreases.
3.3 Numerical examples

Our objective in this section is to illustrate through numerical examples that the solutions of wave propagation problems using the MFS-TRIp discretizations \( p = 1, 2, 3 \) with the Bathe time integration converge to the exact analytical solutions when decreasing the CFL number and are almost independent of the propagation direction if we choose \( h \) such that the wave modes with \( 0 \leq k_h h/\pi \geq p \) are sufficient to accurately approximate the response. We first consider scalar wave propagations in one- and two-dimensional media and then solve the Lamb’s problem [1], where we have different types of wave propagations in a semi-infinite elastic medium. These examples are taken from References [7, 24, 33].

In all numerical examples, the computational domain is sufficiently large so that a wave does not reach the boundary of the computational domain for the time considered. We do not employ, therefore, a technique such as a perfectly matched layer [43–46] to suppress artificially reflected waves at the boundary of the computational domain. For the discretization, like in the dispersion analyses, we uniformly distribute spheres with \( r_I = h \) and use the Bathe time integration.

The computational expense of obtaining two-dimensional scalar wave solutions when using the MFS-TRIp discretizations with the Bathe integration was studied to some degree in Reference [7] where it was observed that for approximately the same relative error measured in the \( L^2 \) norm, the computational times used in the MFS-TRIp schemes are close to the times used for the bi-linear finite element solutions. The reason for this observation is that using the MFS-TRIp schemes a relatively coarse spatial discretization with a similar time step size can be used. The major effort in all solution procedures then pertains to the vector forward reductions and back-substitutions in the time integrations since the evaluations of the stiffness and mass matrices and the matrix factorizations are performed only once. However, since we considered only the solutions of two-dimensional academic problems, a detailed study of various cases in engineering practice, including three-dimensional response solutions, would be valuable.
### 3.3.1 One-dimensional impact of an elastic bar

We consider the one-dimensional wave propagation due to an impact on an elastic semi-infinite bar, as shown in Figure 3-8. The governing equation for the displacement $u$ of the bar is the one-dimensional wave equation where $c = 1 \text{ m/s}$. All initial conditions are zero and a unit step velocity is applied at the left end, see Figure 3-8.

To solve this problem, we calculate all solutions with $h = 0.04 \text{ m}$, which corresponds to the number of spheres $N = 51$ for the computational domain $V = [0, 2]$, see Figure 3-8. Figure 3-9 shows the results obtained. We see that with the different MFS-TRIp schemes we obtain similar results when $\text{CFL} = 1$, and that the solutions are monotonically improved when using the MFS-TRIp schemes by decreasing the CFL number. Using the MFS-TRI2 and MFS-TRI3 schemes surprisingly accurate results are obtained when a small CFL number is used. A more accurate solution with the MFS-TRI1 scheme can be obtained by increasing the number of spheres, i.e. decreasing the value of $h$.

### 3.3.2 Two-dimensional scalar wave propagation

We next consider the two-dimensional scalar wave propagation resulting from a concentrated force (a point excitation) $F_c$ applied at the center of a pre-stressed square
Figure 3-9: Velocity distributions of the bar at $t = 1$ s when decreasing the CFL number: using (a) MFS-LIN, (b) MFS-TRI1, (c) MFS-TRI2 and (d) MFS-TRI3 schemes, with 51 spheres.

The displacement of the membrane $u$ is governed by the two-dimensional wave equation where $c = 1$ m/s. The membrane is initially at rest and for the computation only the domain $V = [0, 1] \times [0, 1]$ is considered, because of symmetry.

First, a Ricker wavelet is considered as the concentrated force

$$F_c = A_F \left[ 1 - 2\pi^2 f_p^2 (t - t_s)^2 \right] e^{-\pi^2 f_p^2 (t-t_s)^2}$$

where $A_F$ is the amplitude of the force, $f_p$ is the peak frequency and $t_s$ is the time shift. In this problem, we use $A_F = 40 \text{ cm/s}^2$, $f_p = 5$ Hz and $t_s = 0.25$ s. The solutions are computed using the schemes MFS-LIN with $31 \times 31$ spheres, MFS-TRI1 with $31 \times 31$ spheres, MFS-TRI2 with $16 \times 16$ spheres and MFS-TRI3 with $11 \times 11$ spheres. Figure 3-11 shows the calculated displacements along the horizontal axis.
at $t = 0.8$ s. We observe that while the MFS-LIN scheme does not give an accurate solution, the solutions of MFS-TRIP converge to the exact solution as the CFL number decreases. Another important observation is that the accuracy of these solutions is almost the same for any angle $\theta$ considered, see Figures 3-12 and 3-13.

Next we consider the wave propagation due to a concentrated force

$$F_c = \begin{cases} 
1.6 \times 10^2 t (0.1 - t) & \text{for } 0 \leq t < 0.1, \\
0 & \text{for } t \geq 0.1.
\end{cases} \quad (3.29)$$

Since much higher wave modes need now be captured to predict the response
Figure 3-11: Displacement distributions of the membrane along horizontal axis at $t = 0.8 \text{ s}$ when decreasing the CFL number: using (a) MFS-LIN scheme with $31 \times 31$ spheres, (b) MFS-TRI1 scheme with $31 \times 31$ spheres, (c) MFS-TRI2 scheme with $16 \times 16$ spheres and (d) MFS-TRI3 scheme with $11 \times 11$ spheres.

accurately we use $55 \times 55$ spheres for the MFS-TRI3 scheme. Figure 3-14 shows the results at $t = 0.9 \text{ s}$, namely, the displacement distributions along the horizontal axis with decreasing the CFL number, the displacement distributions along three different directions when CFL $\approx 0.016$, and the overall displacement distributions in the computational domain when CFL $\approx 0.016$. As in the previous case, a more accurate solution is obtained when decreasing the CFL number and the solution accuracy is almost independent of $\theta$.

### 3.3.3 Two-dimensional elastic wave propagation

In this solution we consider two-dimensional waves propagating in a semi-infinite elastic medium in plane strain conditions [1], as illustrated in Figure 3-15. The concentrated line force located at the free surface of the medium is a Ricker wavelet,
Figure 3-12: Displacement distributions of the membrane along various directions ($\theta = 0^\circ, 22.5^\circ, 45^\circ$) at $t = 0.8$ s: when using (a) MFS-TRI1 scheme with $31 \times 31$ spheres, (b) MFS-TRI2 scheme with $16 \times 16$ spheres and (c) MFS-TRI3 scheme with $11 \times 11$ spheres; CFL $\approx 0.016$. 
Figure 3-13: Snapshots of displacement distributions of the membrane at $t = 0.8$ s when decreasing the CFL number: using (a) MFS-LIN scheme with $31 \times 31$ spheres, (b) MFS-TRI1 scheme with $31 \times 31$ spheres, (c) MFS-TRI2 scheme with $16 \times 16$ spheres and (d) MFS-TRI3 scheme with $11 \times 11$ spheres.
Figure 3-14: Displacement distributions of the membrane at $t = 0.9$ s when using MFS-TRI3 scheme with $55 \times 55$ spheres: (a) displacement distributions along horizontal axis with decreasing the CFL number, (b) displacement distributions along various directions ($\theta = 0^\circ, 22.5^\circ, 45^\circ$) when CFL $\approx 0.016$ and (c) snapshot of displacement distributions when CFL $\approx 0.016$. 

---

Explain the key points from the figure in natural language:
\[ u(x, t = 0) = 0 \text{ m} \quad \rho = 2200 \text{ kg/m}^3 \]
\[ \dot{u}(x, t = 0) = 0 \text{ m/s} \quad c_P = 3200 \text{ m/s} \]
\[ c_S = 1847.5 \text{ m/s} \quad c_R = 1698.6 \text{ m/s} \]

Figure 3-15: The Lamb’s problem [1] - 2D wave propagation in a plane strain elastic half-space: problem description and spatial discretization by finite spheres.

see Equation (3.28), with \( A_F = 2 \times 10^6 \text{ N/m}, f_p = 10 \text{ Hz} \) and \( t_s = 0.1 \text{ s} \). For the material properties, we use the mass density \( \rho = 2200 \text{ kg/m}^3 \), the P-wave velocity \( c_P = 3200 \text{ m/s} \) and the S-wave velocity \( c_S = 1847.5 \text{ m/s} \), which gives the Rayleigh wave velocity \( c_R \approx 1698.6 \text{ m/s} \). The initial displacements and velocities are zero and the computational domain \( V = [0, 3200] \times [0, 3200] \) is considered. The displacements \( \mathbf{u} = [u, v]^T \) of the medium are governed by the Navier equations, which can be expressed in the form of wave equations by the Helmholtz decomposition [47].

We use the MFS-TRI3 scheme with \( 36 \times 36 \) spheres for the solution and calculate the time step size \( \Delta t \) based on the P-wave velocity because then the other types of waves are also accurately simulated, indeed more accurately. Figure 3-16 shows
Figure 3-16: Horizontal and vertical displacements of the elastic medium when using MFS-TRI3 scheme with $36 \times 36$ spheres: (a) displacement histories at $x = (640, 0)$ m with decreasing the CFL number, (b) displacement histories at $x = (1280, 0)$ m with decreasing the CFL number and (c) snapshots of displacement distributions at $t = 1$ s when CFL $\approx 0.016$. 

Horizontal distance $x$ (m)
the predicted horizontal and vertical displacements, namely the calculated displacement histories at two receivers located at $x = (640, 0)$ m and at $x = (1280, 0)$ m with decreasing the CFL number, and the overall displacement distributions in the computational domain at $t = 1$ s when CFL $\approx 0.016$. As expected, the numerical solutions of all three waves monotonically converge to the exact solutions as the CFL number is decreased.

3.4 Concluding remarks

The objective in this chapter was to present a study of the method of finite spheres used with the Bathe method for time integration in the solution of transient wave propagation problems. We analyzed the dispersion property of the method of finite spheres and the numerical effects introduced when using the Bathe method. We illustrated these properties through the solutions of numerical examples. Although we considered only one- and two-dimensional cases, the same analysis can be extended to the three-dimensional case.

The key observation is that we can use the method of finite spheres with the bi-linear polynomials and the trigonometric polynomials as local basis functions to calculate the solution with increasing accuracy as the CFL number is decreased. This is most important and in contrast to the use of the traditional finite elements. Hence very accurate solutions can be obtained when multiple types of waves at different speeds propagate by using the largest wave speed to select the CFL number. Indeed any speed can be chosen for the CFL number if it is larger than all wave speeds. Furthermore, with the uniform sphere arrangements used, the solutions of the problems considered in this paper were accurate in any direction.
Chapter 4

Overlapping finite elements enriched for wave propagation problems

In the previous chapter, the method of finite spheres, a meshless method, enriched for wave propagation problems was used with the Bathe time integration scheme to solve wave propagation problems but uniform spatial discretizations need be used, see also References [4,7]. An important observation is that in the uniform spatial discretizations, a decrease in the time step size leads to a more accurate solution, which is what an analyst intuitively expects, and numerical anisotropy is almost negligible. These are important observations because by using the largest wave speed to establish the time step size, accurate solutions for multiple types of waves can be obtained and regardless of the propagation directions. The details of the mathematical analysis of the solution procedure and illustrative example solutions are given in the previous chapter and References [4,7].

However, the major difficulty in using the method of finite spheres, like other meshless methods, is the very expensive numerical integration for the construction of the mass and stiffness matrices [40,48,49]. The integration cost is clearly prohibitive for irregular discretizations using spheres, see Reference [49]. For uniform discretizations, the numerical integration can be performed only once for a typical sphere and the result can then be reused [48], but this approach can of course not be employed when non-uniform spatial discretizations need be used. The high computational cost
of the method impedes its wide practical use in industry.

In this chapter, for the solution of transient wave propagation problems, we enrich the overlapping finite elements of Reference [50] using trigonometric polynomials and use the Bathe time integration method because of its favorable dissipation properties [26,37]. The same approach has already been applied for use of a traditional finite element [33] and the method of finite spheres [4, 7]. However, as already mentioned above, the use of the method of finite spheres is not efficient in general practical analyses because of the very expensive numerical integrations. For the traditional finite element enriched with trigonometric functions, the solution effort is more acceptable, although high, but the solution accuracy is not as desired because the predicted response sensitively depends on the directions of waves traveling through the mesh and fine meshes or high-order harmonic functions are required.

Our objective in this chapter is to analyze the overlapping finite element enriched with trigonometric polynomials together with the Bathe time integration method and illustrate that the combined spatial and time discretization scheme can be used to solve wave propagations in complex geometries using regular or irregular meshes. Hence, as we also demonstrate, the element can be used with the new paradigm of finite element solutions for CAD. We also show that using the scheme accurate solutions can be obtained for wave propagations in elastic media with crack, heterogeneous media and anisotropic media.

In the next section, we formulate the overlapping finite element for transient wave propagation problems. Then, we study the dispersion properties of the proposed scheme. Thereafter, we provide the calculated solutions of various wave propagation problems to illustrate the capability of the solution scheme. We focus on showing that even when using irregular meshes good results are obtained. Finally, we give the conclusions of our research.
4.1 Spatial approximation scheme

In the new paradigm of finite element analysis, the global analysis domain is discretized by traditional finite elements (that do not overlap) and finite elements that overlap [51, 52]. For every overlap region, the solution variable $u$ is approximated as [49, 50]

$$u \approx u_h = \sum_{i=1}^{q} h_i \psi_i = \sum_{i=1}^{q} h_i \sum_{j=1}^{q} \phi_j^I \sum_{n \in J} p_n a_{jn}$$

(4.1)

where $q$ is the number of nodes in the overlap region, $h_i$ is the shape function used in the traditional finite element [3], $\phi_j^I$ is a partition of unity function, $J_J$ is an index set and $p_n$ is a set of local basis functions (e.g., a polynomial for elliptic problems) which span the local approximation space $V_J^h$ with the corresponding coefficient of node $J$. It is important to note that the function $\phi_j^I$ is a polynomial and hence the computational cost for establishing the stiffness and mass matrices is not high.

For the solution of two-dimensional wave propagation problems, the bi-linear polynomials and trigonometric polynomials are employed for the local approximation space, i.e., at node $J$ we use

$$V_J^h = \text{span} \left\{ 1, x, y, xy, \cos \left( \frac{2\pi x}{\lambda_x} \right), \sin \left( \frac{2\pi x}{\lambda_x} \right), \cos \left( \frac{2\pi y}{\lambda_y} \right), \sin \left( \frac{2\pi y}{\lambda_y} \right), \cos \left( \frac{2\pi x + 2\pi y}{\lambda_y} \right), \sin \left( \frac{2\pi x + 2\pi y}{\lambda_y} \right), \cos \left( \frac{2\pi x}{\lambda_x} - \frac{2\pi y}{\lambda_y} \right), \sin \left( \frac{2\pi x}{\lambda_x} - \frac{2\pi y}{\lambda_y} \right) \right\}$$

(4.2)

where $x$ and $y$ are the Cartesian coordinates, $\lambda_x$ and $\lambda_y$ are the fundamental wavelengths in the $x$- and $y$-directions, respectively, and $p$ is the degree of the trigonometric polynomials used. Hence the number of degrees of freedom per node is 12, 20, 28 when $p = 1, 2, 3$, respectively.

In a one-dimensional wave propagation, the local approximation space is, for node
\[ V_j^h = \text{span} \left\{ 1, x, \cos \left( \frac{2\pi x}{\lambda_x} \right), \sin \left( \frac{2\pi x}{\lambda_x} \right), \ldots, \cos \left( \frac{2\pi px}{\lambda_x} \right), \sin \left( \frac{2\pi px}{\lambda_x} \right) \right\} \] (4.3)

Note that in the local approximation space, the polynomials are included to satisfy linear consistency, ensuring that the approximation can reproduce the linear field (e.g., rigid body displacements and constant strain states). For some experiences using such functions see, for example, References [4,7,32,33].

In this study, we use \( \lambda_x = \lambda_y = 2h \) where \( h \) is the typical size of overlap region and mainly focus on the use of \( p = 1 \). Of course, the use of a higher degree polynomial gives a better solution accuracy for a given mesh, but the solution is also computationally more costly [7,33].

### 4.2 Dispersion analysis

Following the approach given in section 3.2 and Reference [4], we investigate in this section the dispersion properties when solving the standard two-dimensional wave equation with the overlapping finite elements and the Bathe time integration scheme. We compare the results obtained to those when using the traditional finite elements enriched for wave propagation problems [33]. We first consider the dispersion errors caused by only the spatial discretization, and then analyze the additional effects resulting from the temporal discretization using the Bathe implicit time integration.

For the dispersion analysis, we consider a uniform mesh which is translationally-invariant, see Figure 4-1. We also apply the same approach to the scheme discussed in Reference [33], i.e. the triangular traditional finite element enriched with trigonometric polynomials of degree \( p = 1 \).

#### 4.2.1 Spatial discretization error

Figure 4-2 shows the dispersion properties of the overlapping finite elements with the bilinear polynomials and the first degree trigonometric polynomials (OFE-TRI1)
and of the traditional finite elements enriched with the first degree trigonometric polynomials (FE-TRI1). We see that the OFE-TRI1 discretization has practically no dispersion for the wave modes $k_h h/\pi \leq 0.8$ with regardless of the propagation direction. For the wave modes with $0.8 \leq k_h h \leq 1$, the dispersion errors depend on the propagation direction, but the errors are small. On the other hand, these observations are not applicable to the FE-TRI1 scheme, for which the dispersion errors are quite large. To obtain better results higher-order harmonic functions need to be included which however increases the computational complexities [33].

To illustrate the performance of the methods in uniform and distorted meshes, we solve a simple problem in the domain $V = [0, 1] \times [0, 1]$ for which the exact solution is $u = \sin(2\pi x)$, see Figure 4-3. Figure 4-4 shows the $L^2$ relative error norm $e_r$, defined by

$$e_r = \sqrt{\frac{\int_V (u - u_h)^2 \, dV}{\int_V u^2 \, dV}},$$

(4.4)

Figure 4-1: Uniform mesh and propagation angle of a sinusoidal plane wave.
of the solutions when using the OFE-TRI1 and FE-TRI1 discretizations with uniform and distorted meshes. As expected from the dispersion properties shown in Figure 4-2, the OFE-TRI1 discretization with all the uniform meshes considered gives solutions with $\log_{10} e_r < -3$, while using the FE-TRI1 discretization, solutions with
Figure 4-3: Uniform and distorted meshes used for the simple problem for which the exact solution is \( u = \sin(2\pi x); \quad h = 1/N \) where \( N \) is the number of elements along each side.

\( \log_{10} e_r < -3 \) are obtained only when the uniform meshes are used with \( h = 1/64, 1/32 \) \( (\log_{10} h = -1.8, -1.5) \), see Figure 4-4.

When the distorted mesh shown in Figure 4-3 is used, to have a solution with \( \log_{10} e_r < -3 \), the OFE-TRI1 discretization with all the distorted meshes considered can be used, but a very fine mesh is required when using the FE-TRI1 discretization, see Figure 4-4.
4.2.2 Additional effects introduced by the Bathe time integration

Temporal discretizations using time integration methods result in period elongations and amplitude decays [3]. Period elongations lead to errors in the phase velocity (dispersion errors) and amplitude decays result in attenuations of the wave modes (dissipation errors). In this section, we illustrate these additional numerical effects when using the Bathe implicit time integration.

Figure 4-5 shows the dispersion errors and the percentage amplitude decays of
Figure 4-5: Dispersion (left) and dissipation (right) properties of OFE-TR11 scheme for various propagation angles: when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.
Figure 4-6: Dispersion (left) and dissipation (right) properties of FE-TRI1 scheme for various propagation angles: when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.
wave modes for various propagation angles when using the OFE-TRI1 discretization with the Bathe time integration (denoted by OFE-TRI1 scheme). It is important to note that practically all wave modes with $k_h h / \pi \leq 1$ monotonically converge to the exact wave modes when decreasing the CFL number and the accuracy does not depend on the propagation direction. This is not seen for the use of the FE-TRI1 discretization with the Bathe time integration (denoted by FE-TRI1 scheme) as shown in Figure 4-6. The phase velocity shows a significant error, and this error depends on the direction considered, even when using $CFL = 0.25$.

We also observed this characteristic when using the method of finite spheres with the bi-linear polynomials and the trigonometric polynomials and the Bathe time integration scheme, denoted as the MFS-TRI scheme. In practical analyses, however, the MFS-TRI scheme is very expensive to use because of the computational cost when the nodes are distributed irregularly. Since the partition of unity function used in the OFE-TRI scheme is a polynomial, the computational effort in the OFE-TRI scheme is much smaller.

4.3 Numerical examples

In the previous section, we analyzed mathematically the important properties of the OFE-TRI1 scheme. The result is however valid only for the uniform mesh considered. Our objective in this section is to illustrate the important properties of the OFE-TRI1 scheme in both structured (regular) and unstructured (irregular) meshes through numerical examples.

We first consider scalar wave propagations in one- and two-dimensional media. We solve the Lamb’s problem [1] where different waves propagate in a semi-infinite elastic medium. A pre-stressed membrane with circular holes is considered where we employ the meshing procedure proposed in References [51, 52]. We then solve a heterogeneous wave propagation problem and the Lamb’s problem with a crack. Finally, we solve an anisotropic elastic wave propagation problem.

In all numerical examples, the waves do not reach the boundary of the computa-
\[ c = 1000 \text{ m/s} \]
\[ u(x, t = 0) = 0 \text{ m} \]
\[ \dot{u}(x = 0, t > 0) = 1 \text{ m/s} \]
\[ \dot{u}(x, t = 0) = 0 \text{ m/s} \]

Figure 4-7: 1D impact of an elastic bar problem: problem description and mesh used.

We, therefore, do not use a technique such as a perfectly matched layer [33,43-46] to suppress artificially reflected waves at the boundary.

We also note that, as in the discussions given above, in all solutions we use the consistent mass matrices and for the evaluations of all matrices we use the Gaussian quadrature rules of sufficiently high order given in Reference [53].

### 4.3.1 One-dimensional wave propagation

We consider the one-dimensional (1D) wave propagation due to an impact on an elastic semi-infinite bar, see Figure 4-7. The displacement \( u \) is governed by the wave equation with \( c = 1000 \text{ m/s} \), zero initial conditions and a unit step velocity applied at the left end of the bar. The computational domain \( V = [0, 2] \) is considered and discretized with nodes spaced \( \Delta_i, i = 1, 2, \ldots \), apart as shown in Figure 4-7. For the non-uniform mesh, we use

\[
\Delta_i = \frac{\Delta_{\text{max}} - \Delta_{\text{min}}}{N - 1} (i - 1) + \Delta_{\text{min}}, \quad i = 1, 2, \ldots, N
\]
where $\Delta_{\text{max}}$ and $\Delta_{\text{min}}$ are the maximum distances between nodes, respectively, and $N$ is the number of elements.

We first solve the problem using the traditional two-node finite element with the Bathe time integration (denoted by FE scheme). In both uniform and non-uniform meshes, 200 elements are employed. Using a uniform mesh the best result is obtained when $\text{CFL} = 1$, see Reference [24] and Figure 4-8. Using the non-uniform mesh, however, this optimal CFL number cannot be achieved for all elements because each element has a different length. Figure 4-8 shows the velocity distributions calculated using different element sizes to select the CFL number, namely with $h = \Delta_{\text{min}}$, $h = \Delta_{\text{max}}$, and $h = (\Delta_{\text{min}} + \Delta_{\text{max}})/2$. We see that depending on the element size used for the CFL number, spurious oscillations occur in front of or behind the wave front.
Figure 4-9: Velocity distributions of the bar at $t = 0.001$ s calculated using (a) FE-TRI2 scheme and (b) OFE-TRI2 scheme with the uniform and non-uniform meshes; $h$ denotes the element size used for the CFL number.
We then calculate the solutions using the FE-TRI2 and OFE-TRI2 schemes, with the second degree trigonometric polynomials. Using both the uniform and non-uniform meshes, the number of elements is equal to 50 and for the non-uniform mesh the CFL number is chosen by \( h = \Delta_{\text{min}} \). Figure 4-9 gives the results where we see that quite accurate solutions are achieved with both schemes and the OFE-TRI2 scheme performs slightly better.

The monotonic convergence property, as the element sizes and the CFL number decrease, is very important when using a non-uniform mesh. If the smallest element size is used to select the CFL number, the other element sizes correspond to smaller CFL numbers, for which (using each element in a uniform mesh) with our scheme more accurate solutions are achieved. For this reason, we can expect that the OFE-TRI2 scheme accurately solves the problem using a non-uniform mesh with \( h = \Delta_{\text{min}} \) for the CFL number, see Figure 4-9.

### 4.3.2 Two-dimensional scalar wave propagation

We consider the two-dimensional (2D) scalar wave propagation due to a concentrated force (a point excitation) \( F_c \) at the center of a pre-stressed membrane, see Figure 4-10, for which the transverse displacement \( u \) is governed by

\[
\nabla^2 u + F_c(x = 0, t) = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2}
\]

with initial conditions

\[
\begin{align*}
    u(x, t = 0) &= 0 \text{ m} \\
    \dot{u}(x, t = 0) &= 0 \text{ m/s}
\end{align*}
\]

and the wave propagation velocity \( c = 1 \text{ m/s} \). Because of symmetry, we mesh only the domain \( V = [0, 1] \times [0, 1] \) for the solution.

First, a Ricker wavelet is considered as the concentrated force, see Equation (3.28), where we use in this problem the peak frequency \( f_p = 5 \text{ Hz} \) and the time shift
\[ c = 1 \text{ m/s} \]
\[ u(x, t = 0) = 0 \text{ m} \]
\[ \dot{u}(x, t = 0) = 0 \text{ m/s} \]

Figure 4-10: Problem description of 2D scalar wave propagation in a pre-stressed membrane.

t_s = 0.25 \text{ s}. For the solution, we use the OFE-TRI1 scheme with the structured and unstructured meshes shown in Figure 4-11 with the number of elements along each side \( N = 32 \) and the typical element size \( h = 1/N = 0.03125 \text{ m} \).

Figures 4-12 and 4-13 show comparisons of the displacements calculated using the FE-TRI1 and the OFE-TRI1 schemes at \( t = 0.95 \text{ s} \) using the structured mesh. We observe that the FE-TRI1 scheme does not give an accurate solution, whereas the solutions obtained using the OFE-TRI1 scheme converge monotonically to the exact solution as the CFL number decreases and are almost identical for any propagation angle \( \theta \) considered.

In Figure 4-14, we show the solutions using the OFE-TRI1 scheme with the unstructured mesh. As when using the structured mesh, we see that a more accurate solution is obtained with decreasing CFL number and the solution accuracy is practically independent of the propagation direction.

We next consider the concentrated force defined in Equation (3.29). For this load
We consider here the solution of the 2D waves propagating in a semi-infinite elastic medium in plane strain conditions [1], as described in Figure 4-16. For the concentrated line force located at the free surface of the medium, we consider a Ricker wavelet, see Equation (3.28), with the magnitude $2 \times 10^6$, $f_p = 10$ Hz and $t_s = 0.1$ s. The isotropic semi-infinite elastic medium has the mass density $\rho = 2200$ kg/m$^3$, the P-wave velocity $c_P = 3200$ m/s and the S-wave velocity $c_S = 1847.5$ m/s, which gives
Figure 4-12: Contour plots of displacement distributions of the membrane at $t = 0.95$ s calculated using the FE-TRI1 scheme (left) and the OFE-TRI1 scheme (right), both with the structured mesh ($h = 0.03125$ m): when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.
Figure 4-13: Displacement distributions of the membrane along various directions ($\theta = 0^\circ, 22.5^\circ, 45^\circ$) at $t = 0.95$ s calculated using FE-TRI1 scheme (left) and OFE-TRI1 scheme (right), both with the structured mesh ($h = 0.03125$ m): when (a) CFL = 1 (b) CFL = 0.5 and (c) CFL = 0.25.
Figure 4-14: Displacement distributions of the membrane at $t = 0.95$ s calculated using OFE-TRI1 with the unstructured mesh ($h = 0.03125$ m): (a) displacement distributions along horizontal axis with decreasing the CFL number and (b) displacement distributions along various directions when CFL = 0.125.
Figure 4-15: Displacement distributions of the membrane at $t = 0.9$ s calculated using OFE-TRI1 scheme with the structured mesh (left) and with the unstructured mesh (right); in both cases $h = 0.015625$ m: (a) displacement distributions along horizontal axis with decreasing the CFL number, (b) displacement distributions along various directions when CFL = 0.125 and (c) contour plots of displacement distributions when CFL = 0.125.
\[ u(x, t=0) = 0 \text{ m} \]
\[ \dot{u}(x, t=0) = 0 \text{ m/s} \]
\[ \rho = 2200 \text{ kg/m}^3 \]
\[ c_p = 3200 \text{ m/s} \]
\[ c_s = 1847.5 \text{ m/s} \]
\[ c_R = 1698.6 \text{ m/s} \]

Figure 4-16: Problem description of the Lamb's problem [1] - 2D wave propagation in a plane strain elastic half-space.

the Rayleigh wave velocity \( c_R = 1698.6 \text{ m/s} \). The initial displacements and velocities are zero and the computational domain \( V = [0,3200] \times [0,3200] \) is considered with the structured and unstructured meshes shown in Figure 4-11. This is a good test problem for a scheme to solve wave propagations, e.g. see References [24,33].

We solve the problem using the OFE-TRI1 scheme with \( N = 64 \) (\( h = 50 \text{ m} \)), and show the displacement histories at two receivers located at \( x = (640,0) \text{ m} \) and at \( x = (1280,0)\text{ m} \) in Figures 4-17 and 4-18. The time step size \( \Delta t \) is calculated based on the P-wave velocity because the other types of waves are then also accurately predicted. As expected, in both cases, all waves are simulated more accurately as the CFL number is decreased.

To display the response predicted over the complete mesh, Figure 4-19 shows the calculated von Mises stress distributions at \( t = 1 \text{ s} \) using the meshes when \( \text{CFL} = 0.125 \).

4.3.4 Two-dimensional wave propagation in a pre-stressed membrane with circular holes

We consider again the scalar wave propagation in the pre-stressed membrane but in this analysis, we have four circular holes, as shown in Figure 4-20. The objective in this example is to show the performance of the OFE-TRI1 scheme when employed
Figure 4-17: Horizontal displacements (left) and vertical displacements (right) of the elastic medium calculated using OFE-TRII scheme with the structured mesh ($h = 50$ m): (a) displacement histories at $x = (640,0)$ m with decreasing CFL number and (b) displacement histories at $x = (1280,0)$ m with decreasing CFL number.
Figure 4-18: Horizontal displacements (left) and vertical displacements (right) of the elastic medium calculated using OFE-TRI1 scheme with the unstructured mesh \((h = 50\, m)\): (a) displacement histories at \(x = (640, 0)\) m with decreasing CFL number and (b) displacement histories at \(x = (1280, 0)\) m with decreasing CFL number.
Figure 4-19: Snapshots of von Mises stress distributions of the elastic medium at $t = 1 \text{ s}$ calculated using OFE-TRI1 with the structured mesh (top) and with the unstructured mesh (bottom); in both cases $h = 50 \text{ m}$ and $\text{CFL} = 0.125$. Brighter color represents larger stress.
Figure 4-20: 2D scalar wave propagation in a pre-stressed membrane: problem description and the mesh used; the line colored in red is the direction of $\theta = 22.5^\circ$ and the line colored in green is the direction of $\theta = 67.5^\circ$.

The meshing scheme consists of four steps, see References [51, 52]. In the first step the scheme immerses the pre-stressed membrane in a Cartesian grid with $\Delta x$ and $\Delta y$ as distances between lines. In the second step the boundaries are discretized with straight line segments $\Delta s$, during which geometry deficiencies are automatically removed. Thereafter, in the third step the scheme converts the internal cells into traditional finite elements. Finally, in the fourth step the scheme covers the empty space (region near the holes) with overlapping finite elements. Here for the wave propagation analysis, as shown in Figure 4-20, all the traditional finite elements in the inner solution domain are also triangularized and converted to overlapping finite elements. In this example, we use $h = \Delta x = \Delta y = 0.02$ m and $\Delta s = 0.004\pi$ m.
We use a Ricker wavelet (3.28) with the magnitude 0.4, $f_p = 10$ Hz and $t_s = 0.1$ s for the excitation force. The calculated solutions using the OFE-TRE1 scheme are compared with the reference solution obtained using the OFE-TRI1 scheme with a very fine mesh of 28,055 elements.

Figure 4-21 shows the calculated results after the wave passed the hole. We see that, as in the previous example solution, a decrease in the CFL number leads to a more accurate solution and that the wave profiles in the two directions ($\theta = 22.5^\circ$ and $\theta = 67.5^\circ$) are in good agreement with the reference solution. Some snapshots of calculated solutions at various observation times are shown in Figure 4-22.

4.3.5 Two-dimensional scalar wave propagation in a pre-stressed bi-material membrane

In this problem, a scalar wave generated by the concentrated force defined in Equation (3.29) propagates in a pre-stressed bi-material membrane with the wave propagation velocities $c_1 = 1$ m/s for an inner square region and $c_2 = 4$ m/s for the other region, as shown in Figure 4-23. Only the domain $[0, 2] \times [0, 2]$ is used for the OFE-TRI1 solution because of symmetry. The computational domain is discretized differently in each region using $N_1 = 20 (h_1 = 0.025)$ for the inner square and $N_2 = 20 (h_2 = 0.1)$ for the outer square where $N_1$ and $N_2$ are the number of elements along each side, and nodes are distributed regularly and irregularly, see Figure 4-24. For the CFL number, we use $c_1$ and $h_1$ to calculate the time step size, but since $c_2/h_2 = c_1/h_1$, the element in the outer region feels the same CFL number.

Figures 4-25, 4-26 and 4-27 show the displacement distributions along the direction of $\theta = 0^\circ$, $22.5^\circ$, and $45^\circ$ at various observation times with decreasing CFL number, respectively. The reference solution is obtained using the OFE-TRI1 scheme with a very fine mesh of 38,900 elements.

Figure 4-28 shows the contour plots of displacement distributions at different observation times calculated using the structured mesh and unstructured mesh, both when CFL = 0.125. We see that all the waves, the excited wave and the waves
Figure 4-21: Displacement distributions of the membrane with circular holes at \( t = 1 \) s calculated using OFE-TRI1 scheme: (a) displacement distributions along the direction of \( \theta = 22.5^\circ \) with decreasing the CFL number and (b) displacement distributions along the two directions \( (\theta = 22.5^\circ \) and \( \theta = 67.5^\circ ) \) when CFL = 0.125.
Figure 4-22: Snapshots of displacement distributions of the membrane with circular holes at various observation times calculated using OFE-TRI1 scheme; CFL = 0.125.
\[ u(x, t = 0) = 0 \text{ m} \]
\[ \dot{u}(x, t = 0) = 0 \text{ m/s} \]

Figure 4-23: Problem description of 2D scalar wave propagation in a pre-stressed bi-material membrane; \( c_1 = 1 \text{ m/s} \) and \( c_2 = 4 \text{ m/s} \).

transmitted and reflected at the interface of two different materials, are well predicted in both the structured and unstructured meshes.

4.3.6 Two-dimensional elastic wave propagation in a semi-infinite medium with a crack

The Lamb problem is again considered, see Figure 4-16, but in this problem there is a horizontal crack with a length of 640 m located at \( x = (0, -960) \text{ m} \). The same Ricker wavelet is used for the concentrated line force and we solve the problem by applying the OFE-TRI1 scheme with the structured and unstructured meshes with \( N = 50 \) \((h = 62 \text{ m})\). For comparison, the reference solution is calculated using the OFE-TRI1 scheme with a mesh of 20,000 elements.

The calculated displacement history at a receiver located at \( x = (640, 0) \text{ m} \) is
Figure 4-24: Structured and unstructured meshes used for the pre-stressed bi-material membrane problem; $N_1$ and $N_2$ are the number of elements along each side of the inner square and the outer square, respectively.

shown in Figure 4-29. We observe that the wave reflected from the crack is also accurately simulated (when compared to the refined numerical solution) as the CFL number is decreased. The calculated von Mises stress distributions at various observation times are illustrated in Figures 4-30, 4-31, 4-32 and 4-33.

4.3.7 Two-dimensional elastic wave propagation in an anisotropic medium

We solve a 2D anisotropic elastic wave propagation problem. The relation between the stress vector $\sigma$ and the strain vector $\varepsilon$ is written as

$$\sigma = C \varepsilon$$ (4.9)
Figure 4-25: Displacement distributions of the bi-material membrane along the direction of $\theta = 0^\circ$ calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5 \text{ s}$, (b) $t = 0.65 \text{ s}$ and (c) $t = 0.8 \text{ s}$. 
Figure 4-26: Displacement distributions of the bi-material membrane along the direction of $\theta = 22.5^\circ$ calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.
Figure 4-27: Displacement distributions of the bi-material membrane along the direction of $\theta = 45^\circ$ calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.
Figure 4-28: Contour plots of displacement distributions of the bi-material membrane calculated using OFE-TRI1 scheme (CFL = 0.125) with the structured mesh (left) and the unstructured mesh (right) at (a) $t = 0.5$ s, (b) $t = 0.65$ s and (c) $t = 0.8$ s.
Figure 4-29: Time history of displacements of the elastic medium with a horizontal crack at $x = (640, 0)$ m calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right): (a) horizontal displacement and (b) vertical displacement.
Figure 4-30: Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.3$ s calculated using OFE-TRII scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $CFL = 0.125$. Brighter color represents larger stress.
Figure 4-31: Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.5$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $\text{CFL} = 0.125$. Brighter color represents larger stress.
Figure 4-32: Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.7$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases $CFL = 0.125$. Brighter color represents larger stress.
Figure 4-33: Snapshots of von Mises stress distributions of the elastic medium with a horizontal crack at $t = 0.9$ s calculated using OFE-TRI1 scheme with the structured mesh (top) and the unstructured mesh (bottom); in both cases CFL = 0.125. Brighter color represents larger stress.
and in this problem

\[ \sigma = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{12} & C_{22} & 0 \\ 0 & 0 & C_{33} \end{bmatrix} \quad (4.10) \]

where \( C_{11} = 1, C_{12} = 0.25, C_{22} = 0.3 \) and \( C_{33} = 0.2 \). The anisotropic elastic medium has a unit mass density and a vertical point force is applied to it. We use a Ricker wavelet (3.28) with the magnitude \( 0.01, f_p = 5 \) and \( t_s = 0.25 \) for a vertical point force (in the \( y \) direction) and the largest wave propagation speed, here \( \sqrt{C_{11}/\rho} = 1 \), is chosen to establish the time step size.

The computational domain \( V = [-1, 1] \times [-1, 1] \) is considered and two receivers are located at 0.2 from the point force but in different directions, see Figure 4-34. For the solution, we use the OFE-TRI1 scheme with the structured and unstructured meshes with \( N = 60 \) (\( h = 0.0333 \)) and set CFL = 0.125.

Figure 4-35 shows the time history of displacements at the two receivers. Here,
Figure 4-35: Time history of horizontal displacements (left) and vertical displacements (right) of the anisotropy elastic medium calculated using OFE-TRI1 scheme (CFL = 0.125) with the structured mesh and the unstructured mesh: (a) at the receiver 1 and (b) at the receiver 2.
Figure 4-36: Snapshots of horizontal displacement distributions of the anisotropy elastic medium at various observation times calculated using OFE-TRI1 scheme with the structured mesh (left) and the unstructured mesh (right); in both cases CFL = 0.125.
Figure 4-37: Snapshots of vertical displacement distributions of the anisotropy elastic medium at various observation times calculated using OFE-TRII scheme with the structured mesh (left) and the unstructured mesh (right); in both cases $CFL = 0.125$. 
the OFE-TRI1 scheme with a fine mesh of 20,000 elements is used for the reference solution. Figures 4-36 and 4-37 provide snapshots of calculated displacements at various observation times.

4.3.8 Solution times

We list here the computational times required to solve the two-dimensional example problems, in order to give some insight into the computational effort using the OFE-TRI1 scheme. A laptop with a single core Intel 2.40 GHz CPU and 24 GB RAM was used for calculating the solutions.

Table 4.1 presents the CPU times spent to obtain the solutions of the problems when using the OFE-TRI1 scheme with CFL = 0.125. It is seen that most of the computational effort is expended in the time integrations, as expected, and this effort depends on the total number of degrees of freedom, the mean half-bandwidth of the stiffness and mass matrices and the number of time steps. Hence, the use of higher degree trigonometric polynomials could lead to a more efficient solution because for a required accuracy a coarser mesh can be used. However, we recall that this approach may lead to ill-conditioning of the coefficient matrices [33].

4.4 Concluding remarks

The objective in this chapter was to present an overlapping finite element enriched for transient wave propagation problems. We investigated the dispersion and dissipation properties and numerical anisotropy of the proposed overlapping finite element used with the Bathe time integration method in two-dimensional solutions and illustrated these characteristics in regular and irregular meshes through the solutions of numerical examples.

The important property of the present scheme is that the use of a smaller CFL number leads to a more accurate solution irrespective of the propagation direction. Hence multiple types of waves propagating in different directions at different speeds can be accurately calculated at the same time by using the largest wave speed to
Table 4.1: CPU times required to solve the two-dimensional example problems when using OFE-TRI1 scheme with CFL = 0.125.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Number of degrees of freedom</th>
<th>Mean half-bandwidth</th>
<th>CPU time (s) for constructing stiffness and mass matrices</th>
<th>CPU times [s] in the Bathe time integration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>For initial calculations</td>
<td>For calculations in each time step (Number of time steps)</td>
</tr>
<tr>
<td>Membrane</td>
<td>13,968</td>
<td>412</td>
<td>0.3</td>
<td>6.6</td>
</tr>
<tr>
<td>Semi-infinite elastic medium</td>
<td>100,815</td>
<td>1552</td>
<td>3.5</td>
<td>737.3</td>
</tr>
<tr>
<td>Membrane with holes</td>
<td>31,020</td>
<td>844</td>
<td>0.6</td>
<td>73.8</td>
</tr>
<tr>
<td>Bi-material membrane</td>
<td>10,152</td>
<td>356</td>
<td>0.2</td>
<td>4.0</td>
</tr>
<tr>
<td>Semi-infinite elastic medium with crack</td>
<td>62,076</td>
<td>1203</td>
<td>2.1</td>
<td>270.1</td>
</tr>
<tr>
<td>Anisotropic elastic medium</td>
<td>89,304</td>
<td>1452</td>
<td>3.0</td>
<td>557.7</td>
</tr>
</tbody>
</table>
select the CFL number. Furthermore, reasonably accurate solutions can be obtained for wave propagations in heterogeneous and/or anisotropic media.

These desirable solution characteristics have been observed even when using irregular meshes, which must often be employed in practical analysis. Given that the computational effort involved using the overlapping finite element is reasonable also when using irregular meshes, see Reference [50], the proposed scheme shows much promise, including for use in the new solution paradigm of References [51, 52] when transient wave propagation problems need be solved.

While the cost of the numerical integrations of the element matrices was here only a small fraction of the total solution cost (see Table 4.1), a study to identify an optimal scheme for the numerical integration of the element matrices would be valuable.
Chapter 5

Conclusions

The objective in this thesis is to develop novel solution schemes for the analysis of structural dynamics and wave propagation dynamics.

We proposed a novel extension of the Bathe subspace iteration method for the solution of the generalized eigenvalue problem in structural dynamics. Advances have been achieved by exploiting some turning vectors to replace current iteration vectors. The scheme speeds up the Bathe subspace iteration significantly, even a factor larger than 3 has been observed in some analyses, while keeping the attractive characteristics of being suitable for parallel processing.

In this thesis, we did not discuss and show solutions using parallel processing and instead focused on the basic improvements reached without parallel processing. However, the equations used show that the computations in the enriched subspace iteration method can be directly parallelized as in the original basic subspace iteration method [10]. While effective already, with the parallel processing the enriched subspace iteration method presented in the paper will likely provide a very efficient solution scheme.

Regarding structural dynamic analyses, we only focused on improving the solution scheme for the generalized eigenvalue problem. However, importance should be placed on developing novel finite elements so that the lowest modes of a physical system can be accurately and efficiently calculated. One of such elements is, for example, the overlapping element discussed in Reference [50] and extended in this thesis for wave
propagation problems, because it shows good performance for static problems. The detailed study of this overlapping element in structural dynamic problems, however, need be pursued in conjunction with the new solution paradigm of References [51,52].

The rest of this thesis was devoted to studying and developing effective solution schemes for the analysis of wave propagation dynamics.

In chapter 3, we presented a detailed study of the use of the method of finite spheres enriched by harmonic functions and the Bathe time integration method for wave propagation problems. The scheme can be used to solve wave propagation problems with monotonic convergence of the calculated solutions by decreasing the CFL number.

However, we only considered in this thesis uniformly distributed spheres of the same radius. For non-uniform discretizations using spheres, the numerical integration for the construction of the mass and stiffness matrices is very expensive. In practical wave propagation problems, irregular spatial discretizations need be used and the high computational cost involved using the method of finite spheres limits its wide use.

Finally, we developed novel overlapping finite elements for wave propagation problems in order to address this difficulty. Using the proposed scheme, the desirable solution characteristics mentioned above have been identified even with irregular meshes. Considering these attractive characteristics with the reasonable computational effort of the numerical integrations of the element matrices, the scheme is very effective even for heterogeneous and/or anisotropic wave propagation problems.

The field of research considered in this thesis, regarding wave propagation analyses, is very large and important, in particular when general three-dimensional solutions are considered. But we can see that the avenue of using overlapping finite elements in direct time integrations (implicit or explicit integrations, see e.g. Reference [54]) for the accurate solutions of general wave propagation problems, in linear and nonlinear analyses, with multiple waves traveling through anisotropic media is very promising.
Appendix A

Calculation of the amount of vector turning and the turning vectors

In this Appendix we provide an efficient algorithm to estimate the amount of turning and to calculate the turning vectors.

In iteration $k$, after evaluating $\overline{X}_{k+1}$ first calculate matrices $A_{k+1}, B_{k+1}$ and $C_{k+1}$, which are all of order $r_k \times r_k$, and $D_{k+1}$ of order $p_k \times r_k$

$$A_{k+1} = (X_k^a)^T \text{M} \overline{X}_{k+1}^a,$$  \hspace{1cm} (A.1)

$$B_{k+1} = (X_k^b)^T \text{M} \overline{X}_{k+1}^a,$$  \hspace{1cm} (A.2)

$$C_{k+1} = (\overline{X}_{k+1}^a)^T \text{M} \overline{X}_{k+1}^a,$$  \hspace{1cm} (A.3)

$$D_{k+1} = \Phi_k^T \text{M} \overline{X}_{k+1}^a$$  \hspace{1cm} (A.4)

and then construct $Y_k$ of order $n \times r_k$ by following the steps in part (a) and thereafter in part (b):

(a) Calculate in reverse order the amount of turning of the iteration vectors in $\overline{X}_{k+1}^a$, and choose the vectors for which the measure used is larger than the tolerance $\text{tolt}$, i.e., for $i = r_k, r_k - 1, \ldots, 2, 1$, with initially $t_k = 0$, first let

$$l = t_k + 1,$$  \hspace{1cm} (A.5)
\[ h_i = i \] (A.6)

and then calculate for \( m = 1 \ldots l, \)

\[ j = h_m, \] (A.7)

\[
 r_{lm} = C_{ij}^{(k+1)} - \left( a_i^{(k+1)} \right)^T a_j^{(k+1)} - \left( b_i^{(k+1)} \right)^T b_j^{(k+1)} - \left( d_i^{(k+1)} \right)^T d_j^{(k+1)} - \sum_{k=1}^{m-1} \frac{r_{ik} r_{mk}}{r_{kk}} \] (A.8)

where \( a_i^{(k+1)}, b_i^{(k+1)} \) and \( d_i^{(k+1)} \) are \( i \)th column vectors in \( A_{k+1}, B_{k+1} \) and \( D_{k+1}, \) respectively. If \( r_{il}/C_{ii}^{(k+1)} \leq \text{tolt} \) go to the next value of \( i. \) If \( r_{il}/C_{ii}^{(k+1)} > \text{tolt} \) we proceed as follows

\[ t_k = l, \] (A.9)

\[ v_{t_k} = x_i^{(k+1)} \] (A.10)

and now go to the next value of \( i. \)

(b) Next, let \( t_k \) be the last value reached in the above loop in part (a). We now construct \( Y_k = [x_1^{(k)}, \ldots, x_{r_k - t_k}^{(k)}, v_1, \ldots, v_{t_k}] \) and calculate for \( i = 1, \ldots, t_k, \)

\[ l = h_i, \] (A.11)

\[
 \tilde{x}_i = v_i - x_i^{(k+1)}a_i^{(k+1)} - \Phi_k d_i^{(k+1)} - \sum_{j=1}^{r_k - t_k} y_j^{(k)} b_j^{(k+1)} - \sum_{j=r_k - t_k + 1}^{r_k - t_k + i - 1} y_j^{(k)} \left( \left( y_j^{(k)} \right)^T M v_i \right), \] (A.12)

\[
 y_{r_k - t_k + i}^{(k)} = \frac{\tilde{x}_i}{\sqrt{\tilde{x}_i^T M \tilde{x}_i}}, \] (A.13)

Note that the matrix \( A_{k+1} \) and some parts of the matrix \( B_{k+1} \) can be used in the calculation of \( K_{k+1}, \) and the matrices \( C_{k+1} \) and \( D_{k+1} \) can be used in the
construction of $M_{k+1}$, i.e.,

$$
K_{k+1} = \begin{bmatrix}
\Phi_k^T \\
(X_{k+1})^T
\end{bmatrix} K \begin{bmatrix}
\Phi_k & X_{k+1}^a & Y_{k+1}
\end{bmatrix}
\begin{bmatrix}
\Lambda_k & \text{sym.} \\
0 & A_{k+1} \\
0 & Y_{k+1}^T M X_{k+1} & Y_{k+1}^T M Y_k
\end{bmatrix}
$$

where the entries in the first $r_k - t_k$ rows of the two matrices $B_{k+1}$ and $Y_k^T M X_{k+1}$ are the same, and

$$
M_{k+1} = \begin{bmatrix}
\Phi_k^T \\
(X_{k+1})^T
\end{bmatrix} M \begin{bmatrix}
\Phi_k & X_{k+1}^a & Y_{k+1}
\end{bmatrix}
\begin{bmatrix}
I & \text{sym.} \\
D_{k+1}^T & C_{k+1} \\
Y_{k+1}^T M \Phi_k & Y_{k+1}^T M X_{k+1} & Y_{k+1}^T M Y_{k+1}
\end{bmatrix}
$$
Bibliography


