

Abstract

LIM, KENG WIT. Absorbing Boundary Conditions for Corner Regions. (Under the direction of Murthy N. Guddati.)

This thesis contains the work that extends the continued fraction absorbing boundary conditions (CFABC's) to corner regions. We combine the ideas related to optimal discretization of perfectly matched layers (PML's) presented by Asvadurov et al. and the continued fraction expansion of one-way wave equations through finite element discretization given by Guddati to arrive at the new formulation of the CFABC and its extension to corner regions.

It will be shown that CFABC is a special case of the discrete PML, where CFABC is obtained from PML as a result of finite element discretization of the PML via one-point integration with purely imaginary element length. Discretization of the two-dimensional corner region is performed as a tensor product of the two CFABC one-dimensional discretization, which when viewed from the PML approach, is equivalent to discretizing the Helmholtz equation $-\mu(u_{xx} + u_{yy}) - \omega^2 \rho u = 0$ where pure imaginary stretching function and 1×1 integration are used. Extension to non-orthogonal corners by the use of parallelogram elements is also performed.

The result is that the dynamic stiffness matrix is independent of frequency ω , resulting in identical matrix entries in both the wave number-frequency and space-time domain. The dynamic stiffness matrix simply becomes the element stiffness matrix in the space-time domain. This allows for extremely easy finite element implementation for both transient and time-harmonic cases.

An implicit scheme is currently being used with the CFABC. A full explicit scheme is not possible since the mass matrix is singular (the absorbing boundary conditions do not contribute to the matrix \mathbf{M}). The second part of the thesis deals with the exploration of a pseudo-explicit time stepping scheme for the CFABC's. Currently, the implementation is limited to the case of straight computational boundaries. This initial work, together with the computer code developed, will provide a reference for the future improvement of the scheme.

ABSORBING BOUNDARY CONDITIONS FOR CORNER REGIONS

BY
KENG-WIT LIM

A THESIS SUBMITTED TO THE GRADUATE FACULTY OF
NORTH CAROLINA STATE UNIVERSITY
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

DEPARTMENT OF
CIVIL, CONSTRUCTION, AND ENVIRONMENTAL ENGINEERING

RALEIGH
OCTOBER 2003

APPROVED BY:

CHAIR OF ADVISORY COMMITTEE

Biography

Keng-Wit Lim was born in Malaysia in December 13, 1977, the son of Hee-Phang Lim and Kuan-Thye Loh. In July 1999, he was awarded his B. Eng. (Honours) degree in Civil Engineering from the University of Wales, Swansea, United Kingdom. He then returned to Malaysia where he worked as a Civil Engineer in both the construction and consultancy sectors. In January 2002, he entered The Graduate School at North Carolina State University.

Permanent Address: No. 35, Jalan SS19/5F
47500 Subang Jaya
Selangor Darul Ehsan
MALAYSIA

Acknowledgements

The author would like to express his immense gratitude and deep admiration for the supervisor of this thesis Dr. M. N. Guddati who has granted him the opportunity to learn so much through the accomplishment of the present work. The author would also like to thank Dr. T. Hassan and Dr. G. M. Kumar for their time serving as the members of the advisory committee. Their comments and suggestions are gratefully acknowledged.

Table of Contents

List of Figures	vi
List of Tables	viii
1 Introduction	1
1.1 Existing Absorbing Boundary Conditions	2
1.2 Objectives	4
1.3 Outline of Thesis	5
2 Background: CFABC and PML	6
2.1 CFABC: Original Formulation	6
2.1.1 Mixed Form of Absorbing Boundary Conditions	8
2.2 Perfectly Matched Layers	11
3 CFABC: A New Formulation and Approach to Solving	13
3.1 New Formulation of CFABC	13
3.1.1 Reflection Coefficient	20
3.2 Formulation of Corner Absorbing Boundaries	23
3.3 Finite Element Discretization and Time Stepping	27
3.4 Implicit Procedure	29
3.5 Pseudo-Explicit Procedure	30
3.5.1 Non-diagonal C Matrix; Pseudo-Explicit Procedure	32
3.5.2 Presence of Corner Absorbers	36
3.5.3 Computational Performance	38
3.5.4 A Note on Stability Condition	39
3.5.5 Local Truncation Error Analysis	40
4 Numerical Examples	43
4.1 Computational Domains With Corners	43
4.1.1 Explosive Source In An Unbounded Domain	43

4.1.2	Effects of Integration Order	46
4.1.3	Truncated Domain With Oblique Corners	50
4.1.4	Layered Media	52
4.1.5	High Aspect Ratio Computational Domain	55
4.2	Pseudo-Explicit and Explicit Computations	58
4.2.1	One-dimensional Case	58
4.2.2	Two-dimensional Case	64
5	Conclusions	71
5.1	Summary	71
5.2	Future Work	72
	List of References	73
	A Derivation of The Dynamic Stiffness Matrix For Corner ABC's	78
	B XFEP++: Documentation and User Manual	81
B.1	Introduction	81
B.2	Program/Feature Limitations	81
B.3	Caveats and Important Notes	82
B.4	Class Files Description	82
B.5	Finite Elements and Absorbing Elements	83
B.6	Element Sets	84
B.7	Nodes and Node Set	85
B.8	Load Class	86
B.9	Input Manual	87

List of Figures

1.1	CFABC: The new picture.	4
3.1	Replacement of full-space with half-space augmented with ABC's. . .	15
3.2	Replacement of the right half-space with a layer plus another right half-space.	16
3.3	Half-space connected to an infinite number of one-point integrated fi- nite elements/layers.	18
3.4	Truncation of infinite number of auxiliary variables.	20
3.5	Parallelogram absorbing elements at the corner.	24
3.6	Parallelogram element.	25
3.7	Typical discretization for the interior (finite elements) and for the edge and corner absorbers.	27
3.8	Nodes contributing to a tridiagonal block.	34
3.9	Location of Gauss points for the implicit method.	35
3.10	Location of Gauss points for the pseudo-explicit method.	35
3.11	General structure of \mathbf{M}^{eff} for $\theta_{2k} \neq 0^\circ$ (without corner absorbers). The tridiagonal blocks are due to the lumping of the damping terms in the direction of the boundary while the diagonal submatrix is due to the mass lumping in the interior.	36
3.12	Nodes associated with system 1 and 2.	38
3.13	Comparison of reflection coefficients at different values of θ_{2k}	39
4.1	Anti-plane shear wave propagation in an unbounded domain.	44
4.2	Anti-plane shear wave propagation in an unbounded domain (cont'd).	45
4.3	The variation of velocity at Node 302.	46
4.4	The variation of velocity at Node 20300.	47
4.5	Norm of the velocity at $t = 7.5$ ms.	48
4.6	Norm of the velocity at $t = 15.0$ ms.	48
4.7	Norm of the velocity at $t = 19.5$ ms.	49
4.8	Problem schematic: domain with oblique corners.	50

4.9	Anti-plane shear wave propagation in domain with oblique corners. . .	51
4.10	Anti-plane shear wave propagation in domain with oblique corners (cont'd).	52
4.11	Problem schematic: layered media.	53
4.12	Anti-plane shear wave propagation in layered media.	54
4.13	Problem schematic: domain with high aspect ratio.	55
4.14	High aspect ratio computational domain (3 CFABC layers).	56
4.15	High aspect ratio computational domain (9 CFABC layers).	57
4.16	Bar on elastic foundation problem.	58
4.17	Displacement history obtained using $\Delta t = 0.96$	60
4.18	Displacement history obtained using $\Delta t = 0.0001$	60
4.19	Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$, $\text{div} = 1.35$	62
4.20	Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 18^\circ, 36^\circ, 54^\circ, 72^\circ$, $\text{div} = 1.35$	62
4.21	Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$, $\text{div} = 1.35$	63
4.22	Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 18^\circ, 36^\circ, 54^\circ, 72^\circ$, $\text{div} = 1.35$	63
4.23	Wave propagation in an acoustic channel around a cavity.	64
4.24	Finite element mesh for the channel problem.	65
4.25	Explicit method; $\theta_{2k} = 0^\circ, 0^\circ, 0^\circ$: Displacement history at Node 1 (Point <i>A</i>) using $\Delta t = 0.07$	67
4.26	Explicit method; $\theta_{2k} = 0^\circ, 0^\circ, 0^\circ$: Displacement history at Node 1 (Point <i>A</i>) using $\Delta t = 0.001$	67
4.27	Displacement history at Node 31 (Point <i>B</i>) using $\Delta t = 0.07$	68
4.28	Displacement history at Node 31 (Point <i>B</i>) using $\Delta t = 0.001$	68
4.29	Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 1 (Point <i>A</i>) using $\Delta t = 0.07$	69
4.30	Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 1 (Point <i>A</i>) using $\Delta t = 0.001$	69
4.31	Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 31 (Point <i>B</i>) using $\Delta t = 0.07$	70
4.32	Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 31 (Point <i>B</i>) using $\Delta t = 0.001$	70
B.1	Inheritance for the declaration of acoustic and absorbing elements. . .	84
B.2	Inheritance for the declaration of <i>sets</i> of acoustic and absorbing elements. . .	85

List of Tables

B.1	Title block.	88
B.2	Linear solution control block.	88
B.3	Time stepping information block.	89
B.4	Time history definition block.	89
B.5	Material definition block.	89
B.6	Node definition block.	90
B.7	Restraints data block.	90
B.8	Nodal loads data block.	90
B.9	Finite element data block.	91
B.10	Straight continued fraction absorbers data block.	91
B.11	Nodal loads data block.	92

Chapter 1

Introduction

Analysis of unbounded media for arbitrary types of dynamic loading is of prime interest in many fields of engineering such as acoustics, electromagnetics, seismology, and soil-structure interaction. Hyperbolic problems in which the information of the solution moves at finite speed are limited in duration by the return of outwardly propagating features of the solution after interacting with the boundaries of the computational domain.

The simplest way to model unbounded domain is to compute the solution on a domain sufficiently large so that the simulation ends before the return of reflections from the boundary. This leads to significant increase in computational costs in terms of time and storage. Alternatively, the domain can be simply truncated. It is then necessary to find the appropriate boundary conditions that make the artificially created boundary transparent. If all the forces are being applied in the bounded domain, then the solution near the boundary consists of outgoing wave motion, so the boundary conditions should mimic the outward radiation of energy. Absorbing boundary conditions are the boundary conditions that generate little or no reflection.

1.1 Existing Absorbing Boundary Conditions

Absorbing boundary conditions (ABC's) can be broadly classified into two types: differential-equation-based and material-based [34]. Differential-equation-based ABC's can be obtained by factoring the wave equation and allowing a solution which permits only outgoing waves. Material-based ABC's, on the other hand, are realized by surrounding the computational domain with a lossy material that dampens the outgoing waves.

Differential-based ABC's were reported by Engquist and Majda [14, 15], and Lindman [30], who proposed a series of local boundary conditions with increasing order. Their boundary conditions is based on approximating the one-way wave equation in the Fourier domain by rational approximations. Subsequent improvements on these techniques were given by Keys [28] and Higdon [23, 24] in the 1980's, who independently proposed an improved boundary condition based on Engquist-Majda's ABC, which eliminates reflections for arbitrary angles of incidence. Higdon's ABC uses the discretized form of one, or the product of more than one, one-way boundary operator. All these boundary conditions have the same drawback in that their higher-order versions involve high-order derivatives, making it very difficult to implement them in conventional finite element and finite difference settings.

In the context of finite elements, higher order absorbing boundary conditions were not available until the 1990's when Kallivokas and Bielak [27] proposed an implementation that fits well in the finite element settings. However, this boundary condition is only limited to second order. The multi-stage absorbing boundary conditions developed by Safjan [33] do not have this limitation but require that several initial-boundary value problems be solved.

The continued fraction absorbing boundary condition (CFABC), introduced by Guddati and Tassoulas [19], is based on the idea of approximating the dispersion relationship of the one-way wave equation by employing rational approximations and recursive continued fractions, leading to the introduction of auxiliary layers of degrees

of freedom next to the boundary. CFABC's allow, for the first time, a practical way of using arbitrarily high-order ABC's in standard C^0 finite element and finite difference settings.

CFABC is a generalization of the Engquist-Majda boundary conditions, providing accelerated convergence properties using an even-ordered continued fraction expansion of the square-root operator. It is also equivalent to the multi-directional ABC proposed by Higdon. Indeed, it can be shown that the n -th order continued fraction boundary condition is exactly the same as the n -th-order multi-directional boundary conditions with the same choice of angles [19]. The difference between the CFABC and the Engquist-Majda and Lindman ABC's lies in the ease of implementation and that the order of the ABC can be increased by simply adding layers of nodes and elements that can be viewed topologically as a finite element mesh.

The perfectly matched layer (PML) technique, proposed by Bérenger [4] in 1994, can be considered as a material absorber. Traveling waves entering the PML becomes evanescent to the extent that reflected waves from the truncated end of the PML becomes negligible. Different interpretations of the PML were soon given by various workers in the field [8, 9, 32]. In particular, the coordinate stretching interpretation of the PML, given by Chew and Weedon [8, 9], is the one most widely used in the PML literature. Various extensions and generalizations of the PML were subsequently given [5, 6, 13, 16, 29, 31, 36], and research in the optimization of the PML parameters remains active till the present.

Recently (2003), Asvadurov et al. [1] have shown that CFABC is in fact a special and optimal case of discrete PML. It is found that the minimal impedance error (reflection coefficient) can be achieved using pure imaginary coordinate stretching, and that their solution exhibits much smaller reflection coefficients compared to examples of optimized PML's (for the same numbers of discrete layers) known from the literature [11]. From a PML point of view, the addition of auxiliary layers in CFABC is equivalent to a simple mesh extension technique into imaginary space. Figure 1.1 shows where CFABC stands when compared with the differential-based

ABC's described above and PML.

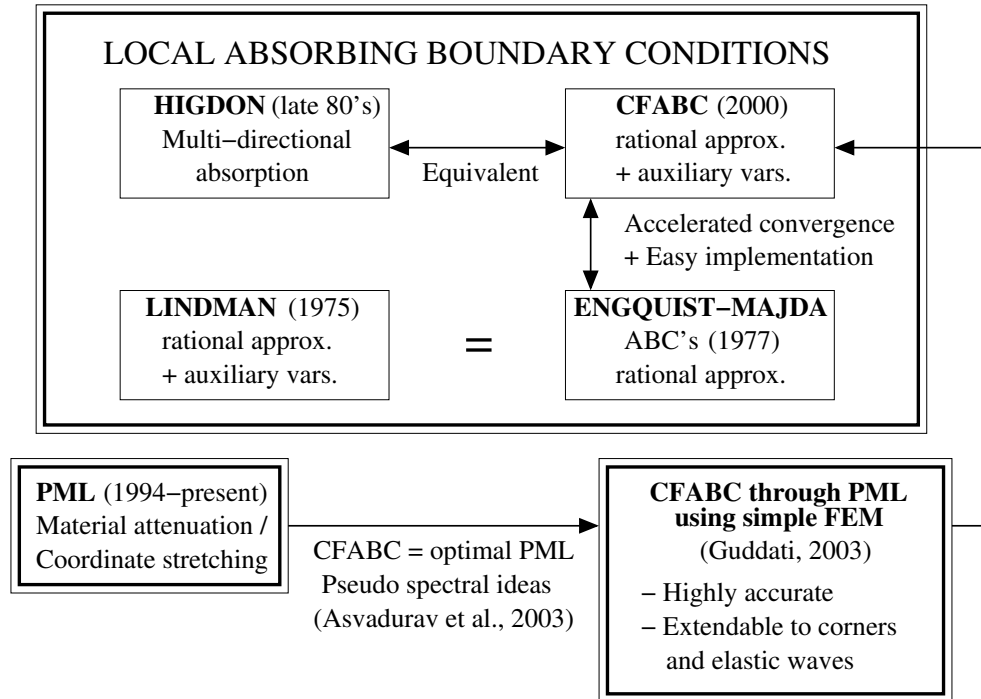


Figure 1.1: CFABC: The new picture.

Here, we combine the ideas related to the optimal discretization of PML's in [1] and the continued fraction expansion of one-way wave equations through finite element discretization in [20] to arrive at the new formulation of the CFABC and its extension to corner regions.

1.2 Objectives

The objectives of the research reported herein are:

- To extend the CFABC to corner regions to allow for transient analysis of wave propagation in fully unbounded media.
- To devise an explicit scheme for the CFABC.

- To study the accuracy and local truncation errors of the system upon the augmentation of the interior with the CFABC's.

1.3 Outline of Thesis

This thesis is organized in five chapters.

Chapter 2 reviews the original derivation of the CFABC. Since the CFABC can be considered as a special case of the PML, a brief description of the PML is also provided. The new derivation of the CFABC and the corner absorbing boundaries are provided in detail in Chapter 3. Chapter 4 describes the results and implementation of an explicit procedure for the CFABC. Finally, Chapter 5 summarizes the conclusions of the investigation and provides some recommendations for future research.

Chapter 2

Background: CFABC and PML

We will first review the original formulation of the CFABC, which will make the new formulation clearer. CFABC can be considered as a special case of PML, and thus it is also helpful to briefly discuss the basic idea of PML.

2.1 CFABC: Original Formulation

We start by considering the scalar wave equation

$$-\mu(u_{xx} + u_{zz}) + \rho u_{tt} = 0 \quad (2.1)$$

The solution to the above equation has the form

$$u = e^{i(kx + ly - \omega t)} \quad (2.2)$$

where k and l are the wave numbers in x and y directions, respectively, and ω is the frequency. Then we have the following duality between the space-time and the Fourier (wave number-frequency) domains:

$$ik \leftrightarrow \frac{\partial}{\partial x}; \quad il \leftrightarrow \frac{\partial}{\partial z}; \quad -i\omega \leftrightarrow \frac{\partial}{\partial t} \quad (2.3)$$

Using the above solution in (2.1) the dispersion relationship

$$k^2 + l^2 = \left(\frac{\omega}{c}\right)^2 \quad (2.4)$$

is obtained, where $c = \sqrt{\mu/\rho}$ is the wave speed. The discussion here is limited to absorbing boundary conditions for straight computational boundaries. We assume that the outward normal is in the positive x direction. Consider an incident wave travelling in the positive x direction. Ideally, the absorbing boundary condition would annihilate all the reflections from the boundary, indicating that the waves should have positive phase velocity in the x direction. This means that the exact solution will contain positive k . The dispersion relationship is rearranged to give:

$$k = \frac{\omega}{c} \sqrt{1 - \left(\frac{l}{\omega/c}\right)^2}$$

The exact boundary condition for the one-way wave equation can then be written as

$$\left(ik - i\frac{\omega}{c} \sqrt{1 - \beta^2}\right) u = 0 \quad (2.5)$$

where $\beta = \frac{il}{i\omega/c}$. For travelling waves β can be viewed as $\sin(\theta)$, where θ is the angle of incidence with respect to the direction normal to the boundary. Although the one-way equation represents the dispersion relationship for exact absorbing boundary condition in the Fourier domain, it cannot be implemented in space-time domain. This is because the inverse Fourier transform of the square-root term results in a pseudo-differential operator that cannot be easily implemented numerically.

To obtain manageable boundary conditions consisting of differential operators, a rational approximation can be used for $\sqrt{1 - \beta^2}$. Then the approximate boundary condition can be written as

$$\left(ik - i\frac{\omega}{c} q_n\right) u = 0 \quad (2.6)$$

where q_n is the n -th order approximation of $\sqrt{1 - \beta^2}$.

Engquist and Majda [15] proposed a series of approximations based on the truncated continued fraction expansions of $\sqrt{1 - \beta^2}$ as follows:

$$q_1 = 1 \quad (2.7)$$

$$q_2 = 1 - \frac{\beta^2}{2} \quad (2.8)$$

$$q_{n+1} = 1 - \frac{\beta^2}{1 + q_n} \quad (2.9)$$

For the continued fraction boundary conditions, a modified continued fraction expansion as follows:

$$q_1 = \cos(\theta_1) \quad (2.10)$$

$$q_2 = \cos(\theta_2) - \frac{\cos^2(\theta_2) - (1 - \beta^2)}{\cos^2(\theta_2) + q_1} \quad (2.11)$$

$$q_{n+1} = \cos(\theta_{n+1}) - \frac{\cos^2(\theta_{n+1}) - (1 - \beta^2)}{\cos^2(\theta_{n+1}) + q_n} \quad (2.12)$$

where the angles θ_n are chosen to accelerate the convergence of the iteration based on the directionality of the wave propagation.

2.1.1 Mixed Form of Absorbing Boundary Conditions

In order to eliminate the higher order spatial and temporal derivatives, a mixed form is presented by defining auxiliary variables. The implementation is restricted to even-order approximations with $\theta_{2k-1} = \theta_{2k}$. The iterative process for even-order approximations takes the form:

$$q_2 = a_1 \quad (2.13)$$

$$q_{2n+2} = a_{n+1} - \frac{b_{n+1}^2}{a_{n+1} + q_{2n}}, \quad n = 1, 2, 3, \dots \quad (2.14)$$

where

$$a_n = \frac{\cos(\theta_{2n})}{2} + \frac{1 - \beta^2}{2 \cos(\theta_{2n})} \quad (2.15)$$

$$b_n = -\frac{\cos(\theta_{2n})}{2} + \frac{1 - \beta^2}{2 \cos(\theta_{2n})} \quad (2.16)$$

Then the one-way wave equation can be written as

$$i k - i \frac{\omega}{c} q_{2n} = 0, \quad n = 1, 2, 3, \dots \quad (2.17)$$

and the boundary condition is expressed as

$$\begin{aligned} \left(i k - i \frac{\omega}{c} q_{2n} \right) u &= 0, \quad n = 1, 2, 3, \dots \\ \left(-\frac{i k}{i \omega/c} + q_{2n} \right) u &= 0 \end{aligned} \quad (2.18)$$

Using (2.14), the above boundary condition can be rewritten as

$$\begin{aligned} \left(-\frac{i k}{i \omega/c} + a_n - b_n \frac{b_n}{a_n + q_{2n-2}} \right) u &= 0, \quad n = 0, 1, 2, 3, \dots \\ \left(-\frac{i k}{i \omega/c} + a_n \right) u + b_n \left(-\frac{b_n}{a_n + q_{2n-2}} \right) u &= 0 \end{aligned} \quad (2.19)$$

From the second term in (2.19), the *auxiliary variable* u_1 is defined as

$$u_1 = \left(-\frac{b_n}{a_n + q_{2n-2}} \right) u \quad (2.20)$$

Equation (2.20) is rearranged to give

$$b_n u + (a_n + q_{2n-2}) u_1 = 0 \quad (2.21)$$

Then (2.19) can be rewritten as

$$\left(-\frac{i k}{i \omega/c} + a_n \right) u + b_n u_1 = 0 \quad (2.22)$$

Equations. (2.21) and (2.22) can be written in matrix form as

$$\begin{bmatrix} -\frac{ik}{i\omega/c} + a_n & b_n \\ b_n & a_n + q_{2n-2} \end{bmatrix} \begin{Bmatrix} u \\ u_1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \quad (2.23)$$

The above procedure can be performed recursively to define additional auxiliary variables u_2, u_3, \dots, u_{n-1} to obtain the following tridiagonal matrix

$$\begin{bmatrix} -\frac{ik}{i\omega/c} + a_n & b_n & & & & \\ b_n & a_n + a_{n-1} & b_{n-1} & & & \\ & b_{n-1} & \ddots & \ddots & & \\ & & \ddots & \ddots & b_2 & \\ & & & & b_2 & a_2 + a_1 \end{bmatrix} \begin{Bmatrix} u \\ u_1 \\ u_2 \\ \vdots \\ u_{n-1} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \quad (2.24)$$

Using Eqs. (2.15) and (2.16) and noting that $\beta = \frac{il}{i\omega/c}$ and $c = \sqrt{\frac{\mu}{\rho}}$ the above equation can be rewritten as:

$$\begin{Bmatrix} ik\sqrt{\mu}u \\ 0 \\ \vdots \\ 0 \end{Bmatrix} - i\omega\sqrt{\rho}(\mathbf{A} + \mathbf{B}) \begin{Bmatrix} u \\ u_1 \\ \vdots \\ u_{n-1} \end{Bmatrix} + \frac{(il)^2\mu}{i\omega\sqrt{\rho}}\mathbf{B} \begin{Bmatrix} u \\ u_1 \\ \vdots \\ u_{n-1} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \quad (2.25)$$

The matrices \mathbf{A} and \mathbf{B} are given by

$$\mathbf{A} = \frac{1}{2} \begin{bmatrix} c_n & -c_n & & & & \\ -c_n & c_n + c_{n-1} & -c_{n-1} & & & \\ & -c_{n-1} & \ddots & \ddots & & \\ & & \ddots & \ddots & -c_2 & \\ & & & -c_2 & c_2 + c_1 & \end{bmatrix} \quad (2.26)$$

this truncation, the boundary condition is exact. With truncation, however, the PML is only approximate; it does have numerical reflection.

It is known that Berenger's PML can be obtained using complex coordinate stretching [9]:

$$\tilde{x}_j = \int_0^{x_j} s_j(\tau) d\tau \quad (2.30)$$

where $j = 1, 2$ for the two-dimensional case, which implies

$$d\tilde{x}_j = s_j(\tau) dx_j \quad (2.31)$$

In general, the stretching function is of the complex form

$$\kappa_j + i \frac{\sigma_j}{\omega} \quad (2.32)$$

where κ_j and σ_j are some non-negative functions, and σ_j is usually termed as the conductivity or attenuation function.

We have, for the case where $\kappa_j = 1$, the solution in the PML given by

$$u(x) = e^{ikx} e^{-k \int_0^x \frac{\sigma(\tau)}{\omega} d\tau} \quad (2.33)$$

Upon the truncation at $x = \delta$ and the Dirichlet boundary condition applied, the reflection coefficient is given by

$$R = e^{2ik \int_0^x \frac{1+\sigma(\tau)}{\omega} d\tau} \quad (2.34)$$

The second exponent in (2.33) provides the exponential decay of the wave in the PML. One can observe that the reflection coefficient is dependent on the incident angle since $k = \frac{\omega}{c} \cos \theta$, where θ is the incidence angle. One might infer from (2.34) that, for a given incidence angle, a large value of σ will make R small. However, this consideration cannot be applied to discretized PML due to numerical dispersion, which is frequency dependent and increases with the increase of σ [1].

Chapter 3

CFABC: A New Formulation and Approach to Solving

3.1 New Formulation of CFABC

Here, a new formulation of the CFABC is presented. We shall henceforth use the same name for a function and its Fourier transform; the use of the time, frequency or wave number domain will always be clear from the context. The discussion here is limited to CFABC's for straight computational boundaries in the vertical z direction. We start by taking the Fourier transform in z and t of (2.1), giving

$$-u_{xx} + \left(l^2 - \left(\frac{\omega}{c} \right)^2 \right) u = 0 \quad (3.1)$$

The above equation, using (2.4), can be written as

$$-u_{xx} - k^2 u = 0 \quad (3.2)$$

In the variational formulation, (3.2) is multiplied by the virtual displacement δu^T and integrated along the length L of the element which results in

$$\begin{aligned} \int_0^L (\delta u)^T [-u_{xx} - k^2 u] dx &= 0 \\ - \int_0^L (\delta u)^T u_{xx} dx - \int_0^L k^2 (\delta u)^T u dx &= 0 \end{aligned} \quad (3.3)$$

Using integration by parts on the first term, the following weak form is obtained

$$[-u_x (\delta u)^T]_0^L + \int_0^L (\delta u_x)^T u_x dx - \int_0^L k^2 (\delta u)^T u dx = 0 \quad (3.4)$$

The element stiffness and mass matrices in Fourier domain will be obtained from the second and third terms of (3.4), respectively. Let's consider that the displacement in an element is to be approximated using linear shape functions as

$$u = \begin{bmatrix} 1 - \frac{x}{L} & \frac{x}{L} \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \end{Bmatrix} \quad (3.5)$$

From the second term of (3.4), the element stiffness matrix in Fourier domain is obtained as

$$\mathbf{k}^e = \int_0^L \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L} \end{bmatrix} \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} dx \quad (3.6)$$

where the superscript e stands for element. Employing one-point integration, the element stiffness matrix in Fourier domain turns out to be

$$\mathbf{k}^e = \frac{1}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (3.7)$$

Similarly, using the third term of (3.4), the element mass matrix in Fourier domain is obtained as

$$\mathbf{m}^e = \frac{L}{4} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (3.8)$$

From (3.7) and (3.8), the following relationship between the displacements and forces are obtained

$$(\mathbf{k}^e - k^2 \mathbf{m}^e) \mathbf{u}^e = \mathbf{f}^e \quad (3.9)$$

which can be written in matrix form as

$$\begin{bmatrix} \frac{1}{L} - \frac{k^2 L}{4} & -\frac{1}{L} - \frac{k^2 L}{4} \\ -\frac{1}{L} - \frac{k^2 L}{4} & \frac{1}{L} - \frac{k^2 L}{4} \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \end{Bmatrix} = \begin{Bmatrix} f_1^e \\ f_2^e \end{Bmatrix} \quad (3.10)$$

where $(\mathbf{k}^e - k^2 \mathbf{m}^e)$ is the dynamic stiffness matrix and u_1^e, u_2^e are the local node displacements.

Now, consider the following situation: we want to replace the full-space with a left half-space augmented with ABC's on the right. This is illustrated in Fig. 3.1. Due to the Fourier transform in z , the dimension of the problem reduces from two to one, in x . To obtain the ABC's we first divide the full-space into two half-spaces, one on the left and the other on the right. Then, we replace the right half-space by a layer corresponding to a one-point integrated finite element *plus* another right half-space as shown in Fig. 3.2.

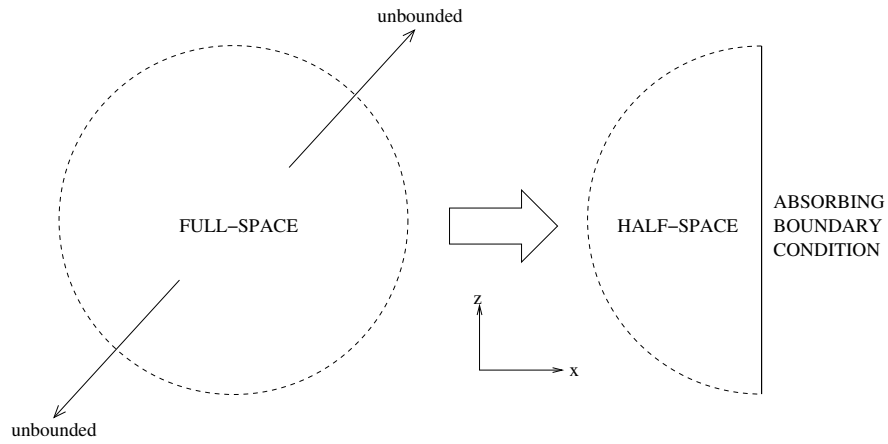


Figure 3.1: Replacement of full-space with half-space augmented with ABC's.

Denoting the displacements on lines 0 and 1 as u_0 and u_1 , respectively, we can write the stiffness relation of the layer as (from (3.10))

$$\begin{bmatrix} \frac{1}{L} - \frac{k^2 L}{4} & -\frac{1}{L} - \frac{k^2 L}{4} \\ -\frac{1}{L} - \frac{k^2 L}{4} & \frac{1}{L} - \frac{k^2 L}{4} \end{bmatrix} \begin{Bmatrix} u_0 \\ u_1 \end{Bmatrix} = \begin{Bmatrix} f_0 \\ f_1 \end{Bmatrix} \quad (3.11)$$

In the above f_0 is the force from the left half-space, while f_1 is the force from the right half-space. The force from the right half-space is given by

$$f_1 = i k u_1 \quad (3.12)$$

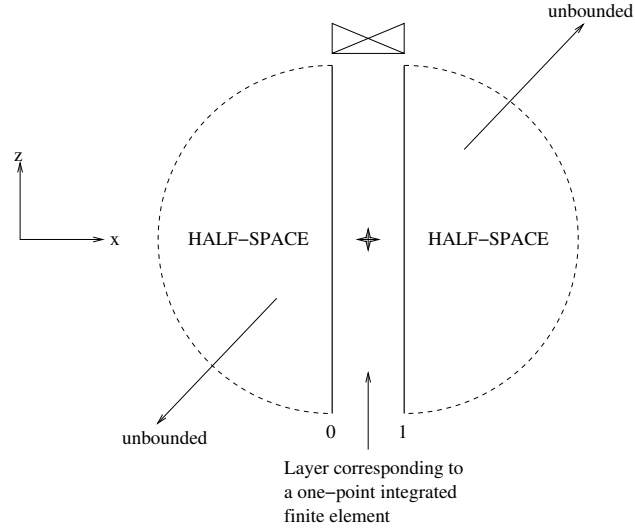


Figure 3.2: Replacement of the right half-space with a layer plus another right half-space.

Substituting (3.12) into (3.11), we obtain the stiffness relation of the layer-right half-space assembly as

$$\begin{bmatrix} \frac{1}{L} - \frac{k^2 L}{4} & -\frac{1}{L} - \frac{k^2 L}{4} \\ -\frac{1}{L} - \frac{k^2 L}{4} & \frac{1}{L} - \frac{k^2 L}{4} - i k \end{bmatrix} \begin{Bmatrix} u_0 \\ u_1 \end{Bmatrix} = \begin{Bmatrix} f_0 \\ 0 \end{Bmatrix} \quad (3.13)$$

By the condensation process, u_1 is eliminated and we obtain

$$f_0 = -i k u_0 \quad (3.14)$$

which is the exact force from the left half-space. We have thus shown that the right-hand side can be replaced by a one-point integrated layer and another half-space without introducing any error. This process can be repeated recursively and the original half-space can be replaced by an infinite number of one-point integrated layers as shown next.

Scaling (3.13) by $\frac{1}{-i\omega/c}$, we have

$$\begin{bmatrix} -\frac{1}{iL\omega/c} + \frac{k^2 L}{4i\omega/c} & \frac{1}{i\omega/cL} + \frac{k^2 L}{4i\omega/c} \\ \frac{1}{i\omega/cL} + \frac{k^2 L}{4i\omega/c} & -\frac{1}{i\omega/cL} + \frac{k^2 L}{4i\omega/c} + \frac{ik}{i\omega/c} \end{bmatrix} \begin{Bmatrix} u_0 \\ u_1 \end{Bmatrix} = \begin{Bmatrix} \frac{f_0}{-i\omega/c} \\ 0 \end{Bmatrix} \quad (3.15)$$

By eliminating u_1 and using (3.14), the following relationship is obtained

$$a - \frac{b^2}{a + \frac{ik}{i\omega/c}} = \frac{ik}{i\omega/c} \quad (3.16)$$

where

$$a = -\frac{1}{iL\omega/c} + \frac{k^2 L}{4i\omega/c} \quad (3.17)$$

$$b = \frac{1}{iL\omega/c} + \frac{k^2 L}{4i\omega/c} \quad (3.18)$$

which can be rearranged to give

$$\frac{ik}{i\omega/c} = \sqrt{a^2 - b^2} \quad (3.19)$$

To approximate the square-root term, the following iteration scheme is employed

$$\begin{aligned} \frac{K_1}{i\omega/c} &= a_1 \\ \frac{K_n}{i\omega/c} &= a_n - \frac{b_n^2}{a_n + \frac{K_{n-1}}{i\omega/c}}, \quad n = 2, 3, 4, \dots \end{aligned} \quad (3.20)$$

where K_n is the even-ordered approximation of the half-space stiffness $K_{\text{exact}} = ik$. Defining $q_{2n} = \frac{K_n}{i\omega/c}$, the above iteration scheme is rewritten as

$$\begin{aligned} q_2 &= a_1 \\ q_{2n} &= a_n - \frac{b_n^2}{a_n + q_{2n-2}}, \quad n = 2, 3, 4, \dots \end{aligned} \quad (3.21)$$

In the above, a_n and b_n are given by

$$\begin{aligned} a_n &= -\frac{1}{iL_n\omega/c} + \frac{k^2 L_n}{4i\omega/c} \\ b_n &= \frac{1}{iL_n\omega/c} + \frac{k^2 L_n}{4i\omega/c} \end{aligned} \quad (3.22)$$

Notice that by using the notation provided in (3.21) and (3.22), (3.15) can be rewritten in the same form as (2.23). Therefore the use of the iteration scheme in (3.21), with $n \rightarrow \infty$, is equivalent to repeating recursively the process of replacing the right half-space in Fig. 3.2 by another layer plus another right half-space infinite number of times. This results in an assemblage of an infinite number of one-point integrated finite elements or layers to the right as shown in Fig. 3.3.

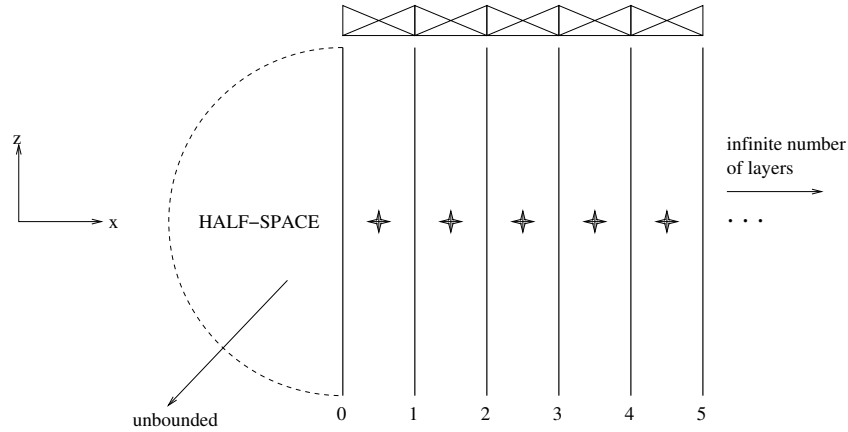


Figure 3.3: Half-space connected to an infinite number of one-point integrated finite elements/layers.

This results in the tridiagonal matrix in (2.24). We observe that the half-space stiffness is exactly represented when we have the wave numbers

$$k = \frac{2i}{L_n}, \quad n = 1, 2, 3, \dots \quad (3.23)$$

where L_n is the length of an element. With this observation, we choose the *imaginary* length

$$L_n = \frac{2i}{\omega/c_n} = \frac{2i}{(\omega/c) \cos \theta_{2n}} \quad (3.24)$$

where c_n are chosen phase velocities for perfect absorption of waves having different

incident angles. The following are then obtained

$$\begin{aligned} a_n &= \frac{\cos(\theta_{2n})}{2} + \frac{(k/(\omega/c))^2}{2 \cos(\theta_{2n})} \\ b_n &= -\frac{\cos(\theta_{2n})}{2} + \frac{(k/(\omega/c))^2}{2 \cos(\theta_{2n})} \end{aligned} \quad (3.25)$$

From the dispersion relationship in (2.4), we have

$$\left(\frac{k}{\omega/c}\right)^2 = 1 - \left(\frac{il}{i\omega/c}\right)^2 = 1 - \beta^2 \quad (3.26)$$

where $\beta = \frac{il}{i\omega/c}$ which, for travelling waves, can be viewed as $\sin(\theta)$, where θ is the angle of incidence with respect to the direction normal to the boundary. Then, a_n and b_n are rewritten as

$$\begin{aligned} a_n &= \frac{\cos(\theta_{2n})}{2} + \frac{1 - \beta^2}{2 \cos(\theta_{2n})} \\ b_n &= -\frac{\cos(\theta_{2n})}{2} + \frac{1 - \beta^2}{2 \cos(\theta_{2n})} \end{aligned} \quad (3.27)$$

Equations (3.21) and (3.27) are exactly the ones used in the approximation of the square-root term $\sqrt{1 - \beta^2}$ for the CFABC presented in Section 2.1. We note here that the use of the imaginary length in (3.24) is equivalent to using pure imaginary stretching for discretization in PML's. Thus CFABC is a special case of discrete PML. With the choice of pure imaginary stretching function and one-point integration, the notion of PML length or thickness becomes non-existent, and the discretization becomes independent of ω and depends only on the incidence angle, i.e., the discretization reduces to the problem of rational approximation of the square-root on $[0, 1]$, ubiquitous in formulations of differential-equation-based ABC's.

Finally, to make the CFABC amenable to computation, the infinite number of auxiliary variables has to be truncated as shown in Fig. 3.4. Due to the truncation, numerical reflections will be generated. The reflections can be quantified using the reflection coefficient as described in the next subsection.

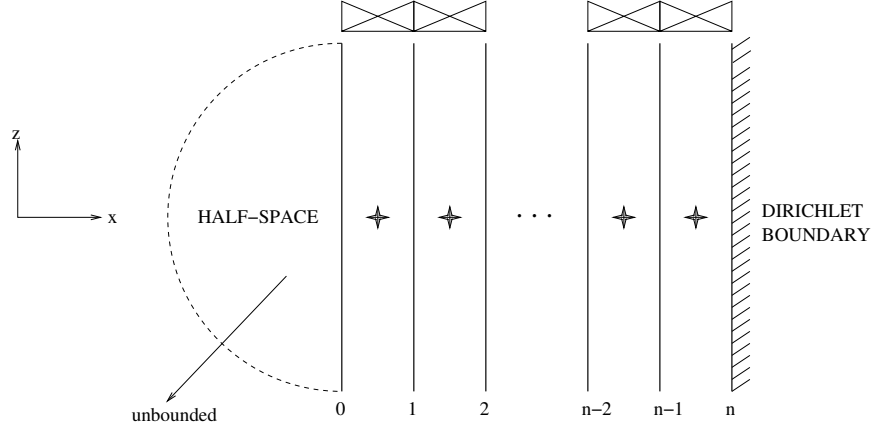


Figure 3.4: Truncation of infinite number of auxiliary variables.

We have thus shown that this new formulation results is equivalent to the original CFABC formulation presented previously. This formulation is better than the original one because it provides a convenient way to formulate the corner absorbers as will be discussed in detail in Section 3.2.

3.1.1 Reflection Coefficient

In this section, the derivation of the reflection coefficient, which gives an indication of the performance of the CFABC, is described. It will also be shown that the CFABC simulates the half-space in a consistent manner. It can be shown that the reflection coefficient is given by

$$R_{2n} = \frac{K_{\text{exact}} - K_n}{K_{\text{exact}} + K_n} \quad (3.28)$$

Substituting for K_n in (3.28) using (3.20), we obtain

$$R_{2n} = \frac{K_{\text{exact}} + \frac{i\omega}{c} \left(a_n - \frac{b_n^2}{a_n + \frac{K_{n-1}}{-i\omega/c}} \right)}{K_{\text{exact}} - \frac{i\omega}{c} \left(a_n - \frac{b_n^2}{a_n + \frac{K_{n-1}}{-i\omega/c}} \right)} \quad (3.29)$$

Using (3.22), the following relationship can be obtained

$$a_n^2 - b_n^2 = \frac{(ik)^2}{(i\omega/c)^2} = \frac{K_{exact}^2}{(i\omega/c)^2} \quad (3.30)$$

After rearranging the terms in (3.29) for K_{exact} and using (3.30), we obtain

$$R_{2n} = \left(\frac{a_n - \frac{ik}{i\omega/c}}{a_n + \frac{ik}{i\omega/c}} \right) \left(\frac{K_{exact} - K_{n-1}}{K_{exact} + K_{n-1}} \right) \quad (3.31)$$

With (3.27), the reflection coefficient becomes

$$R_{2n} = \left(\frac{\cos(\theta) - \cos(\theta_{2n})}{\cos(\theta) + \cos(\theta_{2n})} \right)^2 R_{2n-2} \quad (3.32)$$

Without truncation ($n \rightarrow \infty$), the half-space stiffness is represented exactly resulting in a zero reflection coefficient. In effect, the discretization error is nullified by the use of one-point integration. Thus, one-point integration is important in that it simulates the half-space in a consistent manner.

If one-point integration is not used, a nonzero reflection coefficient will be obtained. For example, consider a two-point integrated finite element, which results in the following dynamic stiffness matrix, scaled by $\frac{1}{-i\omega/c}$,

$$\mathbf{k}^e - k^2 \mathbf{m}^e = \begin{bmatrix} -\frac{1}{iL\omega/c} + \frac{k^2 L}{3i\omega/c} & \frac{1}{i\omega/c L} + \frac{k^2 L}{6i\omega/c} \\ \frac{1}{i\omega/c L} + \frac{k^2 L}{6i\omega/c} & -\frac{1}{i\omega/c L} + \frac{k^2 L}{3i\omega/c} \end{bmatrix} \quad (3.33)$$

and results in

$$a_n = -\frac{1}{iL_n\omega/c} + \frac{k^2 L_n}{3i\omega/c} \quad (3.34)$$

$$b_n = \frac{1}{iL_n\omega/c} + \frac{k^2 L_n}{6i\omega/c}$$

In this case, it can be easily shown that

$$a_n^2 - b_n^2 \neq \frac{K_{exact}^2}{(i\omega/c)^2} \quad (3.35)$$

and the reflection coefficient is nonzero.

Upon the truncation of the infinite number of layers as shown in Fig. 3.4, and using the auxiliary variables u_1, \dots, u_n with the last auxiliary variable $u_n = 0$; $R_0 = 1$, the reflection coefficient for the CFABC is given as

$$R_{2n} = \prod_{k=1}^n \left(\frac{\cos \theta - \cos \theta_{2k}}{\cos \theta + \cos \theta_{2k}} \right)^2 \quad (3.36)$$

In Section 4, the effects of the choice of integration points on the numerical results will be illustrated.

3.2 Formulation of Corner Absorbing Boundaries

The treatment of absorbing boundary conditions at corners for differential-equation-based ABC's is not a new problem. Earlier works were performed by Bamberger et al. [2] and Collino [10]. These earlier approaches formulate the corner absorbers by deriving corner compatibility conditions which are necessary to ensure the regularity of the solution of the problem when the data are smooth. In a more recent work [35], Vacus also describes an algebraic procedure for deriving corner compatibility conditions. It was recently highlighted by Hagstrom [22] that a more straightforward approach would be useful. Here, we present an approach that is simple and leads to an easy implementation of the corner absorbing boundaries. We will formulate the absorbers for non-orthogonal corners, with orthogonal corners being a special case.

The key idea in the ease of the formulation of corner ABC's is the imaginary length of $\frac{2i}{k_n}$ and one-point integration. First, consider the Fourier transform in time of the two-dimensional scalar wave equation in (2.1)

$$-\mu(u_{xx} + u_{zz}) - \omega^2 \rho u = 0 \quad (3.37)$$

Multiplying (3.37) with the virtual displacement δu , and subsequently integrating over the element domain Ω^e using the divergence theorem, we obtain

$$\underbrace{\int_{\Omega^e} \nabla(\delta u) \cdot \mu \nabla u d\Omega}_1 - \omega^2 \underbrace{\int_{\Omega^e} \rho \delta u u d\Omega}_2 = \int_{\Gamma} \delta u \mu \nabla u \cdot \mathbf{n} d\Gamma \quad (3.38)$$

where \mathbf{n} is the unit normal. The element stiffness and mass matrices in the frequency domain will be obtained from the first and second terms of (3.38), respectively. For simplicity, the element stiffness matrix in the frequency domain is derived using isoparametric formulation.

Here, we propose the use of parallelogram elements at the corners when the two intersecting absorbing boundaries are not perpendicular to each other as shown in Fig. 3.5.

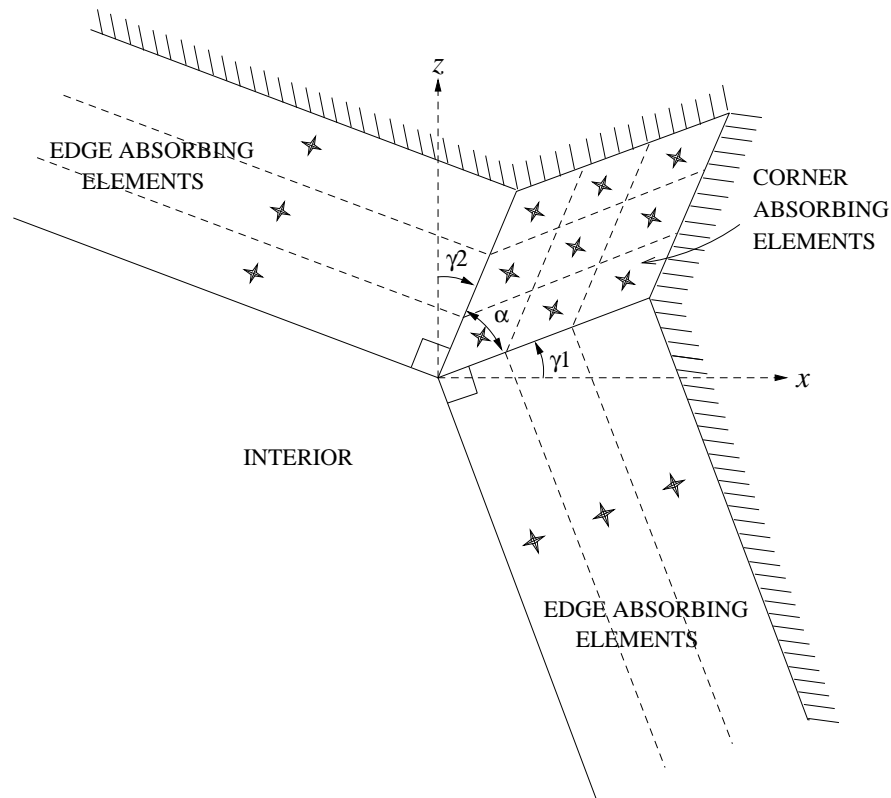


Figure 3.5: Parallelogram absorbing elements at the corner.

The shape of the parallelogram is determined by angle α , which is the angle between the two normals at the corner. For example, in Fig. 3.5, the angle $\alpha = \pi/2 - (\gamma_1 + \gamma_2)$. In our opinion, the parallelogram element, as peculiar as it may be, is a natural choice if we think in terms of the coordinate stretching directions of the two intersecting straight boundaries; the parallelogram element maintains the same stretching directions as in the straight boundaries. Furthermore, it has been observed from several numerical experiments that parallelogram angles determined in this way give the best solution accuracy.

Note that the element stiffness matrix is invariant of the element's orientation with respect to the xz coordinate axes. Thus, the formulation can be based on the parallelogram element as shown in Fig. 3.6. Using standard bilinear finite elements

(see Appendix A), the element stiffness matrix in the frequency domain is given by

$$\mathbf{k}^e = \int_{-1}^{+1} \int_{-1}^{+1} \mu \left(\frac{\partial \mathbf{H}}{\partial \mathbf{r}} \right)^T (\mathbf{J}^{-1})^T (\mathbf{J}^{-1}) \left(\frac{\partial \mathbf{H}}{\partial \mathbf{r}} \right) \det(\mathbf{J}) dr ds \quad (3.39)$$

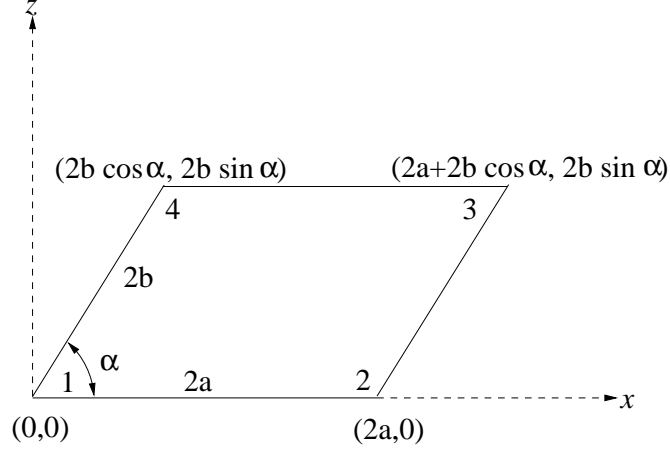


Figure 3.6: Parallelogram element.

where \mathbf{H} is the shape function matrix, $\frac{\partial \mathbf{H}}{\partial \mathbf{r}}$ is the matrix containing the derivatives of the shape functions with respect to the natural coordinates (r, s) , and \mathbf{J} is the Jacobian matrix. Employing 1×1 integration, the entries in \mathbf{k} are obtained as

$$\begin{aligned} k_{11}^e &= k_{33}^e = \frac{\mu}{4ab \sin \alpha} (b^2 - 2ab \cos \alpha + a^2) \\ k_{22}^e &= k_{44}^e = \frac{\mu}{4ab \sin \alpha} (b^2 + 2ab \cos \alpha + a^2) \\ k_{12} &= k_{21} = \frac{\mu}{4ab \sin \alpha} (a^2 - b^2) \\ k_{13}^e &= k_{31}^e = -k_{11}^e \\ k_{14}^e &= k_{41}^e = k_{23}^e = k_{32}^e = -k_{12}^e \\ k_{24}^e &= k_{42}^e = -k_{22}^e \\ k_{34}^e &= k_{43}^e = k_{12}^e \end{aligned} \quad (3.40)$$

Now, using the imaginary lengths

$$\begin{aligned} 2a &= \frac{2i}{k_{2n}^x} = \frac{2i}{(\omega/c) \cos \theta_{2n}^x} \\ 2b &= \frac{2i}{k_{2n}^z} = \frac{2i}{(\omega/c) \cos \theta_{2n}^z} \end{aligned} \quad (3.41)$$

the entries in \mathbf{k} are rewritten as

$$\begin{aligned} k_{11}^e &= k_{33}^e = \frac{\mu}{4 \sin \alpha} \left(\frac{\cos \theta_{2n}^x}{\cos \theta_{2n}^z} - 2 \cos \alpha + \frac{\cos \theta_{2n}^z}{\cos \theta_{2n}^x} \right) \\ k_{22}^e &= k_{44}^e = \frac{\mu}{4 \sin \alpha} \left(\frac{\cos \theta_{2n}^x}{\cos \theta_{2n}^z} + 2 \cos \alpha + \frac{\cos \theta_{2n}^z}{\cos \theta_{2n}^x} \right) \\ k_{12}^e &= k_{21}^e = \frac{\mu}{4 \sin \alpha} \left(-\frac{\cos \theta_{2n}^x}{\cos \theta_{2n}^z} + \frac{\cos \theta_{2n}^z}{\cos \theta_{2n}^x} \right) \\ k_{13}^e &= k_{31}^e = -k_{11}^e \end{aligned} \quad (3.42)$$

$$k_{14}^e = k_{41}^e = k_{23}^e = k_{32}^e = -k_{12}^e$$

$$k_{24}^e = k_{42}^e = -k_{22}^e$$

$$k_{34}^e = k_{43}^e = k_{12}^e$$

The element mass matrix in the frequency domain is similarly obtained as

$$\mathbf{m}^e = \int_{-1}^{+1} \int_{-1}^{+1} \rho \mathbf{H}^T \mathbf{H} \det(\mathbf{J}) \, dr \, ds \quad (3.43)$$

resulting in

$$\mathbf{m}^e = -\frac{1}{\omega^2} \frac{\mu \sin \alpha}{4 \cos \theta_{2n}^x \cos \theta_{2n}^z} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad (3.44)$$

The dynamic stiffness matrix is then given by $\mathbf{k}^e - \omega^2 \mathbf{m}^e$, which is independent of ω and simply becomes the element stiffness matrix \mathbf{k}^e in the space-time domain with

the same entries. Thus, the corner absorbers contribute only to the global stiffness matrix \mathbf{K} . For orthogonal corners, we simply set the angle $\alpha = \pi/2$.

3.3 Finite Element Discretization and Time Stepping

The discretization of the edge CFABC's in the straight boundaries is determined by how the discretization of the interior terminates at the boundary as shown in Fig. 3.7.

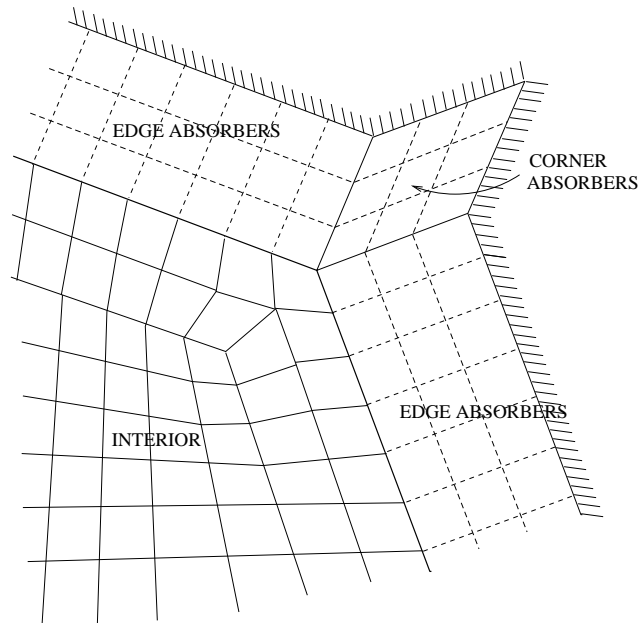


Figure 3.7: Typical discretization for the interior (finite elements) and for the edge and corner absorbers.

As shown in Fig. 3.7, the parallelogram-shaped corner absorbers naturally lend themselves to a discretization that is consistent with the discretization in the edge absorbers. It is evident that the application of the CFABC's can be viewed as adding additional nodes and elements to the existing truncated problem.

The variational form of (2.28) can be written as

$$-\int_{\Gamma_c} \delta u^T \left(\mu \frac{\partial u}{\partial x} \right) d\Gamma \approx \sum_{i=1}^n \left(\int_{\Gamma_c} \delta \mathbf{u}_i^T \mathbf{C}_i \mathbf{v}_i dz + \int_{\Gamma_c} \frac{\partial \delta \mathbf{u}_i^T}{\partial z} \mathbf{R}_i \frac{\partial \mathbf{w}_i}{\partial z} dz \right) \quad (3.45)$$

where the boundary term resulting from integration-by-parts has been set to zero, and the matrices \mathbf{C}_i and \mathbf{R}_i are the contributions from the n discrete number of layers corresponding to the one-point integrated finite elements given by

$$\begin{aligned} \mathbf{C}_i &= \sqrt{\rho\mu}(\mathbf{A}_i + \mathbf{B}_i) \\ &= \frac{\sqrt{\rho\mu}}{2} \left(\begin{bmatrix} c_i & -c_i \\ -c_i & c_i \end{bmatrix} + \begin{bmatrix} 1/c_i & 1/c_i \\ 1/c_i & 1/c_i \end{bmatrix} \right) \end{aligned} \quad (3.46)$$

and

$$\begin{aligned} \mathbf{R}_i &= \mu \sqrt{\frac{\mu}{\rho}} \mathbf{B}_i \\ &= \mu \sqrt{\frac{\mu}{\rho}} \begin{bmatrix} 1/c_i & 1/c_i \\ 1/c_i & 1/c_i \end{bmatrix} \end{aligned} \quad (3.47)$$

with $c_i = \cos(\theta_{2i})$. The subsequent discretization along the computational boundary Γ_c using the 2×1 shape function matrix \mathbf{N}_b for every two nodes on the straight boundary, for a total of $k - 1$ number of nodes, gives

$$\delta \mathbf{U}^T \mathbf{C} \mathbf{U}_t = \delta \mathbf{U}^T \left\{ \sum_{i=1}^n \sum_{j=1}^k \int_0^{\Delta z_j} \begin{bmatrix} \mathbf{N}_b^T & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_b^T \end{bmatrix}_{4 \times 2} \mathbf{C}_i \begin{bmatrix} \mathbf{N}_b & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_b \end{bmatrix}_{2 \times 4} dz \right\} \mathbf{U}_t \quad (3.48)$$

where b stands for the boundary in the z direction. This is the first term on the right-hand side of (3.45). Similarly, the discretization along the boundary gives second term on the right-hand side of (3.45) as

$$\begin{aligned} &\delta \mathbf{U}^T \mathbf{R} \mathbf{W} = \\ &\delta \mathbf{U}^T \left\{ \sum_{i=1}^n \sum_{j=1}^k \int_0^{\Delta z_j} \begin{bmatrix} \frac{\partial \mathbf{N}_b^T}{\partial z} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \mathbf{N}_b^T}{\partial z} \end{bmatrix}_{4 \times 2} \mathbf{R}_i \begin{bmatrix} \frac{\partial \mathbf{N}_b}{\partial z} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \mathbf{N}_b}{\partial z} \end{bmatrix}_{2 \times 4} dz \right\} \int_0^t \mathbf{U} dt \end{aligned} \quad (3.49)$$

The corner CFABC's, as mentioned in Section 3.2, contribute to the \mathbf{K} matrix. Therefore, the interior problem, when augmented with the CFABC's, results in the following integro-differential equation in time

$$\mathbf{M}\mathbf{U}_{tt} + \mathbf{C}\mathbf{U}_t + \mathbf{K}\mathbf{U} + \mathbf{R} \int_0^t \mathbf{U} dt = \mathbf{F} \quad (3.50)$$

3.4 Implicit Procedure

Due to the existence of the time-integral term in (3.50), this discrete evolution equation differs from those of standard wave equations and an appropriate time stepping scheme is devised. The constant-average acceleration method [3] is extended in a consistent manner to handle this situation as follows:

$$\begin{aligned} \mathbf{v}_{n+1} &= \mathbf{v}_n + \frac{\mathbf{a}_n + \mathbf{a}_{n+1}}{2} \Delta t \\ \mathbf{u}_{n+1} &= \mathbf{u}_n + \frac{\mathbf{v}_n + \mathbf{v}_{n+1}}{2} \Delta t \\ \mathbf{w}_{n+1} &= \mathbf{w}_n + \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2} \Delta t \end{aligned} \quad (3.51)$$

where \mathbf{a} is the second derivative, \mathbf{v} is the first derivative and \mathbf{w} is the integral of \mathbf{u} with respect to time.

This scheme can be derived from the Newmark method [3], which uses the following approximations for the displacement and velocity.

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \mathbf{v}_n + \frac{\Delta t^2}{2} [(1 - 2\beta) \mathbf{a}_n + 2\beta \mathbf{a}_{n+1}] \quad (3.52)$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \Delta t [(1 - \gamma) \mathbf{a}_n + \gamma \mathbf{a}_{n+1}] \quad (3.53)$$

We then obtain the integral-of-displacement by integrating (3.52) to obtain

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \Delta t \mathbf{u}_n + \frac{\Delta t}{2} \mathbf{v}_n + \Delta t^3 [(\kappa - \alpha) \mathbf{a}_n + \alpha \mathbf{a}_{n+1}] \quad (3.54)$$

For the constant average acceleration, $\gamma = 1/2$ and $\beta = 1/4$. Using these values and directly integrating (3.52) would give $\alpha = 1/12$ and $\kappa = 1/4$. However, for the

extended constant average acceleration, the values of $\alpha = 1/12$ and $\kappa = 1/4$ are used giving (3.51) which results in unconditional stability of the scheme [37].

For the details on the performance and accuracy of the CFABC, one can refer to [19], where the comparison between the CFABC and exact solutions is made.

3.5 Pseudo-Explicit Procedure

We note that a fully explicit scheme is not possible since the mass matrix is singular (the absorbing boundary conditions do not contribute to the matrix \mathbf{M}). In this section we present the implementation and results obtained from exploring the pseudo-explicit procedure for the CFABC's.

Consider the semi-discrete equation written at time $n\Delta t$

$$\mathbf{M}\mathbf{a}_n + \mathbf{C}\mathbf{v}_n + \mathbf{K}\mathbf{u}_n + \mathbf{R}\mathbf{w}_n = \mathbf{F} \quad (3.55)$$

where \mathbf{a}_n , \mathbf{v}_n , \mathbf{u}_n and \mathbf{w}_n are the acceleration, velocity, displacement and integral-of-displacement vectors at time $n\Delta t$. We consider the interior and exterior (absorber) parts as separate systems. The interior is represented by the mass \mathbf{M} and stiffness matrices \mathbf{K} with the primary variable being the displacement \mathbf{u}_n , while the exterior system is represented by the matrices \mathbf{C} and \mathbf{R} , with the primary variable being the integral-of-displacement \mathbf{w}_n .

First, consider the interior system. Let the acceleration vector be approximated by

$$\mathbf{a}_n = \frac{\mathbf{u}_{n+1} - 2\mathbf{u}_n + \mathbf{u}_{n-1}}{\Delta t^2} \quad (3.56)$$

Now, for the exterior part, since $v = \frac{d^2 w}{dt^2}$, we approximate the velocity as

$$\mathbf{v}_n = \frac{\mathbf{w}_{n+1} - 2\mathbf{w}_n + \mathbf{w}_{n-1}}{\Delta t^2} \quad (3.57)$$

The above two expressions are the explicit parts. Substituting (3.56) and (3.57) into (3.55) and moving all the terms at time $n\Delta t$ to the right-hand side while retaining

the terms at $(n-1)\Delta t$ on the left-hand side, we obtain

$$\begin{aligned} & \frac{1}{\Delta t^2} \mathbf{M} \mathbf{u}_{n+1} + \frac{1}{\Delta t^2} \mathbf{C} \mathbf{w}_{n+1} \\ = & \mathbf{F}_n - \mathbf{K} \mathbf{u}_n - \mathbf{R} \mathbf{w}_n + \frac{1}{\Delta t^2} \mathbf{M} (2\mathbf{u}_n - \mathbf{u}_{n-1}) + \frac{1}{\Delta t^2} \mathbf{C} (2\mathbf{w}_n - \mathbf{w}_{n-1}) \end{aligned} \quad (3.58)$$

Now, we approximate the integral-of-displacement vector \mathbf{w}_{n+1} using the mid-point rule as

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \Delta t \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2} \quad (3.59)$$

This is the implicit part. Substituting (3.59) into (3.58), we obtain, after some rearrangement of terms

$$\mathbf{M}^{eff} \mathbf{u}_{n+1} = \mathbf{F}^{eff} \quad (3.60)$$

where

$$\mathbf{M}^{eff} = \frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \quad (3.61)$$

and

$$\begin{aligned} \mathbf{F}^{eff} &= \mathbf{F}_n - \mathbf{K} \mathbf{u}_n - \mathbf{R} \mathbf{w}_n \\ &+ \frac{1}{\Delta t^2} \mathbf{M} (2\mathbf{u}_n - \mathbf{u}_{n-1}) + \frac{1}{\Delta t^2} \mathbf{C} (2\mathbf{w}_n - \mathbf{w}_{n-1}) \\ &- \frac{1}{\Delta t^2} \mathbf{C} \left(\mathbf{w}_n + \frac{\Delta t}{2} \mathbf{u}_n \right) \\ &= \mathbf{F}_n - \mathbf{K} \mathbf{u}_n - \mathbf{R} \mathbf{w}_n + \frac{1}{\Delta t^2} \mathbf{M} (2\mathbf{u}_n - \mathbf{u}_{n-1}) + \frac{1}{2\Delta t} \mathbf{C} \mathbf{u}_{n-1} \end{aligned} \quad (3.62)$$

From (3.61), if \mathbf{M} and \mathbf{C} are diagonal, then a full explicit computation is achieved. We also see that \mathbf{M}^{eff} is a non-singular matrix even though \mathbf{M} and \mathbf{C} are individually singular. Mass lumping can be performed for the interior problem to achieve a diagonal \mathbf{M} . However, as can be seen from (2.28), \mathbf{C} is only diagonal when the angle $\theta_{2k} = 0$. In the next section, we present the so-called pseudo-explicit procedure for efficient computation when θ_{2k} are non-zero.

3.5.1 Non-diagonal \mathbf{C} Matrix; Pseudo-Explicit Procedure

A preliminary scheme is devised by splitting the damping matrix on the left-hand side of (3.58) into implicit and explicit parts as shown

$$\mathbf{C} = \mathbf{C}^I + \mathbf{C}^E \quad (3.63)$$

where \mathbf{C}^I is taken to be the off-diagonal terms of \mathbf{C} and \mathbf{C}^E is taken to be the diagonal terms of \mathbf{C} . The implicit update of the integral-of-displacement in (3.58) is split into two parts, one for \mathbf{C}^I as in (3.59) and the other for \mathbf{C}^E as the forward Euler formula

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \Delta t \mathbf{u}_n \quad (3.64)$$

The resulting scheme appears to be inconsistent in the sense that the original problem is changed as a result of the splitting procedure. A heuristic proof now follows. We consider only the absorber system, with the following governing equation

$$\mathbf{C}\mathbf{v}_n + \mathbf{R}\mathbf{w}_n = \mathbf{0} \quad (3.65)$$

We assume that the modal equation exists and we are able to uncouple the above equations to obtain the following expression for one of the uncoupled equation as

$$\ddot{w}_n + \lambda^h w_n = 0 \quad (3.66)$$

where λ^h is the eigenvalue associated with that uncoupled equation. Now, apply the central difference formula on \ddot{w}_n using the one-degree-of-freedom analog of (3.57) to obtain

$$w_{n+1} - (2 - \Delta t^2 \lambda^h)w_n + w_{n-1} = 0 \quad (3.67)$$

We now perform a split on w_{n+1} into

$$w_{n+1} := \alpha_I w_{n+1}^I + \alpha_E w_{n+1}^E \quad (3.68)$$

where

$$w_{n+1}^I = w_n + \Delta t \frac{u_n + u_{n+1}}{2} \quad (3.69)$$

$$w_{n+1}^E = w_n + \Delta t u_n \quad (3.70)$$

and $\alpha_I + \alpha_E = 1$. Substituting (3.68) into (3.67) and using (3.69) and (3.70), and multiplying throughout by Δt , the following expression is obtained

$$u_{n+1}\Delta t = 2 \left\{ \frac{2 - \Delta t^2 \lambda^h - \alpha_E}{\alpha_I} - 1 \right\} w_n - \frac{2}{\alpha_I} w_{n-1} - \left(1 + 2 \frac{\alpha_E}{\alpha_I} \right) u_n \Delta t \quad (3.71)$$

The corresponding exact equation is

$$\begin{aligned} \Delta t \tau(t_n) &= \Delta t u(t_{n+1}) - 2 \left\{ \frac{2 - \Delta t^2 \lambda^h - \alpha_E}{\alpha_I} - 1 \right\} w(t_n) \\ &\quad + \frac{2}{\alpha_I} w(t_{n-1}) + \left(1 + 2 \frac{\alpha_E}{\alpha_I} \right) u(t_n) \Delta t \end{aligned} \quad (3.72)$$

By substituting the Taylor series expansion of $u(t_{n+1})$ and $w(t_{n-1})$ into the above expression and using $\alpha_I + \alpha_E = 1$, we obtain the local truncation error as

$$\begin{aligned} \tau(t_n) &= \frac{2}{\alpha_I \Delta t} \left\{ \frac{\alpha_I}{2} \left(1 + \frac{1}{\alpha_I} \ddot{w}_n \right) + \lambda^h w_n \right\} + \Delta t^2 \left(\frac{1}{2} - \frac{1}{3\alpha_I} \right) a_n \\ &\quad + \Delta t^3 \left(\frac{1}{6} + \frac{1}{12\alpha_I} \right) \frac{d a_n}{d t} + O(\Delta t^4) \end{aligned} \quad (3.73)$$

We observe that the first term on the right-hand side is always nonzero, resulting in an inconsistency. Furthermore, the local truncation error changes with λ^h and this can be observed by the addition of absorbers having different values of nonzero θ_{2k} . If $\alpha_I = 1$ ($\alpha_E = 0$) for the case of $\theta_{2k} = 0$, we obtain

$$\begin{aligned} \tau(t_n) &= \frac{2}{\Delta t} \{ \ddot{w}_n + \lambda^h w_n \} + O(\Delta t^2) \\ &= O(\Delta t^2) \end{aligned} \quad (3.74)$$

since the term in the curly brackets $\ddot{w}_n + \lambda^h w_n = 0$ from (3.66) and we recover the second-order convergence property as was shown in the previous section. The result of (3.73) is consistent with the numerical results that are obtained as will be presented in Chapter 4. From (3.73), it appears that no splitting is possible.

For the edge absorbers where we can have $\theta_{2k} \neq 0^\circ$ in the general case, we can at best perform lumping of the damping terms in the direction along the boundary, resulting in a tridiagonal matrix of dimension $n+1$ (for n number of absorber elements normal to the computational boundary) for each node on the edge interface. This is as shown in Fig. 3.8.

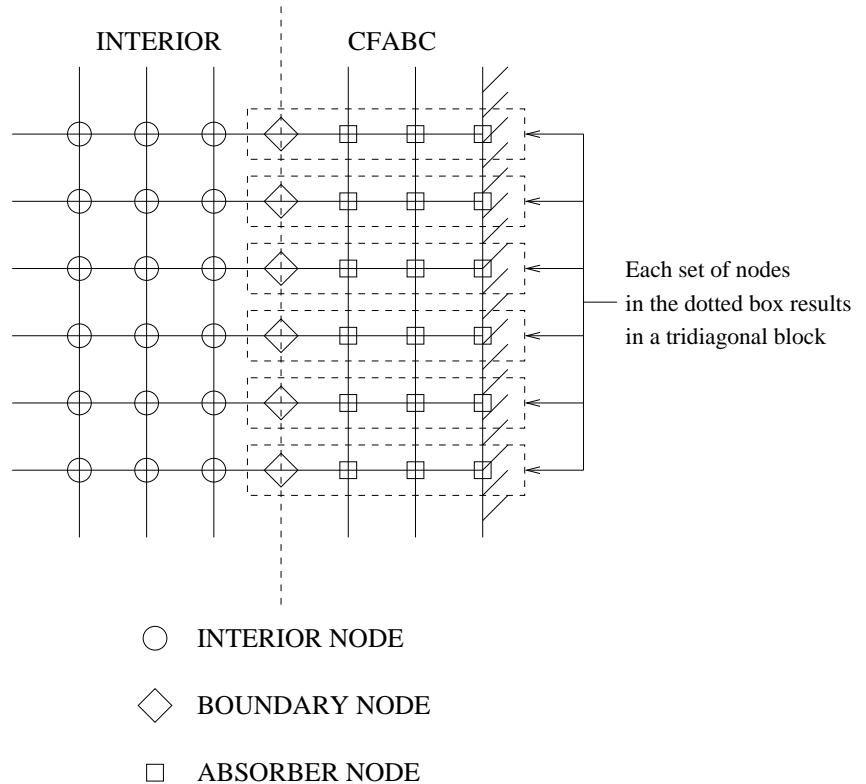


Figure 3.8: Nodes contributing to a tridiagonal block.

This way of lumping can be achieved by moving the Gauss points to the two nodes along the boundary when evaluating (3.48) and (3.49). Fig. 3.9 shows the location of the Gauss points for the implicit method when consistent mass matrix is used in the interior. For the pseudo-explicit method, the Gauss points are shifted to the locations as shown in Fig. 3.10

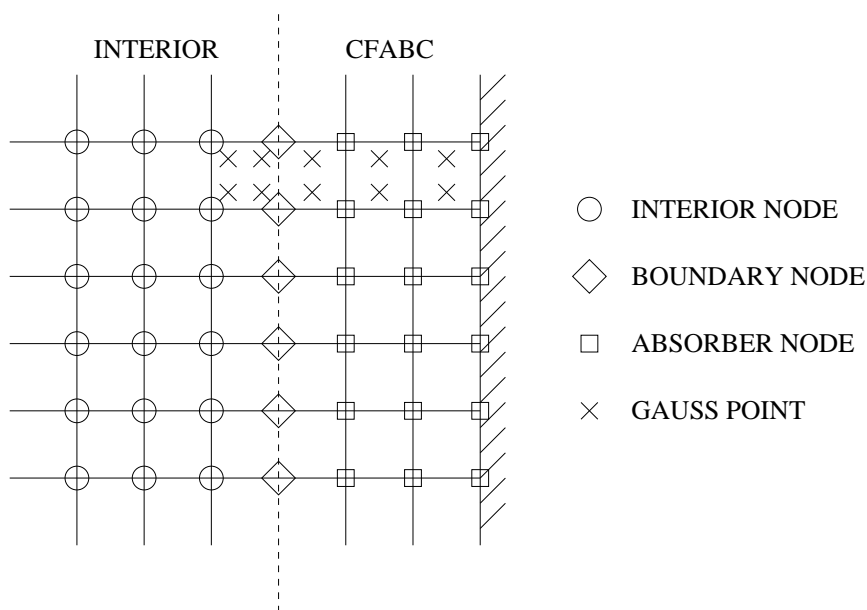


Figure 3.9: Location of Gauss points for the implicit method.

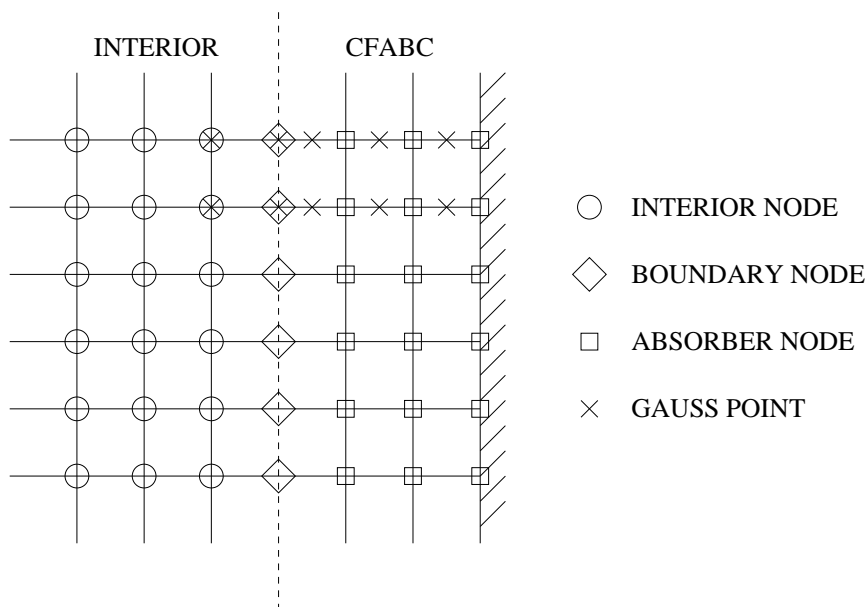


Figure 3.10: Location of Gauss points for the pseudo-explicit method.

As a result of this way of lumping, for $\theta_{2k} \neq 0^\circ$ the matrix \mathbf{M}^{eff} in (3.61) will take the form as shown in Fig. 3.11. For the case of $\theta_{2k} = 0^\circ$, all the tridiagonal blocks will reduce to diagonal blocks, resulting in a diagonal \mathbf{M}^{eff} . The term ‘pseudo’ in the pseudo-explicit procedure refers to the way of solving (3.60) by treating each of the tridiagonal blocks as a smaller effective mass matrix. Thus, a tridiagonal solver can be used to solve each of the system of equations associated with that block. The diagonal submatrix due to the interior can of course be solved explicitly.

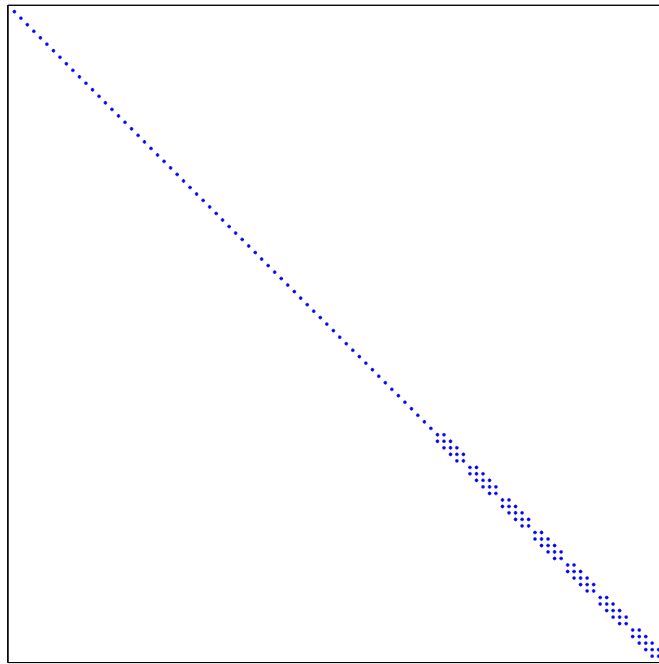


Figure 3.11: General structure of \mathbf{M}^{eff} for $\theta_{2k} \neq 0^\circ$ (without corner absorbers). The tridiagonal blocks are due to the lumping of the damping terms in the direction of the boundary while the diagonal submatrix is due to the mass lumping in the interior.

3.5.2 Presence of Corner Absorbers

From the previous chapter, we realize that the contribution from the corner absorbers will always result in a non-diagonal \mathbf{K} matrix. By augmenting the interior and edge

absorbers with the corner absorbers, we obtain the following effective mass and load vectors

$$\mathbf{M}^{eff} = \frac{1}{\Delta t^2} \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} + \frac{1}{2\Delta t} \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \quad (3.75)$$

and

$$\begin{aligned} \mathbf{F}^{eff} &= \begin{Bmatrix} \mathbf{F}_n \\ \mathbf{0} \end{Bmatrix} - \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{U}_n^1 \\ \mathbf{U}_n^2 \end{Bmatrix} - \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{W}_n^1 \\ \mathbf{W}_n^2 \end{Bmatrix} \\ &+ \frac{2}{\Delta t^2} \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{U}_n^1 \\ \mathbf{U}_n^2 \end{Bmatrix} - \frac{1}{\Delta t^2} \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{U}_{n-1}^1 \\ \mathbf{U}_{n-1}^2 \end{Bmatrix} \\ &+ \frac{1}{2\Delta t} \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{U}_{n-1}^1 \\ \mathbf{U}_{n-1}^2 \end{Bmatrix} \end{aligned} \quad (3.76)$$

where the subscript 1 refers to the system made up of nodes from the interior and the edge absorbers and the subscript 2 refers to the system from the corner nodes *not* connected to the system described by 1. This is illustrated in Fig. 3.12. Therefore, only the submatrices \mathbf{K}_{12} and \mathbf{K}_{21} are nonzero. With this, the first row of (3.60) gives

$$\begin{aligned} \left(\frac{1}{\Delta t^2} \mathbf{M}_{11} + \frac{1}{2\Delta t} \mathbf{C}_{11} \right) \mathbf{U}_{n+1}^1 &= \mathbf{F}_n - \mathbf{K}_{11} \mathbf{U}_n^1 - \mathbf{R}_{11} \mathbf{W}_n^1 \\ &+ \frac{1}{\Delta t^2} \mathbf{M}_{11} (2\mathbf{U}_n^1 - \mathbf{U}_{n-1}^1) + \frac{1}{2\Delta t} \mathbf{C}_{11} \mathbf{U}_{n-1}^1 - \mathbf{K}_{12} \mathbf{U}_n^2 \end{aligned} \quad (3.77)$$

and the second row of (3.60) gives

$$\mathbf{K}_{22} \mathbf{U}_n^2 = -\mathbf{K}_{21} \mathbf{U}_n^1 \quad (3.78)$$

In (3.77), we notice that all the terms except for the last term on the right-hand side are the ones that represents the nodes from the interior, edge absorbers and the nodes from the corners along the interior-edge-corner boundaries. Except for the last term, all the other terms make up the same form as (3.60). Therefore, at each time step, we first solve for \mathbf{U}_n^2 from (3.78) followed by solving for \mathbf{U}_{n+1}^1 . Then, (3.78) is used again to obtain \mathbf{U}_{n+1}^2 .

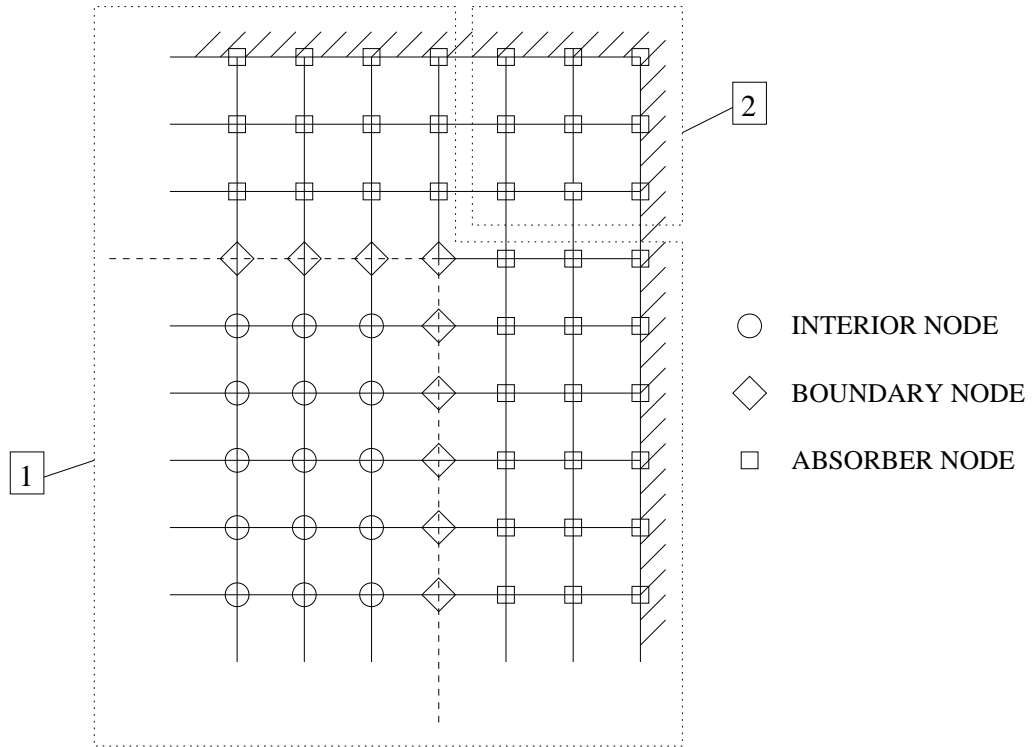


Figure 3.12: Nodes associated with system 1 and 2.

3.5.3 Computational Performance

For the choice of $\theta_{2k} = 0^\circ$, good absorption capabilities can still be achieved by using a few extra CFABC layers as can be inferred from the reflection coefficient in (3.32).

In the figure below, we plot the reflection coefficient against the incident angles for the two cases of $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$ and five CFABC layers at $\theta_{2k} = 0^\circ$.

We observe that a performance that is good for most engineering problems can be achieved with a small increase of computational cost.

For the case of layered media, where the interior is heterogeneous, the case of $\theta_{2k} = 0^\circ$ is somewhat limited, since we may need different phase velocities in different layers to achieve good absorption. The computation of blocks of tridiagonal matrices would only add minimal computational cost. The computation of the corner regions for a

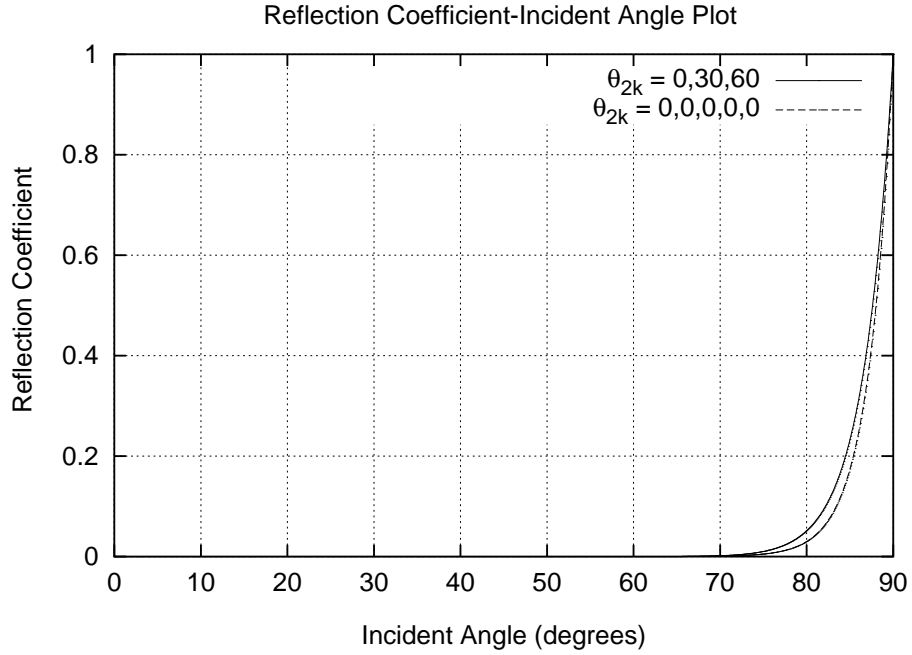


Figure 3.13: Comparison of reflection coefficients at different values of θ_{2k} .

two-dimensional problem requires computing at each time step a partial Neumann-to-Dirichlet map on a $n \times n$ square matrix where n is the number of CFABC layers. This is an operation proportional to the total number of nodes in the corner absorbing region.

3.5.4 A Note on Stability Condition

The stability condition is only governed by the interior problem as will be shown using the following argument. For a two-dimensional wave propagation problem governed by (2.1), rewritten here as

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \quad (3.79)$$

which upon discretization and using lumped masses, we have the required stability condition given by

$$\Delta t \leq \min \left(\frac{\Delta x}{c}, \frac{\Delta z}{c} \right) \quad (3.80)$$

referred to as the CFL condition after R. Courant, K. O. Friedrichs, and H. Lewy [7].

If we consider the absorber as a separate system governed by the one-degree-of-freedom differential equation

$$C \ddot{w} + R w = 0 \quad (3.81)$$

or

$$\frac{\sqrt{\rho\mu}}{2} \left(c_n + \frac{1}{c_n} \right) \frac{\partial^2 w}{\partial t^2} - \mu \sqrt{\frac{\mu}{\rho}} \frac{1}{2 c_n} \frac{\partial^2 w}{\partial z^2} = 0 \quad (3.82)$$

we can rearrange the above to give

$$\frac{\partial^2 w}{\partial z^2} = \frac{1}{\left(c/\sqrt{1+c_n^2} \right)^2} \frac{\partial^2 w}{\partial t^2} \quad (3.83)$$

This is a one-dimensional wave equation with the spatial dimension in the z direction and the wave speed in this case is $c/\sqrt{1+c_n^2}$. Upon the discretization of this equation with 'mass' (damper) lumping in the z direction, the CFL condition gives

$$\Delta t \leq \frac{\sqrt{1+c_n^2}}{c} \Delta z \quad (3.84)$$

Since $\frac{\sqrt{1+c_n^2}}{c} \geq \frac{1}{c}$ we see that (3.80) is the more restrictive condition. As long as the CFL condition is met for the interior problem, instability of the whole system will not arise.

3.5.5 Local Truncation Error Analysis

To simplify the analysis of the accuracy of the aforementioned scheme, we can consider only the one-degree-of-freedom analog of (3.60) given by

$$\left(\frac{1}{\Delta t^2} M + \frac{1}{2\Delta t} C \right) u_{n+1} = F_n - K u_n - R w_n + \frac{1}{\Delta t^2} M (2u_n - u_{n-1}) + \frac{1}{2\Delta t} C u_{n-1} \quad (3.85)$$

We first multiply (3.85) by Δt^2 and set $F_n = 0$ since the accuracy analysis is not affected by the presence of the load vector. Then, after moving all the terms to the left-hand side, we have

$$\left(M + \frac{\Delta t}{2}C\right)u_{n+1} + \Delta t^2 K u_n + \Delta t^2 R w_n - M(2u_n - u_{n-1}) - \frac{\Delta t}{2}C u_{n-1} = 0 \quad (3.86)$$

The corresponding exact expression of the above is of the form

$$\begin{aligned} \left\{M + \frac{\Delta t}{2}C\right\}u(t_{n+1}) + \Delta t^2 K u(t_n) + \Delta t^2 R w(t_n) \\ - M\{2u(t_n) - u(t_{n-1})\} - \frac{\Delta t}{2}C u(t_{n-1}) = \Delta t \tau(t_n) \end{aligned} \quad (3.87)$$

where $\tau(t_n)$ is the local truncation error. The goal is to determine the correct exponent in the entries of τ . To do this, we first perform a Taylor series expansion on $d(t_{n+1})$ and $d(t_{n-1})$

$$u(t_{n+1}) = u(t_n) + \Delta t v(t_n) + \frac{\Delta t^2}{2} a(t_n) + \frac{\Delta t^3}{6} \frac{da(t_n)}{dt} + O(\Delta t^4) \quad (3.88)$$

$$u(t_{n-1}) = u(t_n) - \Delta t v(t_n) + \frac{\Delta t^2}{2} a(t_n) - \frac{\Delta t^3}{6} \frac{da(t_n)}{dt} + O(\Delta t^4) \quad (3.89)$$

Substituting the above expressions into (3.87) and using the relationship $M a(t_n) + C v(t_n) + K u(t_n) + R w(t_n) = 0$, we obtain the final form as

$$\begin{aligned} \Delta t \tau(t_n) &= \frac{\Delta t^4}{6} \frac{da(t_n)}{dt} + O(\Delta t^5) \\ &= O(\Delta t^4) \end{aligned} \quad (3.90)$$

which gives $\tau(t_n) = O(\Delta t^3)$.

Next we need to determine the local truncation errors for the integral-of-displacement, velocity and acceleration terms. The one-degree-of-freedom analog of the implicit update of the integral-of-displacement (3.59) is

$$w_{n+1} = w_n + \Delta t \frac{u_n + u_{n+1}}{2} \quad (3.91)$$

and the corresponding exact equation is

$$\Delta t \tau_w(t_n) = w(t_{n+1}) - w(t_n) - \Delta t \frac{u(t_n) + u(t_{n+1})}{2} \quad (3.92)$$

The Taylor series expansion on $w(t_{n+1})$ gives

$$w(t_{n+1}) = w(t_n) + \Delta t u(t_n) + \frac{\Delta t^2}{2} v(t_n) + \frac{\Delta t^3}{6} a(t_n) + \frac{\Delta t^4}{24} \frac{da(t_n)}{dt} + O(\Delta t^5) \quad (3.93)$$

Using the above and (3.88) in (3.92), we have

$$\begin{aligned} \Delta t \tau_w(t_n) &= \frac{\Delta t^3}{12} a(t_n) - \frac{\Delta t^4}{24} \frac{da(t_n)}{dt} + O(\Delta t^5) \\ &= O(\Delta t^3) \end{aligned} \quad (3.94)$$

which gives $\tau_w(t_n) = O(\Delta t^2)$. The one-degree-of-freedom analog of the velocity and acceleration approximations ((3.57) and (3.56)) can be shown to have a local truncation error of $O(\Delta t^3)$. Therefore, the minimum is of $O(\Delta t^2)$ and the scheme as whole is second-order accurate.

Chapter 4

Numerical Examples

4.1 Computational Domains With Corners

4.1.1 Explosive Source In An Unbounded Domain

In this problem, we simulate the anti-plane shear wave propagation in a two-dimensional unbounded domain. A square domain of size 30×30 m² is discretized using 40000 (200×200) 4-node bilinear elements, and is laced with three CFABC layers ($\theta_{2n} = 0^\circ, 30^\circ, 60^\circ$). The following explosive source obtained from [12] is used

$$f(r, t) = \begin{cases} -2\pi^2 f_0^2 (t - t_0) e^{-\pi^2 f_0^2 (t - t_0)^2} \left(1 - \frac{r^2}{a^2}\right)^3 & \text{if } t \leq 2t_0, r \leq a \\ 0 & \text{otherwise} \end{cases} \quad (4.1)$$

where r is the distance from the center of the source, $a = 5h$ is the radius of the disk, h is the side length of each element, $t_0 = 1/f_0$, $f_0 = V_s/(hN_L)$ is the central frequency, and N_L is the number of points per wavelength. The time step size is determined using $\Delta t = h/V_s$. For this problem, the parameters are $h = 30/200$ m, $N_L = 20$, and $V_s = 2000$ m/s. The explosive source is located at (7.5 m, 7.5 m) with the origin located at the bottom left corner. The snapshots of the solution (absolute value of the velocity $|v|$) are presented in Fig. 4.1 and 4.2. No numerical reflections are visible on the normal scale.

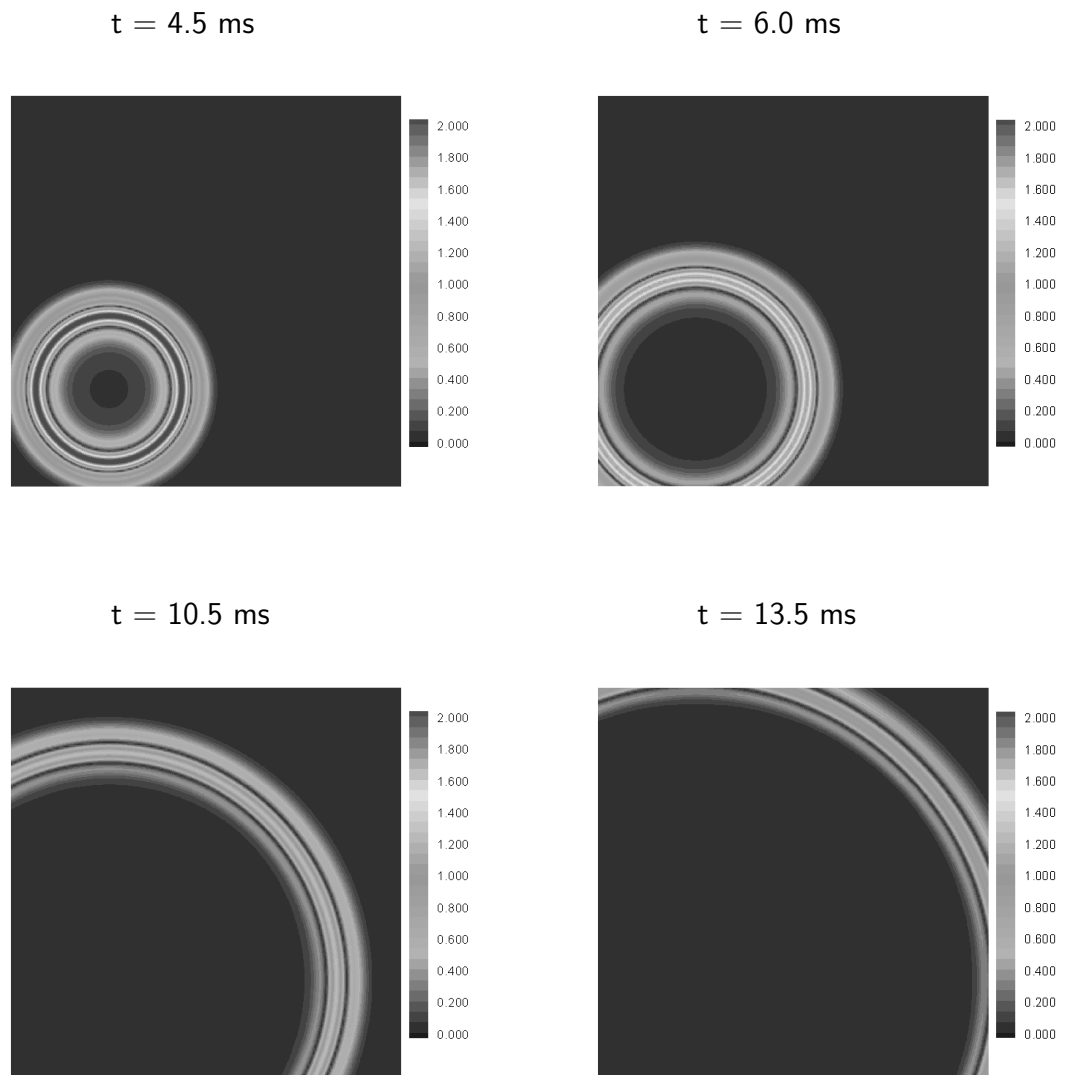


Figure 4.1: Anti-plane shear wave propagation in an unbounded domain.

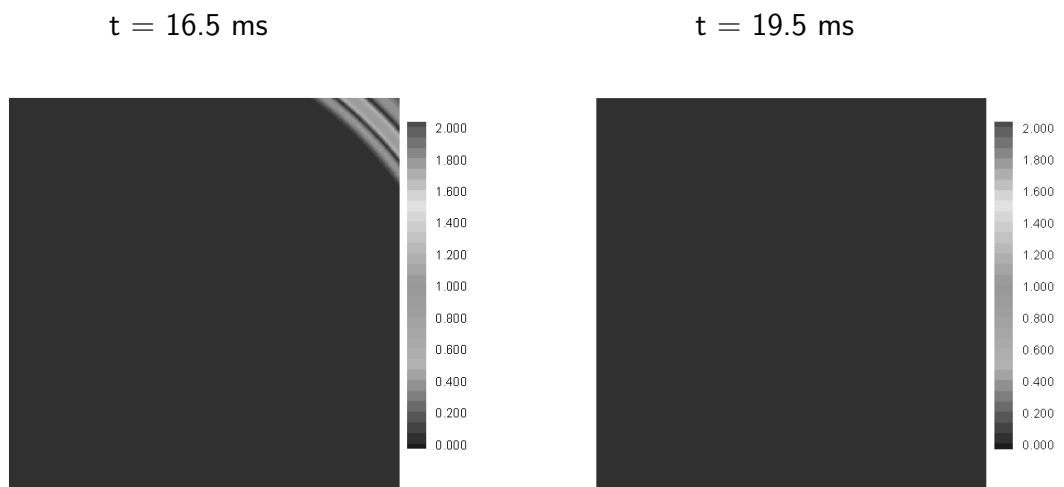


Figure 4.2: Anti-plane shear wave propagation in an unbounded domain (cont'd).

4.1.2 Effects of Integration Order

Here, the effects of integration order on the absorption ability of an ABC are shown. In particular, we will present the comparison between the one-point integrated CFABC and another CFABC where the conventional 2-point integration is used. The velocity at two locations:

- The point which is one cell distance from the bottom boundary (Node 302), with coordinates (15.00 m, 0.15 m).
- The point which is one cell distance from the right boundary (Node 20300), with coordinates (29.85 m, 15.00 m).

are plotted in Figs. (4.3) and (4.4).

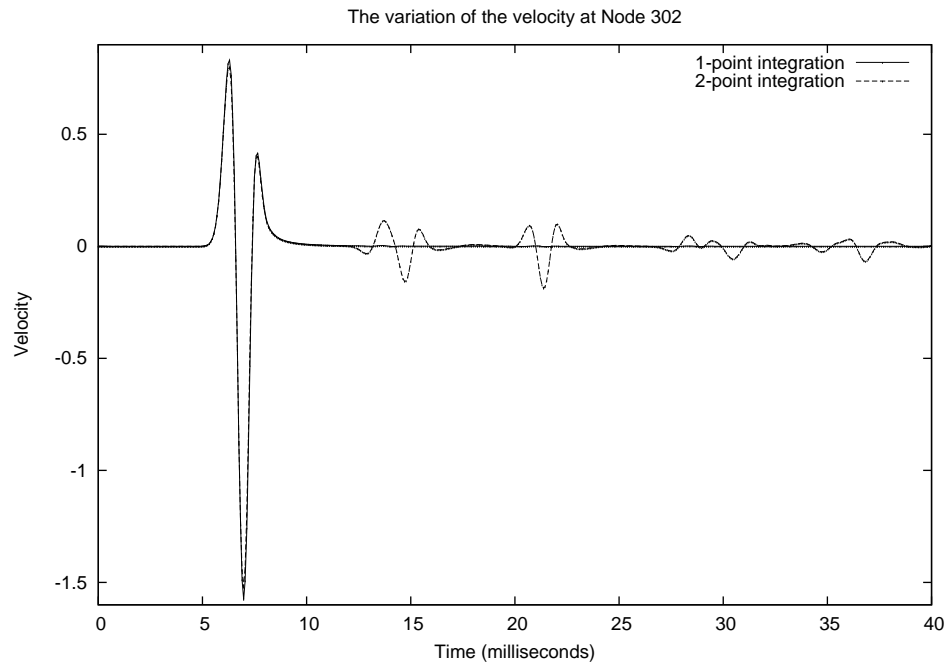


Figure 4.3: The variation of velocity at Node 302.

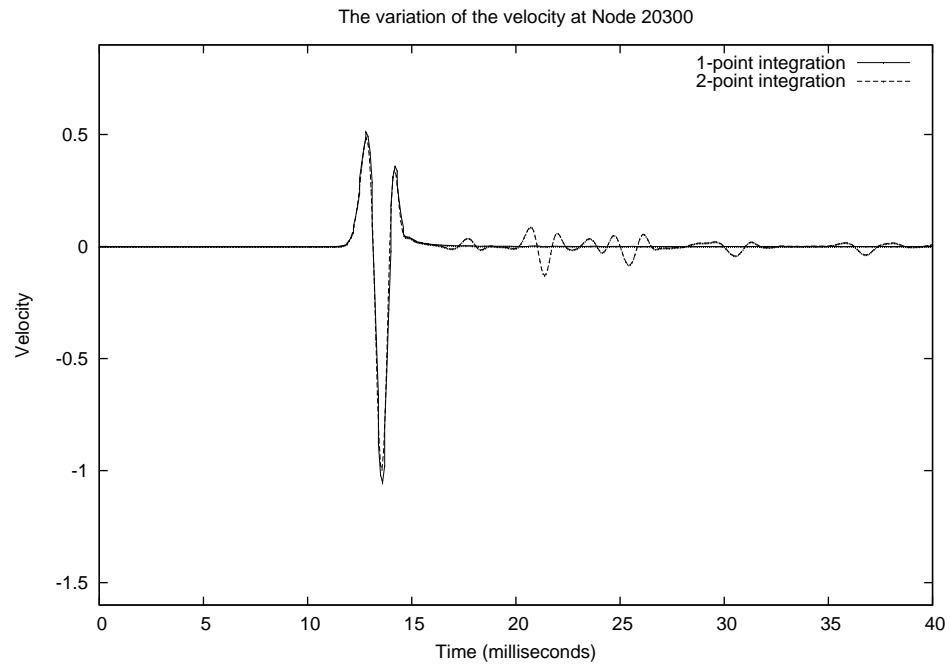


Figure 4.4: The variation of velocity at Node 20300.

The velocity norm contours resulting from using one- and two-point Gauss quadratures are as shown from Fig. 4.5 to Fig. 4.7. To highlight the reflections the contour scale of $[0,0.5]$ is used. We can clearly see from the contours and the velocity plots that two-point Gauss quadrature results in significant reflections.

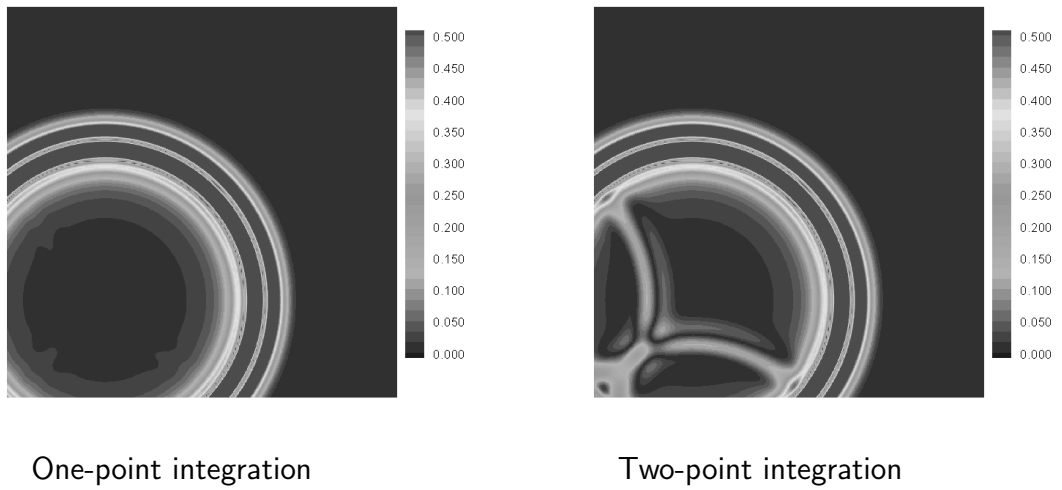


Figure 4.5: Norm of the velocity at $t = 7.5$ ms.

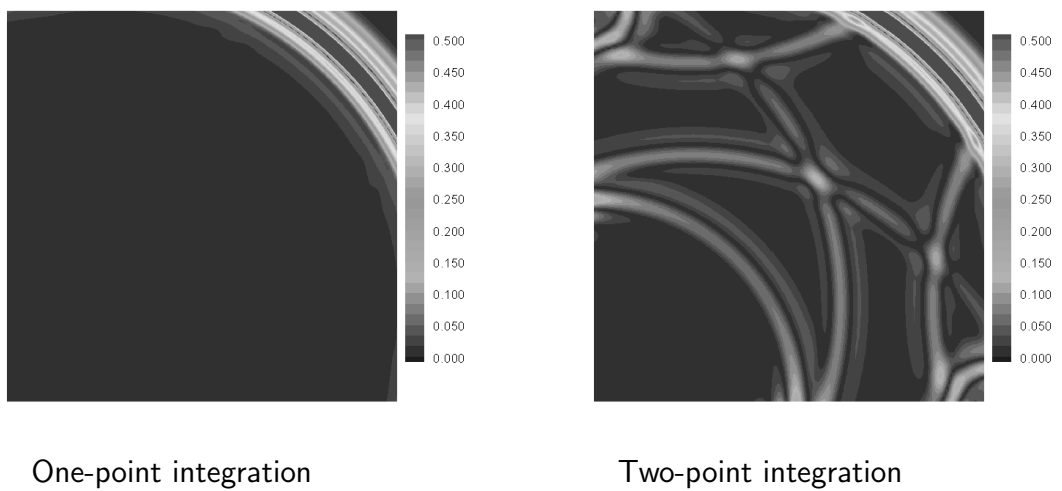


Figure 4.6: Norm of the velocity at $t = 15.0$ ms.

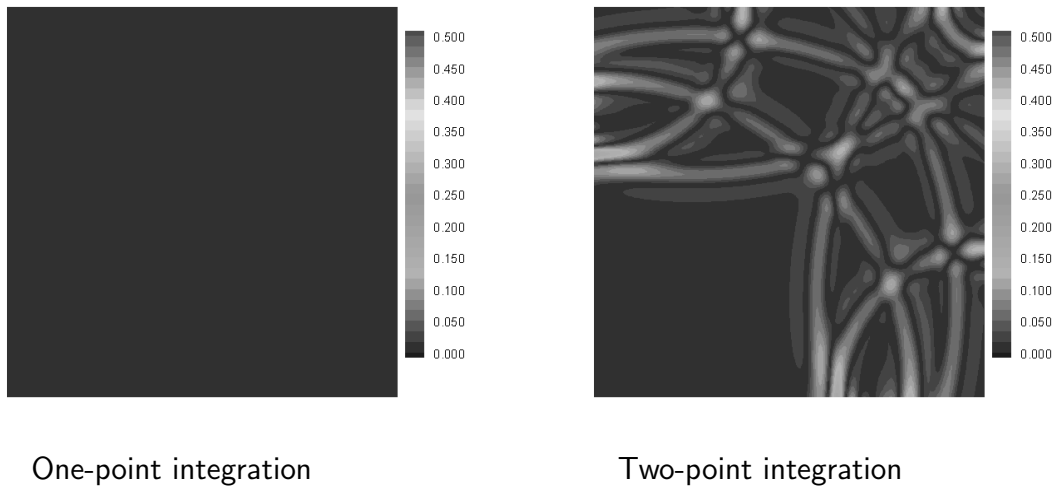


Figure 4.7: Norm of the velocity at $t = 19.5$ ms.

4.1.3 Truncated Domain With Oblique Corners

In this example, the same explosive source in (4.1) is used, located within a computational domain having the oblique corners as shown in Fig. 4.8. The origin is located at the bottom left corner with the edge and corner absorbers as indicated in the figure. Three CFABC layers are used on all boundaries. The explosive source is located near the bottom left corner with coordinates (7.12 m, 0.15 m). The same number of elements and values of the parameters in example 4.1.1 are used here. The parallelogram angles are, starting from the bottom left corner and progressing counterclockwise: 53.13° , 108.44° , 108.44° , and 90.00° . Again, from the contour plots of the velocity norm in Fig. 4.9 and 4.10, no numerical reflections are visible on the normal scale.

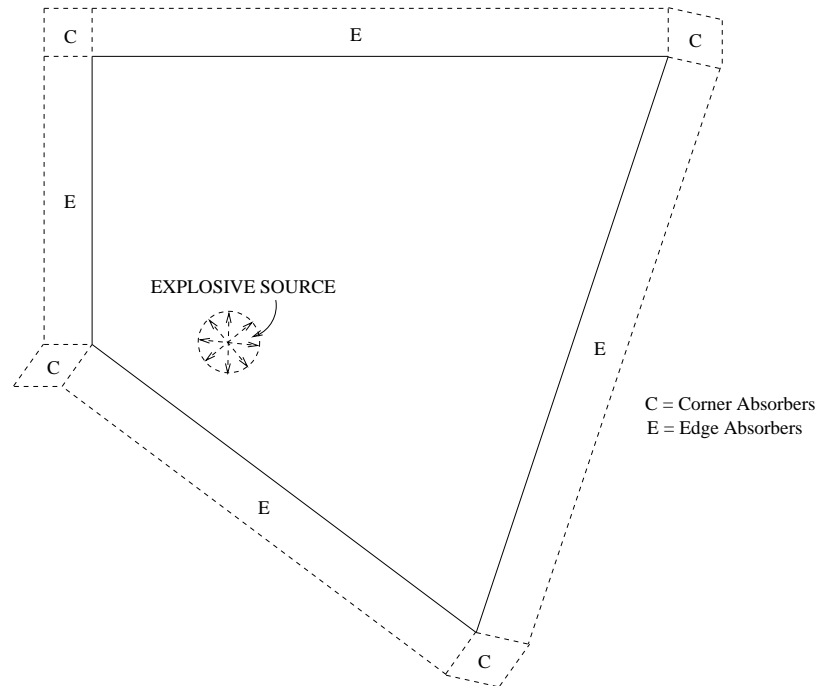


Figure 4.8: Problem schematic: domain with oblique corners.

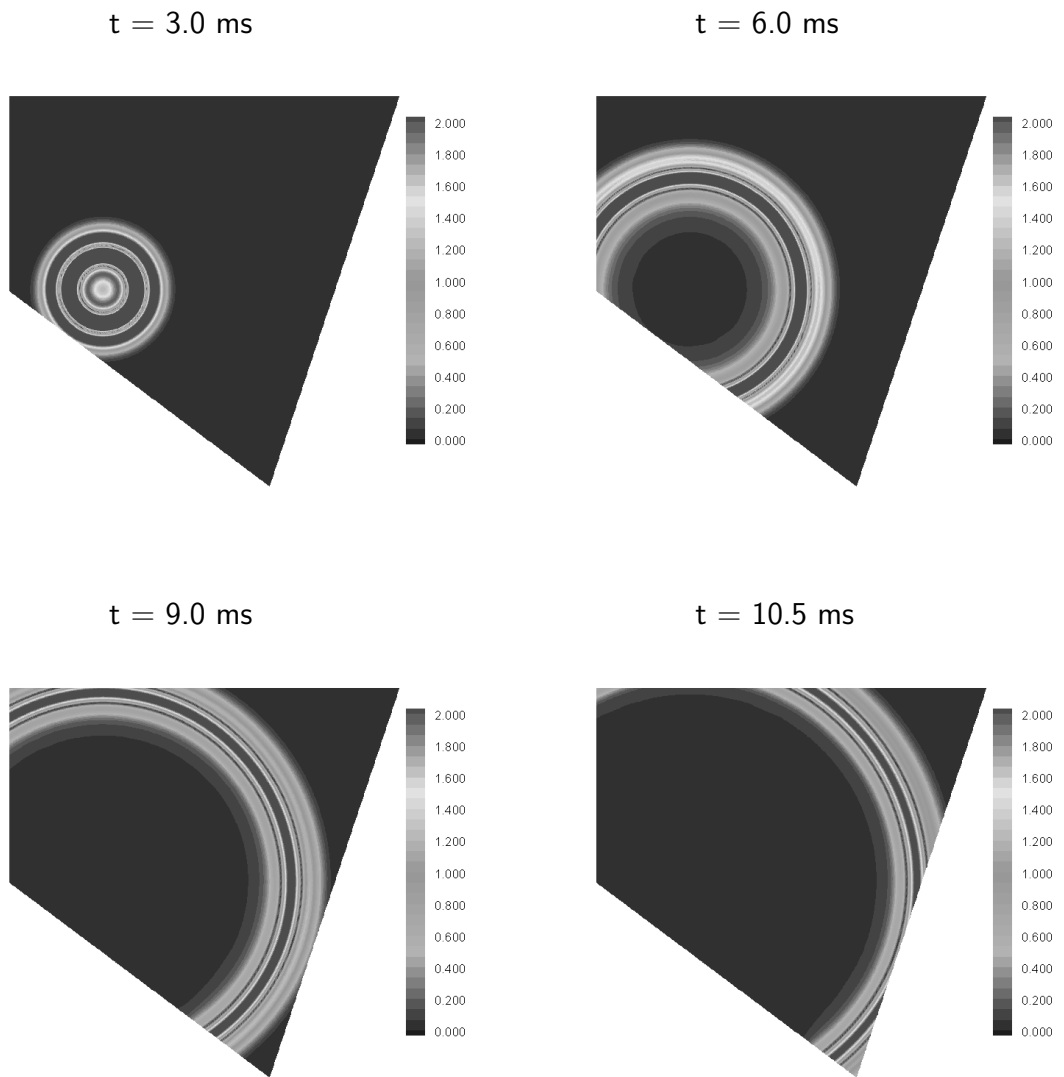


Figure 4.9: Anti-plane shear wave propagation in domain with oblique corners.

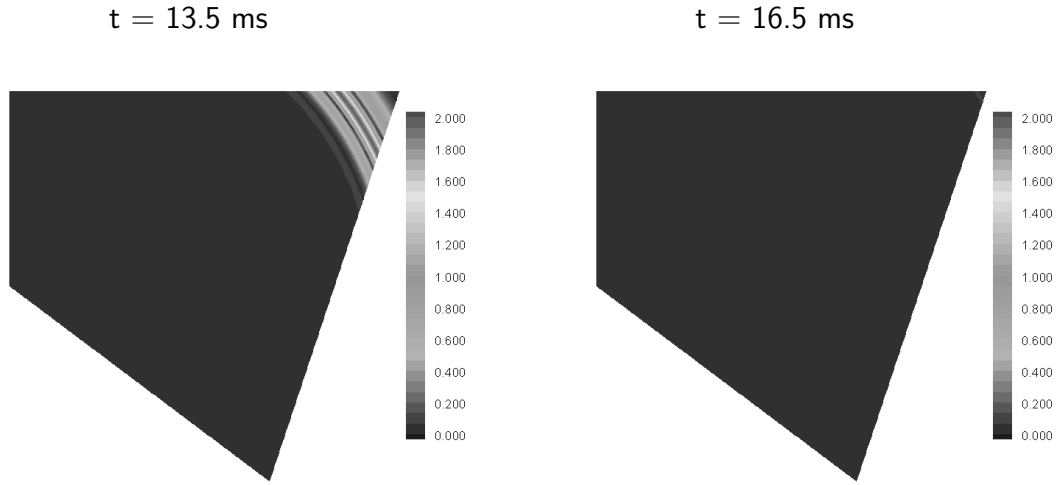


Figure 4.10: Anti-plane shear wave propagation in domain with oblique corners (cont'd).

4.1.4 Layered Media

In this example, we simulate the anti-plane shear wave propagation in a domain having two layers with different wave speeds. The schematic of the problem is as shown in Fig. 4.11. The domain size is 45 m in height and 30 m in width, with the top surface free and the all other boundaries laced with three CFABC layers. The mesh consists of 300×200 bilinear finite elements and the shear wave speeds are specified as 1000 m/s and 2000 m/s for the top and bottom layers, respectively. The layer interface is located at 22.5 m from the top surface and the same explosive source as in (4.1) is located at the center of the layer interface line. The absorption angles are chosen so that the phase velocities in each CFABC layer for the bottom and top media are equal. In this case, for the bottom CFABC layers, $\theta_{2k}^{bottom} = 0^\circ, 30^\circ, 60^\circ$. Using the relationship

$$\text{phase velocity} = \frac{c^{top}}{\cos(\theta_{2k}^{top})} = \frac{c^{bottom}}{\cos(\theta_{2k}^{bottom})} \quad (4.2)$$

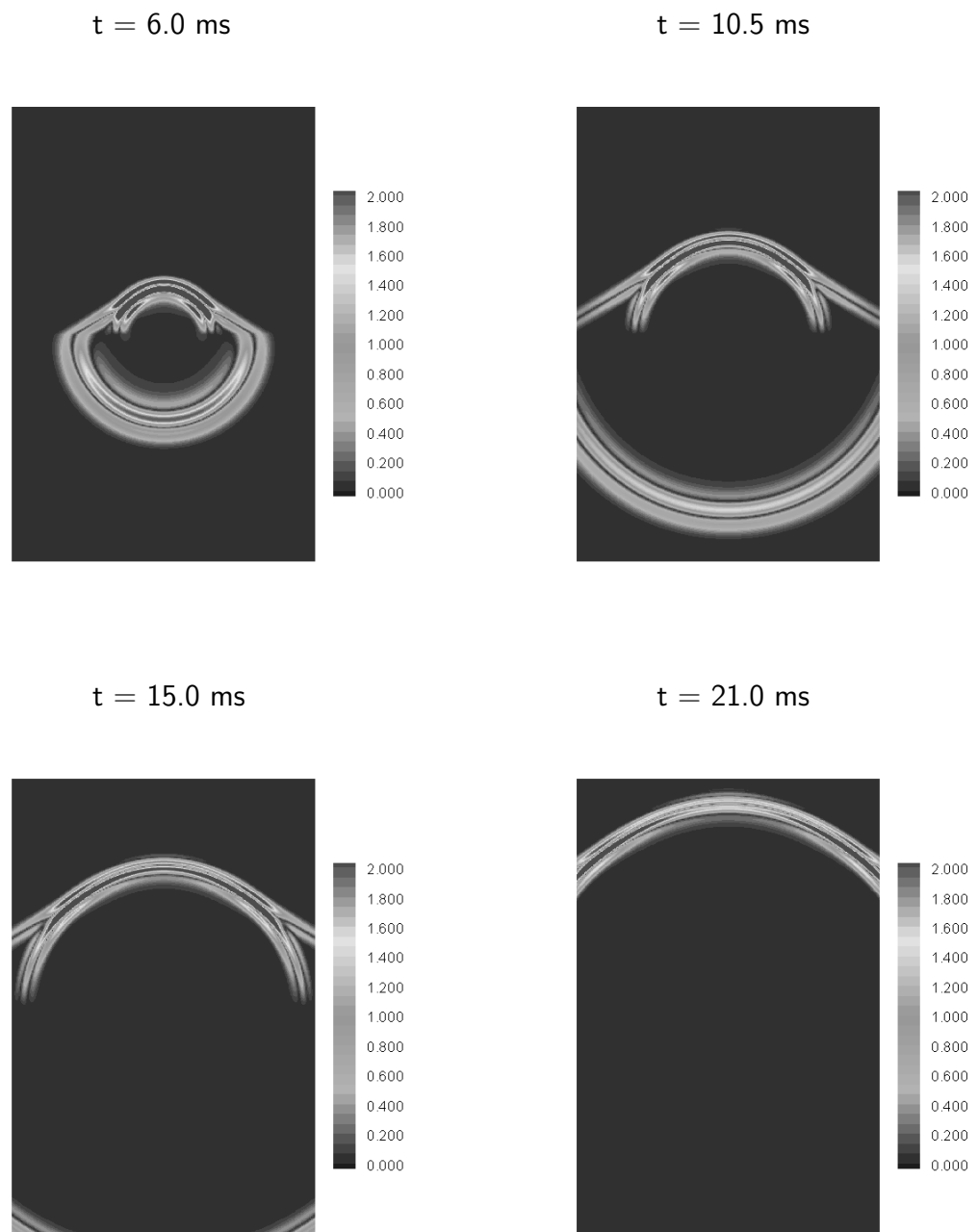


Figure 4.12: Anti-plane shear wave propagation in layered media.

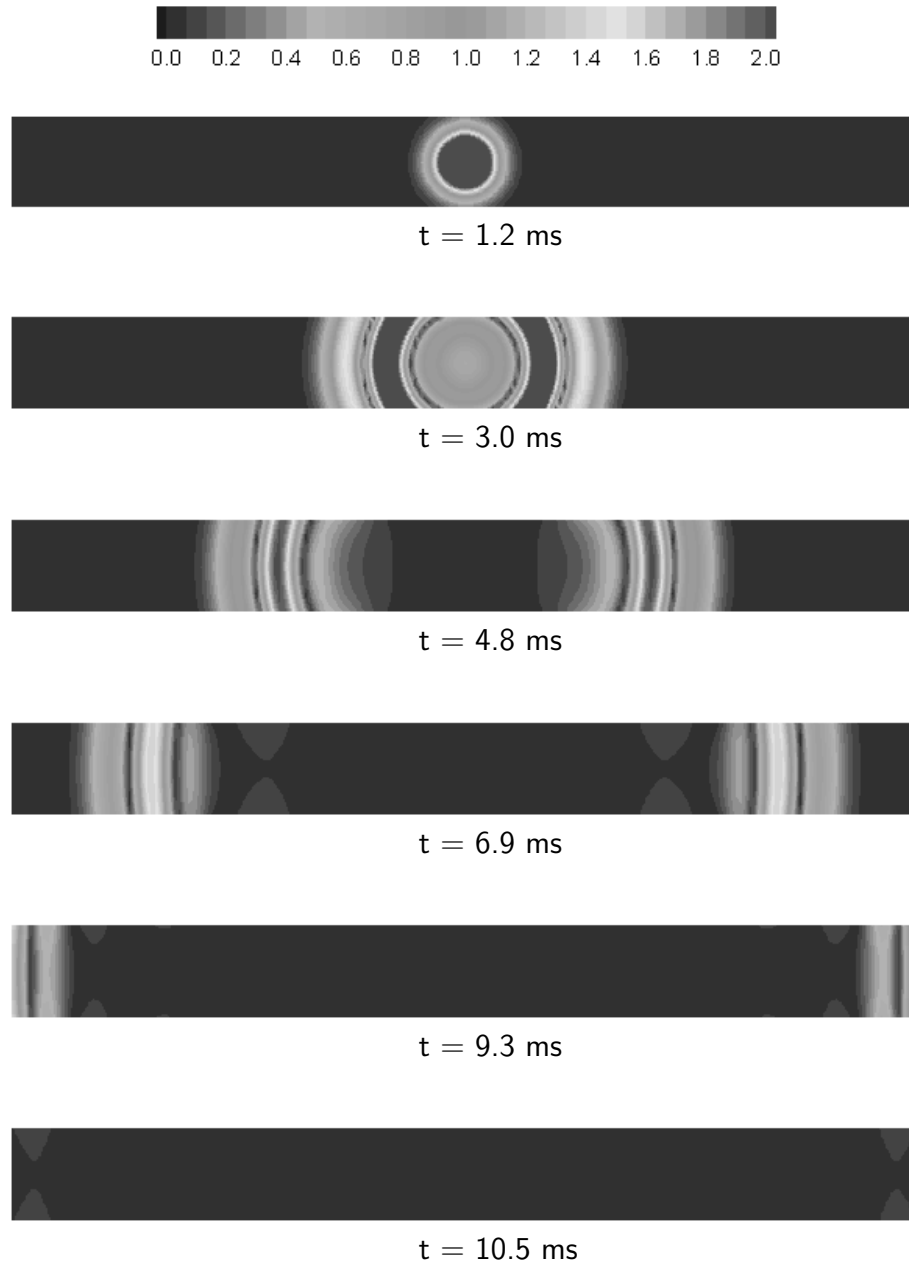


Figure 4.14: High aspect ratio computational domain (3 CFABC layers).

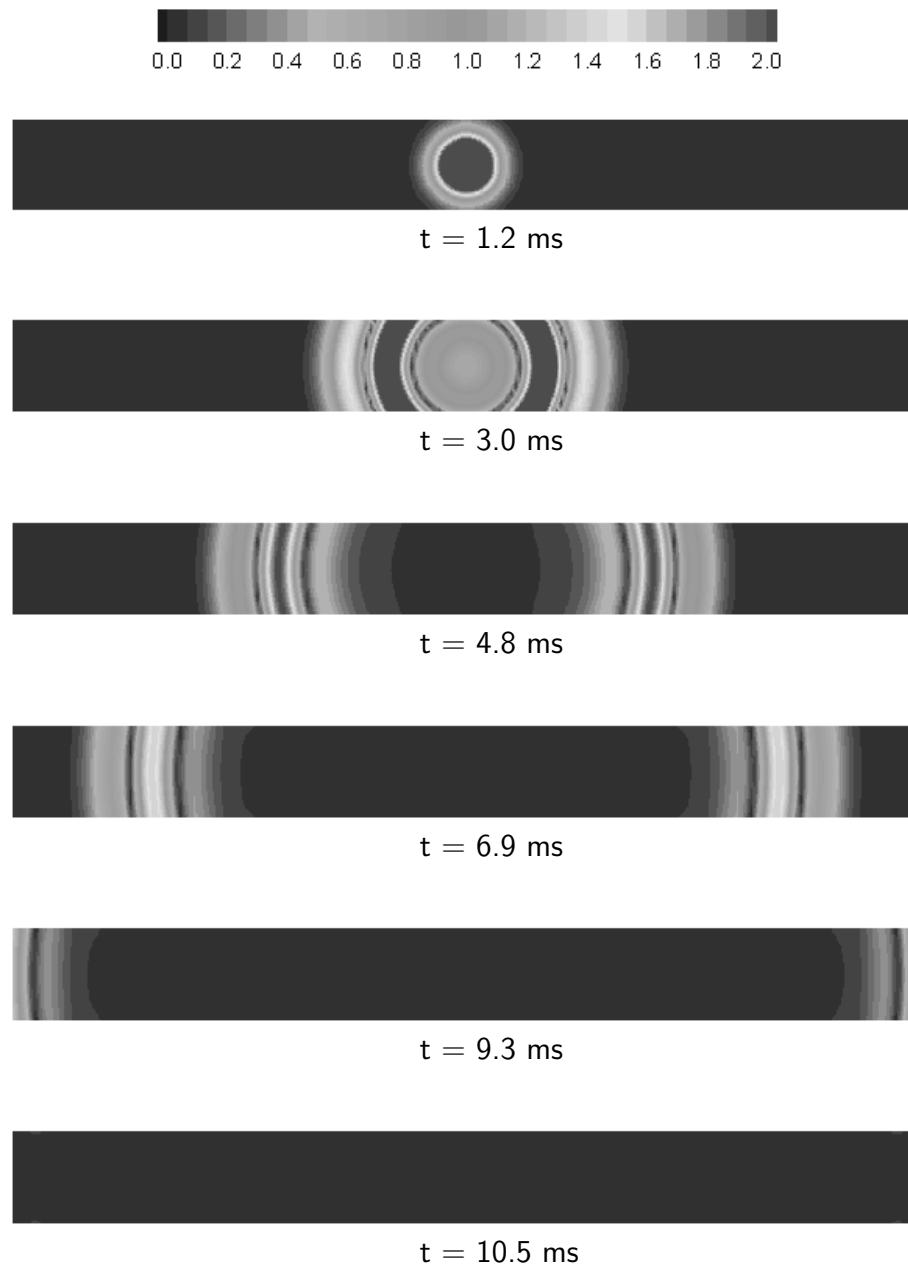


Figure 4.15: High aspect ratio computational domain (9 CFABC layers).

4.2 Pseudo-Explicit and Explicit Computations

4.2.1 One-dimensional Case

Here, we consider the case of a bar on elastic foundation as shown in Fig. 4.16

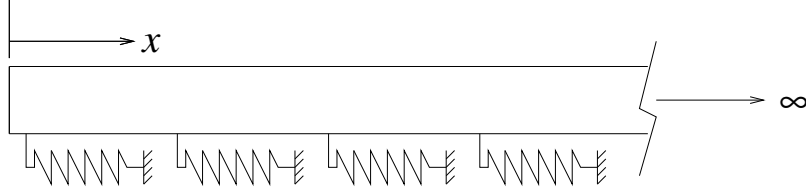


Figure 4.16: Bar on elastic foundation problem.

First, we reduce the spatial dimension from two to one by taking the Fourier transform in z of (2.1) giving

$$-\mu u_{xx} + \mu l^2 u + \rho u_{tt} = 0 \quad (4.3)$$

Multiplying the above equation with the virtual displacement δu , and subsequently integrating over the element length L using integration-by-parts, we obtain

$$\left[-\mu \delta u \frac{du}{dx} \right]_{\delta\Omega^e} + \int_0^L \frac{d\delta u}{dx} \mu \frac{du}{dx} dx + \int_0^L \delta u \mu l^2 u dx + \int_0^L \rho \delta u u_{tt} dx = 0 \quad (4.4)$$

The first term represents the boundary term that will be replaced by the continued fraction absorbing boundary condition. For the absorbers, we reduce the spatial dimension from two to one by taking the Fourier transform in z of (2.28) to obtain

$$\begin{Bmatrix} -\mu \frac{\partial u}{\partial x} \\ 0 \\ \vdots \\ 0 \end{Bmatrix} = \sqrt{\rho\mu} (\mathbf{A} + \mathbf{B}) \frac{\partial}{\partial t} \begin{Bmatrix} u \\ u_1 \\ \vdots \\ u_{n-1} \end{Bmatrix} + \mu l^2 \sqrt{\frac{\mu}{\rho}} \mathbf{B} \int \begin{Bmatrix} u \\ u_1 \\ \vdots \\ u_{n-1} \end{Bmatrix} dt \quad (4.5)$$

Then, the standard finite element formulation leads to the following semi-discrete equation at time $n\Delta t$ given by (3.55) with the element stiffness matrix as

$$\mathbf{k}^e = \frac{\mu}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \mu l^2 \begin{bmatrix} L/3 & L/6 \\ L/6 & L/3 \end{bmatrix} \quad (4.6)$$

where the second matrix on the right-hand side is due to the elastic foundation having a stiffness per unit length of μl^2 , and

$$\mathbf{C} = \sqrt{\rho\mu}(\mathbf{A} + \mathbf{B}) \quad (4.7)$$

$$\mathbf{R} = \mu\sqrt{\frac{\mu}{\rho}}\mathbf{B} \quad (4.8)$$

The problem in Fig. 4.16 is set up where the interior is represented using five elements and the exterior on the right is represented using five CFABC's with $\theta_{2k} = 0$. The parameters are $\mu = \rho = L = l = 1.0$. For the explicit procedure, an eigenvalue calculation on the element level gives the critical time step of $\Delta t_c = 0.9608$. A sine-squared load of

$$f(t) = \begin{cases} \sin^2(t) & \text{if } 0 \leq t \leq \pi \\ 0 & \text{otherwise} \end{cases} \quad (4.9)$$

is applied at Node 1 on the left.

Two time steps are chosen. The first time step is chosen near the stability limit at $\Delta t = 0.96$ and the second time step is chosen at $\Delta t = 0.0001$. The displacement histories for the two time steps at the interior-exterior boundary node (Node 6) is plotted as shown in Figs. 4.17 and 4.18. Simple mass lumping is used for both the implicit and explicit methods, obtained using the element mass matrix of

$$\mathbf{m}^e = \rho L \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} \quad (4.10)$$

We observe that due to the dispersive effects caused by the elastic foundation, the solution at near the critical time step $\Delta t = 0.96$ from both the implicit and explicit methods are erroneous. When the time step is decreased to $\Delta t = 0.0001$, both solutions converge as shown in Fig. 4.18.

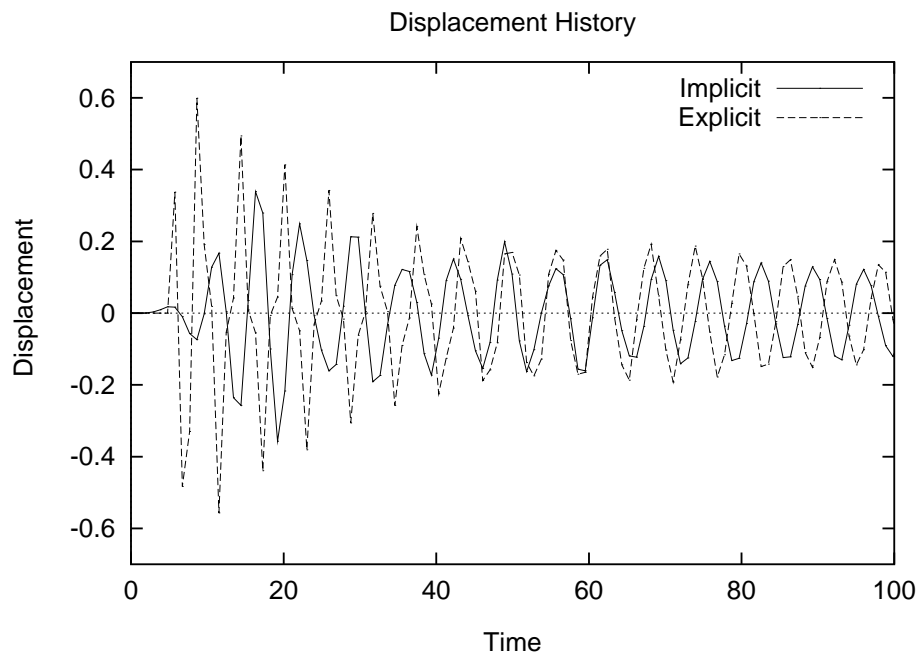


Figure 4.17: Displacement history obtained using $\Delta t = 0.96$.

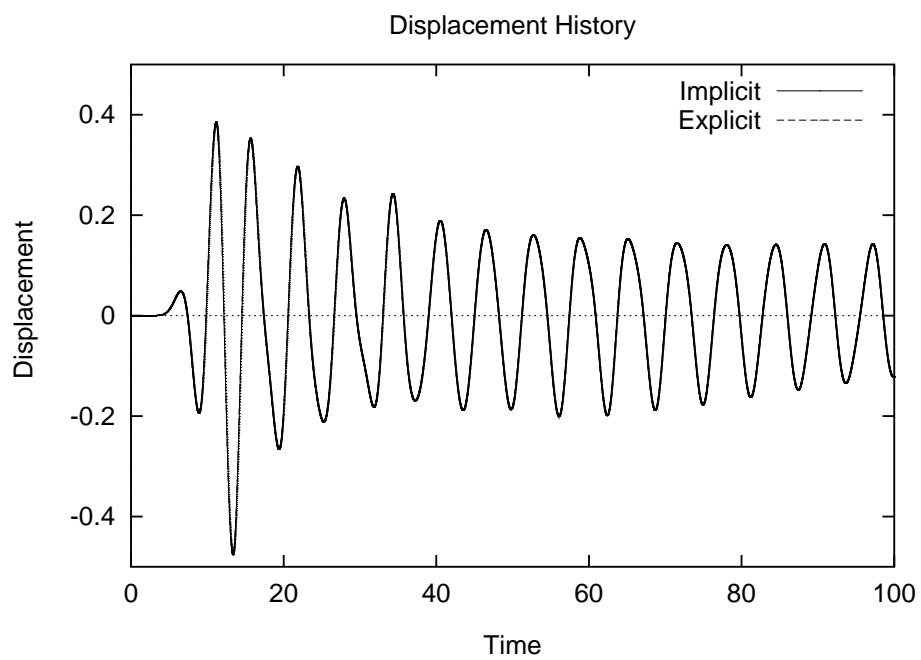


Figure 4.18: Displacement history obtained using $\Delta t = 0.0001$.

Next, we show that the splitting procedure leads to inconsistency. Two cases are considered, namely, $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$ and $\theta_{2k} = 0^\circ, 18^\circ, 36^\circ, 54^\circ, 72^\circ$. The splitting procedure used sums, for each row, a fraction of the off-diagonal terms and the diagonal to obtain \mathbf{C}^I and the remaining matrix $\mathbf{C} - \mathbf{C}^I = \mathbf{C}^E$. The following MATLAB subroutine is used to compute \mathbf{C}^I .

```
function
[C_Implicit]=Get_C_Implicit(ndof,split_factor_damp,DAMP,div)
C_Implicit = zeros(ndof,ndof);

for j=1:ndof
    sum = 0;
    for k = 1:ndof
        if ( j ~= k )
            sum = sum + DAMP(j,k)/div;
        else
            sum = sum + DAMP(j,k);
        end
    end
    C_Implicit(j,j) = sum;
end
```

The variable `div` allows one to select the amount of the off-diagonal terms. For the above two cases, we use the value of `div = 1.35` to achieve a very close fit to the implicit solution. This value is obtained using a trial-and-error process. A time step of $\Delta t = 0.0001$ is used and each of the two cases are plotted against its corresponding implicit as shown in Figs. 4.19 and 4.20. Each of the cases leads to a different maximum value of λ^h in 3.73 and thus the difference in the solutions. Close-ups of the two solutions are shown in Figs. 4.21 and 4.22, showing that the explicit and implicit solutions do not match. Further reduction in the time step size did not result in solution convergence.

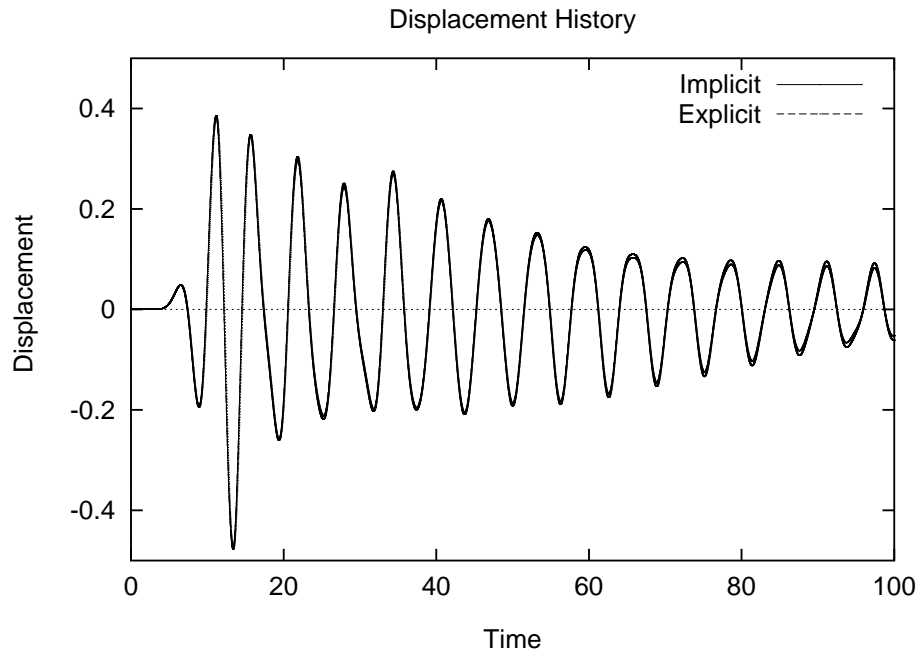


Figure 4.19: Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$, $\text{div} = 1.35$.

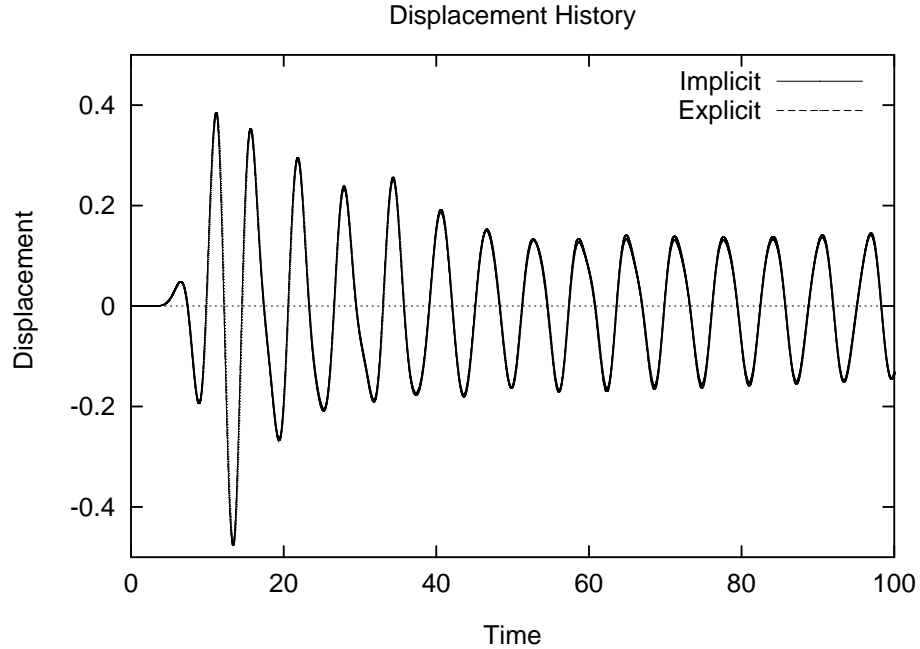


Figure 4.20: Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 18^\circ, 36^\circ, 54^\circ, 72^\circ$, $\text{div} = 1.35$.

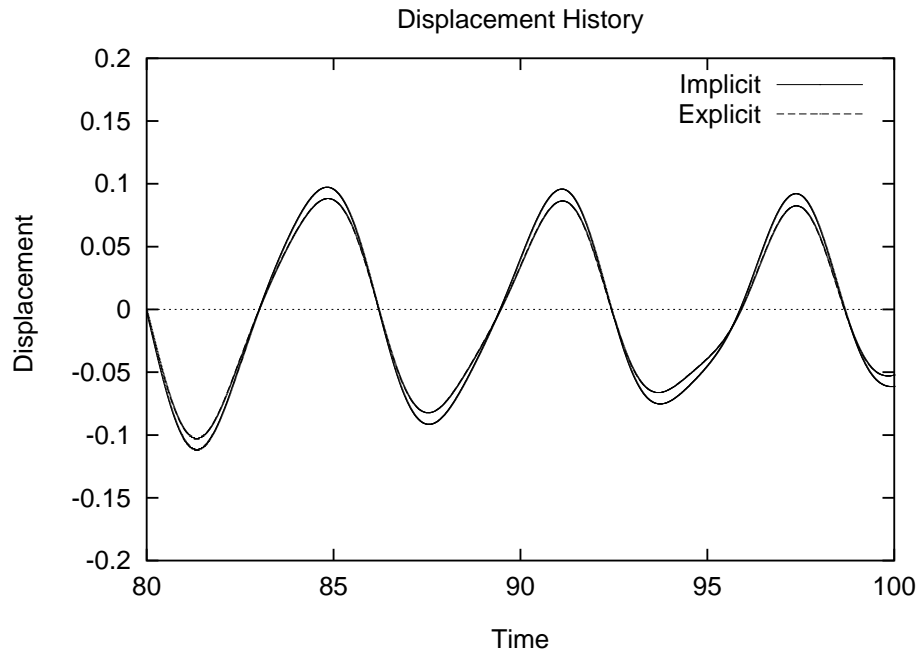


Figure 4.21: Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$, $\text{div} = 1.35$.

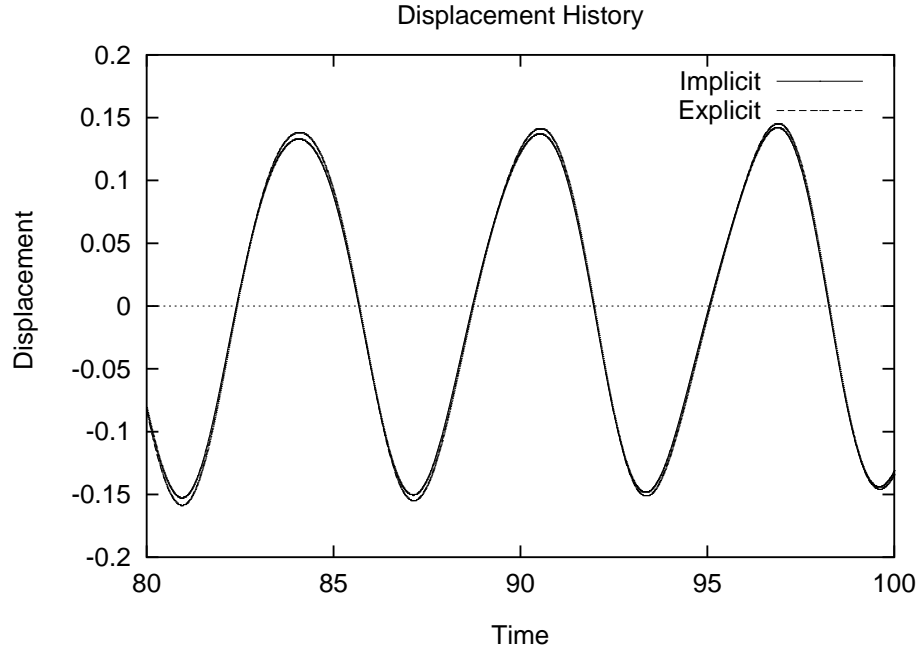


Figure 4.22: Inconsistency due to splitting procedure - $\theta_{2k} = 0^\circ, 18^\circ, 36^\circ, 54^\circ, 72^\circ$, $\text{div} = 1.35$.

Other conceivable ways splitting were also explored and the results remain inconsistent.

4.2.2 Two-dimensional Case

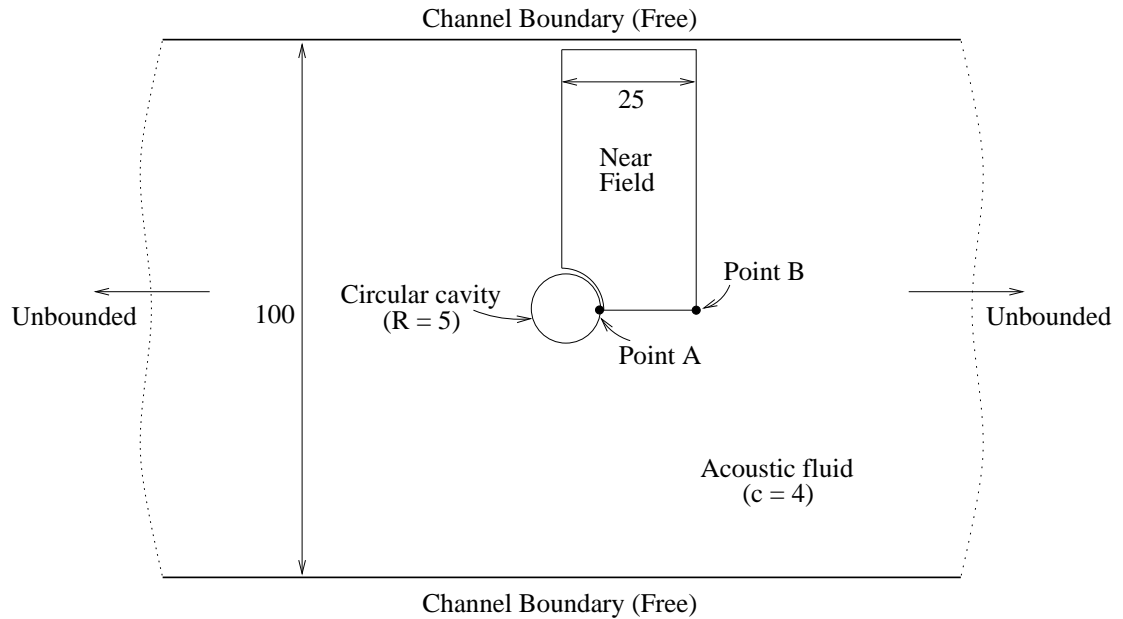


Figure 4.23: Wave propagation in an acoustic channel around a cavity.

In this problem, a cylindrical cavity of radius $R = 5$ is embedded in an infinite channel of width. The channel is filled with acoustic fluid with wave velocity $c = 4$. A squared since radial velocity of (4.9) is applied on the cavity. The problem setting is shown in Fig. 4.23 We are interested in the displacement history on the cavity wall. The wave propagation phenomenon around the cavity is modeled using a 50×100 rectangular region surrounding the cavity. Due to the symmetry of the problem, only one quarter of the near-field will be examined as shown in Fig. 4.23.

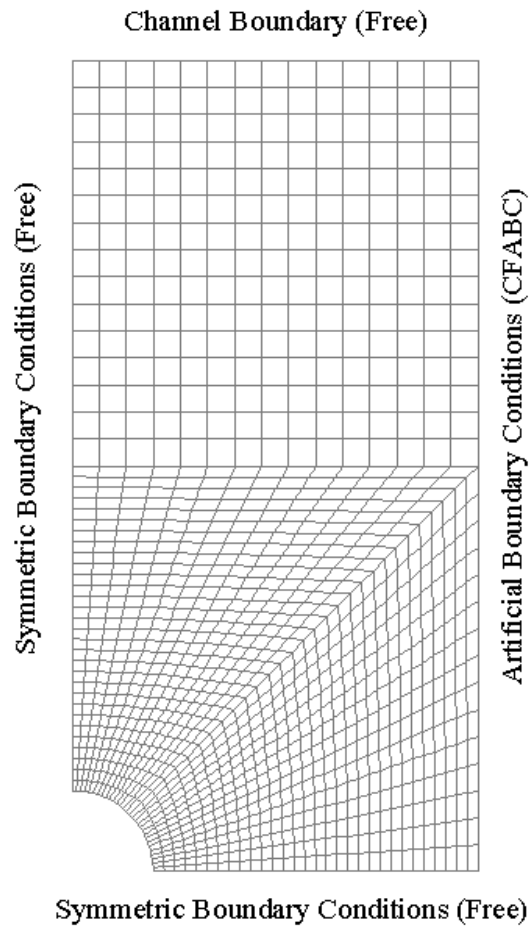


Figure 4.24: Finite element mesh for the channel problem.

The near-field (one quarter) is discretized using 1211 nodes and 1125 bilinear finite elements as shown in Fig. 4.24. Using mass lumping, an eigenvalue calculation on the smallest element gives a critical time step of $\Delta t_c = 0.0764$. We consider the following two cases at Nodes 1 and 31 (Points A and B , respectively):

- Implicit (lumped mass) and explicit (lumped mass) solutions using a time step $\Delta t = 0.07$.
- Implicit (lumped mass) and explicit (lumped mass) solutions at time step $\Delta t = 0.001$.

For both cases, we use three CFABC layers at $\theta_{2k} = 0^\circ, 0^\circ, 0^\circ$. As can be seen from Figs. 4.25 to 4.28, the implicit and explicit solutions using $\Delta t = 0.07$ are very close to each other. This is because, unlike the bar on elastic foundation problem, this two-dimensional system does not have any dispersive effects. At $\Delta t = 0.001$, both solutions match exactly as expected. We repeat the above computations using the pseudo-explicit procedure for $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$. The solutions from the pseudo-explicit scheme are as shown in Figs. 4.29 to 4.32, and both the implicit and explicit solutions converge when the time step is decreased from $\Delta t = 0.07$ to $\Delta t = 0.001$.

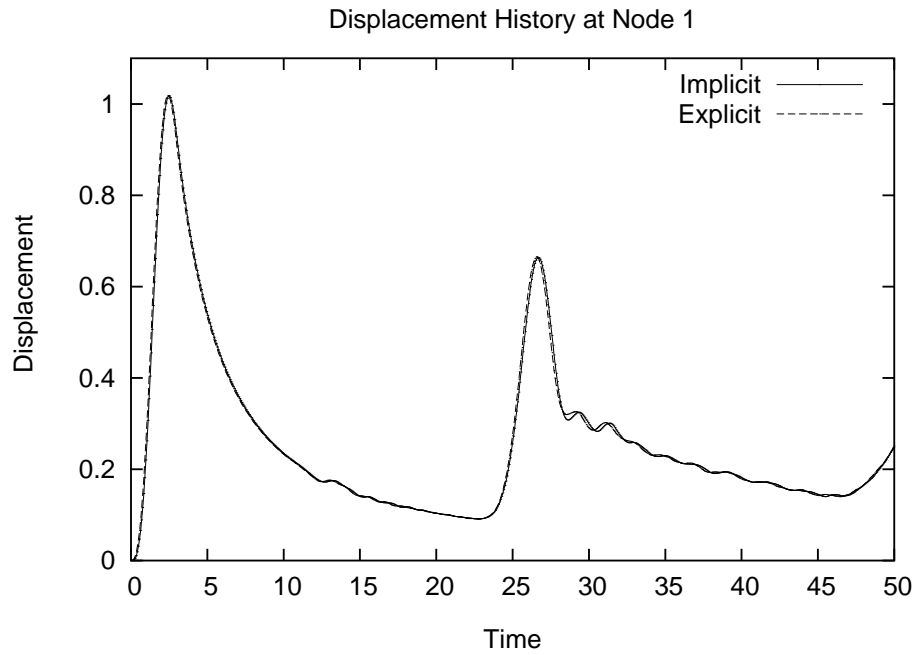


Figure 4.25: Explicit method; $\theta_{2k} = 0^\circ, 0^\circ, 0^\circ$: Displacement history at Node 1 (Point A) using $\Delta t = 0.07$.

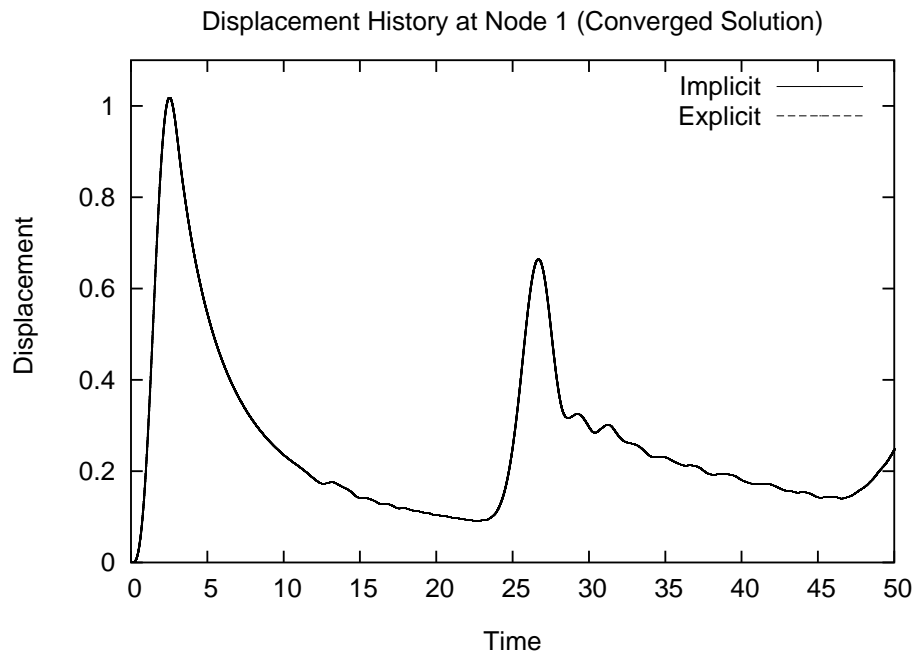


Figure 4.26: Explicit method; $\theta_{2k} = 0^\circ, 0^\circ, 0^\circ$: Displacement history at Node 1 (Point A) using $\Delta t = 0.001$.

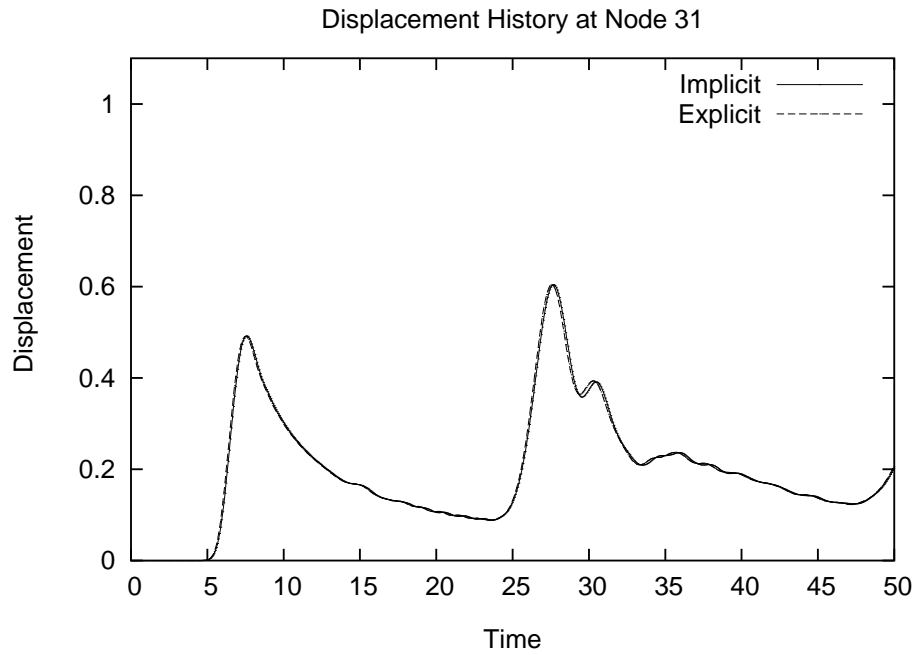


Figure 4.27: Displacement history at Node 31 (Point B) using $\Delta t = 0.07$.

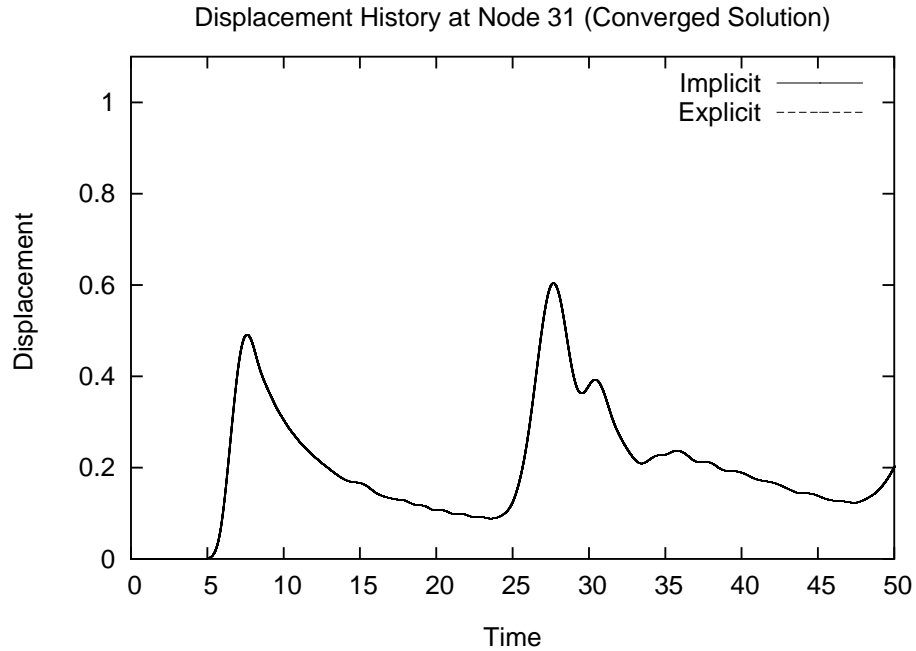


Figure 4.28: Displacement history at Node 31 (Point B) using $\Delta t = 0.001$.

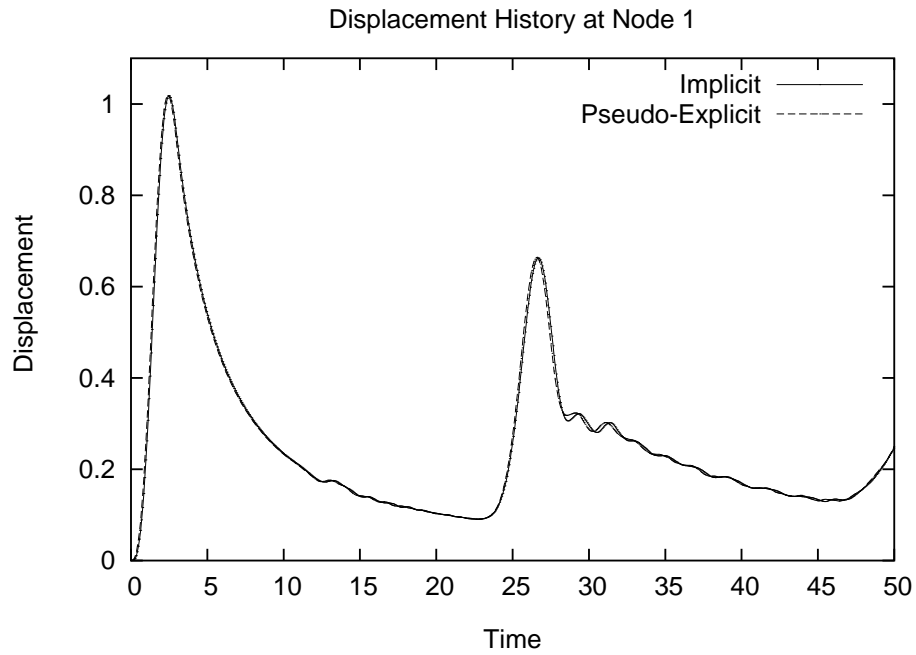


Figure 4.29: Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 1 (Point A) using $\Delta t = 0.07$.

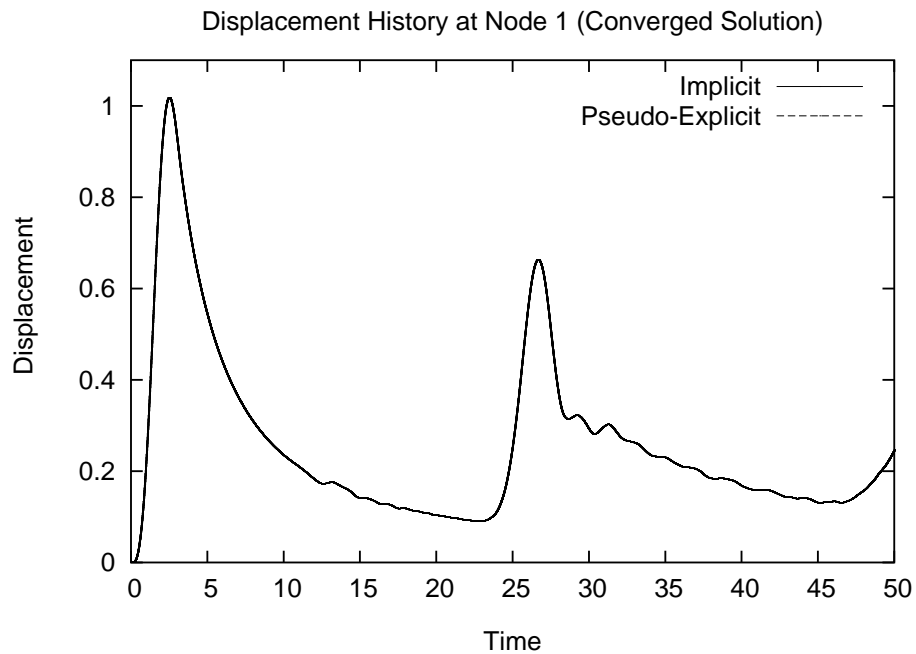


Figure 4.30: Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 1 (Point A) using $\Delta t = 0.001$.

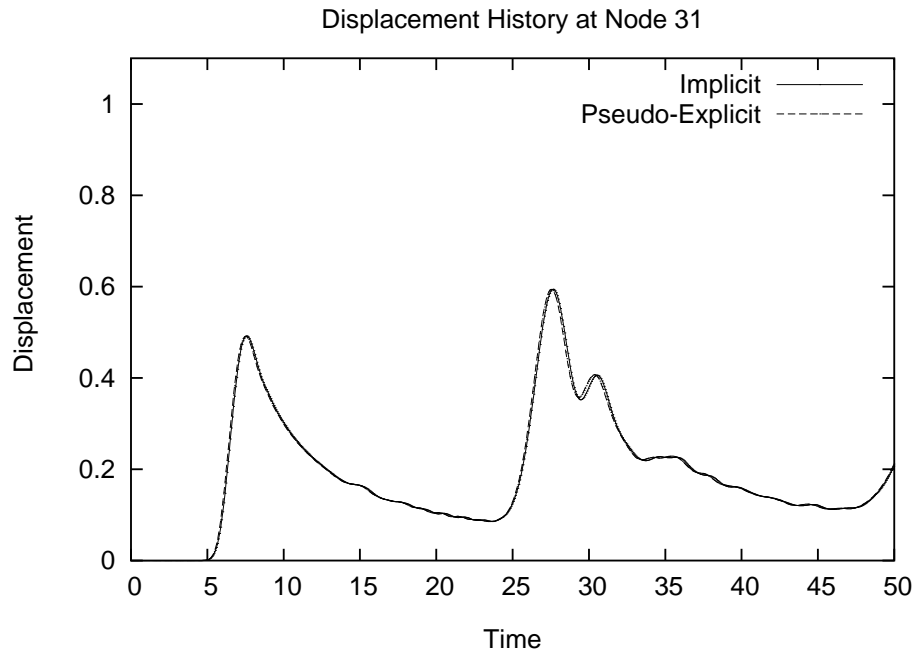


Figure 4.31: Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 31 (Point B) using $\Delta t = 0.07$.

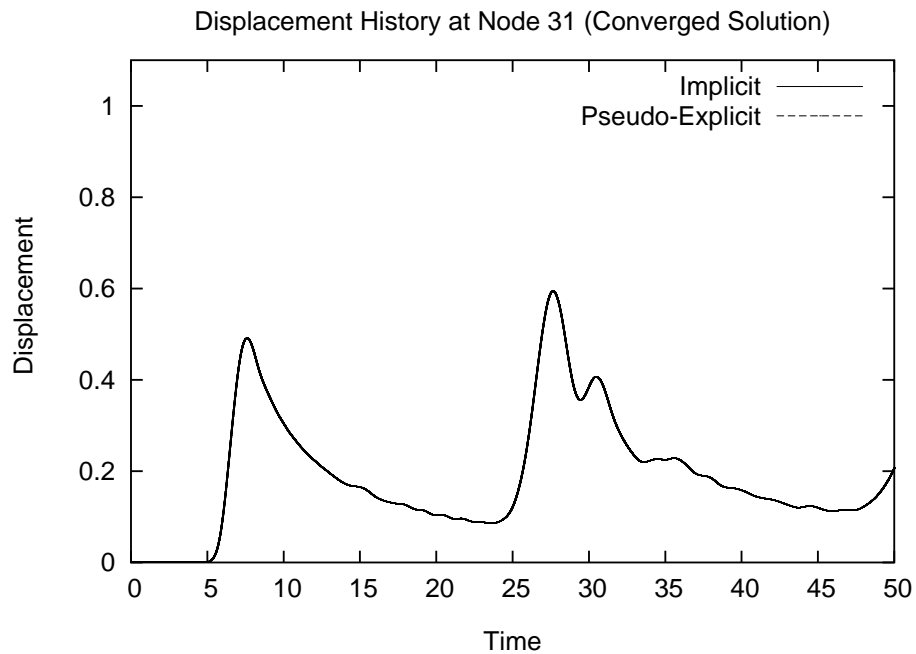


Figure 4.32: Pseudo-explicit method; $\theta_{2k} = 0^\circ, 30^\circ, 60^\circ$: Displacement history at Node 31 (Point B) using $\Delta t = 0.001$.

Chapter 5

Conclusions

5.1 Summary

We have presented a new derivation of CFABC and its extension to corner regions that can be implemented very elegantly in a C^0 finite element setting. The CFABC is shown to be a special case of the discrete PML and when viewed in this manner, facilitates the formulation of the corner absorbing boundary conditions. With pure imaginary stretching, the PML discretization reduces to the problem of rational approximation of the square-root on $[0, 1]$ widely used in differential-equation-based ABC's. Thus, material-based and differential-equation-based ABC's, which have resulted from different approaches, are fundamentally similar and can be considered to be grouped under one larger category where complex stretching functions are used.

The importance of one-point integration to represent the half-space in a consistent manner is highlighted. The corner CFABC's gives contribution only to the stiffness matrix \mathbf{K} which is easily implemented. Numerical results are presented showing that satisfactory performance can be achieved using three CFABC layers, which corresponds to two auxiliary variables.

For the case of $\theta_{2k} \neq 0^\circ$, it has been shown that the splitting of the damping matrix into the implicit and explicit parts leads to inconsistency. By performing lumping of the damping terms in the direction along the edge boundaries and taking advantage

of the resulting tridiagonal blocks, a pseudo-explicit scheme is can be employed to efficiently solve the system. If θ_{2k} are chosen to be zero to allow for a diagonal damping matrix, a full explicit procedure can be obtained and good absorption capabilities can still be achieved using a few additional CFABC layers. This incurs a slight increase in the total number of degree-of-freedom of the system. This disadvantage, however, is negated by the advantages of extremely fast computation and no requirement for storing matrices. Hence, the overall efficiency is still better than the original implicit scheme.

CFABC has the best of both worlds, combining the accuracy of rational or multi-directional absorbers and the flexibility of PML. CFABC is applicable for any convex polygonal computational domain as well as stratified media. CFABC has also been extended to elastic waves [21]. With the improvements proposed in this thesis, CFABC's will be at least as powerful if not superior than the existing ABC's for transient wave propagation analysis.

5.2 Future Work

Ongoing investigations related to CFABC include the following: The investigation of the stability of CFABC for Lamb modes in elastic layers, extension of CFABC to terminate poroelastic media, and the study of errors due to the exterior discretization. CFABC has been shown to be extendible to any second-order wave equations [20] and thus, its use for other forms of wave equations should be explored. Finally, the extension of CFABC to curved boundaries and three-dimensional problems will also be looked into.

List of References

- [1] Sergey Asvadurov, Vladimir Druskin, Murthy N. Guddati, and Leonid Knizhnerman. On optimal finite-difference of PML. *Society for Industrial and Applied Mathematics*, **41** (2003), 287–305.
- [2] A. Bamberger, P. Joly, and J. E. Roberts. Second order absorbing boundary conditions for the wave equation: a solution for the corner problem. *SIAM J. Numer. Anal.*, **27** (1990), 323–352.
- [3] K. J. Bathe. *Finite Element Procedures*. Prentice-Hall, Englewood Cliffs, 1996.
- [4] J. P. Bérenger. A perfectly matched layer for absorption of electromagnetic waves. *Journal of Computational Physics*, **114** (1994), 185–200.
- [5] J. P. Bérenger. Improved PML for the FDTD solution of wave-structure interaction problems. *IEEE Transactions on Antennas and Propagation*, **45** (1997), 466–473.
- [6] J. P. Bérenger. Numerical reflection from FDTD-PMLs: A comparison of the split PML with the unsplit and CFS PMLs. *IEEE Transactions on Antennas and Propagation*, **45** (1997), 466–473.
- [7] R. Courant, K. O. Friedrichs, and H. Lewy. Über die partiellen Differenzgleichungen der mathematischen Physik. *Math. Ann.*, **100** (1928), 32–74.

- [8] W. C. Chew and W. H. Weedon. A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Micro. Opt. Tech. Lett.*, **7** (1994), 599–604.
- [9] W. Chew, J. Jin, and E. Michielssen. Complex coordinate stretching as a generalized absorbing boundary condition. *Microwave Opt. Tech. Lett.*, **15** (1997), 363–369.
- [10] F. Collino. High order absorbing boundary conditions for wave propagation models: Straight line boundary and corner cases. In R. Kleinman et al., editor, *Proceedings of 2nd Int. Conf. on Math. and Numer. Aspects of Wave Prop. Phen.*, SIAM (1993), 161–171.
- [11] F. Collino and P. Monk. Optimizing the perfectly matched layer. *Comput. Methods Appl. Mech. Engrg.*, **164** (1998), 157–171.
- [12] F. Collino and C. Tsogka. Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media. *Geophysics*, **66** (2001), 294–307.
- [13] Steven A. Cummer. A simple, nearly perfectly matched layer for general electromagnetic media. *IEEE Microwave and Wireless Components Letters*, **13** (2003), 128–130.
- [14] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Math. Comput.*, **31** (1977), 629–651.
- [15] B. Engquist and A. Majda. Radiation boundary conditions for acoustic and elastic wave calculations. *Comm. Pure Appl. Math*, **32** (1979), 313–357.
- [16] Jiayuan Fang and Zhonghua Wu. Generalized perfectly matched layer for the absorption of propagating and evanescent waves in lossless and lossy media. *IEEE Transactions on Microwave Theory and Techniques*, **44** (1996), 2216–2221.

- [17] M. N. Guddati and J. L. Tassoulas. FEP++: An Object Oriented Finite Element Program Linear and Nonlinear Transient Analysis. *unpublished technical note*, September 1995.
- [18] M. N. Guddati. Comparison of continued fraction absorbing boundary conditions with perfectly matched layers. In *EM-2000*. ASCE Engineering Mechanics Conference, Austin, TX, May 2000.
- [19] M. N. Guddati and J. L. Tassoulas. Continued fraction absorbing boundary conditions for the wave equation. *Journal of Computational Acoustics*, **18** (2000), 139–156.
- [20] M. N. Guddati. Arbitrarily Wide Angle Wave Equations for complex media. *Wave Motion (submitted)*.
- [21] M. N. Guddati and Md. Anwar Zahid. Continued fraction absorbing boundary conditions for elastic waves (*in preparation*).
- [22] T. Hagstrom. New results on absorbing layers and radiation boundary conditions. In *Proceedings of the LMS-Durham Symposium on Scattering*, Springer-Verlag (2003), to appear.
- [23] R. L. Higdon. Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation. *Math. Comput.*, **47** (1986), 437–459.
- [24] R. L. Higdon. Numerical absorbing boundary conditions for the wave equation. *Math. Comput.*, **49** (1987), 65–90.
- [25] T. J. R Hughes. *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*. Dover Publishers, New York, 2000
- [26] J. M. Jin and W. C. Chew. Combining PML and ABC for the finite-element analysis of scattering problems. *Micro. Opt. Tech. Lett.*, **12** (1996), 192–197.

- [27] L. F. Kallivokas and J. Bielak. Time-domain analysis of transient structural acoustics problems based the finite element method and a novel absorbing boundary element. *J. Acoust. Soc. Am.*, **12** (1996), 192–197.
- [28] R. G. Keys. Absorbing boundary conditions for acoustic media. *Geophysics*, **50** (1985), 892–902.
- [29] M. Kuzuoglu and R. Mittra. Frequency dependence of the constitutive parameters of casual perfectly matched anisotropic absorbers. *IEEE Microwave and Guided Wave Letters*, **6** (1996), 447–449.
- [30] E. L. Lindman. “Free-space” boundary conditions for the time dependent wave equation. *Journal of Computational Physics*, **18** (1975), 66–78.
- [31] J. A. Roden and S. D. Gedney. Convolutional PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. *Microwave and Optical Technology Letters*, **27** (2000), 334–339.
- [32] Z. S. Sacks, D. M. Kingsland, R. Lee, and J.-F. Lee. A perfectly matched anisotropic absorber for use as an absorbing boundary condition. *IEEE Trans. Antennas Propag.*, **43** (1995), 1460–1463.
- [33] A. Safjan. Highly accurate non-reflecting boundary conditions finite element simulation of transient acoustics problems. TICAM Report 95-11, Texas Inst. for Comp. and Appl. Math., The University of Texas at Austin, (1995).
- [34] K. L. Shlager and J. B. Schneider. A Selective Survey of the Finite-Difference Time-Domain Literature. *IEEE Antennas and Propagation Magazine*, **37** (1995), 39–56.
- [35] O. Vacus. Mathematical analysis of absorbing boundary conditions for the wave equation: The corner problem. *Math. Comput.*, (2002), to appear.

- [36] J.-L. Vay. Asymmetric perfectly matched layer for the absorption of waves. *Journal of Computational Physics*, **183** (2002), 367–399.
- [37] Md. A. Zahid, K. W. Lim, K. Piyasatjaboon, M. N. Guddati. Time stepping techniques for unbounded domains using continued fraction absorbing boundary conditions (*in preparation*).

Appendix A

Derivation of The Dynamic Stiffness Matrix For Corner ABC's

For a four-node iso-parametric element, the shape functions are [3, 25]

$$\begin{aligned}H_1 &= \frac{1}{4} (1 - r) (1 - s) \\H_2 &= \frac{1}{4} (1 + r) (1 - s) \\H_3 &= \frac{1}{4} (1 + r) (1 + s) \\H_4 &= \frac{1}{4} (1 - r) (1 + s)\end{aligned}\tag{A.1}$$

The displacement can be interpolated as

$$u = \sum_{j=1}^4 H_j u_j = \mathbf{H}\mathbf{u}\tag{A.2}$$

and

$$\mathbf{H} = \frac{1}{4} \begin{bmatrix} (1 - r) (1 - s) & (1 + r) (1 - s) & (1 + r) (1 + s) & (1 - r) (1 + s) \end{bmatrix}\tag{A.3}$$

Using the chain rule,

$$\frac{\partial}{\partial r} = \frac{\partial}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial}{\partial y} \frac{\partial y}{\partial r}\tag{A.4}$$

$$\frac{\partial}{\partial s} = \frac{\partial}{\partial x} \frac{\partial x}{\partial s} + \frac{\partial}{\partial y} \frac{\partial y}{\partial s}\tag{A.5}$$

or, in matrix form

$$\begin{Bmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial s} \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{Bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{Bmatrix} \quad (\text{A.6})$$

$$\frac{\partial}{\partial \mathbf{r}} = \mathbf{J} \frac{\partial}{\partial \mathbf{x}} \quad (\text{A.7})$$

which gives

$$\frac{\partial}{\partial \mathbf{x}} = \mathbf{J}^{-1} \frac{\partial}{\partial \mathbf{r}} \quad (\text{A.8})$$

where \mathbf{J} is the Jacobian. Using the coordinates in Fig. 3.6, the Jacobian is obtained as

$$\mathbf{J} = \begin{bmatrix} a & 0 \\ b \cos \alpha & b \sin \alpha \end{bmatrix} \quad (\text{A.9})$$

and

$$\det(\mathbf{J}) = a b \sin \alpha \quad (\text{A.10})$$

The inverse of the Jacobian is

$$\mathbf{J}^{-1} = \frac{1}{\det(\mathbf{J})} \begin{bmatrix} b \sin \alpha & 0 \\ -b \cos \alpha & a \end{bmatrix} \quad (\text{A.11})$$

and

$$(\mathbf{J}^{-1})^T \mathbf{J}^{-1} = \frac{1}{[\det(\mathbf{J})]^2} \begin{bmatrix} b^2 & -a b \cos \alpha \\ -a b \cos \alpha & a^2 \end{bmatrix} \quad (\text{A.12})$$

The stiffness matrix is obtained using the term

$$\int_{\Omega} \mu \nabla(\delta u)^T \nabla u \, d\Omega \quad (\text{A.13})$$

from the weak form in (3.38). We have

$$\nabla u = \begin{Bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{Bmatrix} u = \mathbf{J}^{-1} \frac{\partial}{\partial \mathbf{r}} u = \mathbf{J}^{-1} \frac{\partial \mathbf{H}}{\partial \mathbf{r}} \mathbf{u} \quad (\text{A.14})$$

where

$$\frac{\partial \mathbf{H}}{\partial \mathbf{r}} = \begin{bmatrix} \frac{\partial H_1}{\partial r} & \frac{\partial H_2}{\partial r} & \frac{\partial H_3}{\partial r} & \frac{\partial H_4}{\partial r} \\ \frac{\partial H_1}{\partial s} & \frac{\partial H_2}{\partial s} & \frac{\partial H_3}{\partial s} & \frac{\partial H_4}{\partial s} \end{bmatrix} = \frac{1}{4} \begin{bmatrix} -(1-s) & (1-s) & (1+s) & -(1+s) \\ -(1-r) & -(1+r) & (1+r) & (1-r) \end{bmatrix} \quad (\text{A.15})$$

The above relations and the imaginary lengths as defined in (3.41) are substituted into (3.39). The element stiffness matrix in the frequency domain is then obtained after using 1×1 integration. The element mass matrix in the frequency domain is obtained by using (A.3) in (3.43) and subsequently employing 1×1 integration. The dynamic stiffness matrix for the corner is then follows from

$$\mathbf{k}^e - \omega^2 \mathbf{m}^e \quad (\text{A.16})$$

Appendix B

XFEP++: Documentation and User Manual

B.1 Introduction

XFEP++ is a C++ explicit code for the analysis of transient wave propagation. FEP++ can be considered as an extension of the original implicit FEP++ developed by Guddati [17]. Apart from standard wave propagation analysis, XFEP++ also contains the implementation of the continued fraction absorbing boundary conditions, with only chosen absorption angles of zero. Currently, the implementation is only limited to the case of straight absorbing boundaries. Further extension of this scheme is a subject of future research.

B.2 Program/Feature Limitations

For the future improvement of this code, we list down the main limitations of the program as follows:

1. Straight absorbing boundaries only.
2. Works only for homogeneous interior.

3. Running time is a little sloppy due to each element have its own storage memory for the bulk modulus and density (which is identical in all elements since the material of the interior is homogeneous). No use of special solvers to take advantage of the sparse structure of the system matrix. The general and symmetric positive definite solvers are provided by the Math Kernel Library (MKL).
4. Works only for the four-node quadrilateral element in the interior.
5. Lack of special node generation schemes; every single node has to be generated *explicitly*.

B.3 Caveats and Important Notes

The generation of node data in XFEP++ is rather tedious. Currently, the best way to generate the node data is to firstly use the original implicit FEP++ to generate the check (*.chk) file. Subsequently, the data from the check file can be extracted using a spreadsheet tool such as Excel to generate a data file conforming to the data format presented in the last part of this documentation. Note that the data file format for XFEP++ does *not* conform to the data file for the implicit FEP++. The future developer should bear in mind this problem if he or she who to combine XFEP++ and FEP++ into one single program.

B.4 Class Files Description

There are a total of fifteen (15) C++ source (.cpp) files and their fifteen corresponding header (*.h) files. They are in, alphabetical order:

1. Acoustic.cpp
2. AcousticSet.cpp
3. EdgeAbsorber.cpp

4. EdgeAbsorberSet.cpp
5. Element.cpp
6. ElementSet.cpp
7. FESystem.cpp
8. Load.cpp
9. main.cpp
10. Matrix.cpp
11. Node.cpp
12. NodeSet.cpp
13. OutputTable.cpp
14. Quadrature.cpp
15. Vector.cpp

In what follows we provide the essential description of the classes for the finite elements and edge absorbing boundary elements used in the program. For clarity, we use two-row boxes to describe the classes. The first row contains the name of the class and the second row contains the main data member of interest. Member functions are not shown for clarity.

B.5 Finite Elements and Absorbing Elements

The use of inheritance and templates in the definition of the acoustic and absorbing elements is described. The *base class* is defined as `Element` class from which the *sub-classes* `Acoustic` finite element and `EdgeAbsorber` elements are derived from as shown in Fig. B.1.

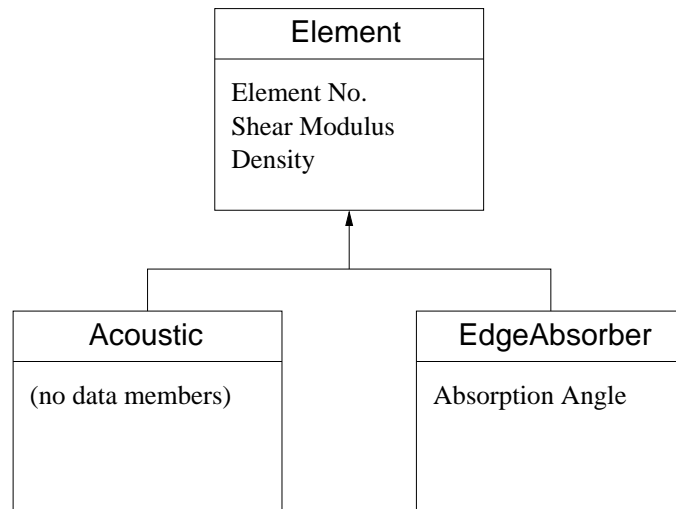


Figure B.1: Inheritance for the declaration of acoustic and absorbing elements.

B.6 Element Sets

The class `ElementSet` is an array class that contains either the `Acoustic` finite elements or `EdgeAbsorber` elements. The type of element is specified using the C++ template facility. For example, the `AcousticSet` is an array containing the `Acoustic` finite elements and is written in C++ as:

```

class AcousticSet : public ElementSet<Acoustic>
{

    // Member functions

};
  
```

which means that the `AcousticSet` is *derived* from the `ElementSet` class with the *typename* specified as `Acoustic`. Similarly, the `EdgeAbsorberSet` is derived from the `ElementSet` class with the *typename* specified as `EdgeAbsorber`, written in C++ as:

```

class EdgeAbsorberSet : public ElementSet<EdgeAbsorber>
  
```

```

{
    // Member functions
    ...
};

```

Note: the arrays of finite elements and absorbing elements are data members of the `ElementSet` class and the sub-classes `AcousticSet` and `EdgeAbsorberSet` contains only the necessary member data and functions for the manipulation of the elements (finite or absorbing) contained in the base class `ElementSet`. The relationship between the classes are as shown in Fig. B.2.

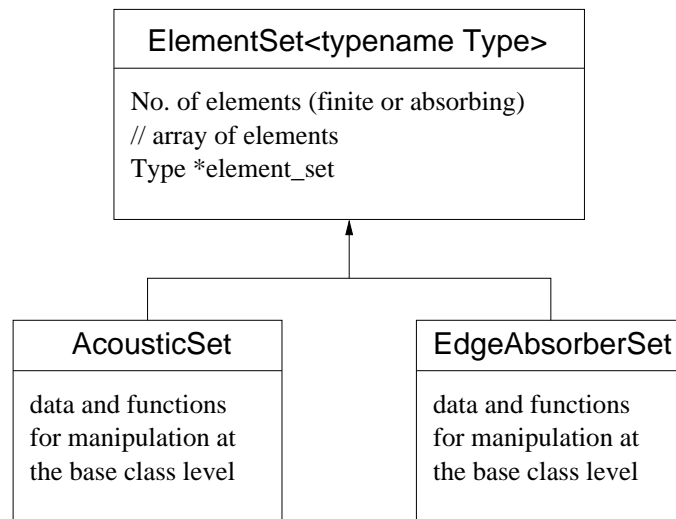


Figure B.2: Inheritance for the declaration of *sets* of acoustic and absorbing elements.

B.7 Nodes and Node Set

Like the `ElementSet` class, the `NodeSet` class is an array class used to contain the nodes defined by the `Node` class. In C++, this is written as:

```
class NodeSet {
```

```

        // Data members
private:
        int num_nodes; // number of nodes
        Node *nodeset; // array of Nodes

        // Member functions
        ...
};

```

B.8 Load Class

The Load class describes the applied loads on the nodes of the interior finite element mesh and holds the following information:

- The applied load history.
- The nodes where the load is applied on.

The applied load history is made up of the following items:

- Number of loads: `num_load_history`
- Vectors (arrays) for the time and corresponding load magnitude: `time_vector`, `force_vector`.

The nodes which are loaded are obtained from the following data members:

- Number of applied nodes: `num_loaded_nodes`.
- Vectors (arrays) containing the number of the applied nodes and corresponding magnification factor of the applied load: `loaded_nodes_vector`, `magnif_factor_vector`.

The above information are contained within the data members of the Load class, written in C++ as:

```

class Load {
        // Data members

```

```

private:

    int num_load_history;          // number of loads
    Vector<double> time_vector;    // time
    Vector<double> force_vector;  // magnitude of load

    int num_loaded_nodes;        // number of loaded nodes
    Vector<int>    loaded_nodes_vector; // applied node number
    Vector<double> magnif_factor_vector;

    // Member functions to retrieve loading
    // information at current time
    ...
};

```

B.9 Input Manual

General notes regarding the input:

- Input contains many **control blocks**. Each control block generally starts with a control string.
- You may have **comments** between (outside) control blocks bounded by % symbols on the left and the right.
- The string `_END_DATA_` **outside** the control block indicates the end of the input data. Everything after this line is completely ignored.
- Some control blocks need to end with the string `_END_`, while others do not need to. A redundant `_END_` outside the control block is ignored. So, it may be a good practice to end every control block with the string `_END_`, whether or not it is required.

In what follows, the input is divided into many input blocks. Most of the time, an input block is identical to a control block, but sometimes it may contain more than one control block. In what follows, a separate table describes each input block. The actual input that goes into the data file is the non-shaded part of the right column.

TITLE	
Control String	TITLE
Allowable values	Character string of length < 80

Table B.1: Title block.

LINEAR SOLUTION CONTROL	
Control String	PENALTY_NUMBER
Value	A number which is many orders of magnitude bigger than the elements of the stiffness matrix

Table B.2: Linear solution control block.

TIME STEPPING INFORMATION	
Control String	NEWMARK
Real values	Newmark parameters for time stepping ($\beta, \gamma, \alpha, \kappa$) - These values are NOT CURRENTLY USED in the program!
Control String	TIMESTEP_DATA
Integer value	Number of time steps
Control String	SPLIT_FACTORS
	For matrix splitting (split_C, split_R, split_K) - These values are NOT CURRENTLY USED in the program!
Control String	FREQUENCY_PRINT
Integer value	Frequency (number of time steps) of printing the history data requested in the OUTPUT REQUEST BLOCK.
Control String	FREQUENCY_PLOT
Integer value	Frequency (number of time steps) of output to the plot file to be used by TECPLOT - It is advised that this is a big number to avoid writing large amounts of data into the plot file.

Table B.3: Time stepping information block.

TIME HISTORY DEFINITIONS	
Control String	TIMEHISTORY_DATA
One or more lines are used to define each time history	Number of points
	For each point the following data is needed
	Time Value
End Flag	_END_

Table B.4: Time history definition block.

MATERIAL DEFINITION	
Control String	MATERIAL_DATA
Only one material is allowed	Material Type, Material Data
	ACOUSTIC Density Bulk modulus
End Flag	_END_

Table B.5: Material definition block.

NODE DEFINITION	
Control String	NODE_DATA
Input consists of many lines	Individual node
	Node number (IN ASCENDING ORDER!)
	Coordinates (X and Y values for the current implementation)
End Flag	_END_

Table B.6: Node definition block.

RESTRAINTS	
Control String	RESTRAINT_DATA
Input consists of many lines	On a single node
	Number of restraints
	Node number to be fixed (all non-specified nodes are free and IN ASCENDING ORDER!)
End Flag	_END_

Table B.7: Restraints data block.

NODAL LOADS	
Control String	NODAL_LOADS
Integer value	Number of loads
For each load (on each separate line!)	Node number
	Multiplication factor for the history
End Flag	_END_

Table B.8: Nodal loads data block.

FINITE ELEMENTS SET	
Control String	ELEMENT_DATA
Integer	Number of loads
Integer	Number of integration points in each direction 2 \rightarrow 2 \times 2 integration 1 \rightarrow 1 \times 1 integration
Element definition over many lines	Single element
	Element number being defined (IN ASCENDING ORDER!)
	Node numbers in the order defined for the element type
End Flag	_END_

Table B.9: Finite element data block.

STRAIGHT CONTINUED FRACTION ABSORBERS	
Control String	ELEMENT_DATA
Integer	Number of elements in the set
Element definition over many lines	Single element
	Element number being defined (CONTINUE THE NUMBERING FROM THE ELEMENT_DATA BLOCK AND IN ASCENDING ORDER!)
	Node numbers in the order defined for the element type
	Angle (θ_{2k} in degrees) associated with absorber
End Flag	_END_

Table B.10: Straight continued fraction absorbers data block.

SELECTIVE OUTPUT REQUEST	
Control String	OUTPUT_CONTROL
Integer	Number of outputs
Input consisting of many lines	Output associated with nodes
	Node number
	DOF control value = 1 for displacement (currently works only for displacement output!)
End Flag	_END_

Table B.11: Nodal loads data block.