

# **The Bipenalty Method for Explicit Structural Dynamics**

Jack Hetherington

PhD Thesis

University of Sheffield

Department of Civil and Structural Engineering

Supervisor: Harm Askes

February 2014



# Declaration of authorship

All work presented within this thesis is my own work, except where specific reference has been made to the work of others.

Signed:

Date:



# Summary

The penalty method is a versatile and widely used technique for imposing constraints in finite element analysis. Traditionally, it is implemented by adding artificial stiffness to the system equations. However, this leads to large eigenfrequencies being introduced, which, when used in explicit dynamics, can significantly decrease the critical time step of an analysis. This in turn vastly increases computational expense, increases the chances of instability, and generally leads to a less robust formulation. The mass penalty method, a less widely used penalty technique that operates on the mass matrix of a dynamic system, does not introduce large eigenfrequencies, but is less accurate and less versatile than the traditional stiffness penalty method.

In this thesis, the two methods are combined to form the bipenalty method. A general formulation is provided that can be used to describe any number of arbitrary multipoint constraints. Mathematical proofs are developed that show that the bipenalty method, like the stiffness penalty method, introduces extra eigenfrequencies into the system, but that they can be carefully controlled by manipulation of the stiffness and mass penalty parameters. It is shown that it is a simple matter to select these parameters such that the critical time step is unaffected by the constraints.

The constraint imposition accuracy of the method is analysed, and found to be comparable to that of the traditional stiffness penalty method. An algorithm that describes how to select the penalty parameters for maximum accuracy is provided. Low errors are attainable due to the fact that very large penalty parameters can be used without causing instability.

The method is then applied for the first time to two problem types commonly solved using penalty methods: crack propagation (modelled using cohesive surfaces), and contact-impact. A series of examples are presented that demonstrate the stability and accuracy of the method.



# Acknowledgements

First and foremost, thanks to my supervisor Harm for the incredible support and undeserved patience that he has provided me with during the past four years. Thanks to Antonio for making my time in Barcelona so enjoyable despite all the linear algebra, and for his many useful contributions to this project.

Thanks to everyone in the UoS Civil Engineering department, especially Terry, since I think he might have been my supervisor for a while; I'm not really sure, but he did a good job anyway. Thanks to everyone in and around D120a who sat with me while I ate cheese sandwiches.

Thanks to Fraser, Imo, Liam and Stook for making me laugh. Thanks to Dr. Carl, Dr. Nic and Dan Esq. for doing it all before. And extra special thanks to my Mum and Dad, who have been awesome my whole life, including these past four years.

Finally, thanks to EPSRC for funding the Doctoral Training Grant and thanks to the Department of Civil and Structural Engineering for awarding it to me.

“Picking the correct penalty parameter is a challenge.”

T. Belytschko, W. K. Liu and B. Moran

*Nonlinear Finite Elements for Continua and Structures*

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Constraints in finite element analysis . . . . .	2
1.1.1	Direct imposition . . . . .	3
1.1.2	Lagrange multipliers . . . . .	3
1.1.3	Penalty methods . . . . .	4
1.2	Time integration . . . . .	6
1.2.1	Implicit and explicit methods . . . . .	6
1.2.2	Prescribed displacements . . . . .	9
1.2.3	The critical time step . . . . .	9
1.3	The bpenalty method . . . . .	13
1.3.1	Literature review . . . . .	13
1.4	Aims and objectives . . . . .	15
1.5	Outline . . . . .	16
<b>2</b>	<b>Formulation of the bpenalty method</b>	<b>17</b>
2.1	The stiffness penalty method . . . . .	17
2.2	The mass penalty method . . . . .	21
2.3	The bpenalty method . . . . .	24
2.4	The bpenalty method with damping . . . . .	26
2.5	Matrix partitioning for the central difference method . . . . .	27
<b>3</b>	<b>Time step stability</b>	<b>31</b>
3.1	The bpenalised eigenproblem . . . . .	33
3.1.1	Mathematical proofs on eigenvalues and the penalty ratio . . . . .	33
3.1.2	Illustration: eigensolutions of a four-noded quadrilateral . . . . .	35
3.2	Effect of damping penalties . . . . .	37
3.3	Selection of stable penalty ratio . . . . .	39

---

3.3.1	The critical penalty ratio . . . . .	39
3.3.2	Calculating a penalty ratio from a stable time step . . . . .	40
3.3.3	Example: time step instability due to a single penalty constraint	42
<b>4</b>	<b>Accuracy</b>	<b>45</b>
4.1	Constraint imposition error . . . . .	48
4.1.1	Comparison of penalty methods . . . . .	48
4.1.2	Effect of penalty ratio . . . . .	51
4.2	Matrix ill-conditioning . . . . .	53
4.3	Selection of penalty parameters . . . . .	56
4.4	Summary . . . . .	59
<b>5</b>	<b>Bipenalty interface elements based on the theory of cohesive surfaces</b>	<b>61</b>
5.1	Element formulation . . . . .	63
5.1.1	Elastic stiffness matrix . . . . .	64
5.1.2	Constitutive relations and damage law . . . . .	67
5.1.3	Mass matrix . . . . .	70
5.2	Elastic wave propagation through 2D bipenalty interface elements . . .	72
5.3	Crack propagation in a single-edge-notched beam . . . . .	77
5.4	Dynamic analysis of PMMA plate with pre-existing crack . . . . .	81
<b>6</b>	<b>Modelling of contact-impact using the bipenalty method</b>	<b>87</b>
6.1	Literature review . . . . .	88
6.2	Problem statement . . . . .	90
6.3	Contact-impact in one dimension . . . . .	92
6.3.1	Finite element implementation . . . . .	93
6.3.2	Numerical example: Huněk bar impact . . . . .	95
6.4	Contact-impact in two dimensions . . . . .	102
6.4.1	Finite element implementation . . . . .	102
6.4.2	Contact detection . . . . .	105
6.4.3	Numerical example: Huněk bar in 2D . . . . .	106
6.4.4	Numerical example: Collision between dissimilar cylinders . . .	107
<b>7</b>	<b>Conclusions and future research</b>	<b>113</b>
7.1	Future research . . . . .	115

# Chapter 1

## Introduction

In order to model a physical phenomenon numerically, a certain amount of information is required about the problem under consideration. This information may be split into categories based on its nature. Firstly, we must know the geometry of the problem: the size, shape and extent of the physical bodies we wish to simulate. Then, we require a reasonable model of how those materials behave under the conditions that we would like to investigate. Finally, we must consider the boundary conditions of the problem. Boundary conditions are a set of constraints which provide information about what is happening along the boundaries of well-understood regions. For example, in the field of solid mechanics problems are often posed in the form of a boundary value problem: a differential equation together with a set of boundary conditions. The differential equation (e.g., the wave equation) describes displacements within regions of material, while the boundary conditions are used to model supports, external forces and any other additional constraints that must be applied to the model.

The application of boundary conditions, therefore, is relevant to almost every problem within the field of engineering. Indeed, many currently ongoing research topics are heavily concerned with the proper application and implementation of suitable boundary conditions, from the modelling of structural supports and periodic structures to the simulation of crack propagation and contact-impact. In all of these cases a suitable technique must be selected based on the computational reliability, stability and expense of the various methods, the ease with which they can be implemented, and how accurately the constraints are enforced.

This thesis sets out to investigate the bipenalty method, a new technique that can be used to impose constraints in dynamic finite element analysis. The main focus is the time domain analysis of elastic materials. The remainder of this chapter will introduce some preliminary concepts so that the specific aims and objectives of the work may be clearly stated.

## 1.1 Constraints in finite element analysis

The finite element (FE) method is a numerical technique used to obtain approximate solutions to partial differential equations. In the present work we consider a displacement-based finite element discretisation of the equations of elastodynamics, a full derivation for which is given by, for example, Bathe [9]. In matrix form, the semi-discretised dynamic equations of motion may be written as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (1.1)$$

where  $\mathbf{M}$  and  $\mathbf{K}$  are the mass and stiffness matrices, respectively;  $\mathbf{u}$  is the displacement vector,  $\mathbf{f}$  the vector of external forces, and dot notation is used to indicate derivatives with respect to time. If structural damping is also taken into account, the system is written

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (1.2)$$

where  $\mathbf{C}$  is the structural damping matrix. The choice of whether to include structural damping in a formulation depends entirely on the type of problem under consideration. In the following technical discussion the damping matrix is often omitted for the sake of clarity, although in these cases the general theories also apply to damped systems.

This system of equations (1.1) represents the assembly of all the finite elements required to describe the continuum we are considering. The vector  $\mathbf{u}$  is of length  $n$ , where  $n$  is the number of degrees of freedom (DOF) possessed by the system. In order to enforce additional constraints we also consider a set of  $n_c$  equality constraints, written as

$$\mathbf{h} = \mathbf{G}\mathbf{u} - \mathbf{q} \quad (1.3)$$

where  $\mathbf{G}$  is the constraint matrix of size  $n_c \times n$ , which describes the relationships

between DOF for each constraint,  $\mathbf{q}$  is a vector of prescribed displacements, and  $\mathbf{h} = \mathbf{0}$  implies exact satisfaction of all constraints. The constraints given in (1.3) are linear if  $\mathbf{G}$  is constant in  $\mathbf{u}$ , and non-linear if they depend on the displacements. The remainder of this section will briefly describe the most commonly used methods for enforcing these constraints.

### 1.1.1 Direct imposition

One method of enforcing (1.3) is to transform the system of equations given by (1.1) so that the constraints are exactly satisfied. This modification of the system results in a system of size  $n - n_c$ ; that is, an equation is removed for each constraint added, with the number of DOF decreasing for an increasingly constrained system [22].

For example, imposing the constraint  $u_i = 0$  (and, by implication  $\ddot{u}_i = 0$ ) can be achieved by simply removing the  $i$ th equation from the system of equations (1.1), i.e.,

$$\sum_j (M_{ij}\ddot{u}_j + K_{ij}u_j) = f_i \quad (1.4)$$

along with the corresponding columns in  $\mathbf{K}$  and  $\mathbf{M}$ , obtaining a so-called *reduced system*. This reduced system is then solved for the remaining unknown displacements. The reaction force  $f_i$  can be obtained if required via (1.4) once the unknown displacements (and accelerations) have been computed.

For such simple constraints this method is simple and effective, and it has the additional advantage that the system size is reduced, but for constraints involving more than one DOF the necessary transformations involve complicated matrix manipulation (see Reference [22] for details). If the constraints change over time this can be especially problematic. However, transformation does provide a robust method for computing exact solutions when simple constraints are used, and will be used in the present work (where possible) for the purposes of comparison, verification, validation and error evaluation.

### 1.1.2 Lagrange multipliers

The Lagrange multiplier method (LMM) is another exact method of constraint enforcement, but with a higher level of versatility. It is executed by solving the system equations (1.1) and the additional constraint equations (1.3) simultaneously. Thus,

the size of the system increases from  $n$  to  $n + n_c$ .

The introduction of Lagrange multipliers into the equations of motion gives

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} + \mathbf{G}^T\boldsymbol{\lambda} = \mathbf{f} \quad (1.5)$$

$$\mathbf{G}\mathbf{u} = \mathbf{q} \quad (1.6)$$

where the vector  $\boldsymbol{\lambda}$  contains the Lagrange multipliers, which must be determined along with the unknown displacements. The multipliers generally have a physical interpretation as constraint forces (e.g. reaction forces, contact forces).

The use of Lagrange multipliers creates a mixed method, where force-type DOF are introduced alongside the spatial DOF as unknowns in the problem. For problems with a large number of constrained nodes this results in a large increase in the size of the system and consequently the computational expense. Use of Lagrange multipliers also results in a loss of positive definiteness which may cause numerical difficulties, depending on the solution method employed. Despite these issues, the LMM is popular for a wide range of problem types due to its accuracy and its high adaptability, for instance the ease with which geometrically nonlinear constraints can be handled [35].

Other techniques based on the LMM included the augmented Lagrangian [70] and the perturbed Lagrangian [72].

### 1.1.3 Penalty methods

The traditional penalty method has long been used in a wide range of fields. Its discovery is sometimes attributed to a 1943 paper by Courant [23], although elsewhere it has been suggested that the concept is “so basic that it would be facetious to attribute its initial use to any single person” [62]. The method takes a minimisation problem (in the case of structural analysis, the minimisation of potential energy) and adds a set of *penalty functions*, one for each constraint. These functions multiply the constraint violation by a large coefficient, called the *penalty parameter*. As this penalty parameter is increased, the violation is penalised more and more severely, forcing the solution closer and closer to the solution of the constrained problem. However, the constraints are only satisfied exactly in the limit of the penalty parameter tending to infinity. In practise infinite parameters cannot be used and hence the constraints

are only approximately imposed.

The penalty method introduces no new solution variables, and so the size of the system is unaffected by their introduction. When applied, the full system of equations becomes

$$\mathbf{M}\ddot{\mathbf{u}} + (\mathbf{K} + \mathbf{G}^T \mathbf{P} \mathbf{G}) \mathbf{u} = \mathbf{f} + \mathbf{G}^T \mathbf{P} \mathbf{q} \quad (1.7)$$

where  $\mathbf{P}$  is a diagonal matrix containing the penalty parameters, one for each constraint equation. If these parameters are large, a violation of the constraints results in a large correcting force  $\mathbf{G}^T \mathbf{P} \mathbf{G} \mathbf{u}$ ; however, large parameters also lead to ill-conditioning of the penalised stiffness matrix, effectively setting an upper limit on the magnitude of the entries in  $\mathbf{P}$ . This leads to some (hopefully small) finite error in the enforcement of the constraint (i.e., in Equation (1.3),  $\mathbf{h} \neq \mathbf{0}$ ).

The method described above is by far the most common form of the penalty method, but for dynamic systems there is an alternative form of the penalty method which operates on the mass matrix of the system instead of the stiffness matrix. The mass penalty equivalent of (1.7) is

$$(\mathbf{M} + \mathbf{G}^T \mathbf{P} \mathbf{G}) \ddot{\mathbf{u}} + \mathbf{K} \mathbf{u} = \mathbf{f} + \mathbf{G}^T \mathbf{P} \ddot{\mathbf{q}} \quad (1.8)$$

where, as before,  $\mathbf{P}$  is a matrix of penalty parameters. Stiffness penalty methods are in general more accurate than mass penalty methods since they are activated by constraint violations in the displacement vector, rather than the acceleration vector. In addition, mass penalties for constraints concerning multiple DOF introduce non-diagonal terms into the mass matrix, which can be problematic. However, compared to stiffness penalties they have a very different effect on the eigenvalues of the system, which can be very useful for certain problem types (for example, calculating natural frequencies using the Rayleigh-Ritz method) [39, 47].

Compared with the LMM, penalty methods have the advantage that no extra solution variables are required. However, non-physical displacements will in general occur due to the approximate nature of the technique. Penalty parameters must be selected so that they are large enough to ensure errors are within acceptable bounds, but not so large that they cause ill-conditioning. The ad hoc nature of this choice gives rise to uncertainty and is often considered the main drawback of the method. In addition, when traditional penalty methods are used in an explicit dynamics set-

ting they often have an adverse effect on the critical time step, as we shall see in the following chapters.

## 1.2 Time integration

So far we have considered only the semi-discretised equations of motion, meaning that the problem has been discretised in space (via the finite element method) but not in time. In order to achieve full discretisation, a time integration scheme must be employed.

In the present work we set out to solve the equations of motion (1.1) for  $\mathbf{u} = \mathbf{u}(t)$  at discrete intervals over a time period  $t \in [0, T]$  using a direct time integration scheme. Direct time integration is a method by which the equations of motion are integrated using a numerical step-by-step procedure without any prior transformation of the equations. Practically, this means finding an equilibrium solution for (1.1) at time  $t = 0$  using the known initial conditions of the problem, and then using this solution to seek a solution for the next time step,  $t = \Delta t$ , and so on. This is achieved by making certain assumptions about how displacements, velocities and accelerations vary over the course of a time step, with different assumptions giving rise to different time integration schemes.

### 1.2.1 Implicit and explicit methods

The majority of time integration procedures fall into one of two main categories: implicit or explicit. Explicit methods are defined by the ability to calculate displacements at time  $t + \Delta t$  in terms of accelerations and displacements at time  $t$  only. If a diagonal mass matrix is used, and damping is neglected, no simultaneous equations need to be solved. Conversely, for implicit methods the displacement calculation at time  $t + \Delta t$  also involves accelerations at time  $t + \Delta t$ , which leads to a set of simultaneous equations that must be solved at each step [57]. For this reason, explicit methods tend to be much more efficient per time step.

However, another important characteristic of explicit methods is that they are *conditionally stable*. This means that the stability of the procedure is dependent on the time step size; more precisely, there exists a *critical time step* above which solutions will display instability. Many implicit methods are formulated so that they are *uncon-*

*ditionally stable*, meaning that much larger time steps can be used to obtain a stable results. In some cases, this saving makes up for the relatively expensive displacement computation. On the other hand, larger time steps lead to lower resolution in the time domain and the loss of high frequency wave information. Implicit schemes are therefore most useful when only the overall low mode structural response is required (for example in the fields of vibration and earthquake engineering) since in this case large time steps can safely be used [42]. When a detailed picture of stress wave propagation is required (e.g. damage mechanics, blast and impact engineering) time steps must be kept relatively small, making explicit methods much more practical.

Examples of both implicit and explicit direct integration methods can be derived from the *Newmark family*, which provide key assumptions for the calculation of displacement and velocity, expressed as

$$\mathbf{d}_{t+\Delta t} = \mathbf{d}_t + \Delta t \mathbf{v}_t + \frac{\Delta t^2}{2} [(1 - 2\beta)\mathbf{a}_t + 2\beta\mathbf{a}_{t+\Delta t}] \quad (1.9)$$

$$\mathbf{v}_{t+\Delta t} = \mathbf{v}_t + \Delta t [(1 - \gamma)\mathbf{a}_t + \gamma\mathbf{a}_{t+\Delta t}] \quad (1.10)$$

where  $\mathbf{d}_t$ ,  $\mathbf{v}_t$  and  $\mathbf{a}_t$  are the time-discretised approximations of the displacement, velocity and acceleration at time  $t$ , and  $\Delta t$  is the time step. The Newmark parameters  $\beta$  and  $\gamma$  dictate properties of the scheme, such as stability, accuracy and efficiency. Combined with the fully discretised governing equations

$$\mathbf{M}\mathbf{a}_{t+\Delta t} + \mathbf{C}\mathbf{v}_{t+\Delta t} + \mathbf{K}\mathbf{d}_{t+\Delta t} = \mathbf{f}_{t+\Delta t} \quad (1.11)$$

(with damping now included for increased generality) and initial conditions,

$$\mathbf{d}_0 = \mathbf{u}(0) \quad (1.12)$$

$$\mathbf{v}_0 = \dot{\mathbf{u}}(0) \quad (1.13)$$

we have all the relations required to calculate displacements, velocities and accelerations at discrete intervals  $t = 0, \Delta t, 2\Delta t, \dots, T - \Delta t, T$ . Some commonly used integration schemes emerge by using certain values for  $\beta$  and  $\gamma$ , such as the linear acceleration method ( $\beta = 1/6, \gamma = 1/2$ ) and the Fox-Goodwin method ( $\beta = 1/12, \gamma = 1/2$ ).

As an example, consider the constant average acceleration method, which emerges

when taking Newmark parameters  $\beta = 1/4$  and  $\gamma = 1/2$ . By manipulating (1.9) and (1.10) we may obtain expressions for  $\mathbf{a}_{t+\Delta t}$  and  $\mathbf{v}_{t+\Delta t}$  in terms of  $\mathbf{d}_{t+\Delta t}$ . Substituting these relations into (1.11) gives

$$\left( \frac{4}{\Delta t^2} \mathbf{M} + \frac{2}{\Delta t} \mathbf{C} + \mathbf{K} \right) \mathbf{d}_{t+\Delta t} = \mathbf{f}_{t+\Delta t} + \mathbf{M} \left( \frac{4}{\Delta t^2} \mathbf{d}_t + \frac{4}{\Delta t} \mathbf{v}_t + \mathbf{a}_t \right) + \mathbf{C} \left( \frac{2}{\Delta t} \mathbf{d}_t + \mathbf{v}_t \right) \quad (1.14)$$

which may then be solved for the unknown displacements  $\mathbf{d}_{t+\Delta t}$ . Clearly, this requires the solution of a set of simultaneous equations due to the presence of the non-diagonal stiffness matrix  $\mathbf{K}$  on the left hand side of the equation. However, as an unconditionally stable method, stability is ensured for any choice of time step  $\Delta t$ .

The central difference method (CDM) is perhaps the most commonly used explicit solution algorithm for time domain FE analysis, providing the basis for commercial solvers such as LS-DYNA [36]. It can be derived from the Newmark family by setting  $\beta = 0$  and  $\gamma = 1/2$  [43], or from the standard difference expressions for acceleration and velocity,

$$\mathbf{a}_t = \frac{1}{\Delta t^2} (\mathbf{d}_{t-\Delta t} - 2\mathbf{d}_t + \mathbf{d}_{t+\Delta t}) \quad (1.15)$$

$$\mathbf{v}_t = \frac{1}{2\Delta t} (-\mathbf{d}_{t-\Delta t} + \mathbf{d}_{t+\Delta t}) \quad (1.16)$$

Substituting these expressions into the damped system at time  $t$ ,

$$\mathbf{M}\mathbf{a}_t + \mathbf{C}\mathbf{v}_t + \mathbf{K}\mathbf{d}_t = \mathbf{f}_t \quad (1.17)$$

we obtain

$$\left( \frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right) \mathbf{d}_{t+\Delta t} = \mathbf{f}_t - \left( \mathbf{K} - \frac{2}{\Delta t^2} \mathbf{M} \right) \mathbf{d}_t - \left( \frac{1}{\Delta t^2} \mathbf{M} - \frac{1}{2\Delta t} \mathbf{C} \right) \mathbf{d}_{t-\Delta t} \quad (1.18)$$

which can then be solved for  $\mathbf{d}_{t+\Delta t}$ . Note that the stiffness matrix is present on the right-hand side only. If it is appropriate to neglect structural damping (for short, highly dynamic wave propagation problems it usually is) then the equations simplify to give

$$\mathbf{d}_{t+\Delta t} = \Delta t^2 \mathbf{M}^{-1} (\mathbf{f}_t - \mathbf{K}\mathbf{d}_t) + 2\mathbf{d}_t - \mathbf{d}_{t-\Delta t} \quad (1.19)$$

Furthermore, if the mass matrix is diagonal (e.g, when using a lumped mass matrix)

then the matrix inversion is trivial and the computation is simplified even further; there is no need to solve a linear system of equations (see Bathe [9, §9.2.1] for more details).

### 1.2.2 Prescribed displacements

Explicit time integration routines give another option for the enforcement of simple constraints, beyond those discussed in Section 1.1. If displacements and/or velocities are provided as prescribed functions of time, these can be imposed by simply setting the necessary values at each time step (if prescribed displacements are known, velocities must be obtained through numerical differentiation). However, when boundary conditions are given as linear or nonlinear algebraic equations relating two or more DOF the implementation is complicated considerably. The most common alternative methods in this case are penalty methods or the LMM [12].

### 1.2.3 The critical time step

The main advantage of the CDM is its low computational cost. However, the apparent computational efficiency of the method is offset by the fact that it is conditionally stable. That is, if the time step  $\Delta t$  is set larger than some critical value  $\Delta t_{\text{crit}}$  the solution will exhibit instability, rendering the results meaningless. For the CDM this critical time step is given by [43]

$$\Delta t_{\text{crit}} = \frac{2}{\omega_{\text{max}}} \quad (1.20)$$

where  $\omega_{\text{max}}$  is the maximum eigenfrequency of the system. The eigenfrequencies of a system are determined by solution of the generalised eigenvalue problem

$$(\mathbf{K} - \omega_i^2 \mathbf{M}) \mathbf{u}_i = \mathbf{0} \quad (1.21)$$

where the eigenvectors  $\mathbf{u}_i$  and corresponding eigenfrequencies  $\omega_i$  form the  $n$  eigensolutions  $(\mathbf{u}_i, \omega_i)$ . These eigensolutions are assumed to be ordered such that  $\omega_1 \leq \omega_2 \leq \dots \leq \omega_n$ , so that  $\omega_{\text{max}} \equiv \omega_n$ . The eigenvalues of the system  $\lambda_i$  are related to the eigenfrequencies by  $\lambda_i = \omega_i^2$ .

For a solid mechanics FE problem the maximum eigenfrequency is dependent on

the material properties and the FE mesh. For example, for a single two-noded linear finite element,  $\omega_{\max} = 2c_e/h$ , where  $c_e = \sqrt{E/\rho}$  is the elastic wave speed in the bar and  $h$  is the element length. In other words, a stiffer material or a more highly refined mesh leads to an increase in the maximum eigenfrequency, while a more dense material leads to a decrease. This holds true as a general principle for all element types, although the eigenfrequencies of more complex 2D and 3D elements also involve other parameters (e.g., Poisson's ratio, element shape measures) and closed form expressions are often unobtainable. An attempt to find analytic expressions for the maximum eigenfrequencies of some common finite element types can be found in Askes et al. [5], while Lin gives a theory for computing upper bounds for various element types [53]. Investigations of this type show that for distorted 2D elements, 3D elements and higher order elements, analytical solutions are usually impossible to retrieve.

In practice, the maximum eigenfrequency of an assembled FE system can only be obtained numerically for all but the simplest of cases. Although this can be achieved relatively efficiently using direct iteration (which allows the maximum eigenfrequency may be found without solving the full eigenvalue problem), this approach may still prove prohibitively expensive for large systems. A common solution is to use the element eigenfrequency inequality, which can provide an upper bound on the maximum eigenfrequency of the system [14]. This theorem states that

$$\omega_{\max} \leq \omega_{\max}^e \quad (1.22)$$

where  $\omega_{\max}^e$  is the maximum eigenfrequency of all finite elements. Thus, a conservative estimate for  $\omega_{\max}$  may be found by finding the maximum eigenfrequency of all the individual elements in a mesh.

The time step limit (1.20) is often referred to as the Courant-Friedrichs-Lewy (CFL) condition after the authors of the paper in which it was first reported [24, 25]. The phenomenon is also explored in Reference [57], which gives an enlightening and intuitive explanation of time step stability in implicit and explicit schemes. It is suggested that the CFL stability limit can be thought of as “a requirement that the physical information flow rate (wave speed) not exceed the computational information flow rate” through the system [57]. Diagrams of information flow are provided

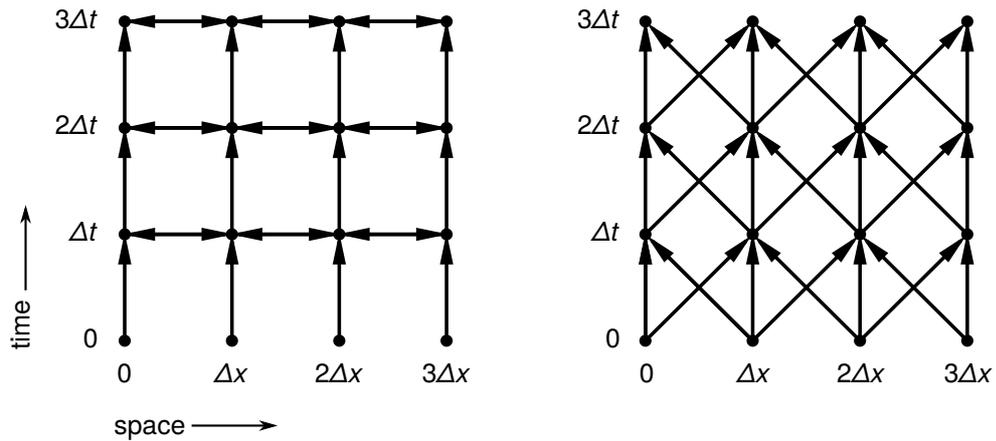


Figure 1.1: Visualisation of information flow for a typical implicit (left) and explicit (right) time integration procedure for a one-dimensional example. Adapted from Reference [57].

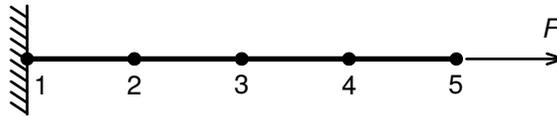


Figure 1.2: A one-dimensional bar with one fixed end and constant force  $F = 1$  N

in Figure 1.1 for an implicit, unconditionally stable scheme and the explicit CDM, in the context of a one-dimensional mesh. The right-hand diagram shows that the solution at any point is dependent on values from the previous time step only, and that information is passed from node to node at a rate of  $\Delta x / \Delta t$ . If physical information (e.g., a displacement wave) is passed faster than computational information, the scheme breaks down. Hence, both the wave speed and the mesh size directly affect the critical time step. Information flow in the implicit constant average acceleration method (shown on the left-hand side of the plot) does not pass exclusively from one time step to the next, but also directly from node to node at each time step. This means that (in a computational sense) information is transferred instantaneously, resulting in an infinite information flow rate. Hence, time step size has no effect on the stability of the scheme in this case.

To demonstrate the effects of time step instability, we now consider elastic wave propagation in a one-dimensional bar modelled with four two-noded linear finite elements, as shown in Figure 1.2. The bar has Young's modulus  $E = 100$  Pa, mass density  $\rho = 1$  kg/m<sup>3</sup>, and cross-sectional area  $A = 1$  m<sup>2</sup>. Each element has a length of  $h = 1$  m, giving a critical time step of  $\Delta t_{\text{crit}} = 0.1$  s. Figure 1.3 shows the tip displace-

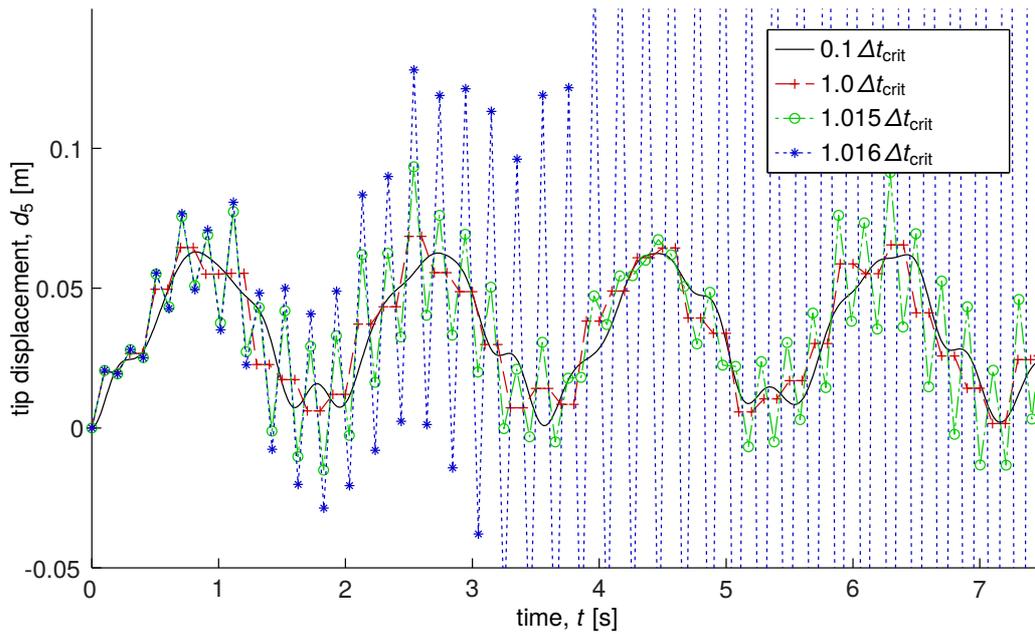


Figure 1.3: Tip displacement of bar for a range of sub- and super-critical time steps

ment over time. For super-critical time step values the solution displays instability, manifesting here as a spurious oscillation of large amplitude. Time step instability is characterised by unbounded exponential growth in displacements. Consequently, it is usually obvious when time step instability occurs, but if the critical time step condition (1.20) is violated by a small amount, and/or for a short period of time, it may be more difficult to detect.

Penalty methods can have a drastic effect on the critical time step. Traditional penalty methods add artificial stiffness to the system, as described by Equation (1.7). This can introduce eigenvalues that are orders of magnitude larger than those associated with the unpenalised system [46]. This in turn *decreases* the critical time step,  $\Delta t_{\text{crit}}$ . If this is not accounted for the solution may become unstable; if it is accounted for (by decreasing  $\Delta t$ ) the total computational cost of the analysis may be significantly increased. In addition, the increase in cost is greater for larger penalty parameters, creating a trade-off between the accuracy of constraint imposition and computational expense. The mass penalty method, on the other hand, does not increase the maximum eigenfrequency associated with the system [39, 47]. Its use therefore does not decrease the critical time step.

### 1.3 The bipenalty method

The bipenalty method is a formulation for constraint enforcement that sets out to combine the stiffness and mass penalty methods. Stiffness penalties provide good constraint enforcement, but tend to lower the critical time step of the analysis. On the other hand, mass penalties are relatively inaccurate, and unsuitable for certain problem types, but they do not have an adverse effect on  $\Delta t_{\text{crit}}$ . Through the simultaneous use of stiffness- and inertia-type penalties it is possible to ensure that the critical time step of an analysis is unaffected (or at least, not decreased) by the constraints imposed upon it, while providing performance that is comparable to the traditional stiffness penalty method [5, 40, 41, 49].

Use of the bipenalty method results in penalty parameters being added to both the stiffness and mass matrices of an FE system. Returning to the notation of Section 1.1, we have

$$(\mathbf{M} + \mathbf{G}^T \mathbf{P}_m \mathbf{G}) \ddot{\mathbf{u}} + (\mathbf{K} + \mathbf{G}^T \mathbf{P}_s \mathbf{G}) \mathbf{u} = \mathbf{f} + \mathbf{G}^T \mathbf{P}_m \ddot{\mathbf{q}} + \mathbf{G}^T \mathbf{P}_s \mathbf{q} \quad (1.23)$$

where  $\mathbf{P}_m$  and  $\mathbf{P}_s$  now contain two different sets of penalty parameters: the mass penalty parameters  $(\alpha_m)_i$  and the stiffness penalty parameters  $(\alpha_s)_i$  (where  $i = 1 \dots n_c$ ). The so-called penalty ratio for each constraint is then defined as

$$R_i = \frac{(\alpha_s)_i}{(\alpha_m)_i} \quad (i = 1 \dots n_c) \quad (1.24)$$

In the common case that penalty parameters are the same for all constraints,  $\alpha_s$  and  $\alpha_m$  are used to refer to the stiffness and mass parameters values, respectively. Likewise, when the penalty ratio is equal for all constraints it is referred to simply as  $R$ . A full derivation of the bipenalty method is presented in Chapter 2.

#### 1.3.1 Literature review

The present investigation builds upon earlier work that developed a bipenalty formulation for imposing single absolute constraints [5]. This study showed that the critical time step of an analysis can be preserved so long as the penalty ratio is kept below a certain value. This value is known as the *critical penalty ratio* (CPR), denoted by  $R_{\text{crit}}$ . However, the methodology for determining this value was dependent

on the finite element shape functions used and necessitated the analytical solution of eigenvalue problems, which becomes impossible for complex element types. In addition, the method of CPR determination was apparently only valid for single absolute constraints, and as such could only be applied to a small range of problems.

An alternative treatment of the bipenalty method, derived from a variational FE formulation, can be found in Reference [61]. In this work, a constraint on the time derivative of a prescribed displacement constraint is added to the weak form of the governing equations. The bipenalty method is therefore described as a “variationally consistent” form of standard penalty methods, because constraints on both displacement *and its time derivative* (i.e., velocity) are included in the variational formulation. It is even stated that for time domain dynamics, both the pure stiffness and the pure mass penalty methods “are not, in general, theoretically sound” due to the fact that strictly speaking both constraints should be included in order to guarantee convergence. It is also shown that the so-called “stiffness” of the system (in this context, the ratio of largest to smallest eigenvalue,  $S = \omega_{\max}/\omega_{\min}$ ) can be guaranteed to be kept within the same order of magnitude only when the bipenalty method is used, which is desirable for ensuring accuracy and stability.

More recent work by Ilanko and Monterrubio [48, 49] highlights some of the advantages that the bipenalty method possesses for frequency domain analysis, especially for computing the eigenvalues of constrained systems. Guidance is provided as to how to tune the penalty ratio  $R$  for optimum convergence of the eigenvalue solution (as opposed to any time step stability concerns).

For contact-impact problems, the bipenalty method has been utilised even for implicit, unconditionally stable time integration schemes. The use of penalty methods that enforce displacement constraints only are known to lead to oscillations in the contact forces [45], the origins of which can be “traced in part to the lack of satisfaction of the constraint in the velocities” [1, p. 283]. The bipenalty method has been used in two specific contact-impact formulations [1, 4] in an attempt to combat this phenomenon.

Although using the bipenalty method for manipulation of the critical time step has apparently not yet been fully explored, various other methods exist for artificially reducing the maximum eigenfrequency of a system in order to increase  $t_{\text{crit}}$ . Probably the most popular is mass scaling, which adds extra mass to a system in order to affect

the higher modes and eigenfrequencies [54, 59]. In practise though, this technique can only reliably be used when only the lower modes need to be determined with accuracy (i.e., when global structural behaviour, not the propagation of waves, is of interest).

## 1.4 Aims and objectives

The bipenalty method has the potential to provide a simple and accurate technique for imposing all kinds of constraints in explicit dynamics without any need to adjust the time step of the analysis. The method is variationally consistent [61] and may provide more accurate constraint imposition when compared to standard penalty methods (as has been shown for contact-impact problems [1]). The main objectives of this thesis are as follows:

### Fundamentals

- Develop a bipenalty formulation for an arbitrary set of constraint equations that can be easily applied to a wide variety of problem types.
- Ensure that the maximum ratio of stiffness and mass penalty parameters that can be used without instability can be easily calculated for all finite element types and all constraint formulations.
- Investigate how the use of the bipenalty method affects accuracy, and determine the optimum penalty ratio, if one exists.
- Investigate potential performance increases gained by introducing damping penalties to the bipenalty formulation, as well as its effect on stability.

### Applications

- Through medium-scale numerical analyses, test the performance of the bipenalty method when used as a basis for a new cohesive surface formulation for dynamic crack propagation.
- Determine the nature and extent of any possible performance advantages that the bipenalty method may possess when used as a basis for one- and two-dimensional contact-impact formulations.

## 1.5 Outline

The next chapter provides a full derivation of the biperality method for a set of arbitrary constraints, providing discussion on both the stiffness and mass penalty methods in turn before combining the two techniques. An extension to the biperality method that includes penalties in the structural damping matrix is then introduced, along with a useful technique for efficiently solving a system with non-diagonal mass and damping matrices using the central difference method.

In Chapter 3 attention is turned to time step stability, and especially how to tune the biperality method in order to obtain a stable solution. Mathematical proofs are presented that show the effect of the biperality method on the eigensolutions of an FE system in order to determine how best to avoid stability issues.

The focus of Chapter 4 is the accuracy of the biperality method: especially how it compares to using stiffness- or inertia-type penalty functions in isolation and the effect the penalty ratio has on the accuracy of constraint imposition. The chapter also includes guidelines on how to select penalty parameters for a biperality analysis, taking into account both stability and accuracy of the solution.

At this point the theoretical analysis of the biperality method has concluded, and we move on to practical applications. Chapter 5 examines interface elements, with a focus on their use in dynamic crack propagation. The two main examples presented are the single-edge-notched beam and the cracking of a PMMA plate under fast loading. Comparisons are made between the biperality method and existing techniques. Chapter 6 moves on to test out the biperality method for use in contact-impact problems, focussing on stability and the presence of spurious, non-physical oscillations in the contact force.

Finally, a summary of the key findings of the work and suggestions for further study are given in Chapter 7.

## Chapter 2

# Formulation of the bipenalty method

The traditional penalty method for structural problems, as described in Section 1.1.3, relies on a modification of the system stiffness matrix. It is a technique with a high degree of adoption within the numerical analysis community. In the field of dynamic analysis a less well-known but closely related technique has been developed which modifies the mass matrix of a system instead of its stiffness matrix. Here, we refer to this method as the mass penalty method, while the more traditional technique is called the stiffness penalty method. This chapter sets out to give formulations of both methods, before describing the bipenalty method: the simultaneous use of both techniques.

### 2.1 The stiffness penalty method

The traditional stiffness penalty method can be derived in a number of ways. In structural analysis the most common method is to consider the total potential energy of the system [9, 22, 84],

$$\mathcal{U} = \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} - \mathbf{u}^T \mathbf{f} \quad (2.1)$$

The total potential energy  $\mathcal{U}$  consists of the internal strain energy and the work done by external forces. By minimisation of this energy expression the finite element method describes the quasi-static structural response.

The penalty method proceeds by adding a so-called penalty function to the ex-

pression for potential energy, so that

$$\mathcal{U}^P = \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} - \mathbf{u}^T \mathbf{f} + \frac{1}{2} \mathbf{h}^T \mathbf{P}_s \mathbf{h} \quad (2.2)$$

where  $\mathbf{h}$  is a vector describing the violation of the constraint equations as described in Equation (1.3) and  $\mathbf{P}_s$  is a diagonal matrix containing the stiffness penalty parameters. As these parameters tend to infinity, the minimisation of  $\mathcal{U}^P$  means that the  $\mathbf{h} \rightarrow \mathbf{0}$  and the constraints are, in theory, exactly enforced. The penalty parameters have units of stiffness (N/m), the same as the entries in  $\mathbf{K}$ , to which they will ultimately be added. Minimisation of the penalised total potential energy expression<sup>1</sup> results in

$$\frac{d \mathcal{U}^P}{d \mathbf{u}^T} = (\mathbf{K} + \mathbf{K}^P) \mathbf{u} = \mathbf{f} + \mathbf{f}_s^P \quad (2.3)$$

where  $\mathbf{K}^P = \mathbf{G}^T \mathbf{P}_s \mathbf{G}$  and  $\mathbf{f}_s^P = \mathbf{G}^T \mathbf{P}_s \mathbf{q}$ . For dynamic analyses, kinetic energy, given by

$$\mathcal{T} = \frac{1}{2} \dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}} \quad (2.4)$$

must also be taken into account. In this case, minimisation of the total energy leads to the full dynamic equations of equilibrium,

$$\frac{d}{dt} \frac{\partial \mathcal{T}}{\partial \dot{\mathbf{u}}^T} + \frac{\partial \mathcal{U}^P}{\partial \mathbf{u}^T} = \mathbf{M} \ddot{\mathbf{u}} + (\mathbf{K} + \mathbf{K}^P) \mathbf{u} = \mathbf{f} + \mathbf{f}_s^P \quad (2.5)$$

Therefore, application of the penalty method consists of adding extra terms to the stiffness matrix  $\mathbf{K}$ . The position of those terms with respect to the global stiffness matrix is decided by the constraint equations, and their magnitude is largely controlled by the magnitude of the penalty parameters in  $\mathbf{P}_s$ . If  $\mathbf{q}$  is non-zero, extra terms must also be added to the external force vector,  $\mathbf{f}$ .

By considering the Rayleigh quotient of this system, we may gain some insight into the effect that the penalty matrix  $\mathbf{K}^P$  has on the system's associated eigenvalues. The Rayleigh quotient for a dynamic system is given by

$$\mathcal{R}(\boldsymbol{\psi}) = \frac{\boldsymbol{\psi}^T \mathbf{K} \boldsymbol{\psi}}{\boldsymbol{\psi}^T \mathbf{M} \boldsymbol{\psi}} \quad \boldsymbol{\psi} \in \mathbb{R}^n, \boldsymbol{\psi} \neq \mathbf{0} \quad (2.6)$$

<sup>1</sup>Minimisation of  $\mathcal{U}^P$  is expressed by taking the derivative with respect to  $\mathbf{u}$  and setting it equal to zero. The notational convention adopted for the derivative of a scalar  $x$  with respect to a vector  $\mathbf{y}$  is  $\frac{dx}{d\mathbf{y}}$  if the result is expressed as a row vector and  $\frac{dx}{d\mathbf{y}^T}$  if the result is a column vector.

If an eigenvector of the system is substituted for the arbitrary vector  $\boldsymbol{\psi}$  the Rayleigh quotient is equal to the associated eigenvalue. In addition, the maximum value of  $\mathcal{R}(\boldsymbol{\psi})$  is equal to the maximum eigenvalue of the whole system, i.e., [43]

$$\lambda_{\max} = \max[\mathcal{R}(\boldsymbol{\psi})] \quad (2.7)$$

where the eigenvalues of the system are given by  $\lambda_1, \lambda_2, \dots, \lambda_N$  when sorted into ascending order and  $\lambda_{\max} \equiv \lambda_N$ . For our stiffness penalised (SP) system, we have

$$\mathcal{R}_{\text{SP}}(\boldsymbol{\psi}) = \frac{\boldsymbol{\psi}^T (\mathbf{K} + \mathbf{K}^{\text{P}}) \boldsymbol{\psi}}{\boldsymbol{\psi}^T \mathbf{M} \boldsymbol{\psi}} \quad (2.8)$$

A detailed analysis of the eigenvalue problem for SP systems is given by Ilanko [46], who builds on the classical theories developed by Rayleigh [65] for vibrating systems. For now it is sufficient to note that the addition of positive stiffness values in the matrix  $\mathbf{K}^{\text{P}}$  cannot decrease the Rayleigh quotient given in (2.8). Thus, stiffness penalties tend to increase the maximum eigenvalue  $\lambda_{\max} = \omega_{\max}^2$  of the system. We might indeed expect this result, considering that the natural frequency of an undamped single degree of freedom mass-spring system is given by  $\omega = \sqrt{k/m}$ , where  $k$  and  $m$  are the stiffness and mass, respectively (meaning that an increase in stiffness leads to an increase in the natural frequency).

For a more intuitive grasp of the stiffness penalty method, we now turn to a commonly cited physical interpretation. It turns out that it is often possible to compute the stiffness penalty matrix  $\mathbf{K}^{\text{P}}$  using the stiffness matrix for a 1D spring of stiffness  $\alpha_s$ ,

$$\mathbf{K}^{\text{spring}} = \alpha_s \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (2.9)$$

Presently we consider as an example a three-element FE model of a bar, fixed at one end, as shown in Figure 2.1(a). For simplicity we use two-noded linear finite elements [84, §1.2] (referred to here as ‘bar elements’) with Young’s modulus  $E$ , cross-sectional area  $A$  and length  $h$ . The full, unconstrained global stiffness matrix is

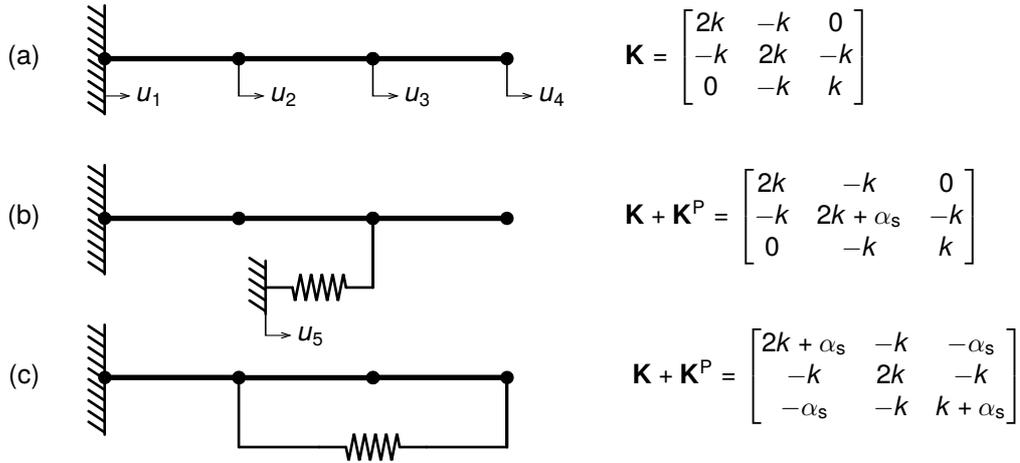


Figure 2.1: Left: physical representation of a 1D bar subject to various constraints, modelled using 2D finite elements and the stiffness penalty method. Right: the relevant reduced global stiffness matrices.

then

$$\mathbf{K}_{(a)}^{\text{full}} = \begin{bmatrix} k & -k & 0 & 0 \\ -k & 2k & -k & 0 \\ 0 & -k & 2k & -k \\ 0 & 0 & -k & k \end{bmatrix} \quad (2.10)$$

where  $k = EA/h$ . We can enforce the support at the left-hand end of the bar (i.e., set  $u_1 = 0$ ) by removing the rows and columns corresponding to the first DOF to obtain the so-called reduced stiffness matrix. The reduced matrix for the unpenalised system is shown in Figure 2.1(a).

In Figure 2.1(b) node 3 has been fixed using the penalty method, which is equivalent to connecting node 3 to a fixed support using a spring element of stiffness  $\alpha_s$ . There is only one occurrence of the penalty parameter  $\alpha_s$  despite the addition of the 2 by 2 matrix given in (2.9) because the additional fixed support has been removed to form the reduced stiffness matrix. The full stiffness matrix, disregarding fixed supports is

$$\mathbf{K}_{(b)}^{\text{full}} = \begin{bmatrix} k & -k & 0 & 0 & 0 \\ -k & 2k & -k & 0 & 0 \\ 0 & -k & 2k + \alpha_s & -k & -\alpha_s \\ 0 & 0 & -k & k & 0 \\ 0 & 0 & -\alpha_s & 0 & \alpha_s \end{bmatrix} \quad (2.11)$$

In this case the first and last rows and columns are deleted, resulting in the 3 by 3 reduced matrix given in 2.1(b).

The system shown in Figure 2.1(c), in which nodes 2 and 4 are connecting by a spring element, is equivalent to using the stiffness penalty method to impose the constraint  $u_2 - u_4 = 0$ .

The two constrained scenarios shown above demonstrate two important kinds of constraint. The first, in Fig. 2.1(b), is an *absolute* or *single-point* constraint, because there is only one DOF featured in the governing constraint equation ( $u_3 = 0$ ). If more than one DOF is involved, as in Fig. 2.1(c), the constraint is known as a *relative* or *multipoint* constraint. Single-point constraints lead to a penalty matrix with diagonal terms only, while multipoint constraints lead to (usually negative) off-diagonal terms.

## 2.2 The mass penalty method

In comparison with the stiffness penalty method, the mass penalty method is relatively unknown. The most detailed analysis from a frequency domain perspective can be found in a contribution by Ilanko [47], which shows that mass penalisation results in convergence towards the fully constrained system for increasing penalty parameter magnitude for both positive and negative penalties. As for stiffness penalties (which in fact may also be either positive or negative [46]) the exact solution is shown to be bounded by the results obtained with positive and negative penalties, allowing for a much greater degree of accuracy than could ordinarily be achieved. A similar study, focusing on time domain analysis, provides proofs of convergence and of boundedness for explicit and implicit time integration schemes [39].

A related and more commonly used technique is that of mass scaling. Essentially, this involves adding artificial or ‘virtual’ off-diagonal entries to the mass matrix of a system in order to decrease the eigenvalues of a system and therefore increase the critical time step. Macek and Aubert [54] and Olovsson et al. [59], each propose using so-called ‘selective mass scaling’, which attempts to target the higher eigenfrequencies so that they are reduced to a greater extent than the lower modes. This method is generally seen as purely heuristic, with little physical basis, although in a recent paper Askes et al. argue that mass penalties for multipoint constraints, mass scaling and the concept of micro-inertia are in fact equivalent for 1D linear finite

elements [6].

The mass penalty method presented here proceeds in a similar way to the stiffness penalty method, but instead of penalising the potential energy of the system we penalise the kinetic energy expression given in (2.4). In order to do this we must differentiate the displacement constraint equations (1.3) with respect to time in order to obtain the corresponding *velocity* constraints, so that

$$\dot{\mathbf{h}} = \mathbf{S}\dot{\mathbf{u}} - \dot{\mathbf{q}} \quad (2.12)$$

We then modify the expression for kinetic energy (2.4) accordingly, giving

$$\mathcal{T}^P = \frac{1}{2}\dot{\mathbf{u}}^T \mathbf{M}\dot{\mathbf{u}} + \frac{1}{2}\dot{\mathbf{h}}^T \mathbf{P}_m \dot{\mathbf{h}} \quad (2.13)$$

Once again minimising total energy, with kinetic energy penalised and potential energy unaltered, we obtain

$$\frac{d}{dt} \frac{\partial \mathcal{T}^P}{\partial \dot{\mathbf{u}}^T} + \frac{\partial \mathcal{U}}{\partial \mathbf{u}^T} = (\mathbf{M} + \mathbf{M}^P)\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} + \mathbf{f}_m^P \quad (2.14)$$

where  $\mathbf{M}^P = \mathbf{G}^T \mathbf{P}_m \mathbf{G}$  and  $\mathbf{f}_m^P = \mathbf{G}^T \mathbf{P}_m \dot{\mathbf{q}}$ . Therefore, the mass penalties in  $\mathbf{M}^P$  are only activated if the computed acceleration vector violates the given constraints. Note that the term  $\mathbf{f}_m^P$  is rarely required, since the prescribed displacement vector  $\mathbf{q}$  is generally constant in time and hence  $\dot{\mathbf{q}} = \ddot{\mathbf{q}} = \mathbf{0}$ .

The physical interpretation of mass penalties is a little more esoteric than that of the stiffness penalty method. While for simple absolute constraints it is tempting to think of simply attaching large masses to the structure, this analogy breaks down when considering relative constraints. What is needed is a mechanical analogy to the spring: a device which applies a force to its two terminals when a relative acceleration is applied. In fact, it seems that such a device was not identified until recently, when the concept of the mechanical inerter was first introduced in 2002 [73]. The inerter does for acceleration what springs and dampers do for displacement and velocity, respectively, without needing large mass or a connection to ground. The symbol for the mechanical inerter is shown in Figure 2.2 and a possible physical implementation is shown in Figure 2.3. In any case, we can think of the mass penalty method as adding non-physical mass (or, more precisely, inertia) at certain points in order to

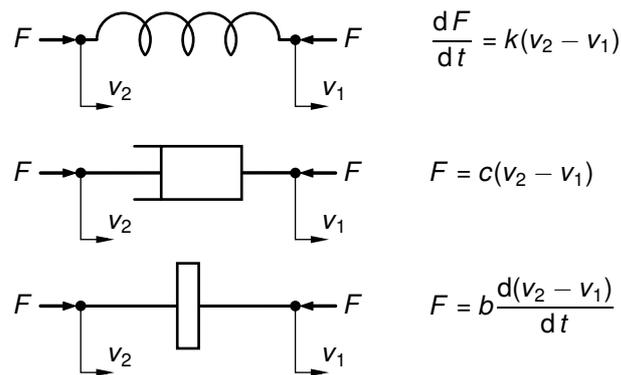


Figure 2.2: Mechanical symbols and governing equations for a spring, damper and inerter [73]

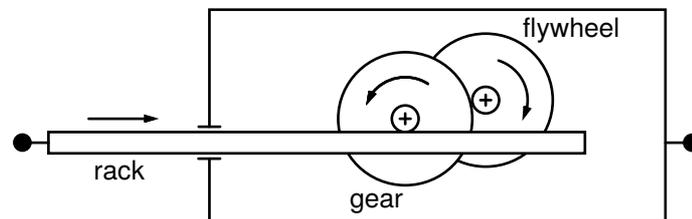


Figure 2.3: Schematic of a simple realisation of the mechanical inerter [73]

restrain movement.

Returning to the simple example of a constrained 1D bar, we see that absolute constraints may be interpreted either as a large mass (Fig. 2.4(b)), or as a direct analogy to the stiffness penalty method with an inerter in place of the spring (Fig. 2.4(c)). Figure 2.4(d) shows a relative constraint between nodes 2 and 4. Note that this again leads to off-diagonal terms in the mass matrix. This would not usually be considered a problem when used in conjunction with a consistent mass matrix. However, when using lumped mass (as we are here) it is possible to make the assumption that the mass matrix is diagonal, which is sometimes useful when selecting a solver. The fact that the mass matrix has off-diagonal terms must therefore be taken into account when imposing relative constraints with the mass penalty method.

Although mass penalties can be used to obtain constraint imposition accuracy equal to that of the stiffness penalty method in some cases, a much larger penalty parameter is generally required, which can lead to ill-conditioning of the mass matrix. According to Paraskevopoulos et al. [61], there is also a “common consensus” that mass penalty methods are “not derived from a rigorous mathematical formulation”, although they argue that the mass penalty method is just as valid as the traditional

stiffness penalty method. Of course, the method cannot be used in statics, which also limits its widespread adoption. However, since adding mass penalties cannot increase the Rayleigh quotient (2.6) of a problem, there is no increase in maximum eigenfrequency [39, 47], which in turn is a great advantage when using explicit time integration.

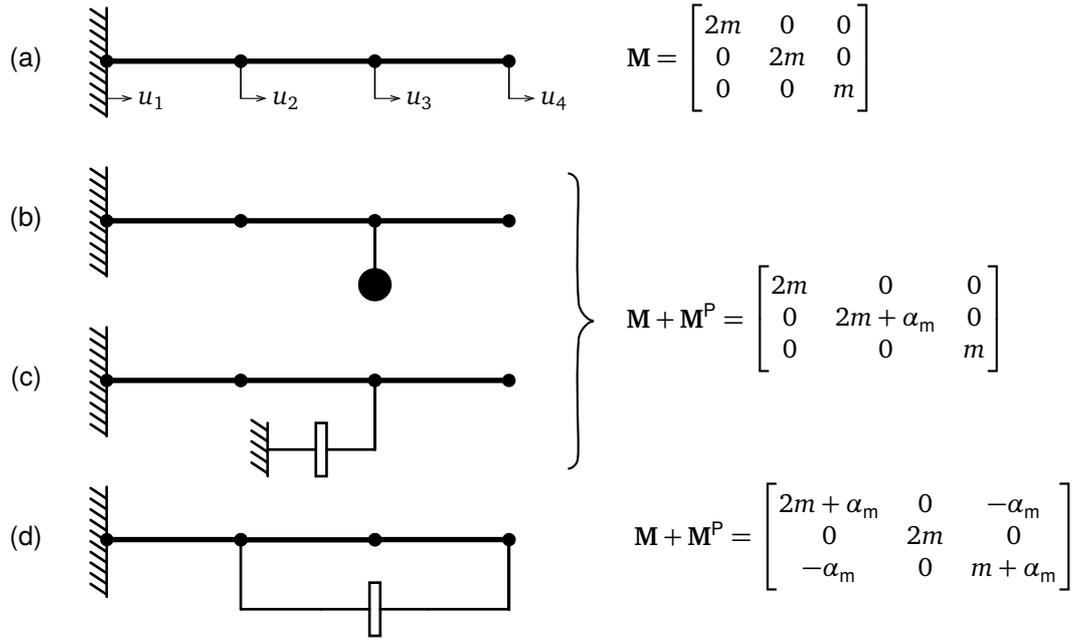


Figure 2.4: Left: physical representation of a 1D bar subject to various constraints, modelled using 2D finite elements and the mass penalty method. Right: the relevant reduced global mass matrices.

### 2.3 The bipenalty method

After introducing these two methods, it is a simple matter to combine them for simultaneous use as the bipenalty method. In this case, penalty functions are applied to both the potential and kinetic energy expressions given in Equations (2.1) and (2.12), giving

$$\mathcal{U}^P = \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} - \mathbf{u}^T \mathbf{f} + \frac{1}{2} \mathbf{h}^T \mathbf{P}_s \mathbf{h} \quad (2.15)$$

$$\mathcal{F}^P = \frac{1}{2} \dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}} + \frac{1}{2} \dot{\mathbf{h}}^T \mathbf{P}_m \dot{\mathbf{h}} \quad (2.16)$$

Minimisation of total energy now yields

$$\frac{d}{dt} \frac{\partial \mathcal{T}^P}{\partial \dot{\mathbf{u}}^T} + \frac{\partial \mathcal{W}^P}{\partial \mathbf{u}^T} = (\mathbf{M} + \mathbf{M}^P) \ddot{\mathbf{u}} + (\mathbf{K} + \mathbf{K}^P) \mathbf{u} = \mathbf{f} + \mathbf{f}^P \quad (2.17)$$

$$\text{where} \quad \mathbf{K}^P = \mathbf{S}^T \mathbf{P}_s \mathbf{S} \quad (2.18)$$

$$\mathbf{M}^P = \mathbf{S}^T \mathbf{P}_m \mathbf{S} \quad (2.19)$$

$$\mathbf{f}^P = \mathbf{S}^T \mathbf{P}_s \mathbf{q} + \mathbf{S}^T \mathbf{P}_m \ddot{\mathbf{q}} \quad (2.20)$$

Note that both the stiffness penalty matrix  $\mathbf{K}^P$  and mass penalty matrix  $\mathbf{M}^P$  have the same form, the only difference being the penalty parameters contained in  $\mathbf{P}_s$  and  $\mathbf{P}_m$ . If we assume that the penalty parameters have the same ratio for all constraint equations, so that

$$R = \frac{\alpha_{s,i}}{\alpha_{m,i}} \quad i \in [1, n_c] \quad (2.21)$$

then the two penalty matrices are in fact proportional to each other, and the stiffness penalty matrix can be written as

$$\mathbf{K}^P = R \mathbf{M}^P \quad (2.22)$$

This will be a common assumption in later chapters since it simplifies the analysis of problems without being too restrictive; while it may be useful to use different parameter *magnitudes* for different types of constraint there is normally no motivation to vary the ratios.

As stated earlier, stiffness penalties increase the maximum eigenfrequency of a system while mass penalties do not. By using both together the effects of one are countered by the effects of the other. In other words, we can tune the penalties used in a bipenalty formulation in order to control the spurious eigenfrequencies that are introduced by the stiffness penalty method without resorting to using the less accurate mass penalty method. This idea has been explored for single absolute constraints on certain finite elements [5] and as a method of controlling the ratio of maximum to minimum eigenfrequency ( $S = \omega_{\max}/\omega_{\min}$ ) [61]. In the frequency domain it has recently been used to improve the convergence properties of a penalised constraint [49]. In Chapter 3 we further explore the effect of the bipenalty method on the eigenfrequencies of a system in a general way, so that clear guidance can be given on the best choice of penalty ratio.

## 2.4 The bipenalty method with damping

So far we have not considered structural damping in our formulation. In an FE analysis this is accounted for using a matrix of damping coefficients,  $\mathbf{C}$ , so that the dynamic equations of motion become

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (2.23)$$

Thus, a velocity-dependent damping force is introduced. The intention is to approximate the overall energy dissipation during the structural response. Although the damping matrix is easily derived by including velocity dependent damping forces in the strong form equation, the elemental damping matrices involve damping parameters that are frequency dependent, which in practice makes the standard FE assembly procedure impossible [9, §4.2.1]. Hence, the damping matrix is often formed from a linear combination of the stiffness and mass matrices; that is,

$$\mathbf{C} = a\mathbf{M} + b\mathbf{K} \quad (2.24)$$

with the constant damping coefficients  $a$  and  $b$  determined by comparison with experimental results. This strategy of using a damping matrix with both a stiffness- and mass-proportional term is known as Rayleigh damping.

Here we propose a penalty formulation with damping penalties alongside the stiffness and mass penalties of the bipenalty method, so that the damped structural equations (2.23) become

$$(\mathbf{M} + \mathbf{M}^P)\ddot{\mathbf{u}} + (\mathbf{C} + \mathbf{C}^P)\dot{\mathbf{u}} + (\mathbf{K} + \mathbf{K}^P)\mathbf{u} = \mathbf{f} + \mathbf{f}^P \quad (2.25)$$

In order to derive the damping penalty matrix  $\mathbf{C}^P$  we utilise Rayleigh damping, but use the stiffness and mass *penalty* matrices as a basis, instead of the full system equations. Since the stiffness penalty matrix  $\mathbf{K}^P$  is a scalar multiple of  $\mathbf{M}^P$  we neglect the damping coefficient  $a$  in Equation (2.24) and define the damping penalty matrix simply as

$$\mathbf{C}^P = D\mathbf{K}^P \quad (2.26)$$

where  $D$  is the damping factor, representing the magnitude of the damping penalty

relative to the stiffness penalty.

It is perhaps technically incorrect to refer to this technique as a penalty method, since it cannot be derived by penalisation of an energy expression. It is included here as an obvious and intuitive extension of the stiffness and mass penalty methods we have previously introduced; if the stiffness penalty method is physically analogous to adding springs to the system (and the mass penalty method corresponds to adding inerters), then damping penalties can be interpreted as the addition of dashpots.

A possible theoretical framework for the formulation presented here is provided by Asano [2–4], who has developed a virtual work principle utilising penalty functions that act on displacements, velocities and accelerations in the context of a contact-impact algorithm. However, more work is required to turn this theory into a generally applicable penalty formulation.

The inclusion of penalty damping has no effect on time step stability, as will be discussed in Section 3.2, but does have the potential to increase accuracy of constraint imposition by placing additional restraint on nodal velocity. Some examples of this in relation to contact-impact problems will be given in Chapter 6.

## 2.5 Matrix partitioning for the central difference method

We wish to utilise the bipenalty method with the CDM time integration scheme in order to maximise the critical time step. However, a major advantage of the CDM is that, when lumped mass is used and the mass matrix is diagonal, no simultaneous equations need to be solved, making it very efficient. As we have seen, the bipenalty method generates off-diagonal terms in the mass matrix when multi-point constraints are introduced, which leads to a linear system of simultaneous equations that must be solved at each time step.

Fortunately, as long as lumped mass is used for the continuum elements in the model, the majority of the mass matrix will remain diagonal as long as the number of constrained DOF is small compared to the total number of DOF associated with the system. In this case, we may use matrix partitioning to reduce the size of the system that must be handled with a linear solver, leaving the majority of the displacements to be calculated in the normal way.

First, assume that the displacement solution variables are ordered so that

$$\mathbf{u} = \begin{bmatrix} \mathbf{f}\mathbf{u} \\ \mathbf{c}\mathbf{u} \end{bmatrix} \quad (2.27)$$

where  $\mathbf{f}\mathbf{u}$  and  $\mathbf{c}\mathbf{u}$  are the displacements for free (unconstrained) DOF and constrained DOF, respectively. Constrained in this case means that the DOF has a non-zero entry in the corresponding column of the constraint matrix  $\mathbf{G}$ . The stiffness and mass matrices are then arranged in a similar manner so that

$$\mathbf{K} = \begin{bmatrix} \mathbf{ff}\mathbf{K} & \mathbf{fc}\mathbf{K} \\ \mathbf{cf}\mathbf{K} & \mathbf{cc}\mathbf{K} \end{bmatrix} \quad \mathbf{K}^P = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{cc}\mathbf{K}^P \end{bmatrix} \quad (2.28)$$

and

$$\mathbf{M} = \begin{bmatrix} \mathbf{ff}\mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{cc}\mathbf{M} \end{bmatrix} \quad \mathbf{M}^P = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{cc}\mathbf{M}^P \end{bmatrix} \quad (2.29)$$

The equations that govern the CDM solution are written in matrix form as

$$\frac{1}{\Delta t^2} (\mathbf{M} + \mathbf{M}^P) \mathbf{d}_{t+\Delta t} = \mathbf{r}_t \quad (2.30)$$

where  $\mathbf{r}_t$  is a residual force vector, which, when damping is neglected, is given by

$$\mathbf{r}_t = \begin{bmatrix} \mathbf{f}\mathbf{r}_t \\ \mathbf{c}\mathbf{r}_t \end{bmatrix} = \mathbf{f}_t - \left[ (\mathbf{K} + \mathbf{K}^P) - \frac{2}{\Delta t^2} (\mathbf{M} + \mathbf{M}^P) \right] \mathbf{d}_t - \frac{1}{\Delta t^2} (\mathbf{M} + \mathbf{M}^P) \mathbf{d}_{t-\Delta t} \quad (2.31)$$

(see Section 1.2.1 or Bathe [9, §9.2.1]). Due to the structure of the mass matrix, the free and constrained systems are uncoupled, and can be written separately as

$$\left( \frac{1}{\Delta t^2} \mathbf{ff}\mathbf{M} \right) \mathbf{f}\mathbf{d}_{t+\Delta t} = \mathbf{f}\mathbf{r}_t \quad (2.32)$$

$$\frac{1}{\Delta t^2} (\mathbf{cc}\mathbf{M} + \mathbf{cc}\mathbf{M}^P) \mathbf{c}\mathbf{d}_{t+\Delta t} = \mathbf{c}\mathbf{r}_t \quad (2.33)$$

Since  $\mathbf{ff}\mathbf{M}$  is diagonal, (2.32) is trivial to solve. On the other hand,  $\mathbf{cc}\mathbf{M}^P$  has off-diagonal terms, and hence (2.33) requires a linear solver. The size of the system that must be solved is dependent on the number of DOF that are constrained using the bipenalty method.

Although structural damping is rarely used with the CDM, we do need to consider the damping penalty matrix suggested in Section 2.4. Therefore, we have

$$\mathbf{C} = \mathbf{0} \quad \mathbf{C}^P = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & {}^{cc}\mathbf{C}^P \end{bmatrix} \quad (2.34)$$

In this case, the CDM equations are decoupled in much the same way, with the free and constrained displacements determined using

$$\left( \frac{1}{\Delta t^2} {}^{ff}\mathbf{M} \right) {}^f \mathbf{d}_{t+\Delta t} = {}^f \hat{\mathbf{f}}_t \quad (2.35)$$

$$\left[ \frac{1}{\Delta t^2} ({}^{cc}\mathbf{M} + {}^{cc}\mathbf{M}^P) + \frac{1}{2\Delta t} {}^{cc}\mathbf{C}^P \right] {}^c \mathbf{d}_{t+\Delta t} = {}^c \hat{\mathbf{f}}_t \quad (2.36)$$

where the residual is now given by

$$\hat{\mathbf{f}}_t = \begin{bmatrix} {}^f \hat{\mathbf{f}}_t \\ {}^c \hat{\mathbf{f}}_t \end{bmatrix} = \mathbf{f}_t - \left[ (\mathbf{K} + \mathbf{K}^P) - \frac{2}{\Delta t^2} (\mathbf{M} + \mathbf{M}^P) \right] \mathbf{d}_t - \left[ \frac{1}{\Delta t^2} (\mathbf{M} + \mathbf{M}^P) - \frac{1}{2\Delta t} \mathbf{C}^P \right] \mathbf{d}_{t-\Delta t} \quad (2.37)$$

Clearly, if a bipenalty formulation is being used, then the addition of damping penalties does not affect the size of the system that must be solved, and hence penalising the damping matrix has a relatively small computational overhead in this case.



## Chapter 3

# Time step stability

The conditional stability of explicit time integration schemes was first reported by Courant, Friedrichs and Lewy [24]. Since then, the theories have been more generally applied (for a summary in the context of FE analysis, see Hughes [43]) and in its most general form the Courant-Friedrichs-Lewy (CFL) condition is given by

$$\Delta t_{\text{crit}} = \frac{\Omega_{\text{crit}}}{\omega_{\text{max}}} \quad (3.1)$$

as previously discussed in Section 1.2.3. Here we have also introduced the so-called critical sampling frequency  $\Omega_{\text{crit}}$ , which allows us to account for various time integration schemes, as shown in Table 9.1.1 of Reference [43]. For the CDM, on which we will be focussing,  $\Omega_{\text{crit}} = 2$  and we will assume this value for the remainder of this chapter.

The effect of the traditional penalty method on this critical time step has also been the subject of attention in the literature. Of course, this is really a study of the effect on the maximum eigenfrequency of the system,  $\omega_{\text{max}}$ , and therefore the underlying eigenvalue problem (as briefly discussed in Section 1.2.3). For a dynamic system with  $n$  DOF, the eigenvalue problem is given by

$$(\mathbf{K} - \lambda_i \mathbf{M}) \mathbf{u}_i = \mathbf{0} \quad i = 1 \dots n \quad (3.2)$$

where  $\lambda_i$  are the  $n$  eigenvalues, sorted into ascending order so that  $\lambda_n \equiv \lambda_{\text{max}}$ , and  $\mathbf{u}_i$  are the corresponding eigenvectors (also referred to as eigenmodes). The eigenfrequencies are simply the positive roots of the eigenvalues,  $\omega_i = \sqrt{\lambda_i}$ . Analytical

solutions for  $\lambda_i$  can be obtained for some single undistorted finite elements with  $n \leq 8$  [5], but for large systems this approach becomes impossible. In general, numerical procedures or approximate estimates must be used to determine system eigenvalues (see Section 3.3.2).

The stiffness penalty method has the effect of increasing the maximum eigenvalue (and therefore eigenfrequency) of the system. For example, Belytschko and Neal show that for a single element, “the introduction of the [stiffness] penalty always decreases the stable time step” [14], while a paper by Hetherington and Askes reaches the same conclusion, and points out that mass penalties, conversely, can never decrease the critical time step [39]. Ilanko’s frequency domain studies of artificial restraints show that this effect is independent of the sign of the penalty for both stiffness and mass penalty methods [46, 47].

The effect of the bipenalty method, however, is not as well understood, although in recent years there have been two studies which investigate the effect on the eigenvalue problem. Firstly, Paraskevopoulos et al. presented their “consistent penalty formulation”, which derives both the stiffness and mass penalty methods via a variational formulation. In terms of eigenvalue analysis, this work concentrates on preserving the so-called “stiffness” (defined in this context as the ratio of the largest eigenfrequency to the lowest,  $S = \omega_{\max}/\omega_{\min}$ ) of the system equations due to its importance with regards to accuracy and stability. They provide a ratio of stiffness and mass penalties which “does not alter the order of the system stiffness, nor the order of maximum and minimum eigenfrequency”.

The second such analysis of the bipenalty method is provided by Ilanko and Monterrubio [49]. Here the main focus is on finding the penalty ratio that gives the highest convergence rate for the Rayleigh-Ritz method. They explore the use of varying penalty ratios in order to obtain bounded results on the exact (fully constrained) solution.

This chapter begins with an study of the precise effect that bipenalisation has on both the eigenvalues and eigenmodes of a FE system by analysis of the underlying generalised (dynamic) eigenvalue problem. Much of this analysis has been recently published by the author and colleagues [40, 41]. We then discuss practical eigenfrequency/time step estimation and how the results can be used in order to select penalty parameters with optimal characteristics with respect to time step stability.

### 3.1 The bipenalised eigenproblem

The eigenvalue problem for a bipenalised system of size  $n$  is

$$[(\mathbf{K} + \mathbf{K}^P) - \tilde{\lambda}_i (\mathbf{M} + \mathbf{M}^P)] \tilde{\mathbf{u}}_i = \mathbf{0} \quad i = 1 \dots n \quad (3.3)$$

where  $\mathbf{K}^P$  and  $\mathbf{M}^P$  are the stiffness and mass penalty matrices as described in Section 2.3 and  $\tilde{\lambda}_i$  and  $\tilde{\mathbf{u}}_i$  are the  $n$  eigenvalues and corresponding eigenvectors for the bipenalised problem (BP). The penalty matrices are formulated such that they impose the  $n_c$  constraints given by

$$\mathbf{h} = \mathbf{G}\mathbf{u} - \mathbf{q} \quad (3.4)$$

as described in Section 1.1. We assume that there are no repeated constraint equations and therefore that the rows of  $\mathbf{G}$  are linearly independent.

Imposing these constraints via transformation of the system equations (i.e., by direct imposition) results in a system of size  $n - n_c$ ; one DOF must be removed for each additional constraint [22]. The fully constrained eigenvalues are bounded by those of the unconstrained system, according to Rayleigh's theorem of separation [14]. This means that

$$\omega_1 \leq \bar{\omega}_k \leq \omega_n \quad \forall k = 1 \dots n - n_c \quad (3.5)$$

where  $\bar{\omega}_k$  are the  $n - n_c$  eigenvalues of the system subject to  $n_c$  exactly enforced constraints. On the other hand, the penalised system does not decrease in size and therefore has  $n$  eigensolutions independent of the number of applied constraints. In addition, these eigenvalues are clearly *not* bounded by those of the unconstrained system. Our goal is to investigate these 'extra' eigensolutions and to find a way of predicting what the associated eigenvalues will be.

#### 3.1.1 Mathematical proofs on eigenvalues and the penalty ratio

**Theorem 1.** *When applying  $n_c$  constraint equations to a system of size  $n$  using the bipenalty method, exactly  $n - n_c$  of the eigenmodes associated with the system satisfy those constraints, while exactly  $n_c$  of the eigenmodes associated with the system do not, for large  $\alpha_m$ .*

*Proof.* First we note that the eigenvectors  $\tilde{\mathbf{u}}_i$  of a finite element system may be scaled

such that they are orthonormal with respect to the mass matrix of the system, giving

$$\tilde{\mathbf{u}}_i^T (\mathbf{M} + \mathbf{M}^P) \tilde{\mathbf{u}}_j = \delta_{ij} \quad (3.6)$$

where  $\delta_{ij}$  is the Kronecker delta. Considering the case where  $i \neq j$  we have

$$\tilde{\mathbf{u}}_i^T \mathbf{M} \tilde{\mathbf{u}}_j + (\mathbf{G} \tilde{\mathbf{u}}_i)^T \mathbf{P}_m (\mathbf{G} \tilde{\mathbf{u}}_j) = 0 \quad i \neq j \quad (3.7)$$

We now rewrite the matrix of penalty parameters so that  $\mathbf{P}_m = \alpha_m \mathbf{D}_m$  where  $\mathbf{D}_m$  is a dimensionless diagonal matrix of size  $n_c$ . Furthermore, we assume that all non-zero entries in  $\mathbf{D}_m$  have the same sign; i.e., all entries are positive, or all entries are negative<sup>1</sup>. Then,

$$\tilde{\mathbf{u}}_i^T \mathbf{M} \tilde{\mathbf{u}}_j + \alpha_m (\mathbf{G} \tilde{\mathbf{u}}_i)^T \mathbf{D}_m (\mathbf{G} \tilde{\mathbf{u}}_j) = 0 \quad (3.8)$$

from which,

$$\lim_{\alpha_m \rightarrow \infty} [(\mathbf{G} \tilde{\mathbf{u}}_i)^T \mathbf{D}_m (\mathbf{G} \tilde{\mathbf{u}}_j)] = 0 \quad (3.9)$$

The assumption that all entries in  $\mathbf{D}_m$  have the same sign means that there cannot be compensation during the above matrix multiplication, and hence we are left with two possibilities for the vector  $\mathbf{G} \tilde{\mathbf{u}}_i$ , assuming large  $\alpha_m$ :

1.  $\mathbf{G} \tilde{\mathbf{u}}_i = \mathbf{0}$ . This is possible for at most  $n - n_c$  of the  $n$  eigenmodes. This can be shown by considering the rank-nullity theorem, which states that  $\text{rank}(\mathbf{A}) + \text{nullity}(\mathbf{A}) = n$  for any  $m \times n$  matrix  $\mathbf{A}$  [55]. Since the rows of  $\mathbf{G}$  are linearly independent, we know that  $\text{rank}(\mathbf{G}) = n_c$  and therefore  $\text{nullity}(\mathbf{G}) = n - n_c$ .
2.  $\mathbf{G} \tilde{\mathbf{u}}_i$  and  $\mathbf{G} \tilde{\mathbf{u}}_j$  are non-zero and orthogonal. This is possible for at most  $n_c$  of the  $n$  modes, since the vector  $\mathbf{G} \tilde{\mathbf{u}}_i$  is of dimension  $n_c$ .

Following directly from the above, we can therefore say that

1.  $\mathbf{G} \tilde{\mathbf{u}}_i = \mathbf{0}$  for exactly  $n - n_c$  of the  $n$  eigenvectors,
2.  $\mathbf{G} \tilde{\mathbf{u}}_i \neq \mathbf{0}$  for exactly  $n_c$  of the  $n$  eigenvectors. □

---

<sup>1</sup>This assumption does not constitute a significant loss in generality, since negative penalties and positive penalties are not usually used in the same analysis; when employed, negative penalties are commonly used for all constraints, usually to complement a separate positive-penalty analysis [39, 46, 47].

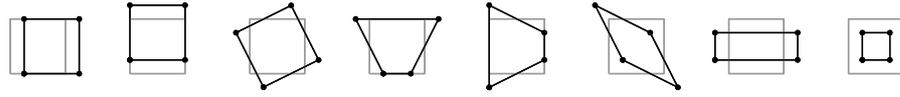


Figure 3.1: Mode shapes for an unconstrained four-noded square finite element.

The constrained modes tend to those of the fully constrained system for large penalty parameters, and therefore, from Rayleigh's theorem of separation, the eigenvalues are bounded by those of the unconstrained system. Hence it is the spurious modes—those which do not satisfy the constraints—that introduce problematic eigenvalues. The second theorem addresses exactly what happens to those eigenvalues.

**Theorem 2.** *For any system subject to  $n_c$  bipenalty constraints with large  $\alpha_m$ ,  $n_c$  of the associated eigenvalues tend to the penalty ratio,  $R$ .*

*Proof.* Consider the Rayleigh quotient of the BP with the arbitrary vectors  $\mathbf{v}$  replaced by the  $n$  eigenvectors of the BP

$$\mathcal{R}_{\text{BP}}(\tilde{\mathbf{u}}_i) = \frac{\tilde{\mathbf{u}}_i^T \mathbf{K} \tilde{\mathbf{u}}_i + \tilde{\mathbf{u}}_i^T \mathbf{K}^p \tilde{\mathbf{u}}_i}{\tilde{\mathbf{u}}_i^T \mathbf{M} \tilde{\mathbf{u}}_i + \tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i} = \frac{\tilde{\mathbf{u}}_i^T \mathbf{K} \tilde{\mathbf{u}}_i + R \cdot \tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i}{\tilde{\mathbf{u}}_i^T \mathbf{M} \tilde{\mathbf{u}}_i + \tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i} \quad (3.10)$$

For eigenmodes with  $\tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i \neq 0$  the penalty terms dominate, so that in the limit

$$\lim_{\alpha_m \rightarrow \infty} \mathcal{R}_{\text{BP}}(\tilde{\mathbf{u}}_i) = \frac{R \cdot \tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i}{\tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i} = R \quad (3.11)$$

That is, the eigenvalue corresponding to any eigenvector that gives a non-zero  $\tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i$  tends to  $R$ , the penalty ratio, for large  $\alpha_m$ . From Theorem 1, exactly  $n_c$  eigenmodes have  $\tilde{\mathbf{u}}_i^T \mathbf{M}^p \tilde{\mathbf{u}}_i \neq 0$ . Therefore, the  $n_c$  eigenvalues associated with these modes tend to  $R$  for large  $\alpha_m$ .  $\square$

### 3.1.2 Illustration: eigensolutions of a four-noded quadrilateral

In order to demonstrate these findings we now consider a four-noded (square) linear finite element. The eight unconstrained mode shapes for the element are shown in Figure 3.1. This set is considered to be comprised of (from left to right) three rigid body modes (two translation and one rotation), two hourglass modes, one shear mode and two volumetric modes. Predicting which mode shape has the highest ei-

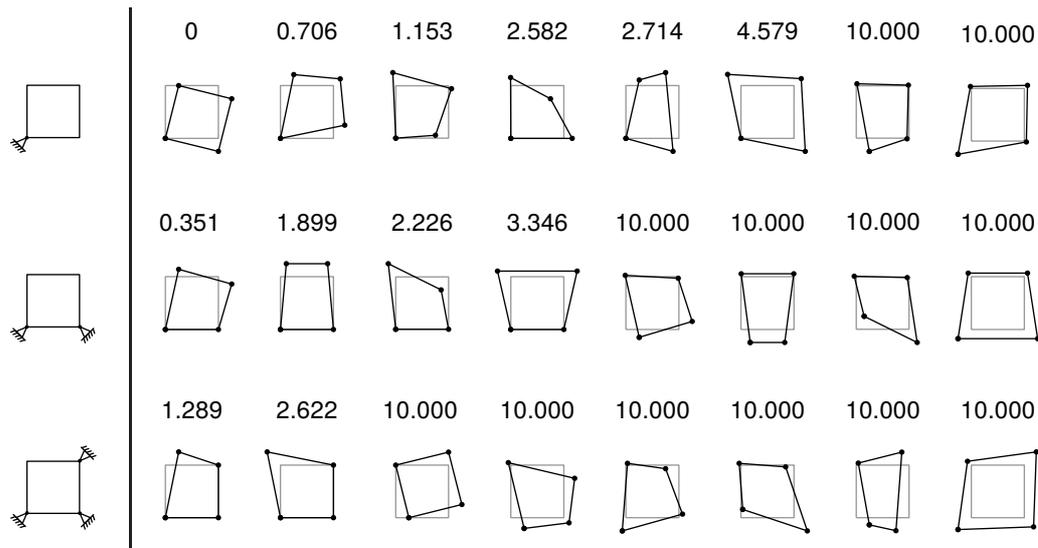


Figure 3.2: Mode shapes and corresponding eigenvalues for a square element: (from top to bottom) one node fixed, two nodes fixed, three nodes fixed. The bipenalty method is used with  $R = 10$ .

genfrequency is difficult, since it depends on many factors such as material properties (e.g., Poisson's ratio), 2D assumption (e.g., plane stress/plane strain) and element distortion. However, it is a simple matter to calculate a solution to the eigenproblem numerically. Figure 3.2 shows the mode shapes for three constrained problems. The top row shows the mode shapes for an element whose bottom left node has been fixed using the bipenalty method. The first six modes appear to comply with the constraint; that is, the displacement of the bottom left node is zero in both the  $x$  and  $y$  directions. Among these, there exists one rigid body mode that conforms with this constraint, which can be identified by its zero eigenvalue. The remaining two modes do not conform to the fixed constraints, and have eigenvalues equal to the penalty ratio,  $R = 10$ .

Of course, we obtain a similar result when using a stiffness or mass penalty method, except for the magnitudes of the spurious eigenvalues. Table 3.1 shows the numerically computed eigenvalues for the stiffness, mass and bipenalty methods. As can be seen, the stiffness penalty method results in two large eigenvalues (approximately proportional to the penalty parameter being used), while for the bipenalty method these eigenvalues tend to  $R = 10$  for large parameters. For the pure mass penalty approach, the 'extra' eigenvalues tend to zero.

Finally, in order to demonstrate how the eigenfrequencies of the element change as bipenalty constraints are introduced, the eigenfrequencies for an increasing num-

Stiffness penalty		Mass penalty		Bipenalty ( $R = 10$ )	
$\alpha_s = 10^3$	$\alpha_s = 10^6$	$\alpha_m = 10^3$	$\alpha_m = 10^6$	$\alpha_s = 10^3$	$\alpha_s = 10^6$
		0.000	0.000		
		0.000	0.000		
0.000	0.000	0.000	0.000	0.000	0.000
0.705	0.706	0.835	0.706	0.705	0.706
1.153	1.153	1.222	1.153	1.153	1.153
2.582	2.582	2.602	2.582	2.582	2.582
2.713	2.714	2.730	2.714	2.713	2.714
4.578	4.579	4.603	4.579	4.578	4.579
4001.289	4000001.289			9.979	10.000
4002.623	4000002.622			9.983	10.000

Table 3.1: Numerically computed eigenvalues of a square element with one fixed node

ber of constrained DOF are shown in Figure 3.3. To produce this plot, absolute constraints are imposed on an increasing number of DOF until all DOF are constrained. At each stage, the eigenfrequencies of the element are calculated numerically for increasing values of  $\alpha_m$  with a constant penalty ratio,  $R = 16$ . The material properties are Young's modulus  $E = 1$  Pa, mass density  $\rho = 1$  kg/m<sup>3</sup>, element side length  $h = 1$  m, Poisson's ratio  $\nu = 0.3$ , and plane stress is assumed.

The bottom figure shows the eigenfrequencies of the unconstrained, unpenalised problem (UP). There are  $n = 8$  eigensolutions in all; in this case there are some repeated eigenfrequencies ( $\omega_1 = \omega_2 = \omega_3$ ,  $\omega_4 = \omega_5$  and  $\omega_6 = \omega_7$ ). As constraints are added, the eigenfrequencies of each constrained problem (indicated by the straight, vertical lines) are clearly bounded by the eigenfrequencies of the less constrained problem below it, as dictated by the eigenvalue separation property. As the penalty parameters are increased,  $n - n_c$  of the BP eigenfrequencies tend to those of the exactly constrained problem; the other  $k$  eigenfrequencies tend to  $\sqrt{R} = 4$ .

## 3.2 Effect of damping penalties

The time step stability condition for the Newmark family of time integration schemes is stated in Equation (3.1) in terms of the maximum eigenvalue  $\omega_{\max}$  and the critical sampling frequency  $\Omega_{\text{crit}}$ . In general, the critical sampling frequency is given by [43]

$$\Omega_{\text{crit}} = \frac{\xi \left( \gamma - \frac{1}{2} \right) + \sqrt{\frac{1}{2} \gamma - \beta + \xi^2 \left( \gamma - \frac{1}{2} \right)^2}}{\frac{1}{2} \gamma - \beta} \quad (3.12)$$

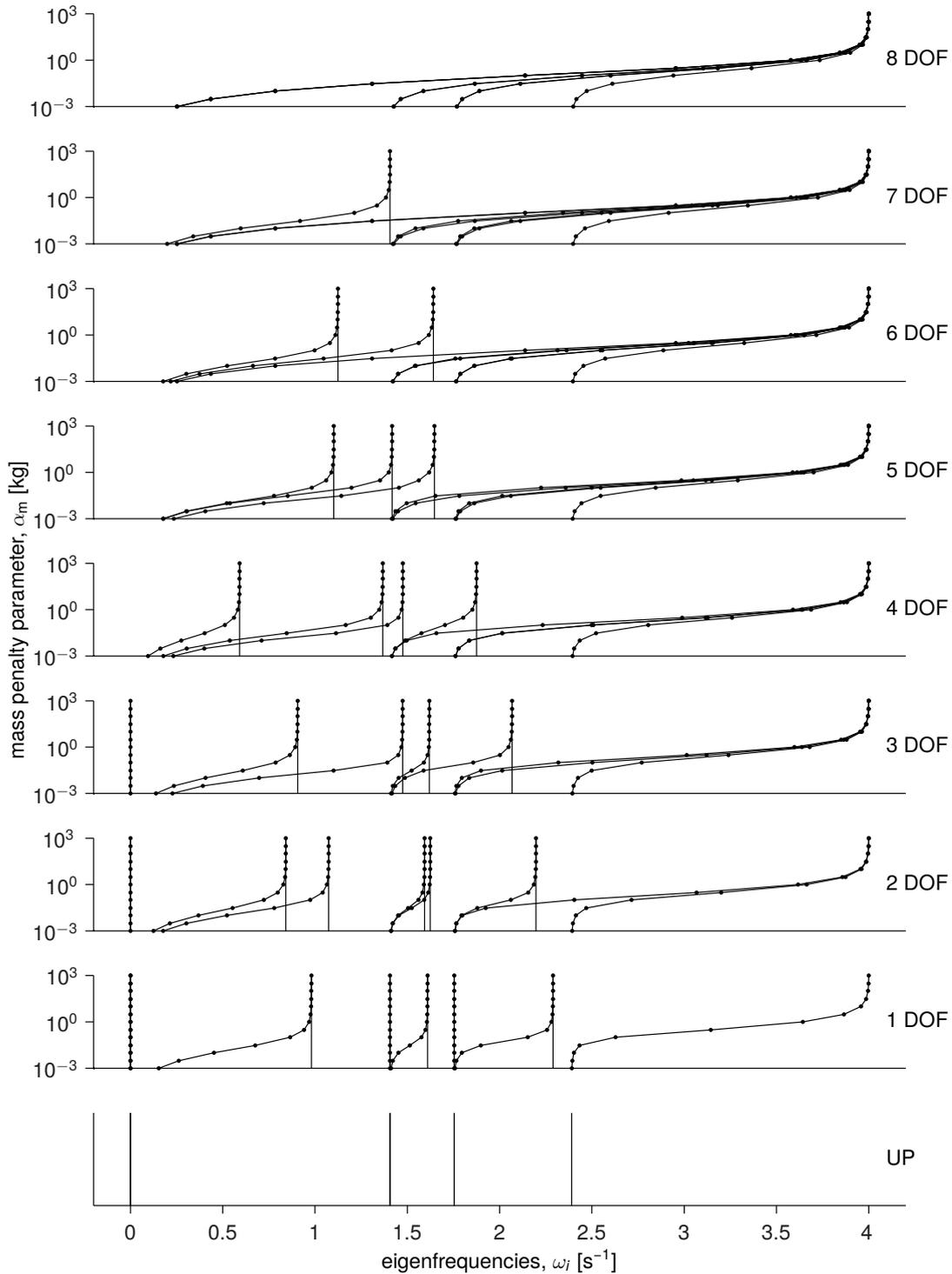


Figure 3.3: The numerically determined eigenfrequencies of a 4-noded quadrilateral element for increasing  $\alpha_m$  and a varying number of constrained DOF. Each plot shows the  $n$  bipenalised eigenfrequencies ( $\text{---}\bullet\text{---}$ ) and the  $n - n_c$  fully constrained eigenfrequencies ( $\text{---}$ ). The penalty ratio is  $R = 16$ .

where  $\xi$  is the damping ratio and  $\gamma$  and  $\beta$  are the Newmark parameters. For the central difference method  $\gamma = 1/2$ , the critical sampling frequency simplifies to

$$\Omega_{\text{crit}} = \frac{1}{\sqrt{\frac{1}{4} - \beta}} \quad \text{for } \gamma = \frac{1}{2} \quad (3.13)$$

and therefore damping has no effect on the critical sampling frequency. For Newmark schemes with  $\gamma > 1/2$ , the undamped critical sampling frequency (with  $\xi = 0$ ) is a conservative estimate for the true value. The natural frequencies (eigenfrequencies)  $\omega_i$  of the system are also unaffected by damping. Therefore, in the vast majority of cases, and certainly when the CDM is being employed for time integration, damping matrices of any form can be used without adversely affecting the critical time step [43, §9.1.2].

### 3.3 Selection of stable penalty ratio

The information presented in this chapter gives a picture of how bipenalty constraints affect the eigenvalues of a dynamic finite element system. Importantly, this allows us to develop guidelines on how to choose penalty parameters such that time step stability of the system is ensured. Since the eigenvalues depend on the penalty ratio  $R$ , we will focus on this quantity. The selection of appropriate penalty magnitudes (i.e., the specific values of  $\alpha_s$  and  $\alpha_m$ ) is more dependent on the accuracy that is required, and hence will be dealt with in Chapter 4.

#### 3.3.1 The critical penalty ratio

The simplest method of ensuring that penalty constraints do not lead to time step instability is to choose a ratio equal to, or less than, the maximum eigenvalue of the unconstrained system. A ratio above this value will lead to an increase in the maximum eigenfrequency, assuming that large parameters are used. The ratio at which this occurs is known as the critical penalty ratio, defined as

$$R_{\text{crit}} = \lambda_{\text{max}}^{\text{UP}} \quad (3.14)$$

As shown in Section 3.1.1, taking  $R = R_{\text{crit}}$  means that the extra eigenvalues introduced by the use of penalty functions will tend to  $R_{\text{crit}}$  for large  $\alpha_m$ , and therefore  $\lambda_{\text{max}}^{\text{BP}} \leq \lambda_{\text{max}}^{\text{UP}}$ . Hence, if the time step used for the analysis is suitable for the *unpenalised* system, it will also be suitable for the bipenalised system.

While theoretically sound, this strategy requires that the maximum eigenvalue of the unpenalised system is known. In practice, the maximum eigenvalue must be computed numerically, which can be expensive. To find a more practical solution, it is useful to examine how the critical time step is estimated when the maximum eigenfrequency of a system is unknown.

### 3.3.2 Calculating a penalty ratio from a stable time step

Selection of a suitable time step is critical for conditionally stable explicit analysis types. An overestimation leads to spurious solutions by introducing instability, while an underestimation may result in an analysis which is needlessly expensive in terms of computer time. For some nonlinear analyses the critical time step may change during the course of the analysis and so must be checked at each time step. This means that an efficient way to find a suitable time step is of great importance.

Generally, the only way to calculate  $\Delta t_{\text{crit}}$  exactly is to compute the maximum eigenfrequency of the whole system. This may be achieved without finding the complete eigensolution by forward iteration [9] (also known as the power method [76]). Forward iteration is a simple algorithm that requires only matrix multiplications (i.e., no matrix decomposition) to find the largest eigenvalue  $\lambda_{\text{max}} \equiv \lambda_n$ . However, convergence is dependent on the ratio of the two largest eigenvalues  $\lambda_{n-1}/\lambda_n$ ; if this ratio is close to 1 then convergence is slow [76].

The element eigenvalue inequality [14] provides an easy method of finding a conservative estimate for  $\omega_{\text{max}}$ . The theorem states that

$$\omega_{\text{max}}^e \geq \omega_{\text{max}} \quad (3.15)$$

where  $\omega_{\text{max}}$  is the maximum system eigenfrequency and  $\omega_{\text{max}}^e$  is the maximum eigenfrequency of all individual elements. This means that by calculating only the maximum eigenfrequencies of the individual elements in the mesh we can obtain a value that is not less than the maximum eigenfrequency of the whole system, which

can then be used to calculate a conservative estimate of the critical time step via

$$\Delta t_{\text{crit}}^e = \frac{2}{\omega_{\text{max}}^e} \leq \Delta t_{\text{crit}} = \frac{2}{\omega_{\text{max}}} \quad (3.16)$$

As an example, consider the time step control employed by the explicit FE code LS-DYNA [36]. Here, time step estimations are conducted on the element level, with the smallest elemental time step size used for the next integration step. To further speed up the process, elemental time steps are calculated directly from the properties of the element (e.g., volume, characteristic length, density etc.) rather than by calculating  $\omega_{\text{max}}$ .

No matter how the elemental time step is calculated, however, there is always the danger of disregarding additional penalty constraints when the time step is computed in an element-wise fashion. For example, the penalty stiffness for the same software's contact-impact algorithm is multiplied by special scale factor, since the addition of the penalty "may cause instabilities unless the time step size is scaled back in the time step calculation" [36, 37]. Likewise, Belytschko and Neal show that for their contact-impact formulation "the introduction of the penalty always decreases the stable time step" [14]. The goal for our bipenalty formulation is to ensure that no penalty constraint can cause a decrease in the critical time step of the analysis.

By rearranging the CFL condition (3.1) we can look at the problem from another angle. If a time step has been chosen for a specific analysis, then we have

$$\omega_{\text{max}} \leq \frac{2}{\Delta t} \quad (3.17)$$

That is, the maximum eigenfrequency must be kept below a value set by chosen time step  $\Delta t$ . If this value has been safely selected according to element-wise calculation described above, then it is the spurious frequencies introduced by the additional constraints which lead to the  $\omega_{\text{max}}$  exceeding the limit.

Fortunately, using the bipenalty method we are able to control the magnitude of these spurious eigenfrequencies. In order to keep them from interfering with the time step  $\Delta t$ , we simply ensure that our penalty ratio complies with

$$R \leq \frac{4}{\Delta t^2} \quad (3.18)$$

### 3.3.3 Example: time step instability due to a single penalty constraint

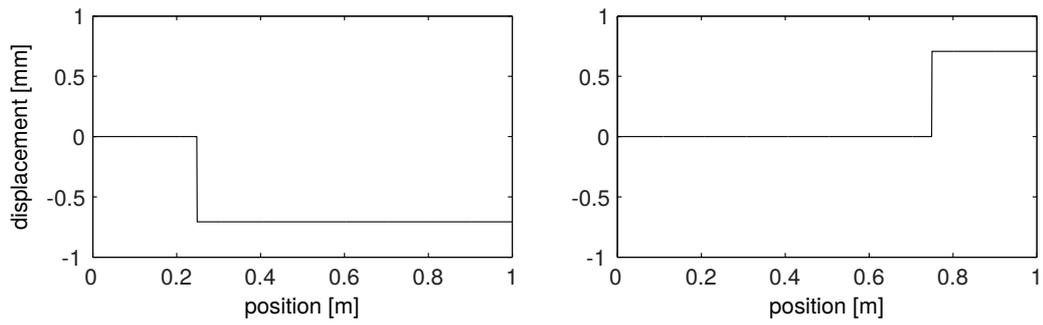
To complete this chapter, we briefly demonstrate the effect of using a super-critical penalty ratio in the case of wave propagation through a one-dimensional bar. The bar in question has a length of 1 m and is discretised in space by 1000 2-noded linear bar elements. The material properties are Young's modulus  $E = 0.01$  Pa, mass density  $\rho = 20000$  kg/m<sup>3</sup> and cross-sectional area  $A = 0.1$  m<sup>2</sup>. This leads to a maximum unpenalised eigenvalue of  $\lambda_{\max}^{\text{UP}} = R_{\text{crit}} = 2$  s<sup>-1</sup>. The time step used is the critical time step so that  $\Delta t = \Delta t_{\text{crit}} = \sqrt{2}$  s. A fixed constraint is applied at node 1 (at  $x = 0$ ) and a force of  $F = 0.001$  N is applied at the opposite end of the bar in the negative  $x$ -direction for the first two time steps only. The loading results in a displacement wave that propagates towards the fixed end of the bar, before being reflected back. The full analysis has a duration of  $T = 5657$  s.

Figure 3.4 shows two different analyses at two points in time. Specifically, Figure 3.4(a) shows the results when a sub-critical penalty ratio of  $R = 0.999R_{\text{crit}}$  is used for the penalty constraint. At time  $t = 1766$  s the displacement wave been reflected at the fixed end and is travelling back towards the free end of the bar. No instability is present at any point in the analysis.

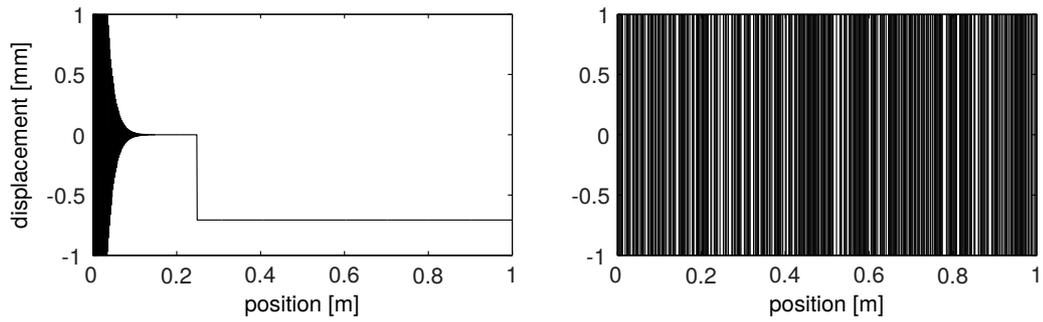
The analysis in Figure 3.4(b) uses a ratio of  $R = 1.001R_{\text{crit}}$ . Instability in the displacement solution is apparent shortly after the displacement wave reaches the constrained node. At time  $t = 5303$  s, near the end of the analysis, the instability completely dominates the solution.

In order to check that damping penalties have no effect on this well-defined bound, the same analysis is conducted with a damping penalty ratio  $D = 0.01$ . The results are shown in Figure 3.5, and show the same pattern as in the undamped case. Although the damping penalties do slightly reduce the amplitude of the oscillations at the onset of instability, they do not effect the stability of the analysis.

These results indicate that there is a sharp bound on the penalty ratio that can be safely used, as predicted by the proofs presented earlier in this chapter. In Chapters 5 and 6 we will apply these theories to larger-scale two-dimensional problems.

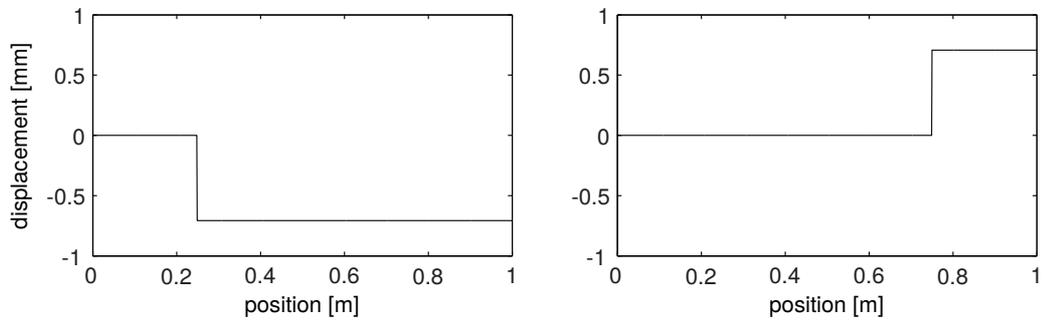


(a) Displacement profile at time  $t = 1766$  s (left) and time  $t = 5303$  s (right) for  $R = 0.999R_{\text{crit}}$

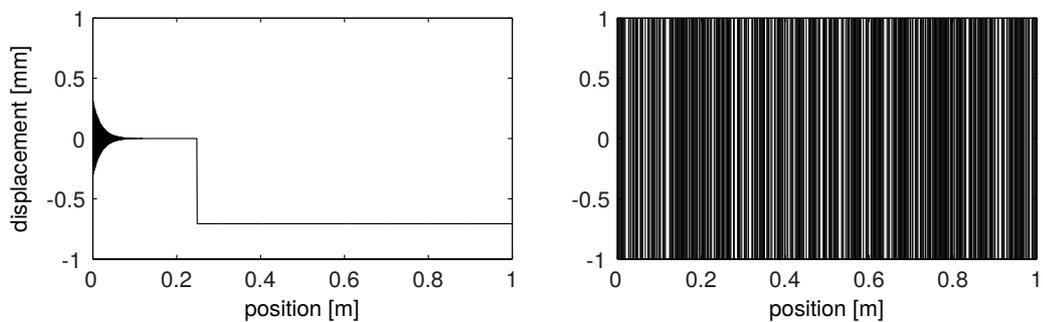


(b) Displacement profile at time  $t = 1766$  s (left) and time  $t = 5303$  s (right) for  $R = 1.001R_{\text{crit}}$

Figure 3.4: Stability testing for the bipenalty method with  $\alpha_s = 10^6$



(a) Displacement profile at time  $t = 1766$  s (left) and time  $t = 5303$  s (right) for  $R = 0.999R_{\text{crit}}$



(b) Displacement profile at time  $t = 1766$  s (left) and time  $t = 5303$  s (right) for  $R = 1.001R_{\text{crit}}$

Figure 3.5: Stability testing for the bipenalty method with damping (stiffness penalty parameter  $\alpha_s = 10^6$  and damping ratio  $D = 0.01$ )



## Chapter 4

# Accuracy

The finite element method provides an approximate solution to a governing differential equation, and as such errors are always present in the solution. Anticipating and quantifying the extent of these errors (i.e., the accuracy of the solution) is obviously vital if we wish to obtain useful solutions. Penalty methods in general introduce additional errors because constraints are not enforced exactly. Depending on the problem type, these errors can manifest in various ways, for example spurious, non-physical displacements at fixed or constrained degrees of freedom; for contact problems, non-physical penetration between nodes and/or surfaces; for cohesive surfaces, unrealistic strain introduced at the interfaces between elements.

This chapter is not concerned with the accuracy of the finite element method itself, but only with the accuracy of the penalty methods used for constraint imposition. Some of the main points of investigation in this chapter are: determining which penalty method leads to the smallest errors, how the bipenalty method affects the accuracy of constraint imposition compared to the stiffness and mass penalty methods, and the possible existence of an optimal penalty ratio which minimises errors for all problems. Simple dynamic FE problems will be used to demonstrate the results so that some general conclusions can be drawn, before larger scale practical problems are investigated in Chapters 5 and 6.

We begin by defining exactly what is meant by the term ‘error’. Instead of comparing the obtained FE solution with an exact analytical solution, we will instead be considering two separate FE solutions: one in which the constraints are enforced using a penalty method, and one in which the constraints have been enforced exactly

(e.g., by using direct imposition or the Lagrange multiplier method). Thus, we have

$$\mathbf{e} = \mathbf{u} - \mathbf{u}^{\text{FC}} \quad (4.1)$$

where  $\mathbf{u}$  is the displacement solution obtained using a penalty method,  $\mathbf{u}^{\text{FC}}$  is the solution of the fully constrained problem and  $\mathbf{e}$  is the resulting error vector. Often it will be more convenient to represent the error using an integral scalar quantity, which gives an indication of the overall error of the solution. A common measure is the  $L_2$  norm, which for nodal quantities is calculated by

$$\|\mathbf{e}\| = \sqrt{e_1^2 + e_2^2 + \cdots + e_n^2} \quad (4.2)$$

The root mean square of a time series variable may be used to give a measure of error over time in an ‘average’ sense. For a time series of  $N$  steps of a variable  $x$ , the root mean square is given by

$$x_{\text{rms}} = \sqrt{\frac{1}{N}(x_1^2 + x_2^2 + \cdots + x_N^2)} \quad (4.3)$$

There are three main ways in which penalty functions may introduce error into an analysis. First and foremost, there is the displacement error which is inherent to the method due to the use of finite penalty parameters. This leads to (hopefully) small non-physical displacements at constrained DOF, and in this chapter will be referred to as *constraint imposition error*. This kind of error will always be present to some extent when penalty functions are used to impose constraints. Secondly, as described in the previous chapter, there may be additional errors introduced by instability of the time integration routine due to the use of large penalty parameters combined with large time steps. Except in the unusual case of short temporary violations, these kinds of errors will rapidly dominate the solution due to the exponential growth that is characteristic of computational instability. Thirdly, there are computational errors caused by the use of finite-precision arithmetic, which may be amplified by ill-conditioning of the system matrices. These errors are always assumed to be present when solving linear systems of equations, but are in danger of becoming significant when penalty methods are employed because large penalty parameters can result in poor scaling of the system matrices.

The situation is complicated by the fact that any number of these sources may combine to give the final error total; in real-world problems it is not a trivial task to correctly identify the source of error, and there may be multiple sources at once. In this chapter we will take a methodical look at each in turn so as to gain insight into the causes, effects and methods of mitigation for each. Normally, the goal of the analyst is not to eliminate errors entirely, but to keep the total error within acceptable bounds.

When comparing the performance of different penalty methods it is often more instructive to use a dimensionless measure of the penalty magnitude which has a similar definition and meaning for all penalty types. Since it is the magnitude of the parameters *relative* to existing entries that dictates their performance, it is somewhat meaningless to merely compare the magnitudes of stiffness and mass penalty parameters, which in any case possess different units. We therefore introduce the dimensionless penalty factors  $p_s$  and  $p_m$ , which represent the magnitude of a stiffness or mass penalty parameter respectively, according to

$$p_s = \frac{\alpha_s}{K_{ii}} \quad (4.4)$$

$$p_m = \frac{\alpha_m}{M_{ii}} \quad (4.5)$$

where  $\alpha_s$  and  $\alpha_m$  are the penalty parameters (with units of N/m and kg respectively),  $\mathbf{K}$  and  $\mathbf{M}$  are the *unpenalised* stiffness and mass matrices introduced in Equation (1.1), and  $i$  is the number of the constrained DOF.

When a single constraint involves multiple DOF (i.e., for multipoint constraints), the definition becomes somewhat more complicated. In this case, the penalty parameter associated with a constraint is added to more than one matrix entry. For example, if a constraint links DOF  $i$  and  $j$ , adjustments must be made to entries  $K_{ii}$ ,  $K_{ij}$ ,  $K_{ji}$  and  $K_{jj}$  (for a stiffness-type penalty). In this case, we use

$$p_s = \frac{\alpha_s}{\max_{i \in P}(K_{ii})} \quad (4.6)$$

$$p_m = \frac{\alpha_m}{\max_{i \in P}(M_{ii})} \quad (4.7)$$

where  $P$  is the set of all DOF numbers associated with the constraint. This is a conser-

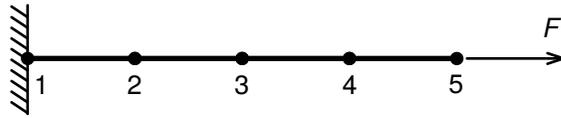


Figure 4.1: System A. A one-dimensional bar consisting of five two-noded linear finite elements, fixed at one end.

vative definition in that it ensures that performance is as expected, or greater than expected, at all DOF. Large variations between the stiffness or mass matrix entries associated with a single constraint only occur if the constrained elements have very different material properties, or there is a large variation in element size.

The dimensionless penalty ratio is defined as

$$r = \frac{p_s}{p_m} \quad (4.8)$$

which gives  $r = R(M_{ii}/K_{ii})$  for a single-point constraint at DOF  $i$ . Accordingly, the ratio  $r_{\text{crit}}$  is the dimensionless ratio corresponding to the dimensioned ratio  $R_{\text{crit}}$  for a given constraint.

## 4.1 Constraint imposition error

The use of finite penalty parameters in the formulation of displacement constraints leads to finite errors in the resulting displacement field. That is, for penalty methods in general, the vector  $\mathbf{h}$  in Equation (1.3) will be non-zero. To investigate this error source, we will initially look at the FE system shown in Figure 4.1, the material properties for which can be found in Table 4.1. A dynamic analysis is undertaken by setting  $u_1 = 0$  by some method in order to fix the structure at node 1, applying a constant force  $F$  at node 5, and then solving for displacements over time using the central difference method. Constraint imposition error in this case is the displacement at node 1, since it should ideally be zero at every time step. For an early investigation into this kind of error with regards to static analysis, see Felippa [29, 30].

### 4.1.1 Comparison of penalty methods

Figure 4.2 shows the displacement error norm for a five element 1D bar under a constant load, fixed at one end using a penalty method. Results from the stiffness penalty

Property	Value
Young's modulus, $E$	100 Pa
Density, $\rho$	1 kg/m <sup>3</sup>
Cross-sectional area, $A$	1 m <sup>2</sup>
Element length, $h$	1 m
Force, $F$	1 N
Critical time step, $\Delta t_{\text{crit}}$	0.1 s

Table 4.1: Material properties and other data for System A

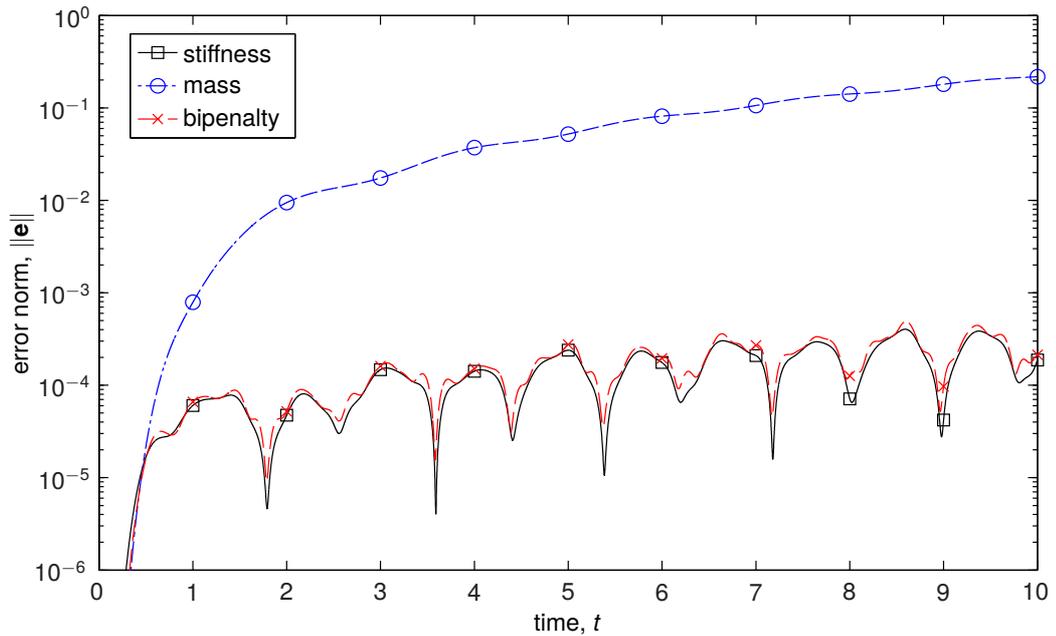


Figure 4.2: Error norm for a simple 1D bar under constant load

method ( $p_s = 1000$ ,  $p_m = 0$ ), mass penalty method ( $p_s = 0$ ,  $p_m = 1000$ ) and bipenalty method ( $p_s = 1000$ ,  $p_m = 1000/r_{\text{crit}}$ ) are shown. In order to avoid possible time step instabilities when using stiffness penalties only, a time step of  $\Delta t = 0.01\Delta t_{\text{crit}}$  is used. The use of the mass penalty method in isolation leads to errors approximately two orders of magnitude larger than the stiffness and bipenalty methods for the majority of the analysis.

Figure 4.3 shows the error only at the constrained node (where displacement should ideally be zero for the duration of the analysis). While the error norm  $\|\mathbf{e}\|$  gives a useful account of how that constraint violation error impacts the predicted behaviour across the whole structure, this plot provides greater insight into exactly how the artificial stiffness and inertia interact at the constraint. In this case there are some oscillations in the error at the constrained node when the bipenalty method is

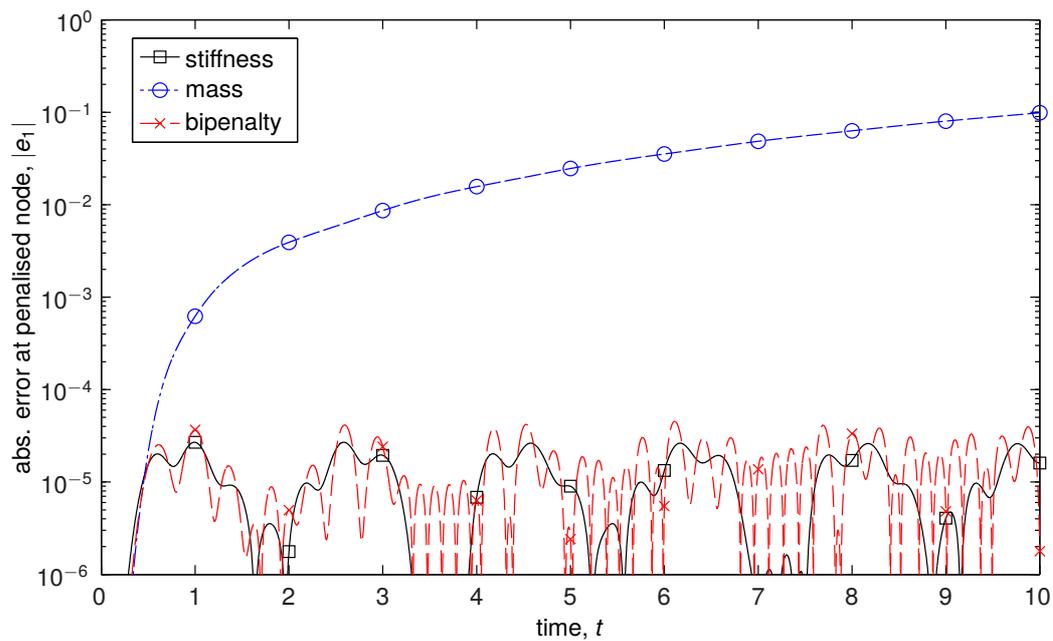


Figure 4.3: Absolute error of the constrained node only for a simple 1D bar under constant load.

used, although Figure 4.2 suggests that this does not have a significant effect on the behaviour of the structure as a whole.

Figure 4.4 shows time series plots of displacement and acceleration at the constrained node for the three methods. Note that for an exactly enforced constraint, both displacement and acceleration are zero throughout the analysis. Stiffness penalties act to correct non-zero displacements, which in turn result in low acceleration at the constrained node. For a pure mass penalty approach, acceleration of the constrained node over time is very similar to the displacement profile of the stiffness penalty method, which leads then to enforcement of the displacement constraint (though it is clearly nowhere near as effective at equal penalty factors). For the bipenalty method, both mechanisms are acting together, with the stiffness penalty functions enforcing zero displacement and mass penalties acting on non-zero accelerations. Unfortunately, these two processes do not always work in harmony to reduce the overall constraint enforcement error, as can be seen in Figure 4.4(c). Typically, the bipenalty method leads to displacement constraint enforcement errors somewhere between the two ‘extremes’ of the stiffness and mass penalty methods. Within these bounds, the error is determined largely by  $R$ : the ratio between stiffness and mass penalty parameters.

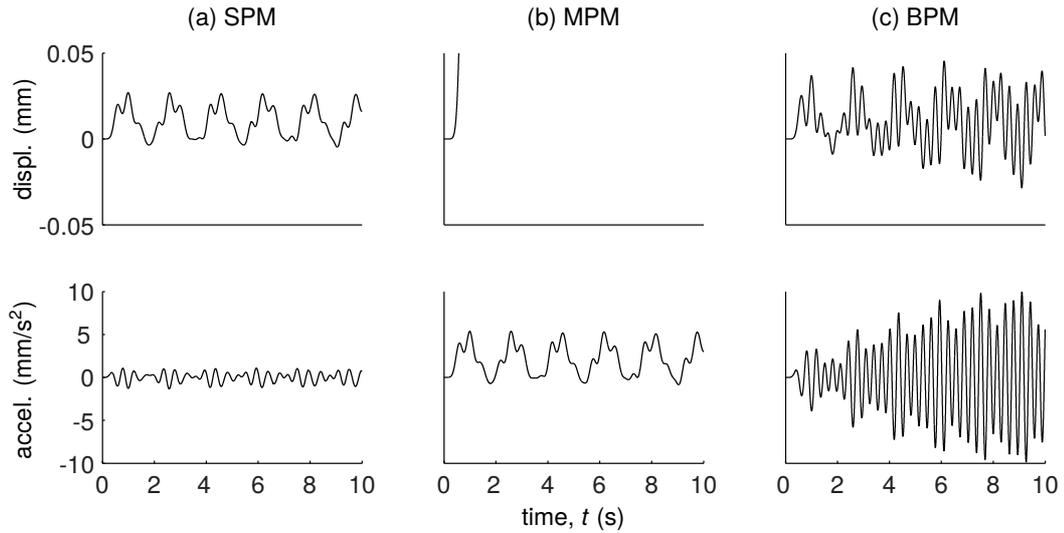


Figure 4.4: Displacement (top) and acceleration (bottom) of the constrained node over time for the stiffness, mass and bipenalty methods.

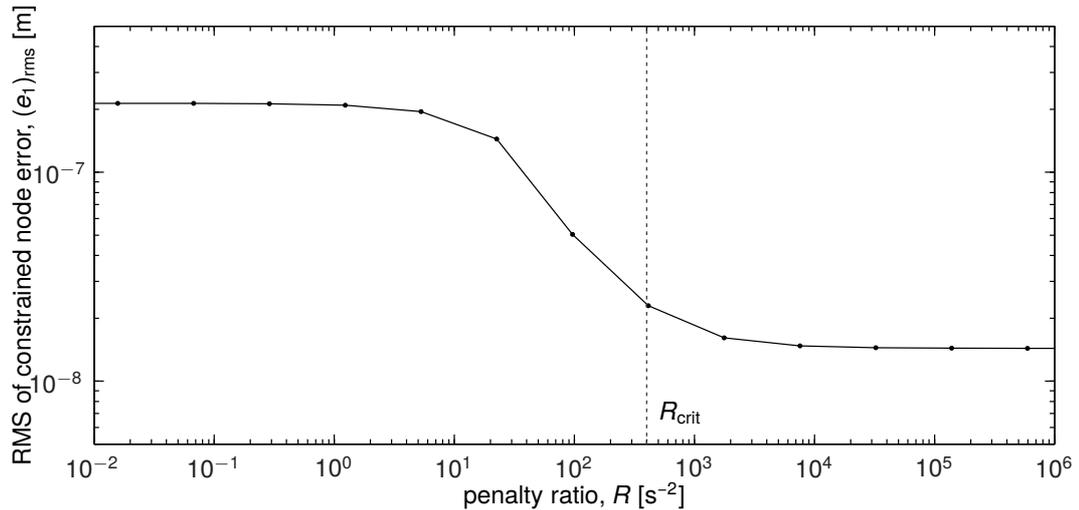


Figure 4.5: RMS of displacement error over time ( $t = 0 \dots 1$  s) at the constrained node for various penalty ratios and constant  $p_T = p_s + p_m = 10^6$ .

### 4.1.2 Effect of penalty ratio

In Chapter 3 we saw how the penalty ratio can affect the eigenfrequencies (and therefore the time step stability) of an FE system. Figure 4.5 shows how the accuracy of constraint enforcement changes for various penalty ratios, while the total penalty factor  $p_T = p_s + p_m$  is kept constant. For low ratios, the solution tends to a pure mass penalty approach ( $p_s = 0$ ,  $p_m = 10^6$ ), while for high ratios the solution displays the higher accuracy of the stiffness penalty method ( $p_s = 10^6$ ,  $p_m = 0$ ); a simple monotonic function connects the two regions. For best accuracy, it is apparently advisable to use the highest penalty ratio that ensures time step stability of the analysis (that

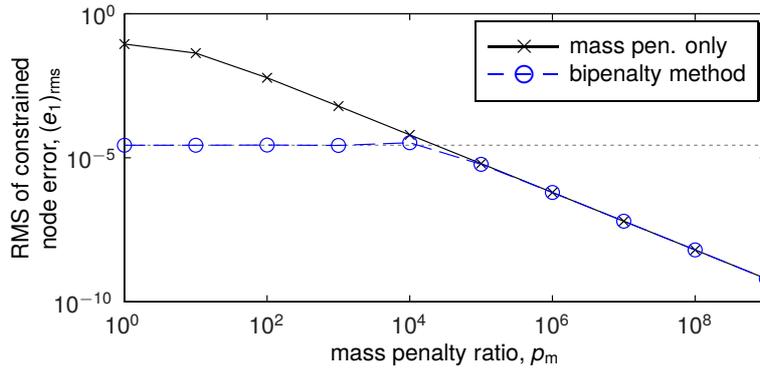


Figure 4.6: Constrained node error as mass penalties are introduced, with constant stiffness penalty  $\rho_s = 10^3$ . The dotted line indicates the error obtained with constant stiffness penalty only.

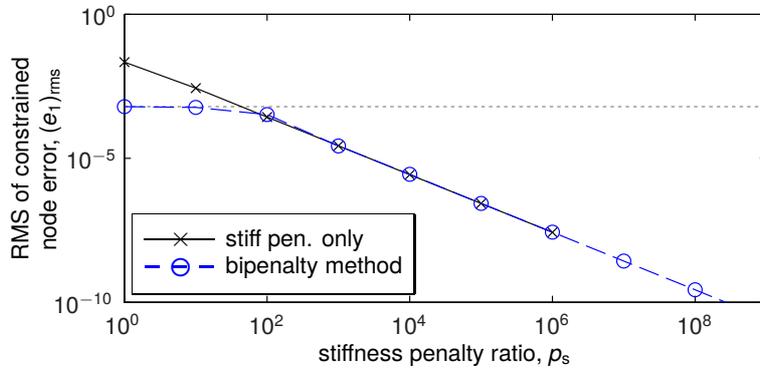


Figure 4.7: Reducing error as stiffness penalties are introduced, with constant mass penalty  $\rho_m = 10^3$ . The dotted line indicates the error obtained with constant mass penalty only.

is, a ratio close to or equal to the critical penalty ratio  $R_{\text{crit}}$ , whose value is also shown in Figure 4.5). However, this analysis assumes that there is a limit on the *total* penalty ratio  $p_T$  that can be used, which is not necessarily the case (as we will examine further in Section 4.2).

Figure 4.6 shows another error analysis with the stiffness penalty factor kept constant at  $p_s = 10^3$ . For low mass penalty values (therefore large penalty ratios) the stiffness penalty dominates and the bipenalty analysis produces errors comparable with a pure stiffness approach (indicated by the dotted line). The addition of mass penalties leads to an *increase* in accuracy once  $p_m$  is greater than approximately  $10^4$ , despite the low penalty ratios in this region. In Figure 4.7, the mass penalty factor is kept constant while the stiffness penalty factor is increased. A similar trend is observed, whereby errors introduced by the bipenalty method closely follow those produced by the dominant stiffness- or inertia-type penalty method.

These figures show that for the bipenalty method (as with traditional penalty

methods) increasing the magnitude of the parameters will always tend to reduce constraint imposition errors as long as an analysis remains stable; to obtain the most accurate solutions, we must make the penalty parameters as large as possible. However, it is stiffness penalties which tend to have the biggest impact on accuracy. Where a choice is available, increasing the stiffness penalty parameters will generally have a larger effect on the constraint imposition accuracy of the solution.

## 4.2 Matrix ill-conditioning

Round-off error is the difference between a mathematically correct value and its approximation, as stored in the computer. Computers store values accurately enough that these errors are usually not significant when dealing with FE systems unless a system is especially sensitive to these kinds of errors. Then, the system is said to be *ill-conditioned*.

Consider the matrix equation  $\mathbf{Ax} = \mathbf{b}$ . If the right-hand side is changed to  $\mathbf{b} + \Delta\mathbf{b}$  (e.g., by round-off error) then the solution becomes  $\mathbf{x} + \Delta\mathbf{x}$ . If a small  $\Delta\mathbf{b}$  leads to a large  $\Delta\mathbf{x}$  then the matrix  $\mathbf{A}$  is ill-conditioned. The extent of this ill-conditioning is measured by the condition number, given by [76]

$$c = \|\mathbf{A}\| \|\mathbf{A}^{-1}\| \quad (4.9)$$

where the matrix norm is defined as

$$\|\mathbf{A}\| = \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|} \quad (4.10)$$

This condition number puts a bound on the solution error that is introduced by a problem error in  $\mathbf{b}$ , expressed as

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq c \frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|} \quad (4.11)$$

or due to a problem error in  $\mathbf{A}$  by

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x} + \Delta\mathbf{x}\|} \leq c \frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|} \quad (4.12)$$

Analysts should be careful to keep the condition number of system matrices within

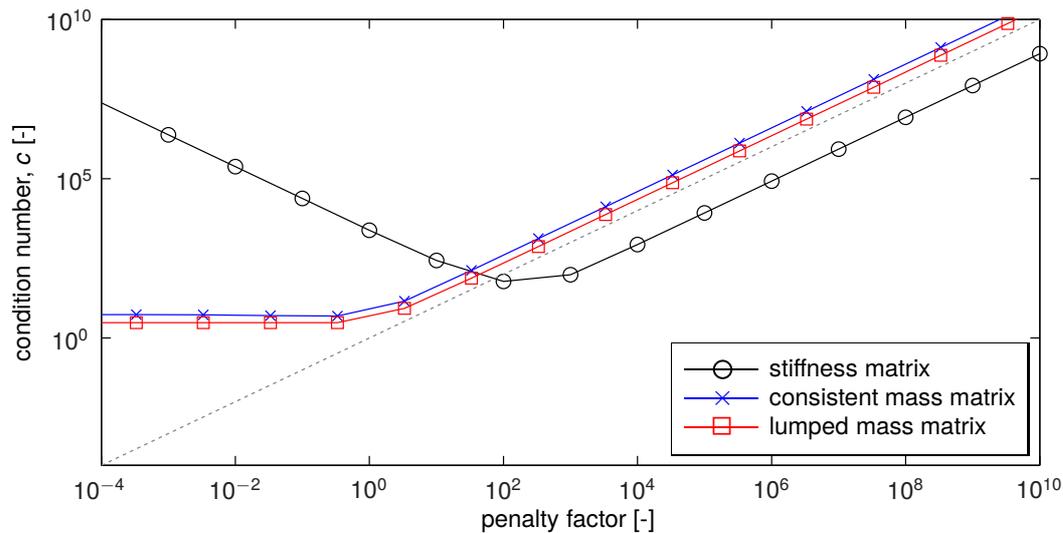


Figure 4.8: Condition number of the stiffness and mass matrices for increasing penalty.

acceptable limits, otherwise round-off errors may lead to unacceptably large errors in the solution vector.

Matrices which are close to singular have large condition numbers. This can be caused by having very large and very small numbers present in a matrix simultaneously, and as such penalty methods tend to worsen the conditioning of the FE system matrices. However, there are no obvious bounds for how large penalty parameters can be, or how high a matrix condition number must get before results can no longer be trusted. A commonly used rule of thumb is that an analyst can expect to lose  $\log c$  decimal places to round-off errors, although the extent to which these errors manifest is dependent on many factors, including the direction of the right-hand side vector and the details of the solution algorithm.

Figure 4.8 shows how stiffness and mass penalties affect the conditioning of their respective system matrices. The system shown in Figure 4.1 is again used as an example. Since the stiffness matrix is singular when unconstrained, the condition number is large for small penalties. On the other hand, both the lumped and consistent mass matrices are positive-definite and therefore have a finite condition number when unpenalised. As larger penalties are introduced the condition number tends to increase proportionally to the penalty factor being added, for both the stiffness and mass matrices.

Nour-Omid and Wriggers [58] have provided an equation that can be used to estimate a suitable parameter for static contact problems that takes into account the

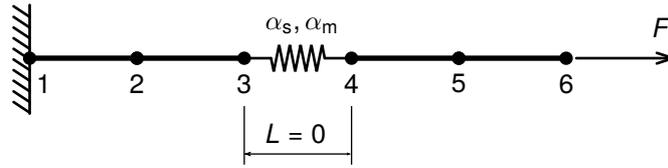


Figure 4.9: System B: 1D bar with zero-length bipenalty tying between nodes 3 and 4.

possibility of matrix ill-conditioning. The estimate for the stiffness penalty is given by

$$\alpha_s = \frac{k}{\sqrt{n\epsilon}} \quad (4.13)$$

where  $k$  is the smallest stiffness of the affected elements,  $n$  is the total number of DOF and  $\epsilon$  represents the machine precision (the smallest number that satisfies  $1 + \epsilon > 1$  in a computation). As with the use of dimensionless penalty factors, this value takes into account the stiffness of the structure, as well as the overall size of the system matrix, which also has an effect on condition number.

Felippa [29] has also provided recommendations for selecting penalty parameters, stating that for single-point constraints the magnitude of the penalty parameter has “no appreciable effect on the computational error due to roundoff”, no matter how large. However, it is also stressed that this theory does *not* apply for constraint equations concerning multiple DOF. Figure 4.9 shows a system that utilises such a constraint and Figure 4.10 shows an error analysis for a series of time domain numerical experiments with the system. The material properties from Table 4.1 are again used, with a time step of  $\Delta t = 0.001$  s and a total analysis duration of  $T = 5$  s. The plot shows that the constraint imposition error (i.e., the displacement gap between nodes 3 and 4) is approximately inversely proportional to the mass penalty factor for the analysis until factors of around  $10^{12}$  are reached, at which point the ill-conditioning of the mass matrix reaches a point where computational errors become extremely pronounced. Since these errors may be introduced repeatedly at each time step, they accumulate as an analysis is extended, and are therefore significantly higher than they would be for a simple static analysis.

By adapting Equation 4.13 we can generate an estimate for a suitable mass penalty factor for this analysis. Recalling (4.7) and noting that  $n = 6$  and  $\epsilon \approx 2.22 \times 10^{-16}$

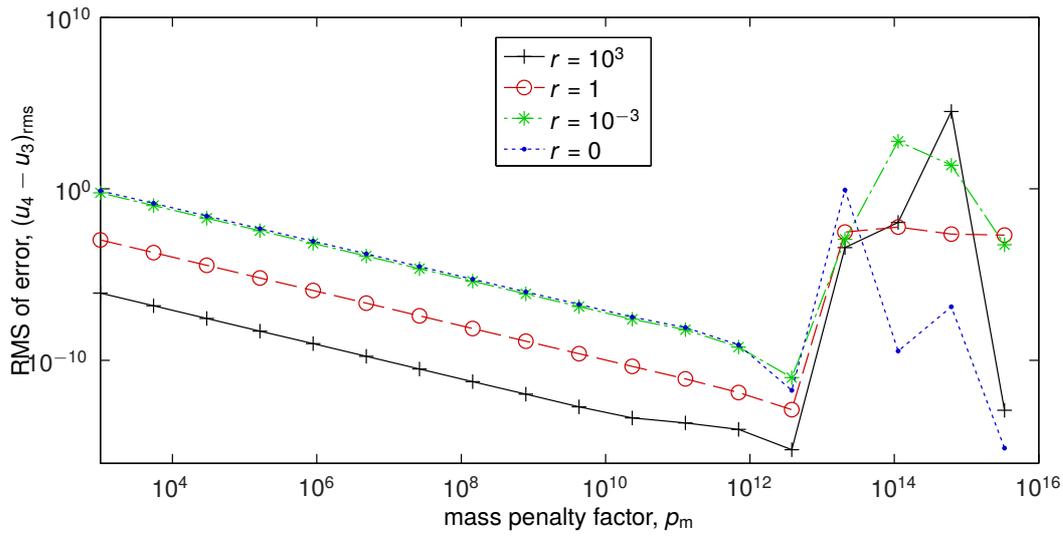


Figure 4.10: Constraint imposition error at a node-to-node tying for increasing mass penalty and a range of penalty ratios.

(for the computer running the analysis), we have

$$p_m = \frac{1}{\sqrt{n\epsilon}} \approx 2.74 \times 10^7 \quad (4.14)$$

which is well within the region unaffected by serious computational errors.

Note that Figure 4.10 shows four different plots with a variety of penalty ratios. When also incorporating large stiffness penalties into the analysis (e.g., for  $r = 10^3$ ) the overall errors are somewhat lower, as we might expect; however, the accumulated computational errors do not show any signs of increasing, or of manifesting at lower penalty magnitudes; in other words, the stiffness penalty parameters appear to have no effect on computational error. In fact, the nature of the central difference method means that mass penalties are the governing factor in the introduction of conditioning-related errors, since the stiffness matrix does not feature in the linear system that must be solved; since no stiffness matrix inversion is required, its conditioning has little effect on the results.

### 4.3 Selection of penalty parameters

As is often cited as a major disadvantage of the penalty method, “the selection of penalty parameters is, in some sense, arbitrary” [7, p. 221]. Ordinarily, there is no mathematically optimal choice for a penalty parameter, and no clear bounds on the

---

**Algorithm 1** Automatic calculation of penalty parameters (maximising  $p_T$ , taking into account conditioning of stiffness matrix).

---

- 1: Determine safe sub-critical time step,  $\Delta t < \Delta t_{\text{crit}}$
  - 2: Compute safe penalty ratio,  $R = F_R \frac{4}{\Delta t^2}$
  - 3: Estimate suitable mass penalty factor,  $p_m = \frac{1}{\sqrt{n\epsilon}}$
  - 4: **for**  $j = 1 \dots n_c$  **do**
  - 5:     Calculate mass penalty parameter  $(\alpha_m)_j$  using Equation (4.5) or (4.7).
  - 6:     Calculate stiffness penalty parameter,  $(\alpha_s)_j = R(\alpha_m)_j$
- 

largest or smallest values that can be used. However, we have now outlined the most important considerations for penalty methods in explicit dynamics, and are in a position to give some useful guidance on penalty parameter selection. As pointed out by Hallquist in relation to contact-impact problems, “pre-empting user control over [the penalty] parameter greatly increases the success of the method” [36, p. 26.2]. Understandably, users do not like to make choices about arbitrary parameters and therefore it is best to make the selection process automatic wherever possible. For explicit dynamic analyses we have established three important considerations that inform the selection process:

1. time step instability caused by the addition of penalties must be avoided,
2. any decrease in critical time step should be avoided or minimised, and
3. computational and constraint imposition errors should be within acceptable bounds.

Broadly, these three points fall under the categories of stability, computational expense, and accuracy, respectively. Algorithm 1 outlines a process by which we may arrive at suitable stiffness and mass penalty parameters for a general bipenalised problem. These steps take into account each of the three considerations outlined above.

Firstly, a time step for the problem is chosen. Crucially, *this choice is not affected by the penalty constraints*; traditional methods of time step estimation (as described in Section 3.3.2) may be used to arrive at a safe sub-critical time step  $\Delta t$ . If higher time resolution is required for analysis, a smaller timestep may be used without issue.

In step 2, a penalty ratio is calculated from the chosen time step (see Section 3.3.2). Recall that using a penalty ratio of  $R$  will introduce an eigenvalue with value  $R$  into the system for large penalties. If this eigenvalue is equal to the maximum allowed by the chosen time step, round-off errors may lead to instability. For this reason, a safety factor  $F_R$  is included in the calculation. Any value  $F_R < 1$  will ensure stability in all cases. Although using  $F_R \geq 1$  will not necessarily lead to instability (since the maximum eigenvalue actually tends to  $R$  from below as  $\alpha_m$  tends to infinity, as shown in Figure 3.3), values greater than 1 are not safe with respect to stability. Constraint imposition accuracy will be reduced as  $F_R$  is reduced, since this will result in a smaller penalty ratio  $R$ . A safety factor of  $F_R = 0.99$  is recommended, since it ensures stability in all cases, while having a negligible effect on the accuracy of the solution.

Since we are using the bipenalty method with a carefully chosen penalty ratio we can be sure that time step stability will be unaffected. We therefore wish to choose our penalty parameters as high as possible without causing computational errors due to ill-conditioning. However, if the central difference method is employed as a solution scheme no stiffness matrix inversion is required and we therefore assume that the conditioning of the stiffness matrix is not an issue. We therefore calculate the mass penalty parameter according to Equation (4.13), first calculating a suitable factor and then converting  $p_m$  to  $\alpha_m$  according to the existing mass matrix entries. Finally, the stiffness penalty parameter is computed from the previously calculated ratio.

The above method maximises the penalty ratio under the assumption that a certain level of stiffness matrix ill-conditioning is acceptable. However, if stiffness matrix conditioning is important to the analyst (for example, if the assembled stiffness matrix is also to be used for a static analysis), it may be desirable to maximise accuracy while ensuring good conditioning of both stiffness and mass matrices. In this case, we are less concerned with maximising the penalty ratio, and more concerned with maximising the total penalty factor  $p_T = p_s + p_m$ . Algorithm 2 describes a process by which to achieve this goal.

Since we wish to use the same penalty ratio for all constraints (as this was assumed in Chapter 3), a ratio must be chosen by first considering all constraints in turn, and finding the ‘ideal’ ratio for that constraint. If those ratios are all greater

---

**Algorithm 2** Automatic calculation of penalty parameters (maximising  $p_T$ , taking into account conditioning of stiffness matrix).

---

- 1: Choose sub-critical  $\Delta t < \Delta t_{\text{crit}}$
  - 2: Calculate  $R_{\Delta t} = \frac{4}{\Delta t^2}$
  - 3: Determine ideal ratios,  $R_j^{\text{ideal}} = \frac{\max_{i \in P_j}(K_{ii})}{\max_{i \in P_j}(M_{ii})}$
  - 4: Find optimal stable ratio,  $R = \min[F_R R_{\Delta t}, \max(R_j^{\text{ideal}})]$
  - 5: **for**  $j = 1 \dots n_c$  **do**
  - 6:     **if**  $R_j^{\text{ideal}} \geq R_{\Delta t}$  **then**
  - 7:          $(\alpha_m)_j = \frac{\max_{i \in P_j}(M_{ii})}{\sqrt{n\epsilon}}$
  - 8:          $(\alpha_s)_j = R(\alpha_m)_j$
  - 9:     **else**
  - 10:          $(\alpha_s)_j = \frac{\max_{i \in P_j}(K_{ii})}{\sqrt{n\epsilon}}$
  - 11:          $(\alpha_m)_j = (\alpha_s)_j / R$
- 

that the ratio required for stability,  $F_R R_{\Delta t}$ , then the stable ratio is chosen. If not, the ratio is chosen based on best performance. The parameters are then chosen by maximising either  $\alpha_s$  or  $\alpha_m$ , depending on the ratio that was chosen.

## 4.4 Summary

The small-scale tests presented in this chapter conclude the more theoretical portion of the thesis. Together with Chapter 3, they provide a basis from which we can make informed decisions about how to implement the bipenalty method for practical applications and large-scale analyses in the coming chapters. Specifically, there are a few rules of thumb which we can make use of when accuracy is a primary concern.

1. For equivalent penalty factors, a pure stiffness penalty approach leads to smaller errors than a pure mass approach.
2. Following from the above, for a given total penalty factor  $p_T = p_s + p_m$ , a larger penalty ratio  $r = p_s/p_m$  leads to smaller errors.
3. Adding mass penalties to a constant stiffness penalty, or visa versa, tends to increase the constraint imposition accuracy of the analysis.

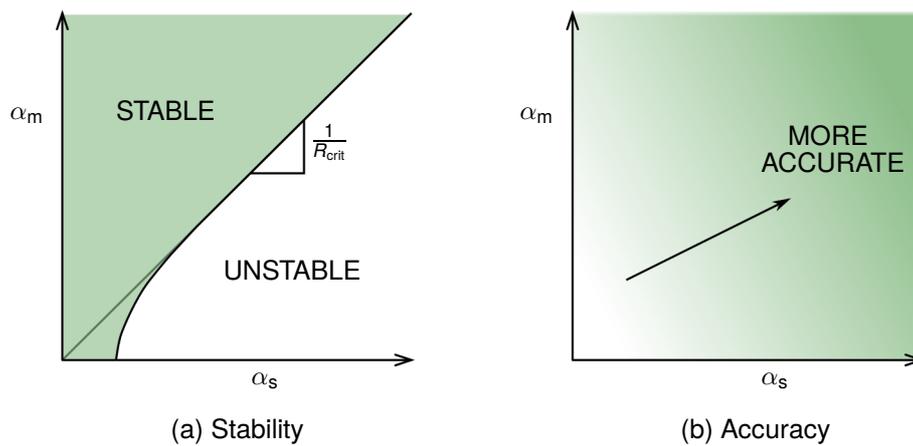


Figure 4.11: Approximate representation of stability (left) and accuracy (right) for a range of stiffness and mass penalties used to apply a single-point displacement constraint.

Taking this into account, along with time step stability considerations, the potential advantages of the bipenalty method can be clearly stated. Figure 4.11 shows in general terms how stiffness and mass penalty parameters can be expected to affect both the stability and accuracy of a typical analysis. The primary goal of the bipenalty method is to ensure time step stability. Examining only the  $x$ -axis of Figure 4.11(a) shows that this is difficult using pure stiffness penalties, since there is a (somewhat poorly defined) point beyond which the critical time step of the analysis is increased beyond the time step that would ideally be used. Furthermore, this point is generally in a region of relatively low accuracy, since the stiffness penalty in question may be quite small. The addition of mass penalties allows us to ensure stability, but also *greatly increase the magnitude of the stiffness penalty parameter* that can be safely used. In this way, the bipenalty method may be used to greatly increase accuracy without decreasing the time step of the analysis.

## Chapter 5

# Bipenalty interface elements based on the theory of cohesive surfaces

In solid mechanics, crack propagation problems are of significant importance to engineers in a wide range of fields, including fracture and damage mechanics, blast and impact engineering, design of mechanical processes, structural stability and others. As a result, a great deal of research effort has been concentrated on the numerical analysis of such problems, and many different modelling strategies have been utilised in order to tackle them. Among the more novel techniques are meshless methods [11, 13] and the boundary element method [16, 20, 64]. Both of these methods are advantageous in that they do not restrict the crack path to follow inter-element boundaries, as is the case with common finite element techniques, meaning that adaptive remeshing is generally not required. However, the finite element (FE) method is well-established and extremely versatile, and remains the mostly widely used. Ongoing research efforts continue to increase the computational efficiency and robustness of FE methods for the treatment of fracture.

In FE analysis the three most common techniques for the modelling of fracture and crack propagation in a dynamic setting are the element deletion method, the extended finite element method (XFEM), and inter-element crack methods [74]. Each of these methods build upon standard FE formulations to include the effects of damage and crack propagation in some way. Element deletion is the simplest of the methods and the most widely used in commercial codes (e.g., ANSYS [50] and LS-DYNA [36]). It requires only an alteration of the constitutive relation of a failing ele-

ment so that the stress in the element is reduced to zero for large strain, effectively removing certain elements during an analysis. The method can also be modified to allow elements to carry compressive but not tensile stresses. It is probably due to the theoretical and computational simplicity that the method has achieved widespread adoption, since its reliability with regards to the prediction of crack paths has been called into question [74]. As with many FE-based approaches, crack paths and the details of crack growth are often highly mesh-dependent [38].

XFEM was first introduced by Belytschko and co-workers in 1999 to tackle crack propagation problems in elastostatics [10, 56]. It uses shape function enrichment in order to introduce discontinuities within finite elements, which overcomes the high mesh dependence of previously existing techniques. This makes it an attractive option for accurately and efficiently predicting crack paths which are not known a priori [21], but has yet to achieve widespread adoption in commercial software.

Inter-element crack methods are a well-established group of techniques which explicitly model cracks on the boundaries of individual finite elements. This can be achieved either by adaptive remeshing or by the addition of interface elements at element boundaries possessing a specially designed traction-displacement relationship, an approach also referred to as the cohesive zone model.

The theory of cohesive zones was first introduced in the 1960s [8, 26] but was not applied to dynamic crack propagation until the 1990s, with publications from Xu and Needleman [80], Camacho and Ortiz [18] and Repetto et al. [66] forming the basis for the formulation described in this chapter. Each of these formulations introduces interface elements, also known as cohesive surfaces, into the FE continuum. A non-linear traction-displacement relationship is then chosen that approximately represents the fracture characteristics of the material. Cracks are thus free to coalesce and propagate as a natural outcome of the simulation.

While this work caused somewhat of a resurgence in the investigation of cohesive zone modelling, the fact that the cohesive surfaces often rely on large stiffness penalties being placed throughout the FE continuum causes some challenges. The elements must initially have very high stiffness in order to avoid artificial compliance (which leads to unrealistic deformations and a retardation of wave speed). With explicit solvers this leads to a reduction of the critical time step, which can be problematic. Camacho and Ortiz [18] avoid the problem by introducing cohesive surfaces

only at the onset of damage, but this requires alterations to the FE discretisation, and thus to the computer memory requirements, as cracks propagate. Ortiz and Pandolfi [60] also select a cohesive law without an initial elastic region based on the fact that this would place “stringent restrictions” on the stable time step. Espinosa and Zavattieri [28] use a large initial stiffness, but urge caution in the selection of the penalty parameter, noting that as the parameter grows “the wave speed in the material asymptotically approaches the speed of the material without interfaces” and hence should be “large enough to be effective but not so large as to provoke numerical instabilities”. It is also acknowledged by the same authors that the penalty will be large enough to have an impact on the critical time step, and as a result, the time step calculation includes an additional limitation in that it must taken into account the cohesive surfaces as well as continuum elements. Because of this, the interface elements may cause sudden changes in time step, which may constitute another source of instability unless contact conditions are computed using a different  $\Delta t$ , leading to the use of an awkward subcycling time integration routine. These additional complications are tolerated, though, since the initial model developed by Xu and Needleman [80] (upon which it was largely based), “induces artificial compliance due to the elasticity of the intrinsic cohesive law”, according to Song et al. [75].

In this chapter we use interface elements formulated using the bipenalty method to investigate crack propagation by the theory of cohesive zones, with the goal of reducing compliance (through the use of high initial stiffness in the cohesive law) whilst also ensuring time step stability. Introducing mass penalties into the formulation means that the interface elements have no effect on the critical time step, and thus eliminates such considerations with regards to the selection of a suitable traction-displacement relationship.

## 5.1 Element formulation

The interface element formulation for this new penalty-type formulation is based on the work of Schellekens [67,68]. In this work, a 4-noded interface element is derived which has an initial volume of zero. The stiffness of the element is controlled by user-defined parameters that describe the constitutive behaviour of the element. For simplicity, we will be working in 2D and using the “linear line interface”—an element

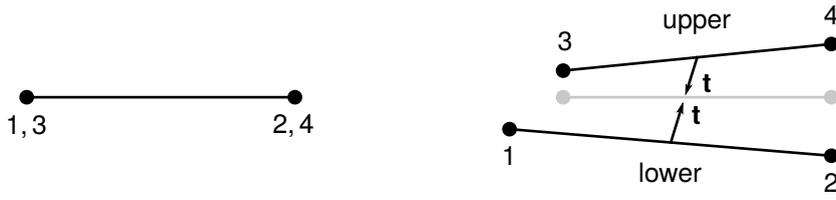


Figure 5.1: Line interface element in an initial (left) and deformed (right) configuration

which is described by normal and tangential tractions and with linear shape functions (see Figure 5.1).

### 5.1.1 Elastic stiffness matrix

We now consider this 4-noded line interface. Each node has two displacement DOF, giving an element nodal displacement vector

$$\mathbf{d} = [d_n^1, d_n^2, d_n^3, d_n^4, d_t^1, d_t^2, d_t^3, d_t^4]^T \quad (5.1)$$

where  $n$  and  $t$  denote the directions normal and tangential to the interface, respectively, and superscripts indicate the node numbers as shown in Figure 5.1. The continuous displacement field is then

$$\mathbf{u} = [u_n^u, u_n^l, u_t^u, u_t^l]^T \quad (5.2)$$

where  $u$  and  $l$  denote the upper and lower sides of the interface, respectively. The relationship between nodal and continuous displacement vectors is given by

$$\mathbf{u} = \mathbf{H}\mathbf{d} \quad (5.3)$$

where  $\mathbf{H}$  is an  $4 \times 8$  matrix containing the interpolation polynomials  $\mathbf{n} = [N_1, N_2]$  and is of the form

$$\mathbf{H} = \begin{bmatrix} \mathbf{n} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{n} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{n} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{n} \end{bmatrix} \quad (5.4)$$

where  $\mathbf{0} = [0, 0]$ . We can find the relative displacements  $\boldsymbol{\delta} = [\delta_n, \delta_t]^T$  from

$$\boldsymbol{\delta} = \mathbf{L}\mathbf{u} \quad (5.5)$$

where the operator matrix  $\mathbf{L}$  is defined through

$$\mathbf{L} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \quad (5.6)$$

The relationship between nodal displacements and relative displacements can then be derived from Equations (5.3) and (5.5) as

$$\boldsymbol{\delta} = \mathbf{LH}\mathbf{d} \quad (5.7)$$

at which point it is useful to define a matrix  $\mathbf{B}$  which directly relates relative displacements to nodal displacements

$$\mathbf{B} := \mathbf{LH} = \begin{bmatrix} -\mathbf{n} & \mathbf{n} & 0 & 0 \\ 0 & 0 & -\mathbf{n} & \mathbf{n} \end{bmatrix} \quad (5.8)$$

For arbitrarily orientated elements, the matrix  $\mathbf{B}$  should be transformed to the local tangential co-ordinate system of the node set.

We now introduce a matrix  $\mathbf{D}$  describing the constitutive traction-displacement relation, so that

$$\mathbf{t} = \mathbf{D}_s \boldsymbol{\delta} \quad (5.9)$$

where  $\mathbf{t} = [t_n, t_t]^T$  is the traction vector for the element (units  $\text{N}/\text{m}^2$ ) and  $\mathbf{D}_s$  is a constitutive matrix of the form

$$\mathbf{D}_s = \begin{bmatrix} d_n & 0 \\ 0 & d_t \end{bmatrix} \quad (5.10)$$

The values  $d_n$  and  $d_t$  (units  $\text{N}/\text{m}^3$ ) represent the ‘stiffness’ of the interface in the normal and tangential directions, although since the term ‘stiffness’ usually refers to a relationship between force and displacement a more accurate description is stiffness per unit area. It is these values that function as the stiffness penalty parameters

for the interface, with  $d_n = d_t = 0$  corresponding to an unenforced constraint, and larger values leading to more accurate constraint imposition (i.e., smaller relative displacements). In the present work we assume that both parameters are equal so that  $d_n = d_t = \alpha_s$  and  $\mathbf{D}_s = \alpha_s \mathbf{I}$ . We postpone until the next section a discussion of how these constitutive relations may change over time (due to damage).

The stiffness matrix  $\mathbf{K}$  can now be obtained by minimisation of the total potential energy. The internal work done in the element is

$$\mathcal{U} = \frac{1}{2} \int_S \boldsymbol{\delta}^T \mathbf{t} \, dS \quad (5.11)$$

where the integration is performed over the surface of the element,  $S$ . This can be rewritten using Equations (5.7)–(5.9) as

$$\mathcal{U} = \frac{1}{2} \mathbf{d}^T \int_S \mathbf{B}^T \mathbf{D}_s \mathbf{B} \, dS \mathbf{d} \quad (5.12)$$

while the external work  $\mathcal{W}$  is given by

$$\mathcal{W} = -\mathbf{d}^T \mathbf{f} \quad (5.13)$$

After setting the variation of the total potential energy ( $\mathcal{U} + \mathcal{W}$ ) to zero we find

$$\mathbf{K} \mathbf{d} = \mathbf{f} \quad (5.14)$$

where the stiffness matrix is given by

$$\mathbf{K} = \int_S \mathbf{B}^T \mathbf{D}_s \mathbf{B} \, dS \quad (5.15)$$

Considering the numerical integration of such elements, we note that the linear shape functions can be written in one isoparametric co-ordinate  $\xi$  as

$$N_1 = \frac{1}{2}(1 - \xi) \quad (5.16)$$

$$N_2 = \frac{1}{2}(1 + \xi) \quad (5.17)$$

which means that we need integrate over only one co-ordinate. The stiffness matrix

can therefore be computed (using 2-point Gaussian integration) via

$$\mathbf{K} = b \int_{-1}^1 \mathbf{B}^T \mathbf{D}_s \mathbf{B} \sqrt{\left(\frac{\partial x}{\partial \xi}\right)^2 + \left(\frac{\partial y}{\partial \xi}\right)^2} d\xi \quad (5.18)$$

where  $b$  is the width of the interface in the out-of-plane direction.

### 5.1.2 Constitutive relations and damage law

The constitutive law for the cohesive surfaces relates traction in the interface to the displacement jump across the surface. As summarised by Xu and Needleman, “the behaviour that needs to be captured is that, as the cohesive surface separates, the magnitude of the traction at first increases, reaches a maximum and then approaches zero with increasing separation” [80, p. 1400]. However, this kind of cohesive law is problematic when used in explicit dynamics, since “the initial elastic slope ... may place stringent restrictions on the stable time step for explicit integration” [60]. In other words, the initial penalty stiffness in the interface causes a significant decrease in the critical time step of the analysis. This effect may be mitigated by decreasing the initial stiffness of the interfaces, but this leads to an increase in artificial compliance (a general decrease in the stiffness of the continuum that leads to unrealistic elastic deformation), especially when cohesive surfaces are embedded throughout the finite element mesh [75,78]. In order to introduce a suitable initial elastic stiffness without affecting the critical time step, our approach is to introduce mass penalties to complement the standard stiffness penalties and control the eigenfrequencies introduced by the interface elements. Mass penalties in cohesive surfaces will be discussed in the next section; first we will develop the framework for stiffness degradation.

We begin by describing the traction-displacement relationship to be employed, first by rewriting the constitutive matrix for the cohesive surface stiffness matrix as

$$\mathbf{D}_s = \gamma_s \mathbf{I} \quad (5.19)$$

where  $\gamma_s$  acts as the penalty parameter, but is now dependent on a set of damage parameters. It is initially set to  $\gamma_s = \alpha_s$ , but may decrease over time as damage occurs and the interface cohesion begins to lessen.

The value of  $\gamma_s$  is determined by two scalar quantities, namely the effective open-

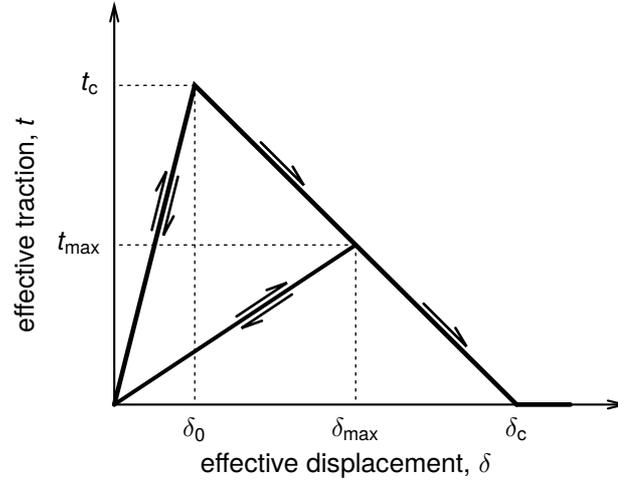


Figure 5.2: Cohesive law for tensile tractions, showing the loading path and a potential unloading path

ing displacement  $\delta$  and the effective traction  $t$ . Inspired by the fracture criteria given by Camacho and Ortiz [18], we have for the effective opening displacement

$$\delta = \begin{cases} \sqrt{\delta_n^2 + \beta^2 \delta_t^2} & \text{if } \delta_n \geq 0 \\ \delta_n & \text{if } \delta_n < 0 \end{cases} \quad (5.20)$$

This value gives a measure of displacement across the interface. The parameter  $\beta$  dictates to what degree tangential displacements are taken into account when assessing damage in the interface. Since no damage is assumed to occur in compression, tangential displacements are not considered when assessing relative displacement for  $\delta_n < 0$ .

The effective traction depends on the current state of the interface and where it lies with regards to the cohesive law shown in Figure 5.2. For undamaged interfaces, the traction-displacement relation is linear-elastic. By defining the history parameter  $\delta_{\max}$  as the maximum effective opening displacement reached during an analysis, we can say that the interface is undamaged if  $\delta_{\max} \leq \delta_0$ , where  $\delta_0$  is the effective displacement value corresponding to the onset of damage. Therefore, during this initial phase,

$$t = \alpha_s \delta \quad \text{if } \delta_{\max} \leq \delta_0 \quad (5.21)$$

where  $\alpha_s$  is the initial elastic stiffness of the interface. If the effective traction should exceed the maximum value of  $t_c$  the interface enters a damaged state and the con-

stitutive relation changes to reflect linear softening in the material, so that

$$t = t_c \left( 1 - \frac{\delta - \delta_0}{\delta_c - \delta_0} \right) \quad \text{if } \delta > \delta_0, \delta = \delta_{\max} \quad (5.22)$$

If the effective displacement reaches the critical value  $\delta_c$  during loading then the interface is broken irreversibly, creating a free surface, and

$$t = 0 \quad \text{if } \delta_{\max} > \delta_c \quad (5.23)$$

If the effective opening rate becomes negative ( $\dot{\delta} < 0$ ) at any time *after* damage has occurred then the interface is said to be unloading. In this state the constitutive behaviour is once again linear-elastic, but with a reduced stiffness. Then,

$$t = \frac{t_{\max}}{\delta_{\max}} \delta \quad \text{if } \delta < \delta_{\max} \quad (5.24)$$

where  $t_{\max}$  is the effective traction corresponding to the effective displacement  $\delta_{\max}$ . Together, these relations describe all four of the possible states for an interface: undamaged (linear-elastic), loading (softening), unloading (linear-elastic, reduced stiffness) and broken (zero traction/free surface).

The effective penalty parameter  $\gamma_s$  of (5.19) is given by

$$\gamma_s = \begin{cases} \alpha_s & \text{if } \delta_{\max} \leq \delta_0 \\ t_{\max}/\delta_{\max} & \text{if } \delta_{\max} > \delta_0 \end{cases} \quad (5.25)$$

We assume that the gap across the interface does not close again after breaking. To model such problems, the definition could be extended so that the interface has stiffness in compression ( $\delta < 0$ ) even after the maximum displacement has been reached ( $\delta_{\max} \geq \delta_c$ ).

At each time step, the effective relative displacement  $\delta$  is computed for each interface so that any changes in the state of the interface may be detected. For affected elements the damage model is implemented by computing the associated effective tractions and then updating the constitutive matrix for those elements via Equation (5.19).

Finally, we introduce the cohesive fracture energy  $G_c$ , a fundamental parameter

of the cohesive zone model regarded as a material constant, which represents the work of separation per unit area of cohesive surface. It is given by the area under the traction-displacement curve, which, for the formulation described above, gives

$$G_c = \int_0^{\delta_c} t \, d\delta = \frac{t_c \delta_c}{2} \quad (5.26)$$

This relationship allows the traction-displacement curve to be fully described by  $t_c$  (representing the yield strength of the material), and the fracture energy  $G_c$ , both of which can be obtained by experimental testing of a specimen.

### 5.1.3 Mass matrix

Thus far, we have formulated a stiffness matrix for the interface, based on the minimisation of total potential energy, as well as a constitutive framework to capture stiffness degradation. In order to obtain a full bipenalty formulation, and thus obtain a suitable mass matrix, we must also consider the kinetic energy of the interface, which is related to velocity. Thus, analogous to the corresponding displacement terms from Section 5.1.1 we have

$$\dot{\mathbf{u}} = \mathbf{H}\dot{\mathbf{d}} \quad (5.27)$$

$$\dot{\boldsymbol{\delta}} = \mathbf{L}\dot{\mathbf{u}} \quad (5.28)$$

$$\dot{\boldsymbol{\delta}} = \mathbf{B}\dot{\mathbf{d}} \quad (5.29)$$

Introducing a momentum vector  $\mathbf{p} = [p_n, p_t]^T$  we can then write a momentum-velocity relation,

$$\mathbf{p} = \mathbf{D}_m \dot{\boldsymbol{\delta}} \quad (5.30)$$

where  $\mathbf{p}$  represents momentum (per unit area) in the normal and tangential directions, and the matrix  $\mathbf{D}_m$  contains mass penalties in the normal and tangential directions (with units  $\text{kg}/\text{m}^2$ ). It is assumed that this matrix is a scalar multiple of the constitutive matrix (5.19), so that  $\mathbf{D}_s = R\mathbf{D}_m$ , since this will simplify the implementation (and in any case, there is no apparent reason for the two penalty types to possess different normal/tangential contributions). The kinetic energy of the inter-

face is then given by

$$\mathcal{T} = \frac{1}{2} \int_S \dot{\boldsymbol{\delta}}^T \mathbf{p} \, dS \quad (5.31)$$

which, after invoking (5.29) and (5.30), becomes

$$\mathcal{T} = \frac{1}{2} \dot{\mathbf{d}}^T \int_S \mathbf{B}^T \mathbf{D}_m \mathbf{B} \, dS \, \dot{\mathbf{d}} \quad (5.32)$$

The equations of motion then follow from the minimisation of energy

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{K}\mathbf{d} = \mathbf{f} \quad (5.33)$$

where the mass matrix is given by

$$\mathbf{M} = \int_S \mathbf{B}^T \mathbf{D}_m \mathbf{B} \, dS = \frac{1}{R} \mathbf{K} \quad (5.34)$$

Since the mass matrix is therefore a scalar multiple of the stiffness matrix, the stability analysis contained in Chapter 3 also applies here, with the assembled interface matrices acting as the penalty matrices  $\mathbf{K}^P$  and  $\mathbf{M}^P$ . In addition, for a bipenalty formulation the integral in (5.34) need not be computed numerically, but instead can be calculated directly from the computed stiffness matrix (or visa versa).

Note that since the entries of the mass matrix sum to zero, no physical mass is added to the system. Just as the mechanical inerter is considered to be massless, so the bipenalty interface element has no mass; rather, it has a mass matrix that applies inertia forces when relative acceleration across the interface is non-zero.

With the inclusion of a damage law, we must also consider how the constitutive matrix  $\mathbf{D}_m$  may change as damage occurs; i.e., identify the damage parameters which determine  $\gamma_m$  in the non-linear constitutive matrix

$$\mathbf{D}_m = \gamma_m \mathbf{I} \quad (5.35)$$

One option is to form a new damage law specifically for the mass matrix of the element as opposed to the traditional traction-displacement used to govern the stiffness of the interface. However, this would effectively constitute a new set of velocity constraints, whereas our goal is to enforce the non-linear displacement constraints (in the form of cohesive surfaces) which have already been derived. Consequently, we

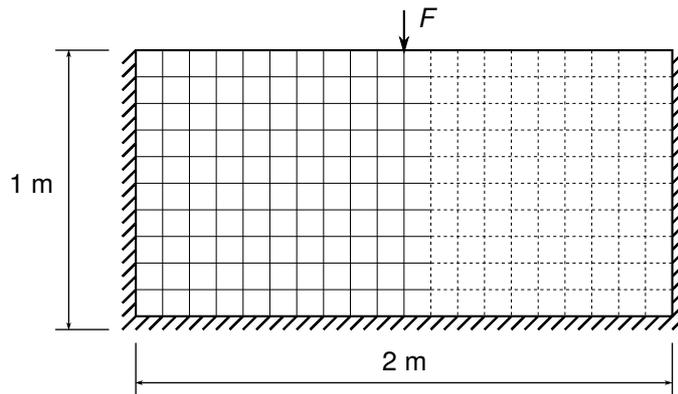


Figure 5.3: Diagram of rectangular elastic region showing fixed supports (indicated by hashed edges) and interface elements (dotted lines), for an element size of  $h = 0.1$  m.

adopt the same cohesive law for both mass and stiffness matrices, which, with the reintroduction of the penalty ratio  $R = \alpha_s/\alpha_m$ , gives  $\gamma_m = \gamma_s/R$ . Once again, the penalty ratio  $R$  may be used to control the relative influence of the stiffness and mass penalty matrices. Note by adopting this method, the assumption that  $\mathbf{D}_s = R\mathbf{D}_m$  (and therefore  $\mathbf{K} = R\mathbf{M}$ ) is valid throughout the analysis.

## 5.2 Elastic wave propagation through 2D bipenalty interface elements

In order to test the interface element formulation, we consider a rectangular region of elastic material modelled using the finite element method. A regular grid of square elements is used to mesh the domain. Two-dimensional interface elements are then inserted between half of the FE continuum elements to observe what effect this has on wave propagation through the medium, as shown in Figure 5.3. At this point, the interfaces are non-breaking and do not suffer damage, and the interface constitutive matrices are given simply by  $\mathbf{D}_s = \alpha_s\mathbf{I}$  and  $\mathbf{D}_m = \alpha_m\mathbf{I}$  regardless of interface tractions. (This corresponds to a cohesive surface formulation where the maximum traction  $t_c$  is never exceeded.)

In the code, interfaces are added by first specifying the geometrical bounds of a ‘process window’ inside which damage may occur. The elements inside this window are then separated from those that surround them. Nodes belonging to the elements within the process window must be repeated in order to achieve this, which increases

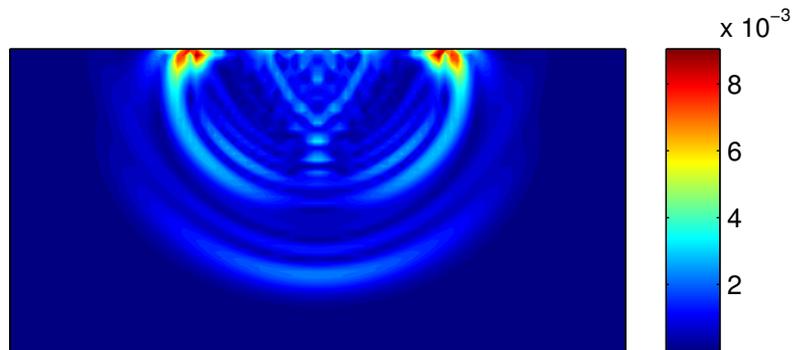
the number of DOF that must be considered (and therefore the size of the system). The significance of this increase depends on the proportion of total elements included in the process window. Once the elements are separated, the global interface matrices are assembled from the individual interface elements needed to connect them. The interfaces have length but no width (initially) and therefore the initial geometry of the problem is not changed. For a bipenalty matrix, the elements each possess a mass matrix, but since the sum of all matrix entries is zero no additional mass is introduced to the system.

Stress wave propagation through the rectangular system is shown qualitatively in Figure 5.4, for a stiffness penalised system. The material has arbitrary properties Young's modulus  $E = 1$  Pa, mass density  $\rho = 1$  kg/m<sup>3</sup>, Poisson's ratio  $\nu = 0$ , and plane stress is assumed. The point load  $F = 10^{-3}$  N is applied from the beginning of the analysis until time  $t = 0.1$  s. The element side length is  $h = 0.02$  m for all elements (giving a total of 5000 elements). The fixed-edge boundary conditions are enforced using direct imposition.

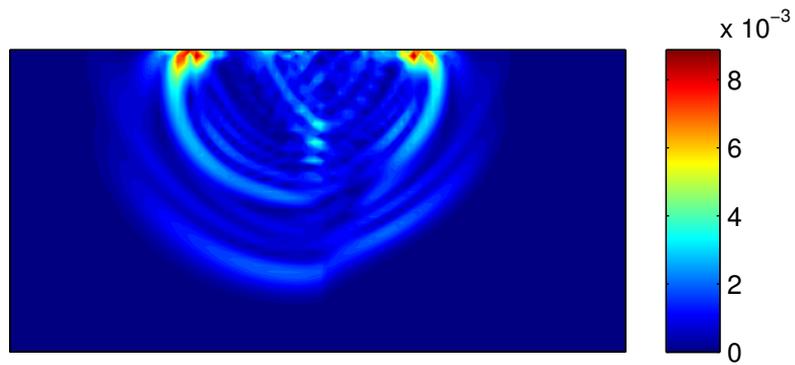
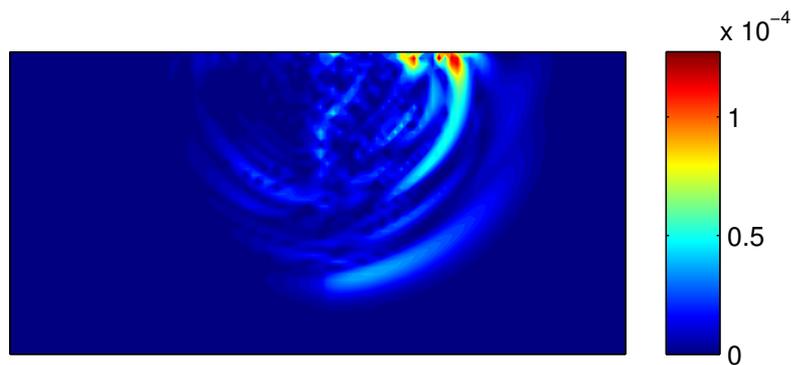
In order to quantify errors, a reference solution (shown in Figure 5.4(a)) is first produced by omitting interface elements entirely. By introducing interface elements with low stiffness penalty parameters, as in Figure 5.4(b), we can easily observe the effects of the added interfaces. Note that the analysis concludes before any wave reflections occur at the boundaries of the region, and hence wave propagation on the left-hand side of the material appears quite unaffected by the interfaces. In the interface process window, however, stress wave propagation is slowed considerably by the additional elastic strain manifesting between the continuum elements, a phenomenon known as artificial compliance. This is clearly evident in the stress error field, shown in Figure 5.4(c).

Figure 5.4 demonstrates the need for well-enforced interface constraints. If no damage has occurred, a continuum that includes cohesive surfaces and the interface-free continuum should ideally behave identically, with zero displacement across an interface. Using penalty methods, this is only possible in the limit as penalty parameters tend to infinity, but cohesive surfaces can be practically transparent if the initial penalty is large enough. Of course, with stiffness-type penalties this introduces the standard concerns with regards to time step stability.

Figure 5.5 gives some idea of the effects of two kinds of catastrophic error (i.e.,

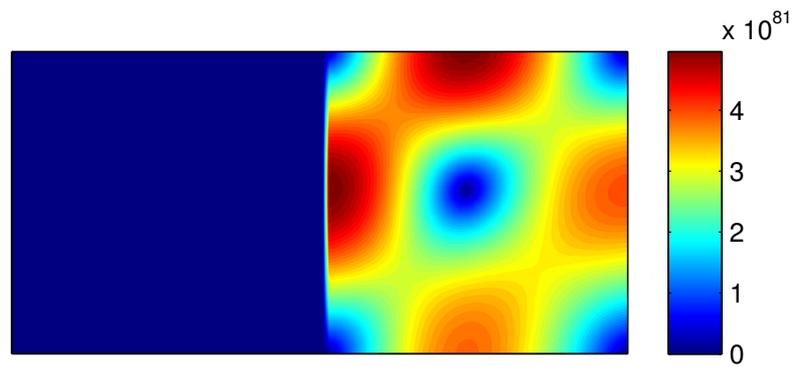


(a) No interfaces

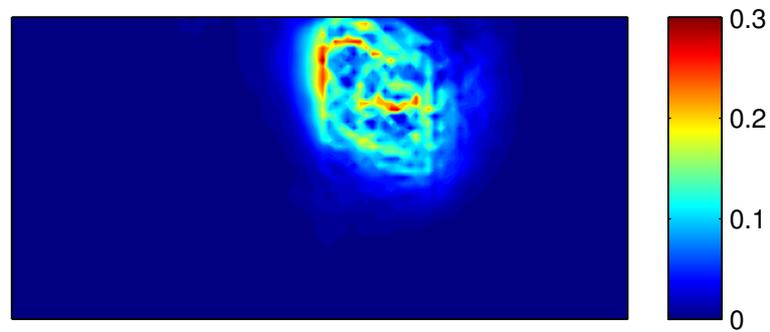
(b) Low stiffness penalty ( $p_s = 100$ )

(c) Absolute error field

Figure 5.4: Von Mises stress profiles at time  $t = 0.8$  s, including error field (units: Pa).



(a) Computational errors due to large mass penalty ( $\rho_s = 0, \rho_m = 10^{14}$ )



(b) Time step instability ( $\rho_s = 1.02 \times 10^4, \rho_m = 0$ )

Figure 5.5: Von Mises stress profiles at time  $t = 0.8$  s for two examples of catastrophic error (units: Pa).

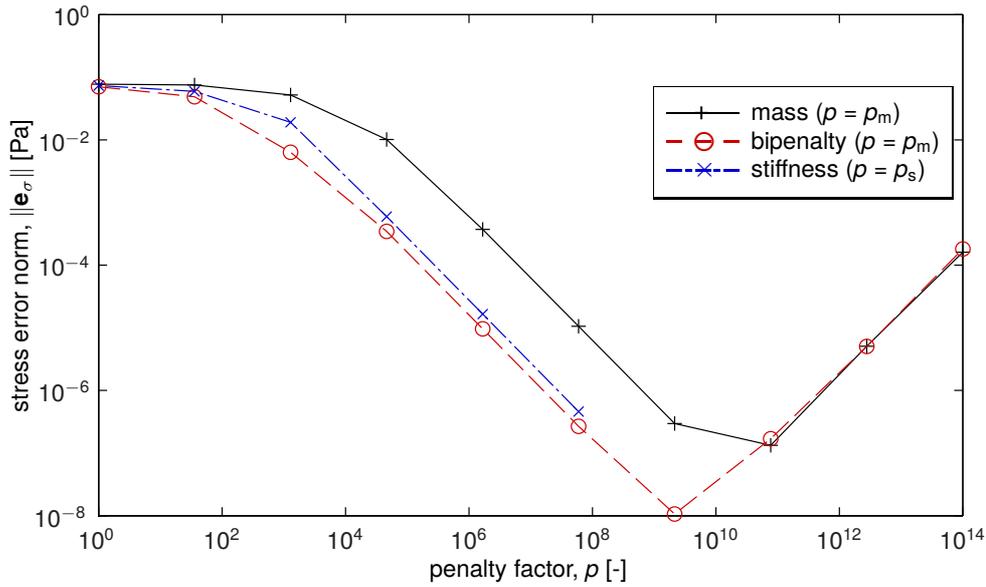


Figure 5.6: Stress error norm at time  $t = 0.8$  s for the stiffness, mass and bipenalty methods.

those which render the results of an analysis so meaningless as to be of no use to an analyst). Figure 5.5(a) shows the manifestation of computational error due to the use of very large mass penalty parameters. Here, round-off errors are exaggerated due to the ill-conditioning of the global mass matrix (the solution of a linear system is required due to the interface elements resulting in a non-diagonal mass penalty matrix). Figure 5.5(b) shows unstable behaviour caused by the use of stiffness-type penalties artificially increasing the maximum eigenvalue of the system.

Before moving on to more complex examples it is worthwhile to compare the performance of the stiffness, mass and bipenalty methods, as well as the influence of penalty parameter magnitude, in the context of this simple 2D problem. Figure 5.6 shows the  $L_2$  norm of the error in stress profile,  $\|e_\sigma\|$ , between two analyses with and without penalty-based interface elements, for a range of penalty factors. Note that the quantity represented by the  $x$ -axis (penalty factor,  $p$ ) represents the penalty factor that has been used as input in each case. For the stiffness penalty analyses this is the stiffness penalty factor  $p_s$ , and for the mass and bipenalty analysis it is the mass penalty factor  $p_m$ , since for the bipenalty method, the stiffness parameters are calculated based on a suitable penalty ratio  $R$  (as described by Algorithm 1 of Section 4.3).

Figure 5.7 shows the timesteps used for the analysis on a logarithmic scale. For mass and bipenalty methods, suitable time steps are estimated using the maximum

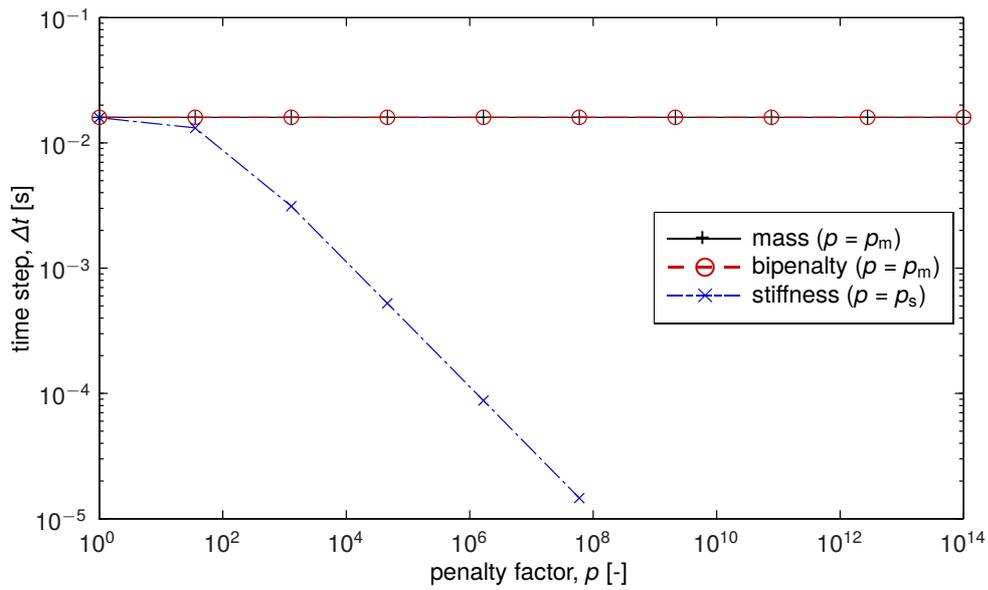


Figure 5.7: Time step used in each analysis (approximately  $0.9\Delta t_{\text{crit}}$ ).

eigenvalue of all individual elements, whereas for the stiffness penalty method the maximum eigenvalue of the full constrained system must be used. This data shows that achieving high accuracy using stiffness-type penalties quickly becomes very expensive as penalty parameters are increased; indeed, for stiffness penalty factors  $p_s > 10^8$  the tests became prohibitively expensive.

The final two plots, shown in Figure 5.8 show the condition number  $\kappa$  for the reduced<sup>1</sup> global stiffness and mass matrices, as estimated numerically using MATLAB. For the mass penalty analysis the stiffness matrix is singular, since many of the continuum elements positioned in the left-hand side of the model are free and unconstrained, but this is perfectly acceptable when using the central difference method as a time integrator. Once again, the condition numbers in both cases are approximately equal to the penalty factors being used.

### 5.3 Crack propagation in a single-edge-notched beam

In this section we introduce breaking interfaces and the cohesive damage model described in Section 5.1.2 in order to investigate dynamic crack propagation. The experimental test under consideration is a fairly well-known problem referred to as

<sup>1</sup>The term ‘reduced’ here refers to the fact that fixed-support boundary conditions are taken into account using direct imposition; the condition numbers are calculated from the reduced matrices after this process, but with penalty methods enforcing the interfacial constraints.

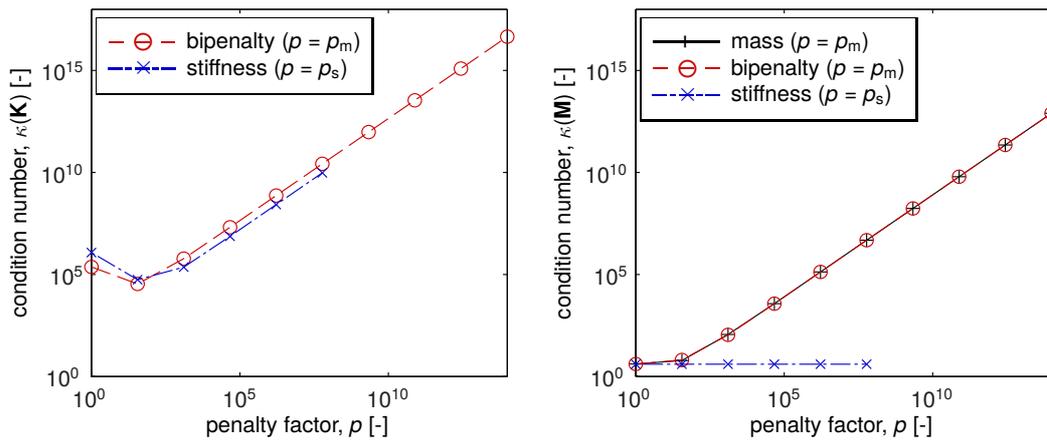


Figure 5.8: Condition numbers of stiffness (left) and mass (right) matrices, estimated using the MATLAB function `condest()`.

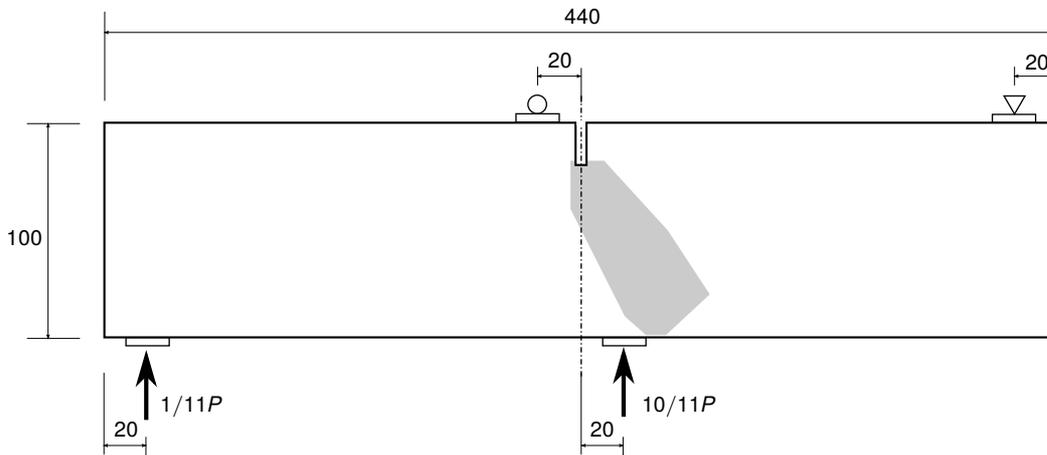


Figure 5.9: The single-edge-notched (SEN) beam experimental setup, showing process window (grey area below notch tip). All dimensions are in mm.

the single-edge-notched (SEN) beam, for which quasi-static experimental results are available in the literature [69]. The geometry and boundary conditions of the problem are shown in Figure 5.9. The beam is assumed to have the following material properties: Young's modulus  $E = 35$  GPa, mass density  $\rho = 1000$  kg/m<sup>3</sup>, Poisson's ratio  $\nu = 0.2$ , and plane stress is assumed. For the damage model, the critical effective traction is taken as  $t_c = 3$  KPa, and the maximum effective displacement is  $\delta_c = 10^{-6}$  m.

Crack paths recorded from experimental tests of the SEN beam have been reproduced in Figure 5.10. In order to reproduce these tests numerically, several FE meshes are created (with the open source meshing software Gmsh [31]) using 3-

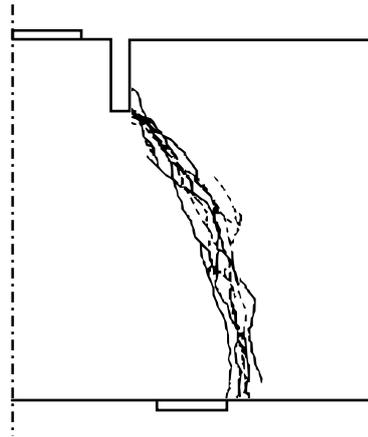


Figure 5.10: Experimental results showing crack path for the SEN beam under quasi-static loading (from Schlangen [69]).

No. of elements	No. of interfaces	Time step, $\Delta t$ [ $\mu\text{s}$ ]	Total CPU time [min]
1617	739	3.608	15.7
3333	1467	3.127	46.3
6428	3045	1.837	177.8
9716	4681	1.367	393.1

Table 5.1: Approximate CPU time spent during the solution phase for each of the analyses shown in Figure 5.11 (AMD Athlon 3.2 GHz CPU).

noded linear triangular elements. Bipenalty interface elements are then introduced between the continuum elements in the central process window, and dynamic tests are carried out using the central difference time integration scheme. Penalty parameters are determined using Algorithm 1 of Section 4.3. The total load  $P$  is set to increase linearly from zero to 30 KPa over a analysis duration of 0.2 s and the analysis is terminated when the crack tip ordinate is less than 10 mm from the bottom edge of the beam. The final crack paths for four different meshes are shown in Figure 5.11. Some details about the computations, including the calculated time step and the total time spent computing the solutions, is shown in Table 5.1.

It should be noted that a true quasi-static analysis is not possible in this case as it would be too computationally expensive to apply the loading at a rate slow enough that inertia effects could reasonably be neglected. Therefore, there are presumably some inertia effects present in the analyses which would not have manifested in the quasi-static experimental tests. Despite this fact, the final crack paths show reasonable agreement with those obtained experimentally.

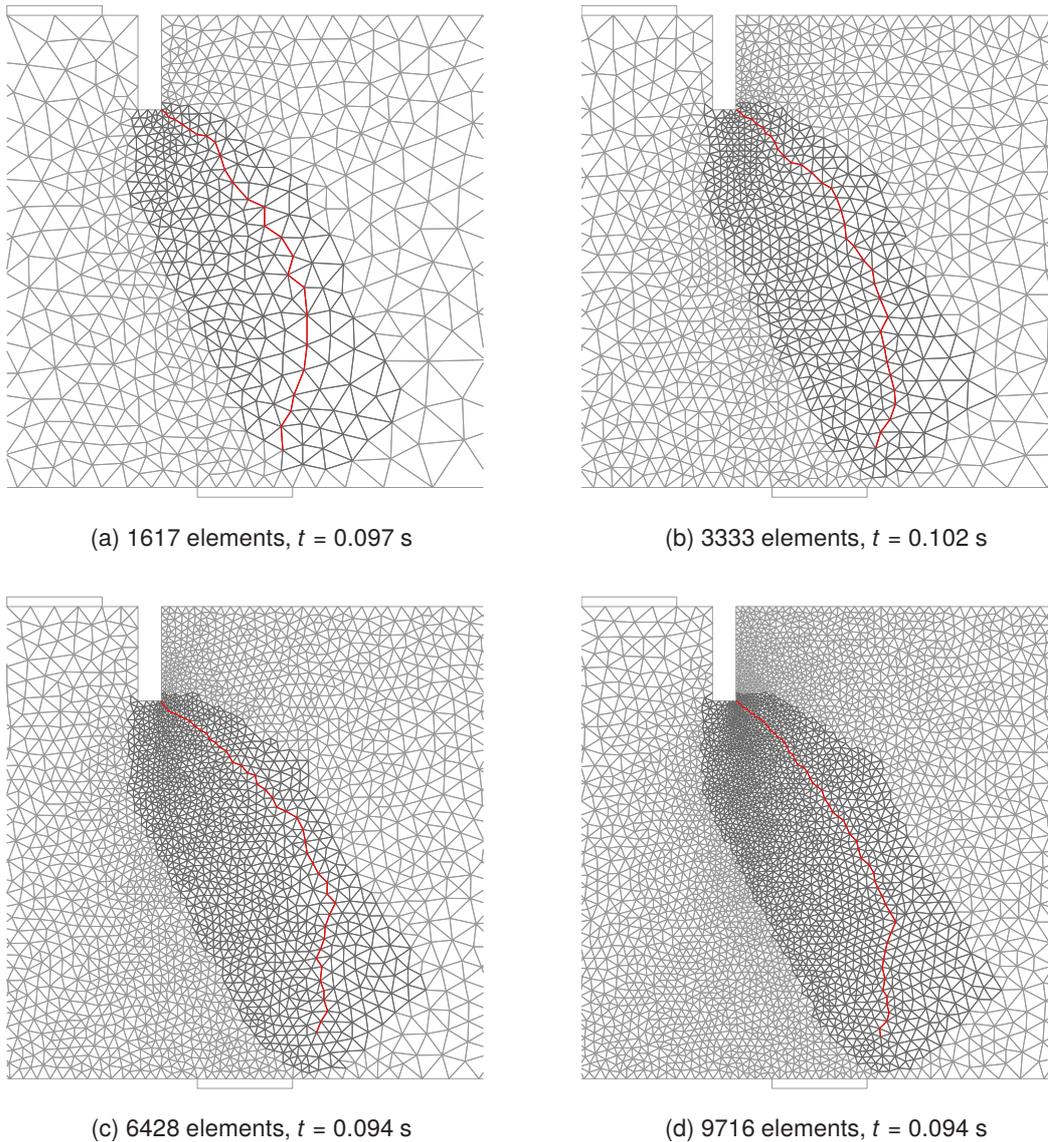
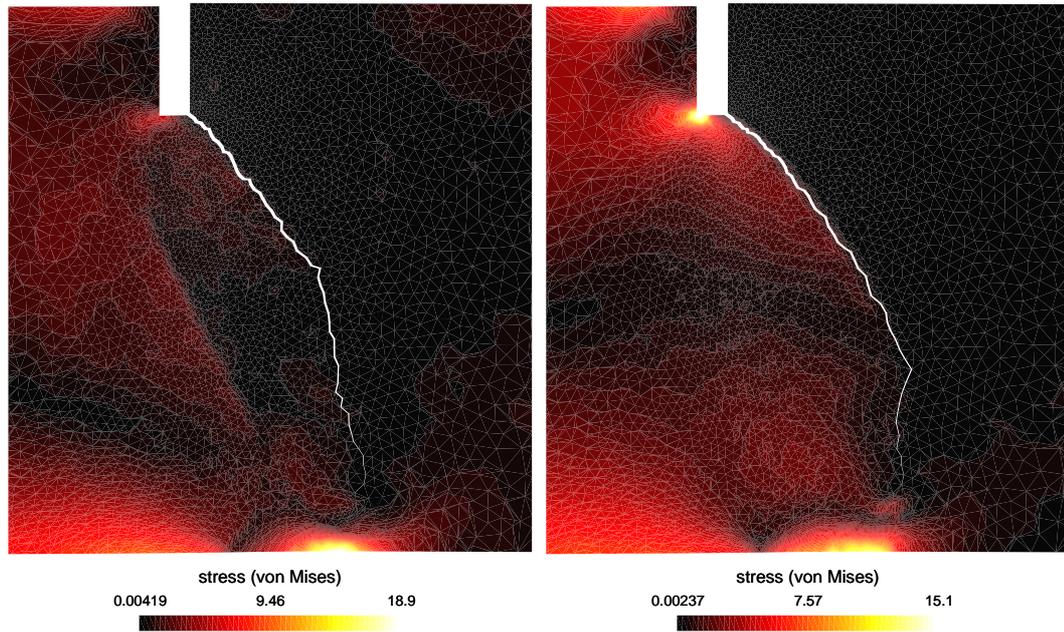


Figure 5.11: Crack paths for increasingly refined mesh.

Mesh dependence is often cited as a major drawback of cohesive zone models, where cracks can only form along element boundaries; undoubtedly, this is a major advantage of meshless methods and also explains why adaptive remeshing is often utilised in these problems. In this case, however, even the coarsest mesh gives a reasonable final crack path, and there is no large variation in the time taken for the crack to extend to the specified end point.

The time step used for these analyses is calculated using the maximum elemental eigenvalue as an estimate for the maximum eigenvalue of the system<sup>2</sup>, with a safety

<sup>2</sup>In fact, in an analysis where cracks are free to form throughout the continuum (leading to a less constrained system) and elements can potentially separate completely, the maximum eigenvalue of the



(a) Stiffness penalty analysis ( $\rho_s = 0.7$ ) at time  $t = 0.080$  s

(b) Bpenalty method at time  $t = 0.094$

Figure 5.12: Von Mises stress profiles for the SEN beam using stiffness and bpenalty methods (units: Pa).

factor of 0.9, giving  $\Delta t \approx 1.3 \mu\text{s}$ . Figure 5.12(a) shows a stiffness penalty analysis using the highest stiffness penalty allowed with this time step. Here, the bounds of the process window are clearly visible in the stress profile, due to the artificial compliance that is introduced in that area. In addition, the crack propagation speed is higher than any of the bpenalty analyses. For comparison, the stress profile and crack path of a bpenalty analysis are shown in Figure 5.12(b). The differences are notable.

## 5.4 Dynamic analysis of PMMA plate with pre-existing crack

For validation of the formulation in a dynamic setting we turn our attention to an experiment carried out by Grégoire et al. [34] (and further investigated by Combescure et al. [21]), which investigates crack propagation through a polymethyl methacrylate (PMMA) plate under impact loading. The experiment uses a Hopkinson bar to ap-

---

system may increase during an analysis. It is therefore dangerous to assume that at the end of the analysis the maximum eigenvalue  $\omega_{\max}$  will be larger than the maximum element eigenvalue  $\omega_{\max}^e$ , which in this case gives a reliable upper bound.

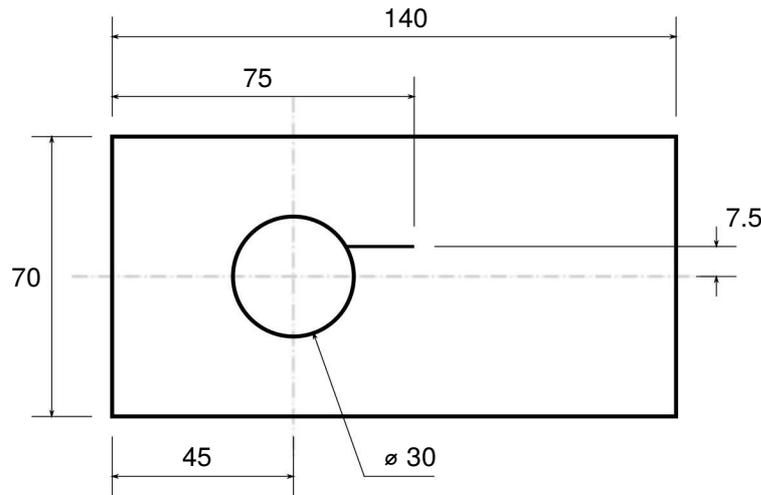


Figure 5.13: Geometry of the PMMA specimen (dimensions in mm).

ply load to the left-hand side of the PMMA specimen shown in Figure 5.13, which features a pre-existing crack emanating from a central hole. The specimen suffers fracture damage during the first 500  $\mu\text{s}$  after impact, and the crack tip position is measured during this time to obtain a detailed crack propagation history.

In this section we simulate the experiment numerically using bipenalty cohesive surfaces. Only the PMMA plate is considered, since the initial contact with the Hopkinson bar on the left-hand side of the specimen can be modelled with prescribed velocities, while contact at the other end is handled with adsorbing boundary conditions. Given material properties for the plate include Young's modulus  $E = 4.25 \text{ GPa}$ , mass density  $\rho = 1180 \text{ kg/m}^3$ , Poisson's ratio  $\nu = 0.42$  and fracture toughness  $K_{IC} = 1.47 \text{ MPa}\sqrt{\text{m}}$ .

For the cohesive bipenalty formulation we require the damage parameters  $t_c$  and  $G_C$  (from which we can calculate  $\delta_c$ ). The cohesive fracture energy can be calculated from the fracture toughness via  $G_C = K_{IC}^2/E$ . The yield stress  $t_c$  is not given, and so initially a range of values are tested, from 10–100 KPa.

The final crack paths are shown in Figure 5.14 for a range of possible  $t_c$  values. Although the total length of the path is not captured by the simulations, the initial portion of the crack is reproduced very well for  $t_c = 15\text{--}21 \text{ KPa}$ . Note that since the crack is required to move along element boundaries it cannot generally move in a perfectly straight line, which adds extra length to the crack path. This in turn adds to the amount of energy required to open the crack, which may help to explain why

the numerical paths are shorter than those obtained experimentally.

The growth of the crack over time is compared to experimental observations in Figure 5.15. The experimental results show the crack beginning to form at around  $200 \mu\text{s}$ , stopping briefly when it reaches  $x \approx 90 \text{ mm}$ , before continuing to its end point. While in the numerical tests crack initiation is about  $25 \mu\text{s}$  early, and crack speed propagation is generally higher than in experiments, the stop-start behaviour of the crack propagation is captured to some extent for  $t_c = 12\text{--}18 \text{ KPa}$ . For comparison purposes, the results from Reference [21], which uses XFEM instead of cohesive surfaces, is shown in Figure 5.16.

The mostly likely reason for disparities in the numerical tests is the accuracy of the bilinear cohesive law. By refining the traction-displacement relationship it may be possible to obtain more accurate results, although this would likely require access to further experimental test data for the material in question. An alternative cohesive model designed for the modelling of PMMA is given by Elices et al. [27].

However, in general the bipenalty cohesive surfaces are well-suited to this problem type, where high loading rates mean that the explicit central difference method is an obvious choice for the solution scheme. Since stiffness-only elements lead to artificial compliance, an artificially lowered time step, or else unexpected time step instabilities, their use is problematic. On the other hand, the bipenalty method allows for time steps close to the critical time step of the unconstrained problem, while allowing for a very high initial stiffness.

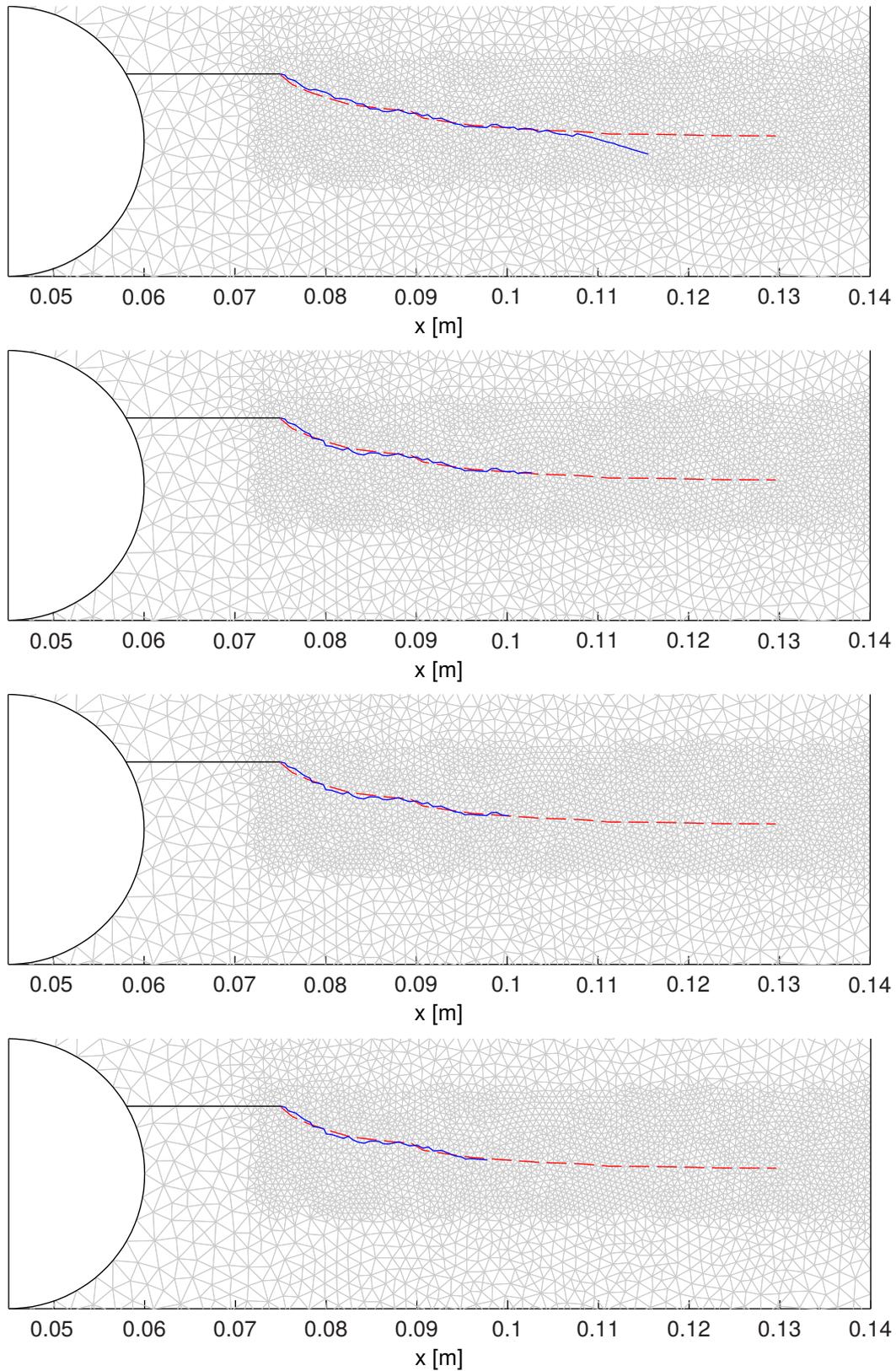


Figure 5.14: Final crack paths. From top to bottom,  $t_c = 12, 15, 18, 21$  KPa

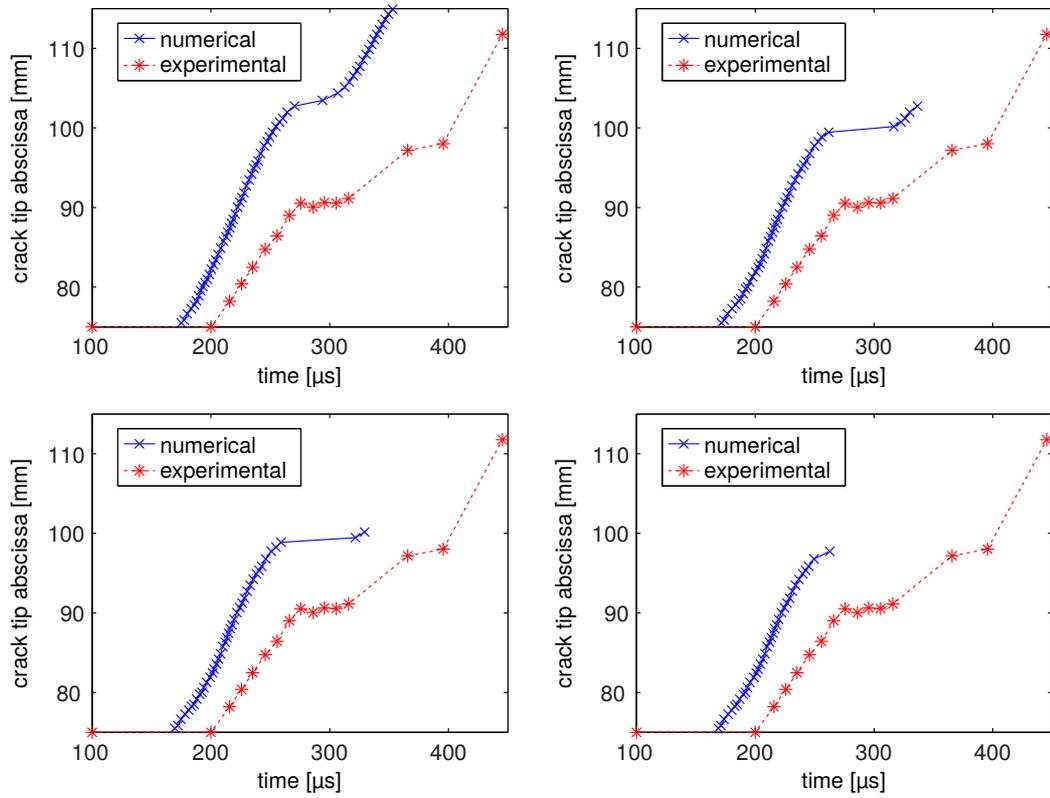


Figure 5.15: Position of crack tip ( $x$ -dir) over time:  $t_c = 12$  KPa (top left),  $t_c = 15$  KPa (top right),  $t_c = 18$  KPa (bottom left),  $t_c = 21$  KPa (bottom right).

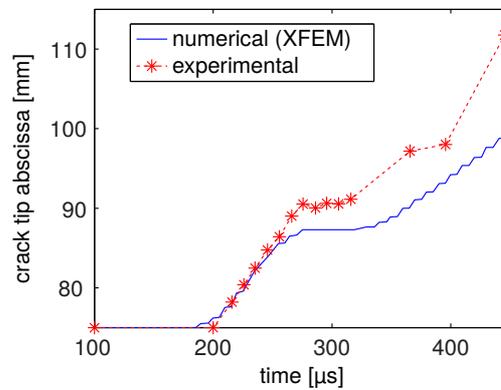


Figure 5.16: Position of crack tip over time (from Combescure et al. [21]).



## Chapter 6

# Modelling of contact-impact using the bipenalty method

The modelling of contact is necessary in a large variety of physical simulations, including impact testing, the analysis of structural elements, and the simulation of mechanical processes, and as such has become a standard feature in commercial finite element (FE) software packages. Especially relevant to the current project is the field of blast and impact, where explicit dynamic FE packages are especially popular, due to their suitability for modelling short, highly dynamic problems.

Contact problems can be roughly defined as any problem for which it is necessary to stipulate that two or more bodies (or, for some large deformation problems, two parts of the same body) cannot occupy the same space at the same time. The term ‘contact-impact’ is often used to describe the subset of these problems for which dynamic effects are taken into account. Compared to the problems considered in previous chapters, contact problems are more difficult to model due to the nature of the constraints and the difficulty in properly detecting and characterising contact as it occurs. Since the present work is concerned with the former of these two areas—the imposition of constraints—the problems in this chapter are selected so as to avoid the necessity for complex search and detection algorithms. Similarly, we assume linear-elastic material behaviour and employ small strain theory in the continuum, so that the contact constraints are the only source of non-linearity in the formulation.

## 6.1 Literature review

Contact and contact-impact are topics that have been covered extensively in the literature, and many different approaches have been developed. For constraint imposition, researchers have called upon many techniques, with the Lagrange multiplier method [19,44,77] and penalty methods [1,3,15,33,37,71] remaining the most popular. A range of implicit [1,44,77] and explicit [14,18,19] time integration schemes have also been employed. In the present study, we focus on explicit dynamics (especially the central difference method) combined with penalty-based formulations. Full overviews (including static contact, implicit methods and alternative methods of constraint enforcement) are available in standard texts [12,79,81] while reviews of the relevant literature can be found in References [17,32,82,83].

The first widely available explicit code that featured the ability to handle contact-impact problems was DYNA, initially developed by John Hallquist and released in 1976 [12]. Early publications by Hallquist and co-workers indicate that penalty methods were favoured due to a lack of mesh hourglassing and the fact that conservation of momentum is achieved without any special impact and release conditions [15,33,37]. LS-DYNA, a popular commercial solver that evolved from these early codes still uses penalty methods as the basis for its contact algorithms [36].

Explicit solvers are well-suited to solving contact-impact problems. Aside from their efficiency, they provide a straightforward treatment of non-linear contact constraints since they “permit the new displacement configuration at each time step to be computed in terms of strictly historical information” [63]. Camacho and Ortiz also choose an explicit scheme based partly on the fact that “explicit contact algorithms are more robust and straightforward than their implicit counterparts” [18]. In fact, a combination of penalty functions (for the constraints) and an explicit solver is probably the simplest method of handling contact-impact in terms of implementation.

Of course, there are disadvantages to this approach. The main downside of using explicit time integrators is the need to consider the critical time step, and when dealing with contact-impact (as with other applications) the use of stiffness penalty parameters complicates the situation. Goudreau and Hallquist in fact noted this problem early on when confronted with unacceptable errors in constraint imposition, stating that

if interface pressures become large, unacceptable penetration may occur. By scaling up the stiffness and scaling down the time step size, we may still solve such problems using the penalty approach. Since a large increase in cost may arise due to more time steps, a sliding only option has been implemented in DYNA3D for treating explosive-structure interaction problems and thereby avoid use of the penalty approach. [33, p. 741]

The estimation of safe critical time steps for contact-impact problems has therefore been the subject of some study [14, 51, 63], with Plesha clearly stating the essence of the problem: “the extreme frequency of the global finite element discretization cannot be bound by the maxima of unconstrained element frequencies, as is conventional practice, since the interface elements have zero mass and hence, infinite frequencies” [63]. The bipenalty method aims to solve this problem by adding a mass matrix and thus setting the eigenfrequencies of the contact elements to known finite values (the elements are given a *distribution of mass*; total mass remains zero).

Beyond time step control, contact-impact algorithms can often be sensitive to penalty parameter values in other ways. Huněk shows in his study on the effect of penalty parameter magnitude that even modest penalty parameter values can lead to considerable (spurious) oscillations in the solution, with the amplitude of the oscillation becoming larger as the penalty parameter is increased [45]. Two possible sources for these undesirable oscillations can be identified in the literature.

The first concerns *regularisation* of the contact constraints. At the time of impact between two bodies, the velocities on the contact interface are discontinuous in time. For methods which exactly enforce the contact constraint, such as the Lagrange multiplier method, this discontinuity may propagate after contact has occurred, causing noise in the solution. Penalty parameters (which are proportional to the magnitude of constraint violation) smooth out these discontinuities, resulting in smoother solutions. In this way, the penalty approach may be thought of as a regularisation procedure [12, §10.5.9]. Lowering the penalty parameter leads to a greater degree of regularisation, with the sacrifice being that non-physical penetration between contacting bodies increases. Conversely, increasing the parameter in order to decrease constraint imposition errors may result in a noisier solution.

If it is true that using large penalties will always result in noisy solutions due to insufficient regularisation, then this aspect must be taken into account too when using the bipenalty method. However, a second source has also been suggested.

Armero and Petocz state that

In situations where an extended time of contact appears, penalty schemes imposing only the gap constraint are known to lead in general to oscillations of the contact forces. These oscillations are also present in traditional schemes, and their origin can be traced in part to the lack of satisfaction of the constraint in the velocities. [1, p. 283]

This sentiment is also expressed by Lee, who uses an iterative scheme to improve the satisfaction of velocity and acceleration continuity on contact surfaces [52]. This implies that by penalising only a violation of the displacement constraint (as traditional stiffness penalties do) we neglect to constrain the velocities (and accelerations) properly at the contact interface. It is therefore feasible that the proposed bipenalty formulation, through the use of mass penalty constraints, can improve performance by reducing spurious oscillations. This theory is supported somewhat by the work of Asano [2–4], who introduces separate penalty parameters for displacement, velocity and acceleration constraints on the contact surface to derive an FE formulation for contact-impact.

The contact-impact formulation used in this chapter is based largely on the work of Belytschko and co-workers [12,14], as well as the node-to-node formulation provided by Huněk [45]. With this work as a basis, we apply the bipenalty method to impose the contact constraints.

## 6.2 Problem statement

Figure 6.1 shows two smooth, continuous bodies,  $\Omega^A$  and  $\Omega^B$ , in contact. The surfaces of the two bodies are denoted by  $\Gamma^A$  and  $\Gamma^B$  and the contact surface by  $\Gamma^C = \Gamma^A \cap \Gamma^B$ , while the outward unit normals are  $\mathbf{n}^A$  and  $\mathbf{n}^B$ .

We wish to derive boundary conditions on the contact surface  $\Gamma^C$  that enforce the impenetrability condition

$$\Omega^A \cap \Omega^B = \emptyset \quad (6.1)$$

which simply states that two bodies cannot occupy the same space. However, translating (6.1) into a useful set of constraint equations for use in an FE implementation is not a trivial task, and many approaches have been suggested.

A common method proceeds by defining a displacement gap function (as shown

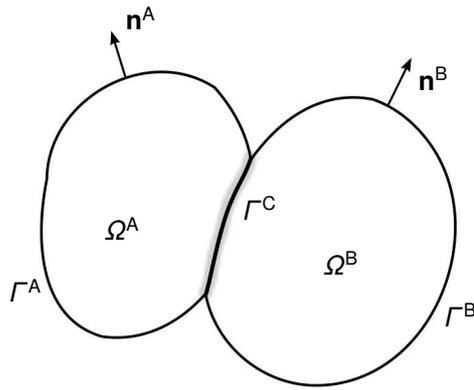
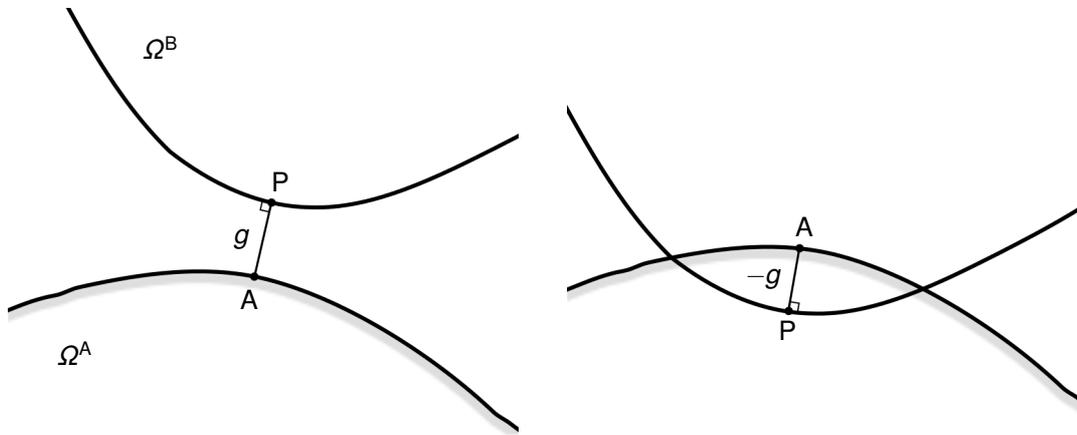


Figure 6.1: Two bodies in contact.

Figure 6.2: The gap function between point P (on  $\Gamma^B$ ) and body A.

in Figure 6.2), given by

$$g_n = (\mathbf{x}^A - \mathbf{x}^P) \cdot \mathbf{n}^P \quad (6.2)$$

where  $\mathbf{x}^P$  and  $\mathbf{x}^A$  are the spatial co-ordinates of points P and A, and  $\mathbf{n}^P$  is the outwards unit normal at point P. The gap function represents the shortest distance from P to surface  $\Gamma^A$ ; that is, the orthogonal projection of P onto  $\Gamma^A$ . When bodies are overlapping, point P is considered to be penetrating body  $\Omega^A$ , and the gap function becomes negative. We choose point P to be the point on  $\Gamma^B$  which minimises  $g$ . The impenetrability condition (6.1) can then be expressed as

$$g_n \geq 0 \quad \text{on } \Gamma^C \quad (6.3)$$

Following the work of Belytschko and co-workers [12, 14], we place a further

constraint on the interpenetration *rate*, expressed as

$$\gamma_n \geq 0 \quad \text{on } \Gamma^C \quad (6.4)$$

which stipulates that the relative normal velocity (or ‘velocity gap’) on the contact surface cannot be negative. The rate constraint (6.4) is sometimes introduced in order to reduce spurious oscillations in the contact force [1]. In the proceeding bipenalty formulation it will be used to derive a suitable mass penalty matrix for the contact constraints.

The normal tractions on  $\Gamma^C$  are given by

$$t_n^A = \mathbf{t}^A \cdot \mathbf{n}^A \quad (6.5)$$

$$t_n^B = \mathbf{t}^B \cdot \mathbf{n}^B \quad (6.6)$$

For simplicity, we consider only frictionless contact so that the tangential tractions are zero and  $t_t^A = t_t^B = 0$ . From conservation of momentum, the tractions on the contact surface must vanish, and so

$$t_n^A - t_n^B = 0 \quad (6.7)$$

By defining the interface normal traction as  $t_n := t_n^A = t_n^B$ , we can write the so-called unitary contact condition

$$t_n g_n = 0 \quad (6.8)$$

which implies that contact tractions can only be non-zero when bodies are in contact ( $g_n = 0$ ).

### 6.3 Contact-impact in one dimension

Implementing robust contact detection and contact constraint enforcement in a multi-dimensional FE model can be problematic. The task is simplified significantly by considering only one spatial dimension. In this setting there is no ambiguity about which nodes are in contact, or about which nodes may come into contact during an analysis, and contact is always node-to-node. Hence, we begin by presenting a one-dimensional (1D) formulation in order to focus on the details of the bipenalty

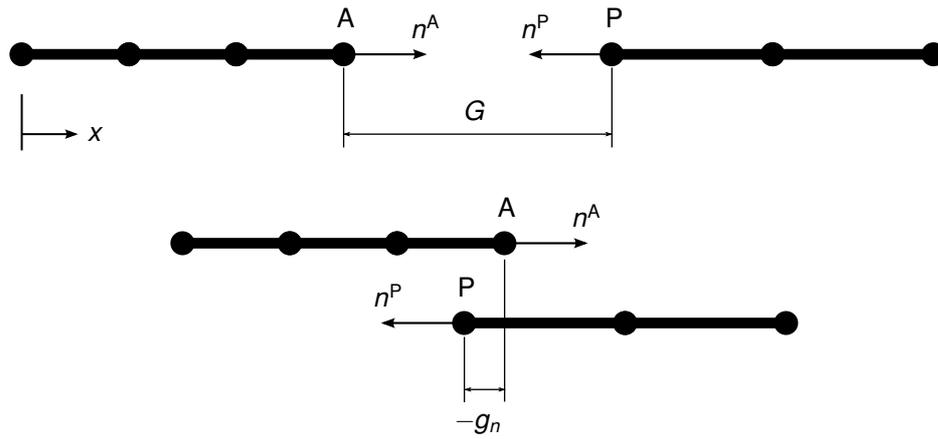


Figure 6.3: Contact between two one-dimensional bodies: initial configuration (top) and during contact (bottom).

method for constraint enforcement. In the next section, we extend these ideas into a two-dimensional (2D) setting.

### 6.3.1 Finite element implementation

In a 1D analysis each node has a single displacement DOF, and the nodal displacements and velocities are scalar values measured in the  $x$ -direction only. Contact surfaces consist of one node (at the end of a body), and the outward unit normals to contact surfaces take a value of  $\pm 1$ . A discretised representation of a typical 1D contact problem is shown in Figure 6.3.

In a 1D FE discretisation, the previously discussed equations governing contact are considered only at nodes. The contact boundary conditions for a contact surface on which node P is penetrating surface A are

$$g_n = (x^A - x^P)n^P = (u^A - u^P)n^P + G \geq 0 \quad (6.9)$$

$$\gamma_n = (v^A - v^P)n^P \geq 0 \quad \text{on } \Gamma^C \quad (6.10)$$

where  $x$ ,  $u$  and  $v$  represent the nodal positions, displacements and velocities, respectively, and  $G$  is the initial gap (at time  $t = 0$ ) between nodes A and P. From these

inequalities we can define the contact constraint equations

$$(u^A - u^P)n^P = -G \quad \text{if } g_n < 0 \quad (6.11)$$

$$(v^A - v^P)n^P = 0 \quad \text{if } g_n < 0 \quad (6.12)$$

Note that Equation (6.11), derived from the gap constraint, contains information about initial configuration of the bodies at the beginning of the analysis, as well as the nodal displacements. The velocity gap constraint (6.12), on the other hand, can be written only in terms of nodal velocities. These constraint equations have the same form used in Section 2.3 to formulate the bipenalty matrices  $\mathbf{K}^P$  and  $\mathbf{M}^P$  and penalty force vector  $\mathbf{f}^P$  and thus can be written in matrix form as

$$\mathbf{h} = \mathbf{S}\mathbf{u} - \mathbf{q} \quad (6.13)$$

$$\dot{\mathbf{h}} = \mathbf{S}\dot{\mathbf{u}} - \dot{\mathbf{q}} \quad (6.14)$$

$$\text{where } \mathbf{S} = n^P(\mathbf{e}^A - \mathbf{e}^P)^T \quad (6.15)$$

$$\mathbf{q} = -G \quad (6.16)$$

where  $\mathbf{e}^A$  is the standard basis vector  $\mathbf{e}_i$  with  $i$  the node number of node A in the global system. Note that the initial gap  $G$  is constant in time and hence  $\dot{\mathbf{q}} = \ddot{\mathbf{q}} = 0$ .

The constraint equations are only applied when a constraint is active. Furthermore, for a system with more than two bodies there may be multiple active contact constraints and hence multiple rows in the constraint matrix  $\mathbf{S}$  and prescribed displacement vector  $\mathbf{q}$ . In order to determine which contact constraints are active a contact detection algorithm is required. For a 1D analysis this process is straightforward. Nodes that may come into contact are identified at the beginning of the analysis. At each time step, and for each contact node pair, the gap function  $g_n$  is calculated via Equation (6.2). If  $g_n < 0$  then penetration is occurring, leading to activation of the contact constraints for that node pair. The penalty matrices and penalty force vector are calculated directly from Equations (2.18)–(2.20).

For a given constraint configuration, the nodal forces can be computed from

$$\begin{aligned} \mathbf{f}^C &= \mathbf{f}^{MP} + \mathbf{f}^{SP} \\ &= \mathbf{M}^P \ddot{\mathbf{u}} + \mathbf{K}^P \mathbf{u} + \mathbf{f}_s^P \end{aligned} \quad (6.17)$$

where  $\mathbf{f}^C$  is the vector of nodal forces due to contact constraints, to which  $\mathbf{f}^{SP} = \mathbf{K}^P \mathbf{u} + \mathbf{f}_s^P$  is the stiffness penalty contribution, and  $\mathbf{f}^{MP} = \mathbf{M}^P \ddot{\mathbf{u}}$  the contribution of the mass penalties (since  $\ddot{\mathbf{q}}$  and therefore  $\mathbf{f}_m^P$  are zero).

Equation (6.17) highlights a potential disadvantage of the bipenalty method when used for contact constraints. As discussed, contact problems are by nature discontinuous, and this discontinuity can cause numerical difficulties. The traditional penalty method helps to mitigate this problem through regularisation; since the contact force is proportional to the displacement gap, the force is applied gradually as penetration increases, until an equilibrium is reached. However, the mass penalty contribution to contact force  $\mathbf{f}^{MP}$  is not proportional to the displacement gap, but instead dependent on accelerations. The regularisation of  $\mathbf{f}^C$  could therefore be compromised if the acceleration gap (and therefore mass penalty force  $\mathbf{f}^{MP}$ ) is significant at the time of contact. The penalty ratio  $R$  can be used to control, to some extent, the contribution of  $\mathbf{f}^{MP}$  and therefore this effect on the regularisation.

In order to examine this effect, and to address the usual concerns with regards to constraint imposition accuracy and time step stability, some numerical examples of 1D contact-impact problems will now be presented.

### 6.3.2 Numerical example: Huněk bar impact

In a study focussing mainly on the effect of penalty parameter magnitude, Huněk uses a penalty formulation very similar to that described above, except that it uses only the traditional stiffness penalty method in order to enforce the contact constraints. To begin our investigation of the bipenalty method for 1D contact-impact, we reproduce a two body impact problem from that study [45].

The problem consists of two bars, one at rest and one with an initial velocity of  $v_0$  in the positive  $x$ -direction, as shown in Figure 6.4. Both bars have Young's modulus  $E = 100 \text{ N/m}^2$ , mass density  $\rho = 0.01 \text{ kg/m}^3$  and cross-sectional area  $A = 1 \text{ m}^2$ . The length of bar 1 is  $L = 10 \text{ m}$ , while bar 2 has length  $2L = 20 \text{ m}$ , and the bars are discretised using linear 2-noded bar elements of length  $h = 0.2 \text{ m}$ . The initial velocity given to bar 1 is  $v_0 = 0.1 \text{ m/s}$  and the initial gap is  $G = 0$ .

For the given geometry, nodes A and P come into contact twice before separating. The analytical solution gives a contact force of 0.05 N for time  $0 < t \leq 0.2$  and  $0.4 < t \leq 0.6$ , and zero otherwise. For  $0.2 < t \leq 0.4$  both nodes are at rest while

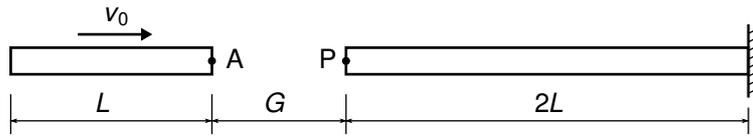


Figure 6.4: Bar impact problem

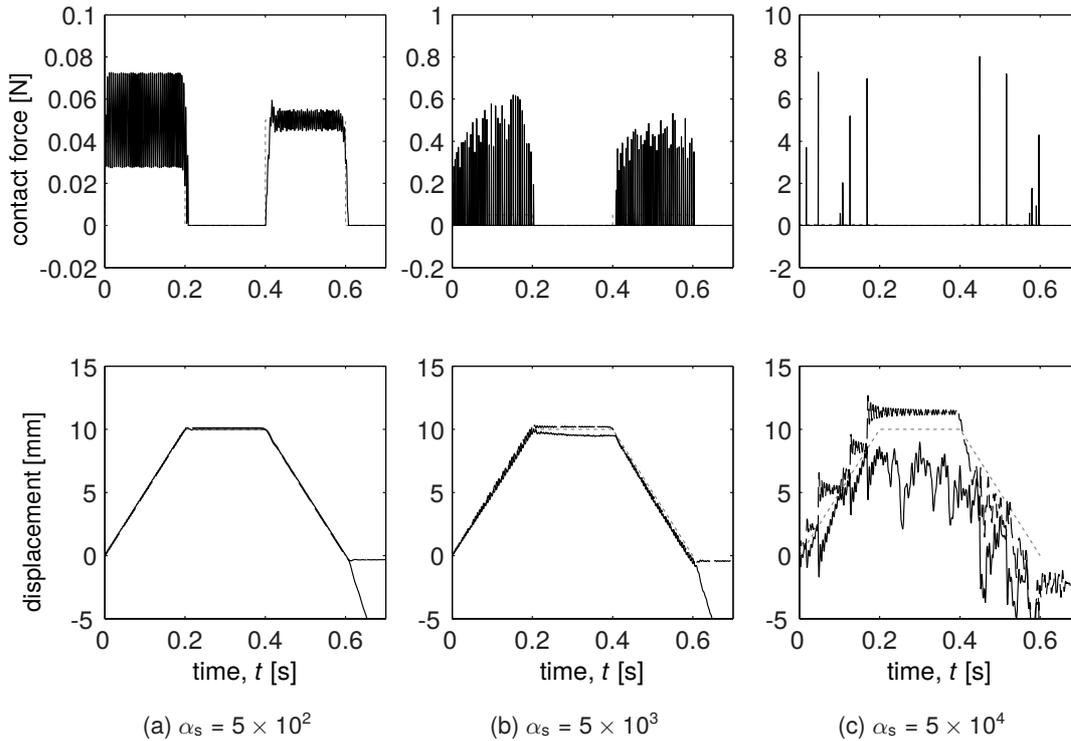


Figure 6.5: Stiffness penalty method: contact force, and displacement of contacting nodes for increasing stiffness penalty parameter. The analytical solution is indicated by dotted line.

the elastic strain wave is reflected back along the length of the second bar. Huněk provides time series plots of the computed contact force for three stiffness penalty parameters  $\alpha_s = 62.5$ , 125 and 500 and numerical results are acceptable in this range, although at  $\alpha_s = 62.5$  the contact forces are somewhat slow to reach the proper value, and with  $\alpha_s = 500$  large oscillations appear in the contact force. This suggests a high sensitivity to penalty parameter magnitude, since there is a relatively small window of penalty parameters (around one order of magnitude) where the general behaviour of the contact forces are comparable to the analytical solution. Compared to the penalty magnitudes used in other methods they are relatively small, with  $\alpha_s = 500$  corresponding to a penalty factor of  $p_s = 1$ .

Presently, we extend this study by further increasing the stiffness penalty para-

meter. Figure 6.5 shows the computed contact force and displacement at nodes A and P for three stiffness penalty analyses with a constant timestep  $\Delta t = 0.0004$  s. The critical time step associated with the system (neglecting the effect of penalty constraints) is  $\Delta t_{\text{crit}} = 0.002$ .

Figure 6.5(a) shows the oscillations that are present in the contact force for  $\alpha_s = 500$ , although this does not seem to cause any serious problems with regards to nodal displacements. Figure 6.5(c) shows the manifestation of time step instability. Note that the displacements are not subject to unbounded exponential growth in this case, as is characteristic of time step instability, since the contact constraints are only applied for very short periods (separation occurs after one or two time steps).

The stability of the middle analysis, with  $\alpha_s = 5000$ , is not so easy to comment on. The numerically computed maximum eigenvalue during contact is  $\lambda_{\text{max}}^{\text{C}} \approx 1.05 \times 10^7$ , giving a critical time step of around  $\Delta t_{\text{crit}}^{\text{C}} = 0.0006$ , suggesting that  $\Delta t$  is marginally sub-critical. However, examination of the total energy of the system reveals a slight increase over the course of the analysis, which is characteristic of instability.

Figure 6.6 shows contact force and displacements for the bipenalty formulation. The penalty ratio is chosen as  $R = 10^6$ , which is equal to the largest eigenvalue of the unpenalised system. There is no evidence of time step instability in the results, and nodal displacements on the contact surface are reasonable, although large amplitude oscillations persist in the contact forces for high values of  $\alpha_s$ . Interestingly, Figure 6.6(a) suggests that with relatively low penalty parameters, the additional constraint offered by the mass penalties can significantly reduce contact force oscillations. For larger values of  $\alpha_s$ , however, the noise is still present.

In order to investigate further, the same problem is repeated with a range of penalty parameter magnitudes, and with several different penalty ratios. A time step of  $\Delta t = 20 \mu\text{s} = 0.01\Delta t_{\text{crit}}$  is used for all analyses. The gap function  $g_n$  should ideally be zero until separation at  $t = 0.6$  s, and hence the root-mean-squared of the gap function between 0–0.6 s gives a measure of the error in constraint imposition. These values are plotted in Figure 6.7. For a pure stiffness penalty approach, time step instability manifests at the larger penalty factors. All bipenalty analyses converge to a constant value as the stiffness penalty is increased.

It should be noted that a problem of this type *cannot* be modelled with mass penalties alone. Contact-impact problems rely on a displacement constraint in order to

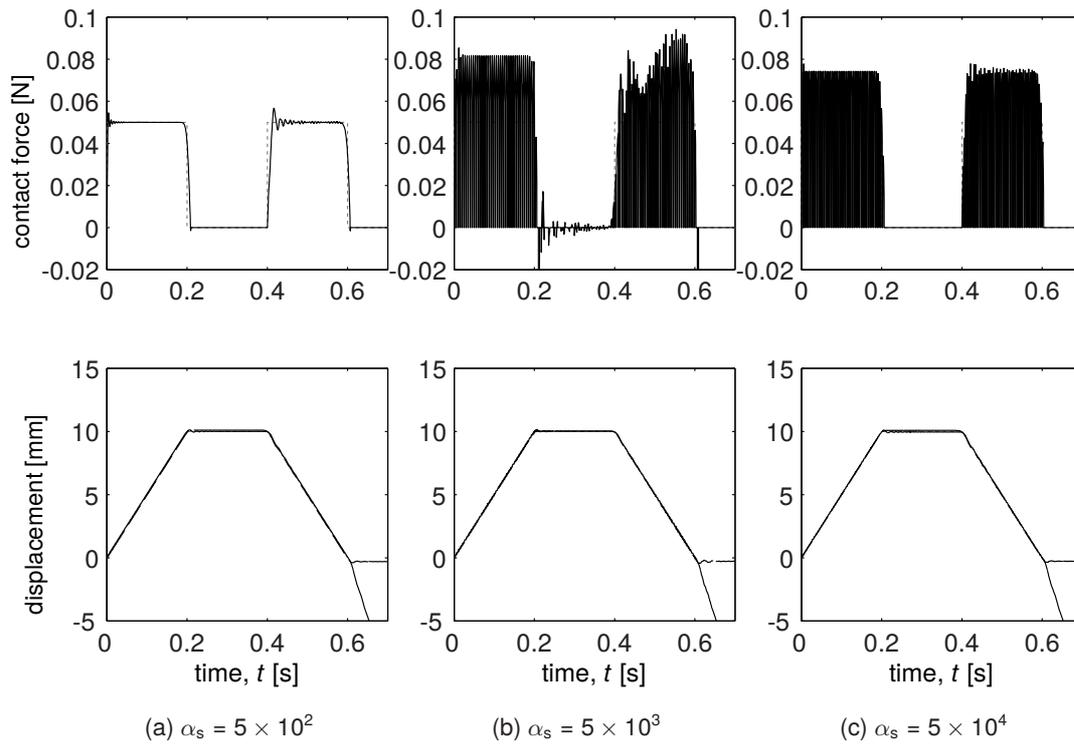


Figure 6.6: Bipenalty method: contact force, and displacement of contacting nodes for increasing stiffness penalty and mass penalty  $\alpha_m = \alpha_s/R_{crit}$ . The analytical solution is indicated by dotted line.

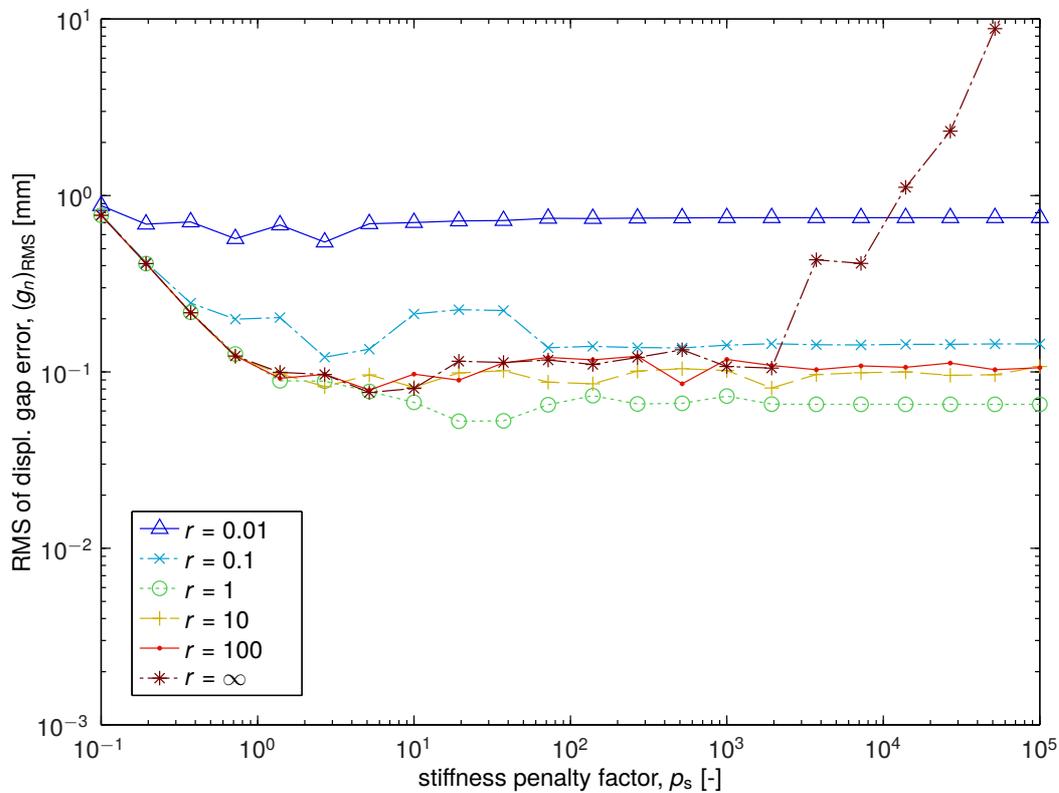


Figure 6.7: Error in the gap function  $g_n$  for a range of penalty ratios.

enforce the kinematic constraints properly. In a bipenalty formulation, contact forces include contributions by both stiffness and mass penalties, as shown by Equation (6.17). Stiffness penalties are activated by displacements that violate the constraint equations, while mass penalties are activated by violations in the nodal accelerations. For the bar problem considered here both bars are travelling with constant velocity at the beginning of the analysis, and since there are no applied forces the acceleration at all nodes is zero. Hence,  $\mathbf{f}^{\text{MP}} = \mathbf{M}^{\text{P}}\ddot{\mathbf{u}} = \mathbf{0}$  and therefore  $\mathbf{f}^{\text{C}} = 0$  for the duration of the analysis (unless stiffness penalties are also included). This explains why, for the tests shown in Figure 6.7, the smallest errors are found with penalty ratios in the range of 1–100, with smaller ratios leading to much larger errors. Some relative acceleration is initially required on the contact surface as contact occurs; in theory, the velocities of the contacting nodes are altered instantaneously, after which the acceleration gap (as well as the displacement and velocity gap) is zero until release. For small ratios, the mass penalty force resists this sudden change in relative velocity by overpowering the stiffness penalty force. It is therefore recommended that a reasonably large penalty ratio be used for contact-impact problems of this type. Typically, we will be using the largest ratio that ensures stability for the time step being employed.

Although the bipenalty method is effective in safeguarding the stability of the analysis, the oscillations in the contact force are still present. Since this has been attributed to the absence of a velocity gap constraint [1], we will finally look at the effect of damping penalties on the results. These are activated by a velocity gap across the contact surface, and as such may help to damp out the noise. Figure 6.8 shows total contact force, displacement at contacting nodes, and the normal gap function for four different values of ‘damping factor’  $D$ , where

$$\mathbf{C}^{\text{P}} = D\mathbf{K}^{\text{P}} \quad (6.18)$$

Both stiffness and mass penalties are also used with a relatively large stiffness penalty factor of  $p_s = 10^5$  and penalty ratio  $R = 10^6 \text{ s}^{-2}$ , which is equal to the maximum eigenvalue of the unconstrained system. Damping penalties help to enforce continuity of velocities across the contact interface, and because of this they are very effective at reducing noise in the contact forces. However, it is difficult to determine what the ideal damping factor is for a given analysis; small differences in  $D$  can often lead to

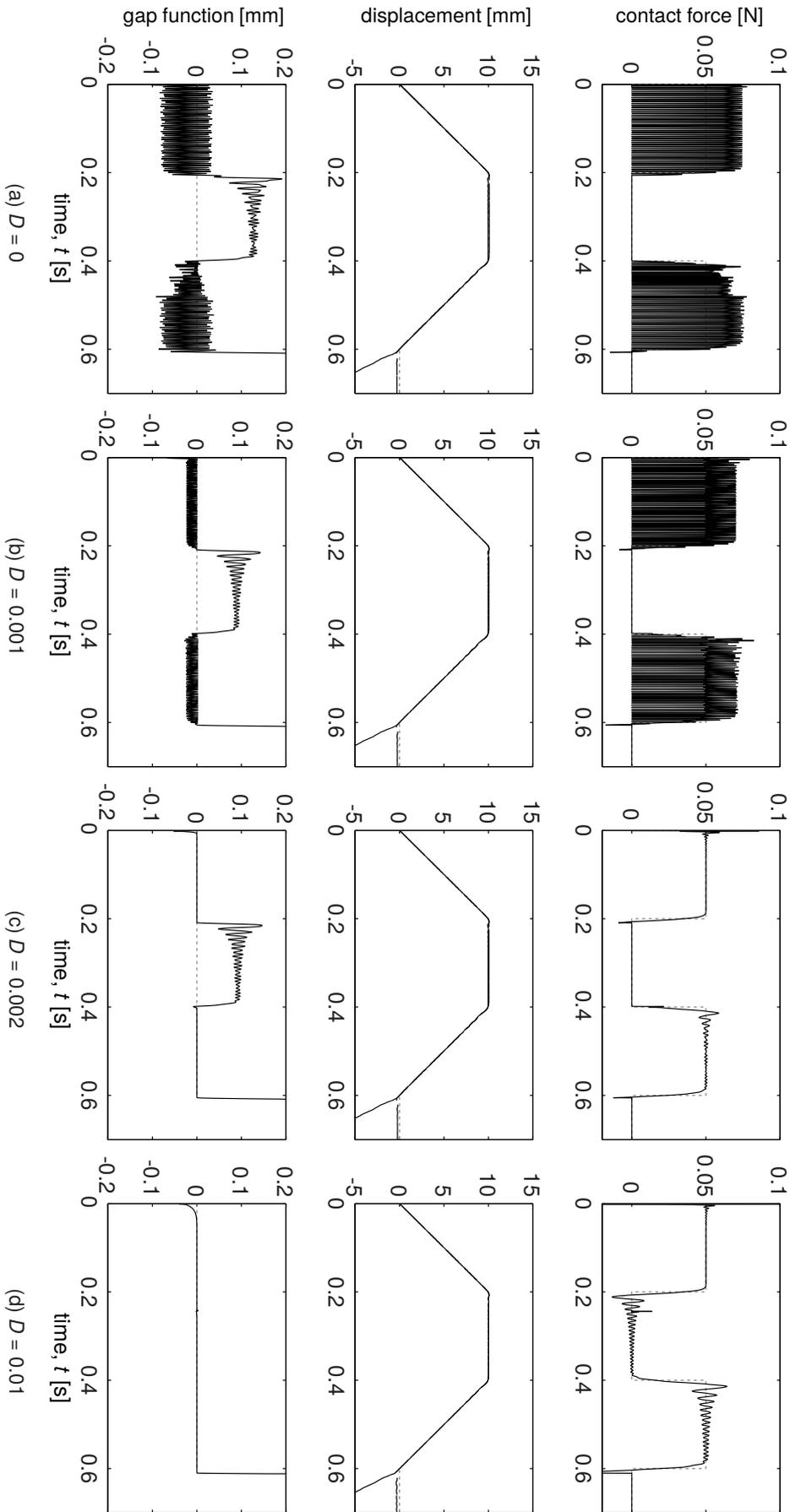


Figure 6.8: Bipenalty method with damping: contact force, displacement and displacement gap of contacting nodes for increasing damping factor. Penalty parameters are calculated from  $\rho_s = 10^5$  and  $R = \lambda_{\max}^6 = 10^6 \text{ s}^{-2}$ . The analytical solution is indicated by dotted line.

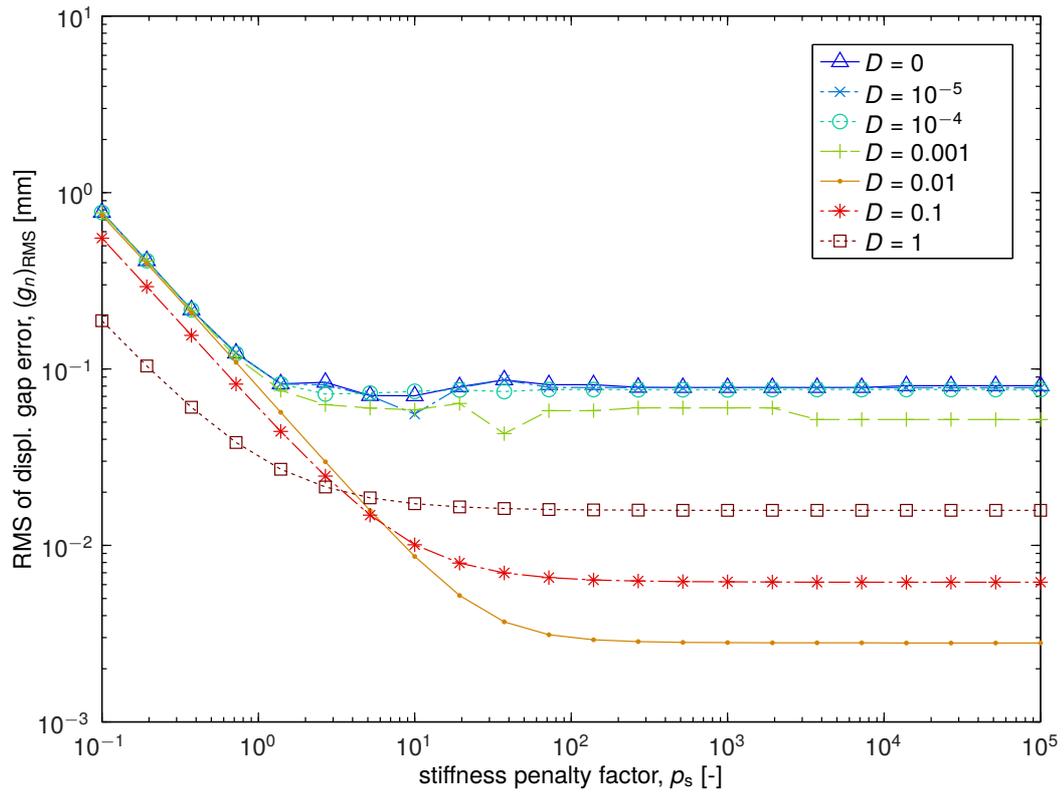


Figure 6.9: Error in the gap function  $g_n$  for a range of damping factors.

significant changes in the results. For example, in Figures 6.8(a)–(c) the two bodies separate at around  $t = 0.2$ , leading to a zero contact force and a small positive displacement gap. Figure 6.8(d) reveals that that for  $D = 0.01$  a very small negative displacement gap is preserved throughout, with contact forces oscillating around zero. It is debatable which analysis captures the behaviour more accurately.

Figure 6.9 examines the effect of the damping ratio in a manner similar to the investigation of penalty ratio shown in Figure 6.7. Once again, results converge to a constant value for large stiffness penalty factors, although with damping penalties included the RMS of the error is generally a lot smaller due to the reduction in noise.

The value of the damping penalty  $D$ , however, remains very arbitrary. Since there is no obvious physical basis upon which to base an estimate, and because increasing the value of  $D$  does not cause convergence towards the exact solution, damping penalties should be used with care. After developing guidelines to simplify the choice of penalty parameters  $\alpha_s$  and  $\alpha_m$ , adding another arbitrary factor that must be chosen using trial and error would appear to be something of a step backwards. On the other hand, the results shown above do display promising characteristics, and the use of

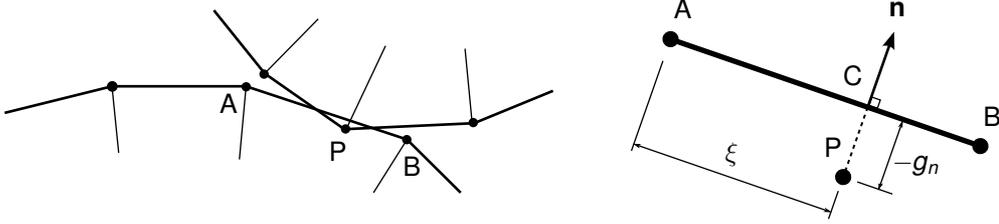


Figure 6.10: Contact between 2D discretised bodies.

damping penalties may be useful in situations where the trial and error approach is viable. For general adoption, however, clear and generally applicable guidelines should be investigated that might aid in the choice of  $D$ .

## 6.4 Contact-impact in two dimensions

In 2D, contact problems become more complicated. The node-to-node formulation can only be extended to 2D if the surface mesh of the bodies is identical and no sliding occurs on the contact surface. A more general solution is the node-to-surface formulation, which will be developed in this section.

### 6.4.1 Finite element implementation

Figure 6.10 shows a 2D FE discretisation in which two bodies are overlapping. Node  $P$  is the penetrating node, while nodes  $A$  and  $B$  are the nodes that define the element edge which has been penetrated. The ‘contact point’, denoted by  $C$ , is the nearest point on the penetrated surface, with position

$$\mathbf{x}^C = \mathbf{x}^A + \xi(\mathbf{x}^B - \mathbf{x}^A) \quad (6.19)$$

where  $\mathbf{x}$  denotes a position vector in the  $x$ - $y$  co-ordinate system and  $\xi$  is the normalised distance from node  $A$  to the contact point, that is,

$$\xi = \frac{|\mathbf{x}^C - \mathbf{x}^A|}{|\mathbf{x}^B - \mathbf{x}^A|} \quad (6.20)$$

This value can be calculated from the positions of nodes  $A$ ,  $B$  and  $P$  via

$$\xi = \frac{(\mathbf{x}^P - \mathbf{x}^A) \cdot (\mathbf{x}^B - \mathbf{x}^A)}{|\mathbf{x}^B - \mathbf{x}^A|^2} \quad (6.21)$$

The gap function  $g_n$  is given by

$$\begin{aligned}
 g_n &= -(\mathbf{x}^C - \mathbf{x}^P) \cdot \mathbf{n} \\
 &= -[(1 - \xi)\mathbf{x}^A + \xi\mathbf{x}^B - \mathbf{x}^P] \cdot \mathbf{n} \\
 &= -[(1 - \xi)\mathbf{u}^A + \xi\mathbf{u}^B - \mathbf{u}^P] \cdot \mathbf{n} - \mathbf{G} \cdot \mathbf{n}
 \end{aligned} \tag{6.22}$$

where  $\mathbf{G}$  is the initial gap between node P and the contact point C. This expression is derived from Equation (6.2), with the additional assumption that  $\mathbf{n}^P = -\mathbf{n}^A$  (since the outward unit normal at nodal points is in general discontinuous for discretisations of curved surfaces). Note that since the contact point depends on displacements, the initial gap is no longer constant in time. The velocity gap is given by

$$\gamma_n = -[(1 - \xi)\mathbf{v}^A + \xi\mathbf{v}^B - \mathbf{v}^P] \cdot \mathbf{n} \tag{6.23}$$

The two-dimensional constraint equations can therefore be written

$$[(1 - \xi)\mathbf{u}^A + \xi\mathbf{u}^B - \mathbf{u}^P] \cdot \mathbf{n} = -\mathbf{G} \cdot \mathbf{n} \quad \text{for } g_n < 0 \tag{6.24}$$

$$[(1 - \xi)\mathbf{v}^A + \xi\mathbf{v}^B - \mathbf{v}^P] \cdot \mathbf{n} = 0 \quad \text{for } g_n < 0 \tag{6.25}$$

However, in two dimensions it is not possible to write these equations directly in the form  $\mathbf{h} = \mathbf{S}\mathbf{u} - \mathbf{q}$ , as was possible for the simple 1D constraints in Section 6.3.1. Instead, we will derive stiffness and mass penalty matrices from the weak form of the contact constraint equations. The process is similar in many ways to the interface element formulation described in Section 5.1. First, we choose the normal vector  $\mathbf{n}$  as the basis for a local co-ordinate system. The displacements of the contact nodes in the direction of the contact surface outward normal are denoted by

$$\mathbf{d} = [d_n^A, d_n^B, d_n^P]^T \tag{6.26}$$

and the gap function is given by

$$g_n = \mathbf{B}\mathbf{d} + G_n \tag{6.27}$$

where  $\mathbf{B} = [\xi - 1, -\xi, 1]$  and  $G_n$  is the initial gap between node P and the contact point in the local co-ordinate system.

We then define a relationship between contact force and displacement gap function,

$$f_n = \alpha_s g_n \quad \text{if } g_n < 0 \quad (6.28)$$

Similar to the traction–relative displacement relationship used for interface elements, this equation adds a corrective force proportional to the magnitude of constraint violation. However, since contact is assumed to act at a single point, a force (units: N) is generated instead of traction (N/m<sup>2</sup>). The work done by the contact connection is given by  $U = f_n g_n / 2$ , which becomes, after substitution of Equations (6.28) and (6.27),

$$\mathcal{U} = \frac{\alpha_s}{2} (\mathbf{B}\mathbf{d} + G_n)^2 \quad (6.29)$$

Minimisation of potential energy then leads to

$$\mathbf{K}\mathbf{d} = \mathbf{q} \quad (6.30)$$

where the stiffness matrix is given by

$$\mathbf{K} = \alpha_s \mathbf{B}^T \mathbf{B} = \alpha_s \begin{bmatrix} (\xi - 1)^2 & \xi(1 - \xi) & \xi - 1 \\ \xi(1 - \xi) & \xi^2 & -\xi \\ \xi - 1 & -\xi & 1 \end{bmatrix} \quad (6.31)$$

and the initial gap vector given by  $\mathbf{q} = \alpha_s G_n \mathbf{B}^T$ .

In the local co-ordinate system of the contact surface the velocity gap is given by

$$\gamma_n = \mathbf{B}\mathbf{v} \quad (6.32)$$

where  $\mathbf{v} = [v_n^A, v_n^B, v_n^P]^T$  and the work done due to the mass penalties is given by

$$\mathcal{T} = \frac{\alpha_m}{2} (\mathbf{B}\mathbf{v})^2 \quad (6.33)$$

Minimisation of total energy ( $\mathcal{U} + \mathcal{T}$ ) then leads to

$$\mathbf{K}\mathbf{d} + \mathbf{M}\dot{\mathbf{v}} = \mathbf{q} \quad (6.34)$$

where the mass penalty matrix is

$$\mathbf{M} = \alpha_m \mathbf{B}^T \mathbf{B} = \frac{1}{R} \mathbf{K} \quad (6.35)$$

The 2D bipenalty contact formulation therefore consists of computing  $\mathbf{K}$ ,  $\mathbf{M}$  and  $\mathbf{q}$  for each penetrating node. These are then rotated into the global  $x$ - $y$  co-ordinate system and assembled to form the penalty matrices  $\mathbf{K}^P$ ,  $\mathbf{M}^P$  and penalty force vector  $\mathbf{f}^P$ .

### 6.4.2 Contact detection

The actual process of detecting contact is largely irrelevant to the method of constraint imposition described above, but a brief overview of the methods used in the developed code will be given.

At the beginning of an analysis, possible contact surfaces are identified. Contact detection at each time step is then a two stage process. Firstly, elements which are potential candidates for contact are identified. Then, individual nodes are checked, and, if found to be penetrating a contact surface, projected back onto a suitable point on that surface. To complete the first stage, every element on a contact surface is given a bounding circle, which contains the element for the duration of the analysis. If any of these bounding circles are overlapping at a given time step then the element pair is considered to be a candidate for penetration. In the second stage, each pair of potentially contacting elements is considered in turn. If a node from the first lies inside the bounds of the second, a contact connection is created. The process is then reversed so that nodes from both elements are checked.

When a node is found to be penetrating an element/surface, the choice of contact point (point C in Figure 6.10) is not always obvious, and therefore must be handled with some care. Figure 6.11 shows three nodes (P, Q and R) penetrating a surface. In the case of node P there is only one possible contact point, which is easily identified. Node Q can be projected at right angled to two different points on the surface. In this case, the contact point closest to the penetrating node is chosen. Finally, node R cannot be projected onto any point on the surface. In this case, a node-to-node connection is established with the closest surface node. This ensures that even if there are multiple possible projections, or no projections at all, a sensible contact

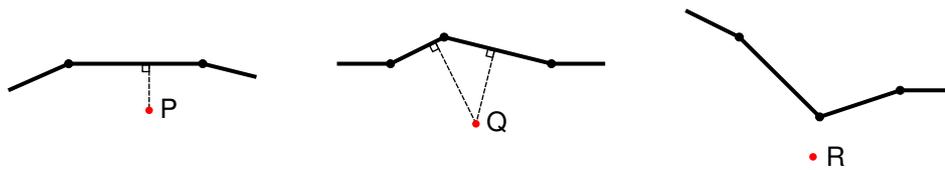


Figure 6.11: Nodes  $P$ ,  $Q$  and  $R$  penetrating discretised 2D surfaces.

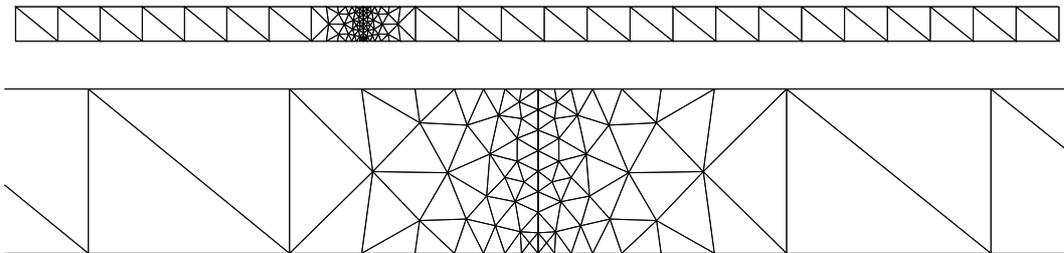


Figure 6.12: Complete 2D mesh for the bar impact problem (top) and detail of mesh around contact surface (bottom)

point is always selected. When non-physical penetrations are small, the first case is by far the most common occurrence.

### 6.4.3 Numerical example: Huněk bar in 2D

For verification of the given formulation, the 1D Huněk bar impact test is repeated in 2D. While the analytical solutions assume a 1D theory of wave propagation, approximate contact forces, displacements and duration of contact should be approximately the same when the bars are modelled with 2D finite elements. The material properties are the same as were used in Section 6.3.2, with a Poisson's ratio  $\nu = 0$ . The mesh is shown in Figure 6.12, and the bar is assumed to have a  $1 \times 1$  m square cross-section. The stable time step is estimated using the maximum element eigenvalue with a safety factor of 0.5 (that is,  $\Delta t = 1/\max_e(\omega_{\max}^e) \approx 2.43 \times 10^{-4}$  s).

Time histories of total contact force on the impacted surface, and the mean displacement on both contact surfaces, are shown in Figure 6.13 for three different penalty formulations, each with  $p_s = 1$ . Penalty magnitudes therefore roughly correspond to  $\alpha_s = 500$  for the 1D analysis. Mass penalty parameters are calculated by taking  $R$  equal to the maximum eigenvalue of the unpenalised system. The main features of the collision are correct, with the contact force in all cases oscillating around 0.05 N. As with 1D contact, the damping penalties are effective in damping out high frequency noise in the solution, although low frequency oscillations are present in all

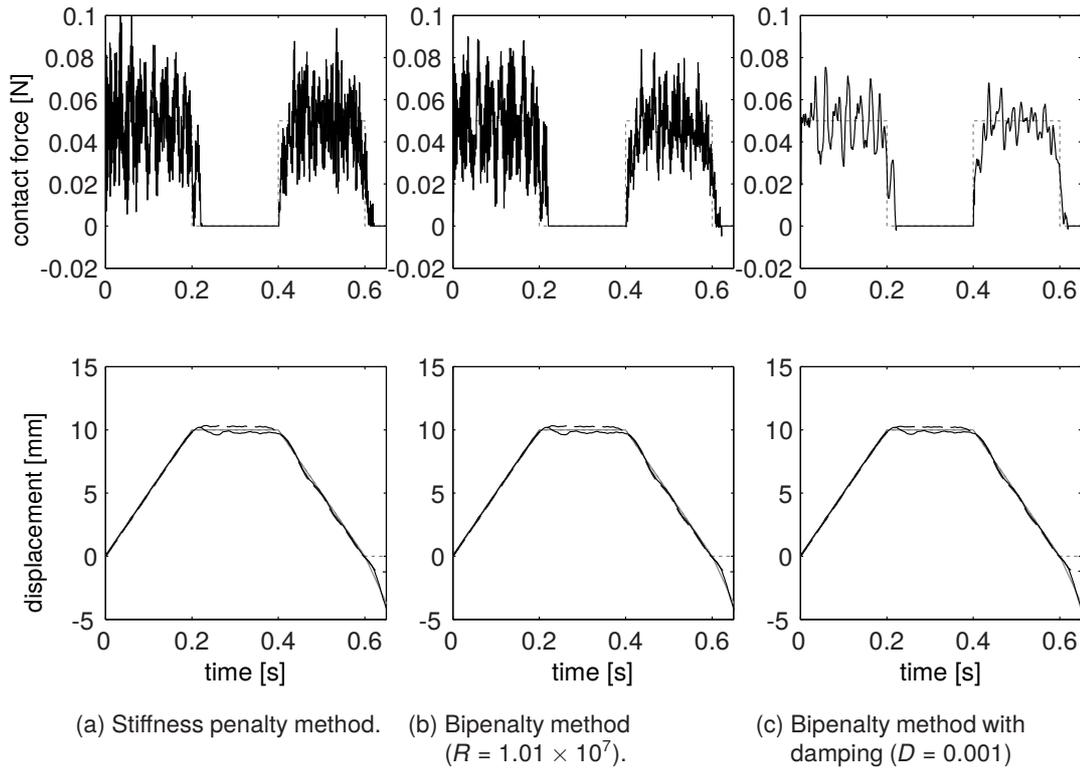


Figure 6.13: Total horizontal contact force and mean horizontal displacement on contact surfaces for the bar impact problem modelled in 2D.

cases. This is partly due to lateral wave propagation which is not taken into account in 1D theory. The displacements on the contact surface show good agreement with the predicted solution.

#### 6.4.4 Numerical example: Collision between dissimilar cylinders

Another contact-impact problem is shown in Figure 6.14. One cylindrical body is at rest, while another impacts its bottom face with a velocity of 2 m/s in the diagonally upwards-right direction. Both bodies have mass density  $\rho = 0.1 \text{ kg/m}^3$  and Poisson's ratio  $\nu = 0.2$  and plane strain is assumed, but body 1 has a Young's modulus of  $E_1 = 1000 \text{ N/m}^2$  while body 2 is considerably less stiff, with Young's modulus  $E_2 = 100 \text{ N/m}^2$ . The diameter of both cylinders is 2 m, and there are no fixed supports and no external forces applied to either of the bodies.

The deformation of the contact surface during the collision is depicted in Figure 6.15. The bipenalty method is used with a stiffness penalty factor of  $p_s = 1$ . The time step  $\Delta t \approx 9.46 \times 10^{-5}$  is calculated from the maximum elemental eigenvalue,

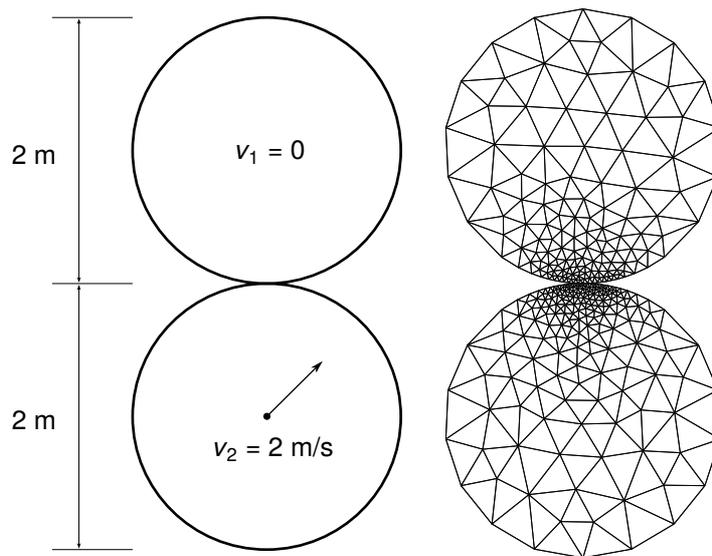


Figure 6.14: Colliding cylinder problem, including FE mesh.

with a safety factor of 0.5, and the penalty ratio  $R \approx 4.02 \times 10^8$  is calculated directly from the time step  $\Delta t$  (with a safety factor of 0.9). Due to the horizontal movement of the bottom cylinder, and the fact that friction forces are neglected, there is considerable sliding along the contact surface. Larger deformations occur in the bottom cylinder due to its lower elastic modulus.

The same analysis was performed with the stiffness and bipenalty methods for a range of penalty parameters, with and without damping penalties included. Figure 6.16 gives an indication of how each of the different methods performed, by showing the maximum penetration of any node during the analysis. Results are not included if the displacement of any node exceeds 1 m at any point; since displacements should not approach such values in the first 0.18 s, numerical problems are assumed to have occurred in these cases. For small parameters there is not much difference between the methods. For higher values of  $p_s$ , both the stiffness and bipenalty methods fail to supply reasonable results.

The failure of the pure stiffness penalty approach can be explained by time step instability occurring due to the penalty contact elements, but poor regularisation of the contact constraints is also an important factor. The bipenalty formulations that include damping show that although penetration does not tend to zero for large parameters, there is no catastrophic failure due to time step instability, even for large penalty factors. Since damping penalties do not effect time step stability, this implies

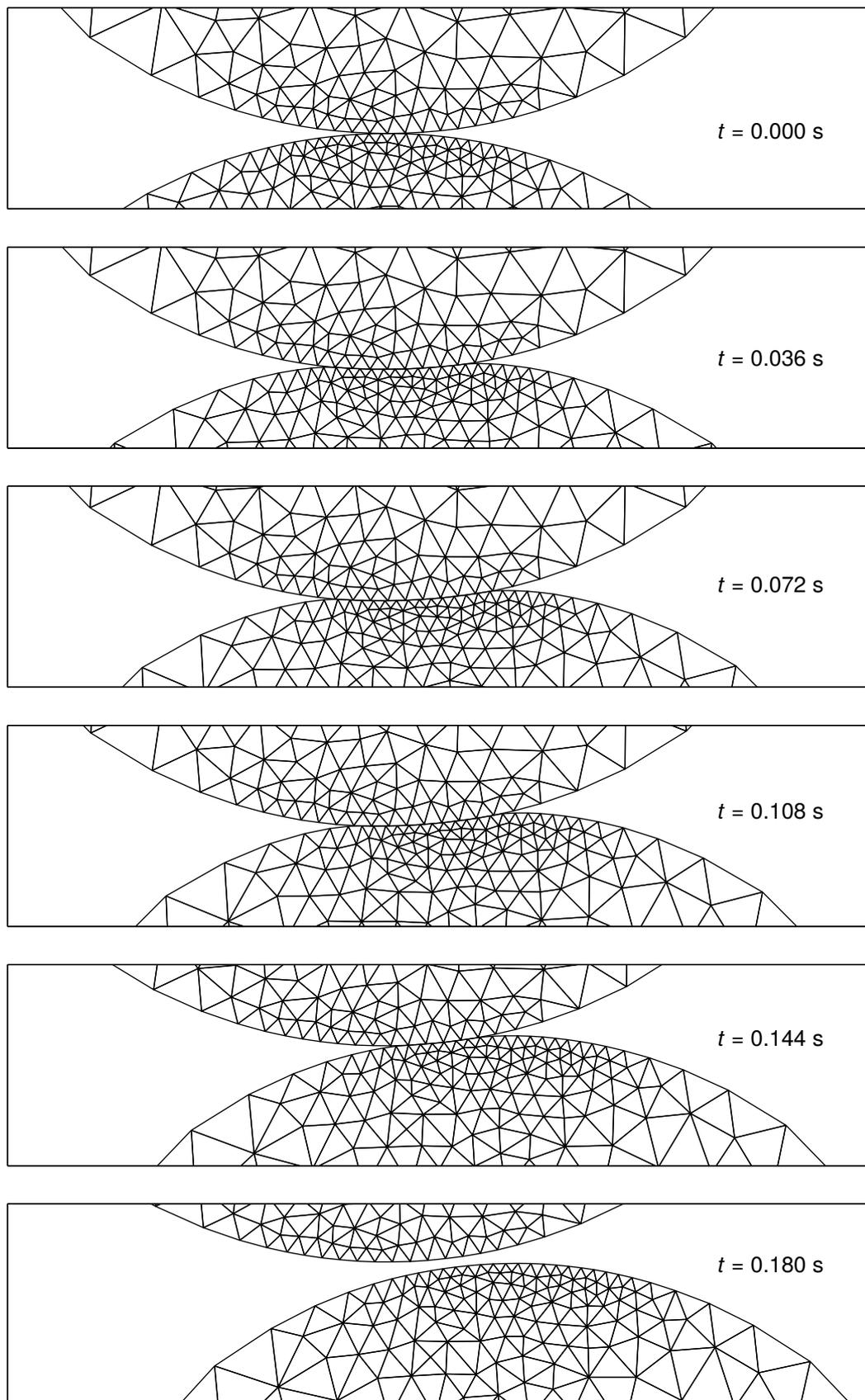


Figure 6.15: Collision of dissimilar cylinders.

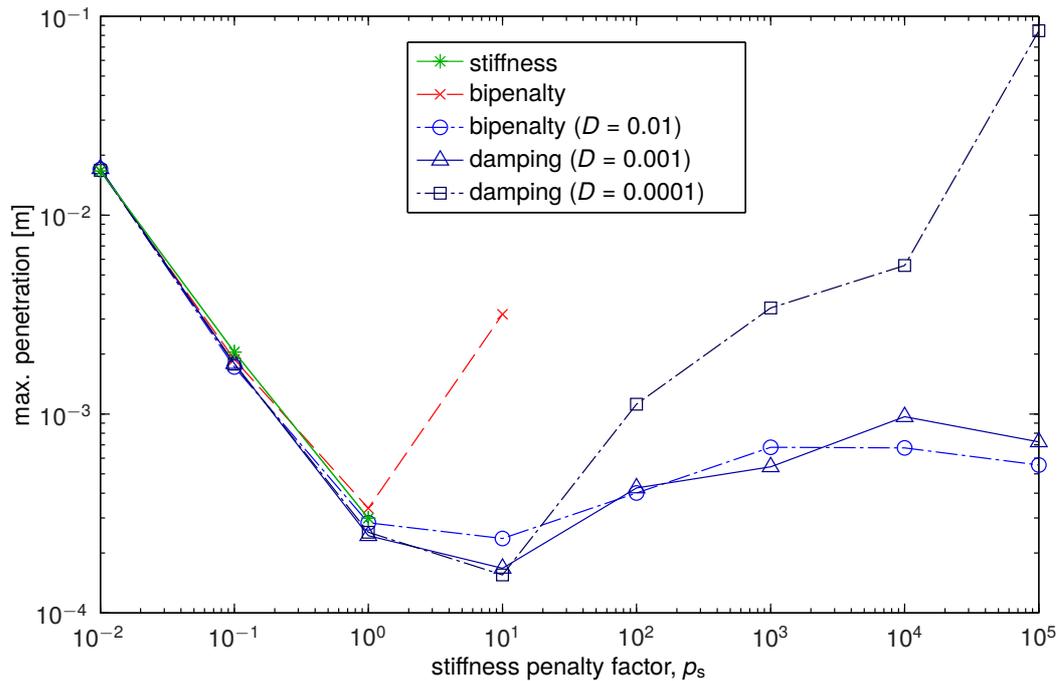


Figure 6.16: Maximum penetration of a penetrating node for increasing penalty magnitude.

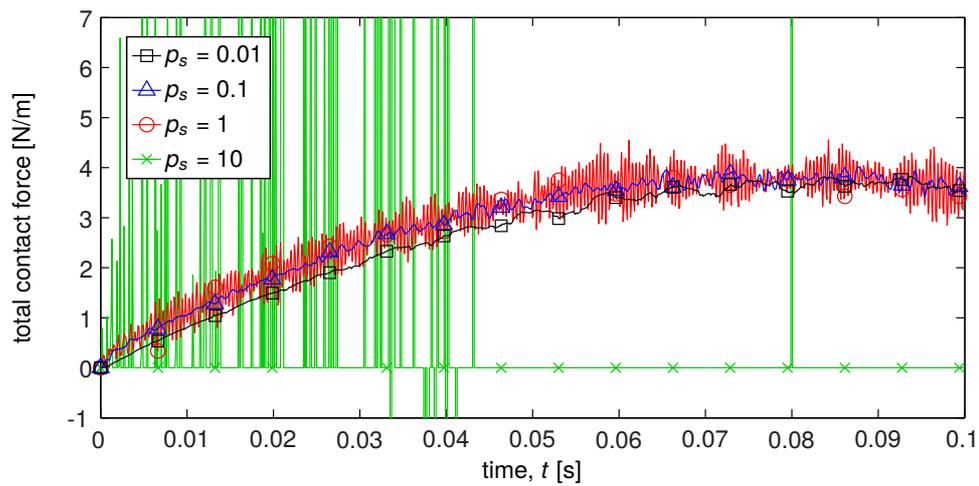
that the bipenalty formulation fails due to a lack of regularisation, creating noise in the solution. When damping penalties are included this noise is reduced (damped out) to varying degrees.

Sensitivity to penalty parameter is therefore high for all formulations; very large parameters lead to large, suddenly applied forces, which lead to spurious oscillations in the solution. Penalty factors in the region of 1–10 appear to be optimal in terms of minimising penetration. Crucially, this is also the region where stiffness penalty formulations begin to fail due to time step instability. The bipenalty method eliminates this possibility, but parameters must still be chosen carefully for best performance. Damping penalties lead to a slight increases in performance, and extend the range of acceptable parameter magnitudes to some degree, but still do not constitute a general fix; they damp spurious oscillations to some degree, but do not solve the problem of poor constraint regularisation.

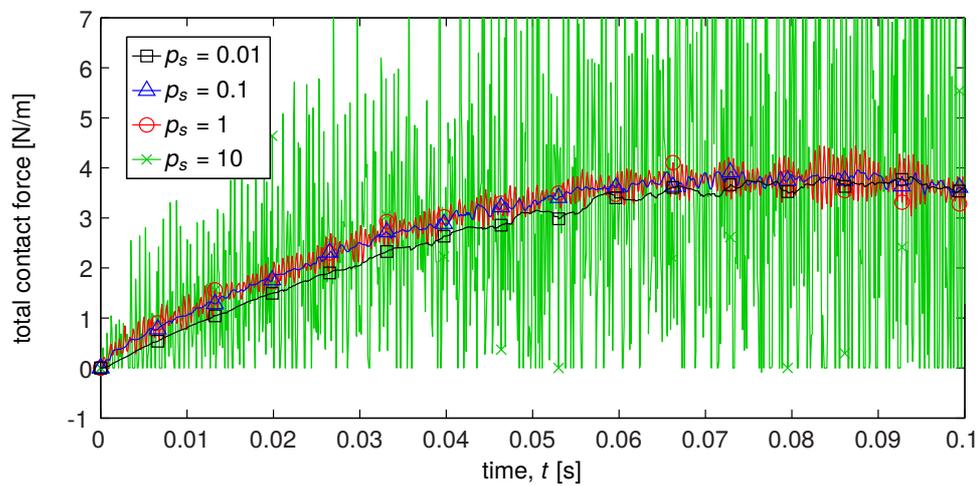
Hallquist et al. [37] give an equation that can be used to calculate an appropriate penalty stiffness that depends on the bulk modulus, volume and area of an element, as well as a ‘scale factor’ to protect against time step instability. The bipenalty method would eliminate the need for the scale factor, but for good performance it is still necessary to use a reasonable penalty stiffness (one which takes into account the

properties of the contacting bodies) for explicit contact-impact problems.

Figure 6.17 shows that total contact force during the initial period of contact between the two cylinders. At  $p_s = 0.01$  for all analyses the contact force is underestimated, leading to large penetration (as seen in Figure 6.16). As the penalty factor is increased, the contact force converges to a solution, but oscillations around that solution also increase. The oscillations are most pronounced for the stiffness penalty method, with the bipenalty method showing slightly reduced noise. With damping penalties included, oscillations are much reduced at  $p_s = 10$ , although the problem inevitably manifests for larger penalty magnitudes, even with the inclusion of damping penalties.



(a) Stiffness penalty method



(b) Bipenalty method

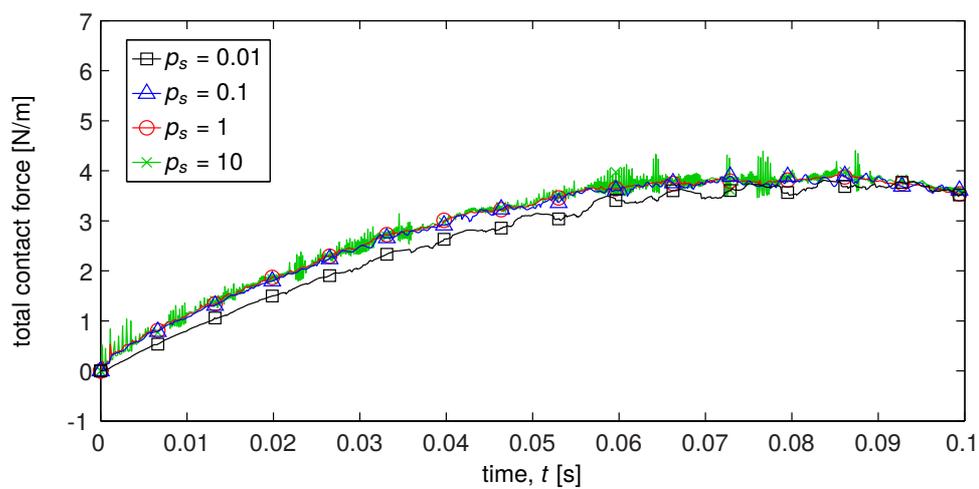
(c) Bipenalty method with damping ( $D = 0.0001$ )

Figure 6.17: Total vertical contact force per metre length for the initial 0.1 s of the analysis.

## Chapter 7

# Conclusions and future research

Penalty methods provide a simple and versatile framework for the imposition of constraints in a wide range of common problem types. They are valued for their conceptual simplicity and the ease with which they can be robustly implemented in finite element codes. They are often favoured in explicit dynamic codes for their computation efficiency. However, there is a certain amount of uncertainty involved with regards to the selection of suitable penalty parameters, since no well-defined optimum values can be identified. In static analysis, one may select parameters that are too low (leading to poor enforcement) or too high (leading to ill-conditioning), but in explicit dynamics a third complication is introduced: the effect on the critical time step. Traditional stiffness penalty parameters tend to decrease the critical time step, although it is hard to predict the extent of this effect; safety factors, whose values are generally determined through trial and error, are common.

The bipenalty method removes uncertainty from the process of applying penalty constraints in explicit dynamics. Beginning with a general bipenalty formulation that allows for any number of arbitrary constraints, this thesis provides mathematical proof of the precise effects that the constraints have on the eigenvalues (and therefore associated critical time step) of a finite element system, and shows that they can be carefully controlled through manipulation of the penalty ratio. This is a very desirable feature of the bipenalty method from the point of view of an analyst. It allows the use of the largest possible critical time step, based on the standard criteria; namely, element size and material properties (as dictated by the CFL condition). It means that an arbitrary parameter used in a contact algorithm or the constitutive

model of a cohesive surface does not have any effect on the critical time step. The bipenalty method certainly helps prevent time step instability, but also has the potential to prevent analysts from being overly cautious when selecting a time step due to uncertainty about the effects that the traditional penalty method might have.

In terms of the accuracy of constraint imposition, the traditional stiffness penalty method is usually superior when compared to the bipenalty method *with penalty parameters of similar magnitude*. However, in explicit dynamics this comparison does not tell the whole story. Traditional penalty methods have a restrictive upper bound on the values that can be used due to their effect on stability, whereas the bipenalty method does not. This means that ill-conditioning once again provides the upper bound for parameter magnitude, as it does in static analysis. In a cohesive surface formulation this is especially important, since it allows for a more realistic constitutive behaviour during the initial elastic region of deformation, before damage has occurred. Previously, analysts would have to use small stiffness penalties (which may introduce a significant amount of artificial compliance in the continuum), introduce the cohesive surfaces only after the onset of damage (which greatly increases the complexity and cost of the analysis, due to constantly changing system size and memory requirements) or decrease the time step of the analysis. The bipenalty method provides a way of using large parameters while keeping the number of DOF constant throughout the analysis; the critical time step of the analysis is not affected.

The bipenalty method possesses some significant disadvantages, however. Potentially the most problematic is that the method results in a non-diagonal mass matrix for the types of constraints most commonly tackled with penalty methods (i.e., constraints that involve multiple DOF). Whether or not this computational overhead is prohibitively large depends entirely on the problem being addressed. For example, if contact occurs only at specific points on the boundary of a large structure, the bipenalty method is certainly a worthy candidate. On the other hand, thin structures modelled with shell elements may have a very large proportion of DOF on a contact surface, and in such extreme cases the potential increase in time step is very likely be offset by the more costly solution procedure. Similar considerations should be made for crack propagation problems using cohesive surfaces; if interface elements must be placed throughout an entire structure then the computational cost of the solution

scheme will rise dramatically.

In the field of contact-impact, the fact that the bipenalty method allows for very large penalty parameters is less useful. Here, very large penalty parameters lead to poor regularisation of the discontinuous contact constraints; large parameters do not necessarily lead to high performance. Of course, the inclusion of mass penalties does guard against time step instability, which is often difficult to detect in contact-impact problems. Furthermore, the inclusion of damping penalties does appear to help damp out the spurious contact force oscillations that are characteristic of dynamic contact problems. There is no significant computation overhead to including damping penalties if mass penalties are already being employed.

## 7.1 Future research

In the field of interface elements and crack propagation this thesis has utilised only a very simple bilinear constitutive model. In order to properly assess the performance of the bipenalty method in this area, research could be undertaken to develop a more sophisticated bipenalty formulation. For contact-impact problems, the use of damping penalties appears to yield promising results in terms of performance. In the trade-off between excessive penetration and poor regularisation, there is currently a fairly small window when penalty parameters are most effective; a window the bipenalty method with damping may enlarge. However, it is not yet clear how to reliably select damping penalty parameters for best performance. Possible areas for research include a method for choosing a damping penalty factor based on material properties, or directly from the stiffness and mass penalty parameter being used for a given constraint.

The work presented in this thesis considers only linear elastic material behaviour; although nonlinear constraints are explored, the material models are quite simplistic, and large deformations are not considered. For the bipenalty method to be widely accepted both stability and accuracy would need to be further investigated in the context of practical nonlinear analysis. This would likely require the implementation of bipenalty techniques into existing codes. Although every attempt has been made to ensure that the bipenalty method as derived in this work is as generally applicable as possible, the performance of the method in highly nonlinear, dynamic analyses is

yet to be fully analysed.

Perhaps the most useful research that could be undertaken at this point is related to the computational efficiency of a bipenalty constrained system with non-diagonal mass matrix terms. Even within the framework of the central difference method, there are many algorithms that can be employed to compute a solution, and the overall efficiency is very much dependent on which algorithm is used, and how the nodal displacements at the free and penalty-constrained DOF are calculated. Further insight in this area would allow guidance to be given regarding how many DOF can be constrained (relative to the size of the complete system) before the bipenalty method is no longer a computationally efficient option. This would require a comprehensive assessment of available algorithms, and possibly the development of new schemes that are designed around the bipenalty formulation.

Even if the computational efficiency of the bipenalty method is too poor for general adoption, some of the work presented here may perhaps inform the analysis of penalty methods in general. The selection of penalty parameters is often challenging, not just because of stability but also because of how difficult it is to predict the resulting error ahead of time. Further research that builds on the ideas presented in this work could lead to clear and robust algorithms for penalty parameter selection, whether the penalties are applied to the stiffness, mass or damping matrix of a system.

# Bibliography

- [1] F. Armero and E. Petocz. Formulation and analysis of conserving algorithms for frictionless dynamic contact/impact problems. *Computer Methods in Applied Mechanics and Engineering*, 158(3-4):269–300, 1998.
- [2] N. Asano. An approximate hybrid type of virtual work principle for two elast-  
oimpact contact bodies. *Bulletin of the JSME*, 26(221):1849–1856, 1983.
- [3] N. Asano. A penalty function type of virtual work principle for impact contact  
problems of two bodies. *Bulletin of the JSME*, 29(257):3701–3709, 1986.
- [4] N. Asano. A virtual work principle using penalty function method for impact  
contact problems of two bodies. *Bulletin of the JSME*, 29(249):731–736, 1986.
- [5] H. Askes, M. Carames-Saddler, and A. Rodriguez-Ferran. Bipenalty method  
for time domain computational dynamics. *Proceedings of the Royal Society A:  
Mathematical, Physical and Engineering Sciences*, 466(2117):1389–1408, 2009.
- [6] H. Askes, D. C. D. Nguyen, and A. Tyas. Increasing the critical time step:  
micro-inertia, inertia penalties and mass scaling. *Computational Mechanics*,  
47(6):657–667, 2011.
- [7] I. Babuška. The finite element method with penalty. *Mathematics of Computa-  
tion*, 27(122):221–228, 1973.
- [8] G. I. Barenblatt. The mathematical theory of equilibrium cracks in brittle frac-  
ture. *Advances in Applied Mechanics*, 7:55–129, 1962.
- [9] K.-J. Bathe. *Finite Element Procedures*. Prentice Hall, Englewood Cliffs, NJ,  
1996.
- [10] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal  
remeshing. *International Journal for Numerical Methods in Engineering*, 45:601–  
620, 1999.
- [11] T. Belytschko, Y. Krongauz, and D. Organ. Meshless methods: an overview and  
recent developments. *Computer Methods in Applied Mechanics and Engineering*,  
139:3–47, 1996.
- [12] T. Belytschko, W. K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua  
and Structures*. Wiley, New York, 2001.
- [13] T. Belytschko, Y. Lu, and L. Gu. Crack propagation by element-free Galerkin  
methods. *Engineering Fracture Mechanics*, 51(2):295–315, 1995.

- [14] T. Belytschko and M. O. Neal. Contact-impact by the pinball algorithm with penalty and Lagrangian methods. *International Journal for Numerical Methods in Engineering*, 31(3):547–572, 1991.
- [15] D. J. Benson and J. O. Hallquist. A single surface contact algorithm for the post-buckling analysis of shell structures. *Computer Methods in Applied Mechanics and Engineering*, 78:141–163, 1990.
- [16] G. E. Blandford, A. R. Ingraffea, and J. A. Liggett. Two-dimensional stress intensity factor computations using the boundary element method. *International Journal for Numerical Methods in Engineering*, 17(3):387–404, 1981.
- [17] N. G. Bourago and V. N. Kukudzhanov. A review of contact algorithms. *Mechanics of Solids*, 40(1):35–71, 2005.
- [18] G. Camacho and M. Ortiz. Computational modelling of impact damage in brittle materials. *International Journal of Solids and Structures*, 33(20-22):2899–2938, 1996.
- [19] N. J. Carpenter, R. L. Taylor, and M. G. Katona. Lagrange constraints for transient finite element surface contact. *International Journal for Numerical Methods in Engineering*, 32(1):103–128, 1991.
- [20] C. Chen, E. Pan, and B. Amadei. Fracture mechanics analysis of cracked discs of anisotropic rock using the boundary element method. *International Journal of Rock Mechanics and Mining Sciences*, 35(2):195–218, 1998.
- [21] A. Combescure, A. Gravouil, D. Grégoire, and J. Réthoré. X-FEM a good candidate for energy conservation in simulation of brittle dynamic crack propagation. *Computer Methods in Applied Mechanics and Engineering*, 197(5):309–318, 2008.
- [22] R. D. Cook, D. S. Malkus, and M. E. Plesha. *Concepts and Applications of Finite Element Analysis*. Wiley, 3rd edition, 1989.
- [23] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society*, 1943.
- [24] R. Courant, K. Friedrichs, and H. Lewy. Ueber die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, 100(1):32–74, 1928.
- [25] R. Courant, K. Friedrichs, and H. Lewy. On the partial difference equations of mathematical physics. *IBM Journal of Research and Development*, 11(2):215–234, 1967.
- [26] D. Dugdale. Yielding of steel sheets containing slits. *Journal of the Mechanics and Physics of Solids*, 8(2):100–104, 1960.
- [27] M. Elices, G. Guinea, J. Gómez, and J. Planas. The cohesive zone model: advantages, limitations and challenges. *Engineering Fracture Mechanics*, 69(2):137–163, 2002.

- [28] H. D. Espinosa and P. D. Zavattieri. A grain level model for the study of failure initiation and evolution in polycrystalline brittle materials. Part I: Theory and numerical implementation. *Mechanics of Materials*, 35(3-6):333–364, 2003.
- [29] C. A. Felippa. Error analysis of penalty function techniques for constraint definition in linear algebraic systems. *International Journal for Numerical Methods in Engineering*, 11(4):709–728, 1977.
- [30] C. A. Felippa. Iterative procedures for improving penalty function solutions of algebraic systems. *International Journal for Numerical Methods in Engineering*, 12(5):821–836, 1978.
- [31] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.
- [32] G. Gilardi. Literature survey of contact dynamics modelling. *Mechanism and Machine Theory*, 37(10):1213–1239, 2002.
- [33] G. L. Goudreau and J. O. Hallquist. Recent developments in large-scale finite element lagrangian hydrocode technology. *Computer Methods in Applied Mechanics and Engineering*, 33:725–757, 1982.
- [34] D. Grégoire, H. Maigre, J. Réthoré, and A. Combescure. Dynamic crack propagation under mixed-mode loading – Comparison between experiments and X-FEM simulations. *International Journal of Solids and Structures*, 44(20):6517–6534, 2007.
- [35] M. a. Gutiérrez and H. Askes. Simulation of special loading conditions by means of non-linear constraints imposed through Lagrange multipliers. *Communications in Numerical Methods in Engineering*, 18(10):699–709, 2002.
- [36] J. O. Hallquist. *LS-DYNA Theory Manual*. Livermore Software Technology Corporation, Livermore, California, 2006.
- [37] J. O. Hallquist, G. Goudreau, and D. Benson. Sliding interfaces with contact-impact in large-scale Lagrangian computations. *Computer Methods in Applied Mechanics and Engineering*, 51(1-3):107–137, 1985.
- [38] R. Hambli. Comparison between Lemaitre and Gurson damage models in crack growth simulation during blanking process. *International Journal of Mechanical Sciences*, 43(12):2769–2790, 2001.
- [39] J. Hetherington and H. Askes. Penalty methods for time domain computational dynamics based on positive and negative inertia. *Computers & Structures*, 87(23-24):1474–1482, 2009.
- [40] J. Hetherington, A. Rodríguez-Ferran, and H. Askes. A new bipenalty formulation for ensuring time step stability in time domain computational dynamics. *International Journal for Numerical Methods in Engineering*, 90(3):269–286, 2012.
- [41] J. Hetherington, A. Rodríguez-Ferran, and H. Askes. The bipenalty method for arbitrary multipoint constraints. *International Journal for Numerical Methods in Engineering*, 2012.

- [42] H. M. Hilber, T. J. R. Hughes, and R. L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5(3):283–292, 1977.
- [43] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, New York, 1987.
- [44] T. J. R. Hughes, R. L. Taylor, J. Sackman, and A. Curnier. A finite element method for a class of contact-impact problems. *Computer Methods in Applied Mechanics and Engineering*, 8:249–276, 1976.
- [45] I. Huněk. On a penalty formulation for contact-impact problems. *Computers & Structures*, 48(2):193–203, 1993.
- [46] S. Ilanko. Existence of natural frequencies of systems with artificial restraints and their convergence in asymptotic modelling. *Journal of Sound and Vibration*, 255(5):883–898, 2002.
- [47] S. Ilanko. Introducing the use of positive and negative inertial functions in asymptotic modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 461(2060):2545–2562, 2005.
- [48] S. Ilanko and L. E. Monterrubio. On the bipenalty method: why is it advantageous to add stiffness and mass. In *International Symposium on Vibrations of Continuous Systems*, pages 25–27, 2011.
- [49] S. Ilanko and L. E. Monterrubio. Bipenalty method from a frequency domain perspective. *International Journal for Numerical Methods in Engineering*, 2012.
- [50] P. Kohnke, editor. *ANSYS, Inc. Theory Reference: ANSYS release 9.0*. ANSYS, Inc., Canonsburg, 2004.
- [51] R. F. Kulak. Critical time step estimation for three-dimensional explicit impact analysis. Technical report, Argonne National Laboratory, Argonne, IL, 1989.
- [52] K. Lee. A numerical solution for dynamic contact problems satisfying the velocity and acceleration compatibilities on the contact surface. *Computational Mechanics*, 15(3):189–200, 1994.
- [53] J. Lin. An element eigenvalue theorem and its application for stable time steps. *Computer Methods in Applied Mechanics and Engineering*, 73(3):283–294, 1989.
- [54] R. W. Macek and B. H. Aubert. A mass penalty technique to control the critical time increment in explicit dynamic finite element analyses. *Earthquake Engineering and Structural Dynamics*, 24(10):1315–1331, 1995.
- [55] C. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, Philadelphia, PA, 2000.
- [56] N. Moes, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, 1999.
- [57] R. Mullen and T. Belytschko. An analysis of an unconditionally stable explicit method. *Computers & Structures*, 16(6):691–696, 1983.

- [58] B. Nour-Omid and P. Wriggers. A two-level iteration method for solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 54:131–144, 1986.
- [59] L. Olovsson, K. Simonsson, and M. Unosson. Selective mass scaling for explicit finite element analyses. *International Journal for Numerical Methods in Engineering*, 63(10):1436–1445, 2005.
- [60] M. Ortiz and A. Pandolfi. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. *International Journal for Numerical Methods in Engineering*, 44(9):1267–1282, 1999.
- [61] E. A. Paraskevopoulos, C. G. Panagiotopoulos, and G. D. Manolis. Imposition of time-dependent boundary conditions in FEM formulations for elastodynamics: critical assessment of penalty-type methods. *Computational Mechanics*, 45(2-3):157–166, 2010.
- [62] D. A. Pierre. *Optimization Theory with Applications*. Dover Publications, Inc., New York, 2nd edition, 1986.
- [63] M. Plesha. Eigenvalue estimation for dynamic contact problems. *Journal of engineering mechanics*, 113(3):457–462, 1987.
- [64] A. Portela, M. H. Aliabadi, and D. P. Rooke. The dual boundary element method: Effective implementation for crack problems. *International Journal for Numerical Methods in Engineering*, 33(6):1269–1287, 1992.
- [65] J. W. S. Rayleigh. *The Theory of Sound*. Macmilland and Co., London, 2nd edition, 1894.
- [66] E. A. Repetto, R. Radovitzky, and M. Ortiz. Finite element simulation of dynamic fracture and fragmentation of glass rods. *Computer Methods in Applied Mechanics and Engineering*, 183(1-2):3–14, 2000.
- [67] J. C. J. Schellekens. *Computational strategies for composite structures*. PhD thesis, Delft University of Technology, Delft, 1992.
- [68] J. C. J. Schellekens and R. De Borst. On the numerical integration of interface elements. *International Journal for Numerical Methods in Engineering*, 36(1):43–66, 1993.
- [69] E. Schlangen. Experimental and numerical analysis of fracture processes in concrete, 1993.
- [70] J. C. Simo and T. A. Laursen. An augmented Lagrangian treatment of contact problems involving friction. *Computers & Structures*, 42(1):97–116, 1992.
- [71] J. C. Simo, P. Wriggers, K. H. Schweizerhof, and R. L. Taylor. Finite deformation post-buckling analysis involving inelasticity and contact constraints. *International Journal for Numerical Methods in Engineering*, 23(5):779–800, 1986.
- [72] J. C. Simo, P. Wriggers, and R. L. Taylor. A perturbed Lagrangian formulation for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 50(2):163–180, 1985.

- 
- [73] M. Smith. Synthesis of mechanical networks: the inerter. *IEEE Transactions on Automatic Control*, 47(10):1648–1662, 2002.
- [74] J.-H. Song, H. Wang, and T. Belytschko. A comparative study on finite element methods for dynamic fracture. *Computational Mechanics*, 42(2):239–250, 2007.
- [75] S. H. Song, G. H. Paulino, and W. G. Buttlar. A bilinear cohesive zone model tailored for fracture of asphalt concrete considering viscoelastic bulk material. *Engineering Fracture Mechanics*, 73(18):2829–2848, 2006.
- [76] G. Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, Wellesley, MA, 3rd edition, 2003.
- [77] R. L. Taylor and P. Papadopoulos. On a finite element method for dynamic contact/impact problems. *International Journal for Numerical Methods in Engineering*, 36(12):2123–2140, 1993.
- [78] M. Tijssens. *On the cohesive surface methodology for fracture of brittle heterogeneous solids*. PhD thesis, Delft University of Technology, 2000.
- [79] P. Wriggers. *Computational Contact Mechanics*. Springer, Berlin, 2nd edition, 2006.
- [80] X.-P. Xu and A. Needleman. Numerical simulations of fast crack growth in brittle solids. *Journal of the Mechanics and Physics of Solids*, 42(9):1397–1434, 1994.
- [81] Z.-H. Zhong. *Finite Element Procedures for Contact-Impact Problems*. Oxford University Press, Oxford, UK, 1993.
- [82] Z.-H. Zhong and J. Mackerle. Static contact problems—a review. *Engineering Computations*, 9(1):3–37, 1992.
- [83] Z.-H. Zhong and J. Mackerle. Contact-impact problems: A review with bibliography. *Applied Mechanics Reviews*, 47(2):55, 1994.
- [84] O. C. Zienkiewicz and R. L. Taylor. *The finite element method: Vol. 1. The basis*. Butterworth-Heinemann, Oxford, UK, 2000.