

**Mathematical Physics**  
Edited by Aleksandar Sedmak  
and Dragoslav Kuzmanović  
University of Belgrade

**Volume II - Numerical Methods**

**Contributors:**

**Aleksandar Sedmak**

**Simon Sedmak**

**Nikola Mladenović**

**Rade Vignjević**

**Sreten Mastilović**



# **Mathematical Physics**

**Numerical Methods**

**A. Sedmak, S. Sedmak, N. Mladenović,  
R. Vignjević, S. Mastilović**

ISBN: 9788831482561

Copyright © 2022

ESIS

[HTTPS://WWW.STRUCTURALINTEGRITY.EU](https://www.structuralintegrity.eu)

*First edition, November 2022*

# Contents

## I Numerical method - FDM-FEM

<b>Chapter 1</b>	<b>Finite difference method and Finite element method</b>	<b>11</b>
1.1	Finite difference method . . . . .	11
1.1.1	Finite difference method for parabolic partial differential equations . . . . .	11
1.1.2	Consistency and convergence . . . . .	12
1.1.3	Stability . . . . .	13
1.1.4	Parabolic equations – application to diffusion equation . . .	14
1.1.5	Explicit finite difference method . . . . .	15
1.1.6	The program . . . . .	18
1.1.7	Application of the finite difference method to hyperbolic partial differential equations – one-dimensional wavelength	22
1.1.8	Application of the finite difference method to elliptical differential equations . . . . .	23
1.1.9	Neumann problem . . . . .	24
1.1.10	Curvilinear boundaries . . . . .	26
<b>Chapter 2</b>	<b>Finite element method</b>	<b>29</b>
2.1	Finite element application to solving of one-dimensional problems .	29
2.1.1	Variation formulation . . . . .	35
2.1.2	Galerkin’s method . . . . .	36
2.1.3	Finite element base functions . . . . .	37
2.2.1	Interpolation error . . . . .	40
2.3	Finite element approximation . . . . .	42
2.4	Determining of finite element matrices and finite element system matrices . . . . .	42
2.4.1	Application of finite element method to parabolic and hyperbolic partial differential equations . . . . .	45

<b>Chapter 3</b>	<b>Comparison of finite element method and finite difference method</b>	<b>49</b>
3.1	Approximate analytical solution . . . . .	50
3.1.1	Finite difference method solution . . . . .	51
3.1.2	Finite element method solution . . . . .	52
3.2	Finite element method – two-dimensional problem . . . . .	55
3.2.1	Introduction . . . . .	55
3.2.2	Physical base of the problem . . . . .	55
3.2.3	Two-dimensional elliptical boundary problem . . . . .	57
3.3	Variation formulation of the boundary problem . . . . .	60
3.4	Finite element interpolation . . . . .	62
3.4.1	Interpolation within triangles . . . . .	64
3.4.2	Interpolation error . . . . .	67
3.5	Finite element approximations . . . . .	67
3.5.1	Determining of the finite element matrix . . . . .	74
3.5.2	Calculating of finite element matrices . . . . .	79
3.6	Program for solving of elliptical problems . . . . .	82
3.6.1	Uvodne napomene . . . . .	82
3.6.2	Subprogram structure . . . . .	83
3.6.3	Examples . . . . .	85
	<b>Bibliography</b>	<b>105</b>

<b>Chapter 4</b>	<b>Finite volume method</b>	<b>109</b>
4.1	Introduction . . . . .	109
4.2	Solution of Euler equations . . . . .	110
4.2.1	Explicit numerical scheme . . . . .	110
4.2.2	Implicit numerical scheme . . . . .	115
4.2.3	Godunov scheme . . . . .	123
4.3	Solution of Navier–Stokes equations . . . . .	149
4.3.1	Implicit numerical scheme . . . . .	149
	<b>Bibliography</b>	<b>155</b>

<b>Chapter 5</b>	<b>Review of Development of the Smooth Particle Hydrodynamics (SPH) Method</b>	<b>161</b>
5.1	Introduction . . . . .	161
5.2	Basic Formulation . . . . .	165
5.3	Conservation Equations . . . . .	167
5.4	Kernel Function . . . . .	169
5.5	Variable Smoothing Length . . . . .	170
5.6	Neighbour Search . . . . .	171
5.7	SPH Shortcomings . . . . .	172
5.7.1	Consistency . . . . .	172
5.8	Derivation of Normalised Corrected Gradient SPH formula . . . . .	172
5.9	Tensile Instability . . . . .	176
5.9.1	Stability analysis of conventional (Eulerian) SPH . . . . .	178
5.10	Zero-Energy Modes . . . . .	181
5.11	Non-Local properties of SPH . . . . .	182
5.11.1	Theoretical Background of Strain-Softening . . . . .	183
5.11.2	Main Aspects of the Smoothed Particle Hydrodynamics (SPH) Method as Nonlocal Regularisation Method . . . . .	187
5.11.3	Numerical Experiments for the Evaluation of the SPH Method . . . . .	189
5.11.4	Numerical Results of the Strain-Softening in FE . . . . .	193
5.11.5	Strain-Softening in SPH Numerical Results of the Strain-Softening in SPH . . . . .	194
5.12	Summary . . . . .	201
	<b>Bibliography</b>	<b>203</b>

<b>Chapter 6</b>	<b>Introduction to Computational Mechanics of Discontinua</b>	<b>215</b>
6.1	Introduction . . . . .	215
6.2	Molecular Dynamics . . . . .	221
6.2.1	Basic Idea of MD . . . . .	222
6.2.2	Empirical Interatomic Potentials . . . . .	223

6.2.3	Integration Algorithms . . . . .	228
6.2.4	Calculation of Macro-parameters of State . . . . .	229
6.2.5	MD Simulation Cell and List of Neighbors . . . . .	234
6.2.6	Temperature and Pressure Control . . . . .	235
6.2.7	Advantages and Disadvantages of Traditional MD . . . . .	237
6.3	Particle Dynamics . . . . .	239
6.3.1	Basic Idea of PD . . . . .	239
6.3.2	Interparticle Potentials . . . . .	240
6.4	Lattices . . . . .	244
6.4.1	Basic idea of Lattice Models . . . . .	245
6.4.2	Lattices with Central Interactions ( $\alpha$ Models) . . . . .	245
6.4.3	Lattices with Central and Angular Interactions ( $\alpha - \beta$ Models) . . . . .	249
6.4.4	Lattices with Beam Interactions . . . . .	251
6.4.5	Various Aspects of Lattice Modeling . . . . .	255
6.5	Discrete Element Methods . . . . .	257
6.5.1	Basic idea of DEM . . . . .	257
6.5.2	Contact Algorithms . . . . .	259
6.5.3	DEM Modeling of Particulate Systems . . . . .	261
6.5.4	DEM Modeling of Solid Materials . . . . .	264

<b>Bibliography</b>	<b>273</b>
---------------------	------------

<b>Index</b>	<b>283</b>
--------------	------------

# Preface

This book is mainly based on the material initially published in Serbian, in 2021, by the University of Belgrade, Faculty of Mining and Geology, under the title *Mathematical Physics (Theory and Examples)*. For the purpose of this book the material from the Serbian edition was reviewed, amended, and translated, with new material added in two final chapters in the second volume. We have divided text into two separate volumes:

Mathematics of Physics - Analytical Methods and  
Mathematics of Physics - Numerical Methods.

**The first volume consists of 8 chapters:**

- The first 7 chapters were written by Dragoslav Kuzmanović, Dobrica Nikolić and Ivan Obradović, and correspond to the text from Chapters 1-8 of the Serbian edition, translated by Ivan Obradović.
- The material of Chapter 8, which is of a monographic character, corresponds to the material of Chapter 9 in the Serbian edition, but was thoroughly reviewed and rewritten in English by Mihailo Lazarević.

**The second volume consists of 6 chapters:**

- The first 3 chapters were written by Aleksandar Sedmak and correspond to Chapter 10 of the Serbian edition, restructured and reviewed, and then translated by Simon Sedmak.

- Chapter 4 corresponds to the text of Chapter 11 of the Serbian edition, written and translated by Nikola Mladenović.
- Chapters 5 and 6, written by Rade Vignjević and Sreten Mastilović, respectively, offer completely new material.

Chapters 4,5 and 6 are of a monographic character.

# Numerical method - FDM-FEM

## **1 Finite difference method and Finite element method . . . . . 11**

1.1 Finite difference method

## **2 Finite element method . . . . . 29**

2.1 Finite element application to solving of one-dimensional problems

2.3 Finite element approximation

2.4 Determining of finite element matrices and finite element system matrices

## **3 Comparison of finite element method and finite difference method . . . . . 49**

3.1 Approximate analytical solution

3.2 Finite element method – two-dimensional problem

3.3 Variation formulation of the boundary problem

3.4 Finite element interpolation

3.5 Finite element approximations

3.6 Program for solving of elliptical problems

## **Bibliography . . . . . 105**

II

**FVM**

**107**



# 1. Finite difference method and Finite element method

## 1.1 Finite difference method

The basic idea behind the finite difference method is to replace the derivatives of a given function with their approximate values. In order to achieve this goal, points which form a mesh of nodes are introduced, and the solution is determined for them.

The basic concept of finite different method and its realization will be shown using examples which involve parabolic, hyperbolic and elliptical partial differential equations.

### 1.1.1 Finite difference method for parabolic partial differential equations

A **mesh** in plane  $xt$  is a set of points  $(x_n, t_j) = (x_0 + nh, t_0 + jk)$ , where  $n$  and  $j$  are integers,  $(x_0, t_0)$  is the referent point, and  $(x_n, t_j)$  are called mesh points or **nodes**. Positive numbers  $h$  and  $k$  are mesh steps along the  $x$  and  $t$  directions, respectively. If  $h$  and  $k$  are constants, the mesh is uniform, and if they are equal, the mesh is quadratic. If we use the compact designation

$$u_{nj} = u(x_n, t_j). \quad (1.1)$$

We can now write the following

$$u_t(x_n, t_j) = \frac{u_{n,j+1} - u_{nj}}{k} + O(k), \quad (1.2)$$

$$u_{xx}(x_n, t_j) = \frac{u_{n+1,j} - 2u_{nj} + u_{n-1,j}}{h^2} + O(h^2) \equiv \frac{\delta_x^2 u_{nj}}{h^2} + O(h^2), \quad (1.3)$$

which introduce the differential difference operator  $\delta_x^2$  which is analogous to the differential operator  $\frac{\partial^2}{\partial x^2}$ . Equation (1.2) is a two-level equation with respect to  $t$ , since it only includes two subsequent values of  $j$ .

Let area  $\Omega$  within the plane  $xt$  be covered by a mesh  $(x_n, t_j)$ . If all derivatives in a **partial differential equation**, given by:

$$L[u] = f \quad (x, t) \in \Omega. \quad (1.4)$$

Replaced with their finite differences, we obtain the finite difference equation:

$$D[U_{nj}] = f_{nj} \quad (x_n, t_j) \in \Omega. \quad (1.5)$$

We can say that equation (1.4) was discretized in order to obtain (1.5). Whose solution,  $U_{nj}$ , approximately represents the unknown  $u(x, t)$  in element nodes  $(x_n, t_j)$

### 1.1.2 Consistency and convergence

In order to obtain good approximation via discretization, the solution of (1.4) needs to be as close as possible to satisfying the condition given by (1.5), for sufficiently small  $h$  and  $k$ . Local rounding error is represented by the following difference:

$$T_{nj} = D[u_{nj}] - f_{nj}. \quad (1.6)$$

Finite difference equation is **consistent** with the partial differential equation (1.4) under the following condition:

$$\lim_{k,h \rightarrow 0} T_{nj} = 0. \quad (1.7)$$

Besides consistency, it is necessary for the approximate solution accuracy to increase when  $h, k \rightarrow 0$ . If  $U_{nj}$  is the exact solution of (1.5), and  $u_{nj}$  is the solution of (1.4) at point  $(x_n, t_j)$ , discretization error is defined as the difference  $U_{nj} - u_{nj}$ . Finite difference method is considered convergent if:

$$\lim_{h,k \rightarrow 0} |U_{nj} - u_{nj}| = 0 \quad (x_n, t_j) \in \Omega. \quad (1.8)$$

### 1.1.3 Stability

Let  $U_{nj}$  be the solution of equation (1.5), with initial values of  $U_{n0}$  and certain boundary values. Let  $V_{nj}$  be the solution of a system of finite difference equations, which differs from (1.5) only in the sense of initial values, i.e. the relation  $V_{n0} \equiv U_{n0} + E_{n0}$ , holds, where  $E_{n0}$  is the "error", i.e. the initial difference (deviation). It can be show that  $E_{n0}$  propagates with an increase in  $j$ , towards a homogeneous partial differential equation, with given homogeneous boundary conditions:

$$D[E_{nj}] = 0.$$

When the finite difference equation (1.5) is applied to approximate determining of  $u(x, T)$  for a fixed  $T = t_0 + jk$ , it is clear that  $h, k \rightarrow 0$  requires that  $j \rightarrow \infty$ . In addition, in order for equation (1.5) to be applicable to a fixed mesh for the purpose of approximate determining of  $u(x_n, t_j)$ , for increasing  $t_j$ , it is once again necessary for  $j \rightarrow \infty$ . For partial equations with limited solutions, it is said that the **solution** of (1.5) is **stable** if  $E_{nj}$  is uniformly limited along  $n$  when  $j \rightarrow \infty$ , i.e. when the following holds:

$$|E_{nj}| < M \quad (j > J) \quad (1.9)$$

where  $M$  is an arbitrary constant, and  $J$  is a positive integer. If  $h$  and  $k$  are functionally dependent in order to ensure that (1.9) is fulfilled, than the solution of the finite difference equation is conditionally stable. The stability of this solution also implies its convergence.

Let us also define the matrix criteria for stability. For this purpose, we will consider a boundary problem with the initial condition including  $N$  nodes along the  $x$  direction, and let us define a vector-column of errors at level  $j$ ,  $\mathbf{E}_j = (E_{1j}, \dots, E_{Nj})^T$ . For a two-level finite difference method, errors at levels  $j$  and  $j + 1$  are related by the following expression:

$$\mathbf{E}_{j+1} = \mathbf{C}\mathbf{E}_j, \quad (1.10)$$

where  $\mathbf{C}$  is a matrix of  $N \times N$  order. Let  $\rho(\mathbf{C})$ , the spectral radius of  $\mathbf{C}$ , denote the highest eigenvalue of matrix  $\mathbf{C}$ . In this case, the matrix criteria of stability can be defined as:

*a two-level finite difference method for a boundary problem with an initial condition with a limited solution is stable (in matrix terms) if  $\rho(\mathbf{C}) \leq 1$ .*

Matrix criteria is a necessary condition for two-level finite difference method stability, but also becomes a sufficient condition if  $\mathbf{C}$  is symmetrical or nearly symmetrical, whereas all of its eigenvalues are real.

### 1.1.4 Parabolic equations – application to diffusion equation

One-dimensional diffusion equation:

$$u_t = a^2 u_{xx} \quad (1.11)$$

is used as a typical parabolic partial differential equation, for which we will derive the corresponding finite difference equation. For a mesh  $(x_n, t_j) = (nh, jk)$ , there are several possible finite difference equations. Here we will consider three of the most commonly used two-level equations, whose solution is known during step  $j$ , and is explicitly or implicitly used in order to determine a new solution in  $j + 1$ :

1) **Explicit method** ("forward difference")

$$\frac{U_{n,j+1} - U_{nj}}{k} = a^2 \frac{U_{n+1,j} - 2U_{nj} + U_{n-1,j}}{h^2} \quad (1.12)$$

or, in short:

$$U_{n,j+1} = (1 + r\delta_x^2)U_{nj} \quad (r = a^2 k/h^2).$$

2) **Implicit method** ("backward difference")

$$\frac{U_{n,j+1} - U_{nj}}{k} = a^2 \frac{U_{n+1,j+1} - 2U_{n,j+1} + U_{n-1,j+1}}{h^2} \quad (1.13)$$

or, in short:

$$(1 - r\delta_x^2)U_{n,j+1} = U_{nj} \quad (r = a^2 k/h^2).$$

3) **Implicit method** (Crank - Nicolson)

$$\frac{U_{n,j+1} - U_{nj}}{k} = \frac{a^2}{2} \frac{\delta_x^2 U_{nj} + \delta_x^2 U_{n,j+1}}{h^2} \quad (1.14)$$

or, in short:

$$(1 - \frac{r}{2}\delta_x^2)U_{n,j+1} = (1 + \frac{r}{2}\delta_x^2)U_{nj}.$$

Implicit methods are unconditionally stable with local rounding error of  $O(k + h^2)$  for the forward method and  $O(k^2 + h^2)$  for the Crank – Nicolson method, whereas the explicit method is conditionally stable ( $r \leq 1/2$ ) and has a local rounding error of  $O(k + h^2)$ . Fir two-dimensional **diffusion equation**:

$$u_t = a^2(u_{xx} + u_{yy}) \quad (1.15)$$

the mesh is  $(x_m, y_n, t_j) = (mh, nh, jk)$  i  $U_{mnj} \approx u_{mnj} = u(x_m, y_n, t_j)$ . Analogous to the previous equations, we now have the following:

$$U_{mn,j+1} = [1 + r(\delta_x^2 + \delta_y^2)]U_{mnj} \quad (1.16)$$

$$[1 - r(\delta_x^2 + \delta_y^2)]U_{mn,j+1} = U_{mnj} \quad (1.17)$$

$$[1 - \frac{r}{2}(\delta_x^2 + \delta_y^2)]U_{mn,j+1} = [1 + \frac{r}{2}(\delta_x^2 + \delta_y^2)]U_{mnj} \quad (1.18)$$

Local rounding errors and stability are the same as in the case of the one-dimensional diffusion equation, with the only difference being that the explicit method is stable for  $r \leq 1/4$ .

Under the condition that it is stable, the explicit method has a considerable advantage over the implicit ones, since it directly provides a step-by-step solution. Implicit method, even though unconditionally stable, require the inversion of corresponding matrices in each time step, i.e. the solving of corresponding equation systems, which requires a far greater computing capacity.

### 1.1.5 Explicit finite difference method

We will now perform a thorough analysis in order to show that expression given for explicit finite difference method hold, including error estimation and stability conditions. For this purpose, we will write the expressions for  $u_t$  and  $u_{xx}$ , obtained by deriving function  $u(x, t)$  into a Taylor series:

$$u(x, t + k) = u(x, t) + u_t(x, t)k + u_{tt}(x, \tilde{t})\frac{k^2}{2}, \quad (1.19)$$

series:

$$u(x, t + k) = u(x, t) + u_t(x, t)k + u_{tt}(x, \tilde{t})\frac{k^2}{2}, \quad (1.20)$$

i.e.

$$u_{n,j+1} = u_{nj} + u_t(x_n, t_j)k + u_{tt}(x_n, \tilde{t}_j)\frac{k^2}{2}, \quad (1.21)$$

from which follows that:

$$u_t(x_n, t_j) = \frac{u_{n,j+1} - u_{nj}}{k} - u_{tt}(x_n, \tilde{t}_j)\frac{k}{2} \quad (1.22)$$

where  $u_{tt}(\tilde{x}_n, \tilde{t}_j)\frac{k}{2} = O(k)$  is the local rounding error.

Expression for  $u_{xx}$  can also be obtained by deriving  $u(x, t)$  into a Taylor series:

$$\begin{aligned} u(x + h, t) = & u(x, t) + u_x(x, t)h + u_{xx}(x, t)\frac{h^2}{2} + \\ & + u_{xxx}(x, t)\frac{h^3}{6} + u_{xxxx}(\tilde{x}, t)\frac{h^4}{24} \end{aligned} \quad (1.23)$$

where  $x < \tilde{x} < x + h$ ; i.e.

$$u(x - h, t) = u(x, t) - u_x(x, t)h + u_{xx}(x, t)\frac{h^2}{2} - u_{xxx}(x, t)\frac{h^3}{6} + u_{xxxx}(\tilde{x}, t)\frac{h^4}{24} \quad (1.24)$$

where  $x - h < \bar{x} < x$ . By adding the above, we obtain:

$$u_{n+1,j} + u_{n-1,j} = 2u_{n,j} + h^2 u_{xx}(x_n, t_j) + \frac{h^4}{24} [u_{xxxx}(\bar{x}, t) + u_{xxxx}(\hat{x}, t)]$$

from which follows that:

$$\begin{aligned} u_{xx}(x_n, t_j) &= \frac{u_{n+1,j} - 2u_{n,j} + u_{n-1,j}}{h^2} + \frac{h^2}{12} u_{xxxx}(\hat{x}, t) = \\ &= \frac{\delta_x^2 u_{nj}}{h^2} + O(h^2) \end{aligned} \quad (1.25)$$

where  $\bar{x} < \hat{x} < \tilde{x}$ . In this way it was also shown that the local rounding error is  $O(k + h^2)$ . It can now be shown that the local rounding error is  $O(k^2 + h^2)$ , if  $k = h^2/6a^2$ , where  $a$  is the diffusion coefficient. For this purpose, we write the following:

$$(u_t - a^2 u_{xx})_{nj} = \frac{u_{n,j+1} - u_{nj}}{k} - a^2 \frac{\delta_x^2 u_{nj}}{h^2} - \frac{k}{2} u_{tt}(x_n, \bar{t}_j) + a^2 \frac{h^2}{12} u_{xxxx}(\bar{x}_n, t_j) \quad (1.26)$$

where  $t_j < \bar{t}_j < t_{j+1}$  and  $x_{n-1} < \bar{x}_n < x_{n+1}$ . Solution error is:

$$T_{nj} = \frac{k}{2} u_{tt}(x_n, \bar{t}_j) - a^2 \frac{h^2}{12} u_{xxxx}(\bar{x}_n, t_j) = O(k + h^2) \quad (1.27)$$

if  $u_{tt}$  and  $u_{xxxx}$  are limited. By applying the expression for Taylor series and the condition that  $(u_t - a^2 u_{xx})_{nj} = 0$ , we obtain:

$$\frac{u_{n,j+1} - u_{nj}}{k} - a^2 \frac{\delta_x^2 u_{nj}}{h^2} = \left[ \frac{k}{2} u_{tt} - a^2 \frac{h^2}{12} u_{xxxx} \right]_{nj} + O(k^2) + O(h^4) \quad (1.28)$$

since  $u_t = a^2 u_{xx}$ ,  $u_{tt} = a^2 u_{xxt} = a^2 (u_t)_{xx} = a^2 (a^2 u_{xx})_{xx} = a^4 u_{xxxx}$ , which shows that the expression  $\left[ \frac{k}{2} u_{tt} - a^2 \frac{h^2}{12} u_{xxxx} \right]_{nj}$  can be written as:

$$\left( \frac{k}{2} a^4 - a^2 \frac{h^2}{12} \right) u_{xxxx}(x_n, t_j), \quad (1.29)$$

which is equal to 0 for  $k = h^2/6a^2$ .

We will now apply the matrix criteria of stability in order to show that the explicit method is stable if and only if  $r \leq 1/2$ . As an example, we will use the boundary problem given by:

$$\begin{aligned} u_t &= a^2 u_{xx} & 0 < x < 1, & \quad t > 0, \\ u(x, 0) &= f(x) & 0 < x < 1, & \\ u(0, t) &= u(1, t) = 0 & t > 0. & \end{aligned} \quad (1.30)$$

Let  $(x_n, t_j) = (nh, jk)$  ( $n = 0, 1, \dots, N; j = 0, 1, 2, \dots$ ) with  $Nh = 1$ , and let the vector column  $\mathbf{U}_j$  be defined as:

$$\mathbf{U}_j = [U_{1j}, \dots, U_{N-1,j}]^T.$$

Explicit method can be expressed as the following matrix equations:

$$\mathbf{U}_{j+1} = \mathbf{C}\mathbf{U}_j \quad (j = 0, 1, 2, \dots), \quad (1.31)$$

where  $\mathbf{U}_0 = [f_1, f_2, \dots, f_{N-1}]^T$ , and  $\mathbf{C}$  is the three-diagonal square matrix of the  $N-1$  order.

$$\begin{bmatrix} 1-2r & r & 0 & 0 & 0 & 0 \\ r & 1-2r & r & 0 & 0 & 0 \\ & r & 1-2r & r & 0 & 0 \\ & & r & 1-2r & \ddots & 0 \\ \mathbf{0} & & & \ddots & \ddots & r \\ & & & & r & 1-2r \end{bmatrix} \quad (1.32)$$

Let us assume that error  $E_{nj}$  was introduced at  $x_n$  in time step  $j$ , ( $n = 1, \dots, N-1$ ), such that the solution of equation (1.31) becomes  $\mathbf{U}_j + \mathbf{E}_j$ , where  $\mathbf{E}_j$  is the vector-column whose  $n$ -th component is  $E_{nj}$ . In this case, the following holds:

$$\mathbf{U}_{j+1} + \mathbf{E}_{j+1} = \mathbf{C}\mathbf{U}_j + \mathbf{C}\mathbf{E}_j \quad \text{or} \quad \mathbf{E}_{j+1} = \mathbf{C}\mathbf{E}_j \quad (1.33)$$

i.e., after a total of  $m$  steps

$$\mathbf{E}_{j+m} = \mathbf{C}^m \mathbf{E}_j \quad (1.34)$$

$\lambda_1, \dots, \lambda_{N-1}$  and  $\mathbf{V}_1, \dots, \mathbf{V}_{N-1}$  be the eigenvalues and their respective linearly independent eigenvectors of a symmetrical matrix  $\mathbf{C}$ . If  $\mathbf{E}_j$  is written as a linear combination of  $\mathbf{V}_k$ :

$$\mathbf{E}_j = \sum_{k=1}^{N-1} a_k \mathbf{V}_k, \quad (1.35)$$

and if the following is used

$$\mathbf{C}\mathbf{V}_k = \lambda_k \mathbf{V}_k \quad (1.36)$$

we obtain:

$$\mathbf{E}_{j+m} = \sum_{k=1}^{N-1} \lambda_k^m a_k \mathbf{V}_k \quad (1.37)$$

Equation (1.37) indicates that the error  $E_{nj}$  remains limited if and only if  $|\lambda_k| \leq 1$  for  $k = 1, \dots, N-1$ . Knowing that the eigenvalues of a real symmetrical

triangular matrix of the  $N \times N$  order are all within the interval  $(p - |2q|, p + |2q|)$ , where  $p = 1 - 2r$ ,  $q = r$ , we obtain:

$$\lambda_{max} \approx (1 - 2r) + |2r| = 1, \quad (1.38)$$

$$\lambda_{min} \approx (1 - 2r) - |2r| = 1 - 4r, \quad (1.39)$$

from which follows that  $1 - 4r \geq -1$ , i.e  $r \leq 1/2$ .

### 1.1.6 The program

The simplicity of the finite difference method will be illustrated using a computer program which solved the following problem:

$$u_t = u_{xx} \quad 0 < x < 1 \quad t > 0, \quad (1.40)$$

$$u(x, 0) = 100 \sin(\pi x) \quad 0 < x < 1, \quad (1.41)$$

$$u(0, t) = u(1, t) = 0 \quad t > 0. \quad (1.42)$$

For  $t = 0.5$ , the result will be compared to the exact solution:  $u = 100e^{-\pi^2 t} \sin(\pi x)$ .

Application of the explicit method is reduced to using the expression (1.12), whereas the application of implicit methods requires additional analysis. If we introduce the weight factor  $W$ , we can write the following:

$$U_{n,j+1} - U_{nj} = r[(1 - W)\delta_x^2 U_{nj} + W\delta_x^2 U_{n,j+1}] \quad (1.43)$$

which is reduced to the backward method for  $W = 1$ , and to the Crank-Nicolson method for  $W = 0.5$ . By introducing contour and initial conditions in (1.43), we obtain a three-diagonal system of equations:

$$\begin{bmatrix} 1 + 2Wr & -Wr & 0 & 0 & 0 \\ -Wr & 1 + 2Wr & -Wr & 0 & 0 \\ 0 & -Wr & 1 + 2Wr & -Wr & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & -Wr & 1 + 2Wr & -Wr \\ 0 & 0 & 0 & -Wr & 1 + 2Wr \end{bmatrix} \begin{bmatrix} U_{1,j+1} \\ U_{2,j+1} \\ U_{3,j+1} \\ \dots \\ U_{N-2,j+1} \\ U_{N-1,j+1} \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \dots \\ D_{N-2} \\ D_{N-1} \end{bmatrix} \quad (1.44)$$

where

$$D_n = U_{nj} + (1 - W)r\delta_x^2 U_{nj} \quad (n = 2, 3, \dots, N - 2), \quad (1.45)$$

$$D_1 = U_{1j} + (1 - W)r\delta_x^2 U_{1j} + WrU_{0,j+1}, \quad (1.46)$$

$$D_{N-1} = U_{N-1,j} + (1 - W)r\delta_x^2 U_{N-1,j} + WrU_{N,j+1}. \quad (1.47)$$

Programs for implicit and explicit method are shown in the following section, along with their corresponding results. In the case of the explicit method, a total of ten steps were used and parameter  $r$  was equal to  $1/6$ , and  $1/2$ , whereas for the implicit method ( $W = 0.5$ , and  $W = 1$ ), this parameter was  $r = 1/2$ , with the same number of steps. Results have shown that the best compliance was obtained for  $r$  equal to  $1/6$ , in the case of the explicit method.

$W = .50$			$W = 1.00$		
$N = 10$	$VK = .005000$		$N = 10$	$VK = .005000$	
$T = .50$	NUMERICKI	TAČNO	$T = .50$	NUMERICKI	TAČNO
$Z = .0$	.000000	.000000	$Z = .0$	.000000	.000000
$Z = .1$	.231190	.222242	$Z = .1$	.259880	.222242
$Z = .2$	.439749	.422730	$Z = .2$	.494321	.422730
$Z = .3$	.605263	.581837	$Z = .3$	.680375	.581837
$Z = .4$	.711529	.683991	$Z = .4$	.799828	.683991
$Z = .5$	.748146	.719190	$Z = .5$	.840989	.719190
$Z = .6$	.711529	.683991	$Z = .6$	.799828	.683991
$Z = .7$	.605263	.581837	$Z = .7$	.680375	.581837
$Z = .8$	.439749	.422729	$Z = .8$	.494321	.422729
$Z = .9$	.231190	.222242	$Z = .9$	.259880	.222242
$Z = 1.0$	.000000	.000000	$Z = 1.0$	.000000	.000000

```

PROGRAM i1djd
DIMENSION a(101) , b(101) , c(101) , d(101) , u(0:101)
OPEN (1,FILE='ULAZ')
OPEN (2,FILE='IZLAZ')
pi = 4.*atan(1.)
READ (1,99005) n , vk , w
DO 100 i = 0 , n
    h = 1./n
    z = i*h
    u(i) = 100.*sin(pi*z)
100 CONTINUE
l = n - 1
200 DO 300 i = 1 , l
    a(i) = -w*vk*n*n
    b(i) = 1. + 2.*w*vk*n*n

```

## 20 Chapter 1. Finite difference method and Finite element method

---

```

        c(i) = -w*vk*n*n
        d(i) = u(i) + (1-w)*vk*n*n*(u(i+1)-2*u(i)+u(i-1))
300  CONTINUE
    CALL tridi(a,b,c,d,l)
    t = t + vk
    DO 400 i = 1 , n - 1
        u(i) = d(i)
400  CONTINUE
    u(0) = 0.
    u(n) = 0.
    IF ( abs(0.5-t).GT.vk/2 ) GOTO 200
    WRITE (2,99001) w
    WRITE (2,99002) n , vk
    WRITE (2,99003) t
    DO 500 i = 0 , n
        h = 1./n
        z = i*h
        WRITE (2,99004) z , u(i) , 100.*exp(-pi*pi*t)*sin(pi*z)
500  CONTINUE
    STOP
99001 FORMAT ('W=',F5.2)
99002 FORMAT ('N=',I5,T18,'VK=',F10.6)
99003 FORMAT ('T=',F5.2,T18,'NUMERICKI',T35,'TACNO',)
99004 FORMAT ('Z=',F4.1,T13,F13.6,T30,F13.6)
99005 FORMAT (I5,F15.6,F5.2)
    END

    SUBROUTINE tridi(a,b,c,d,l)
    DIMENSION a(101) , b(101) , c(101) , d(101)
    DO 100 i = 2 , l
        rt = -a(i)/b(i-1)
        b(i) = b(i) + rt*c(i-1)
        d(i) = d(i) + rt*d(i-1)
100  CONTINUE
    d(1) = d(1)/b(1)
    DO 200 i = l - 1 , 1 , -1
        d(i) = (d(i)-c(i)*d(i+1))/b(i)
200  CONTINUE
    RETURN
    END
```

$N = 10$	$VK = .001667$		$N = 10$	$VK = .005000$	
$T = .50$	NUMERICKI	TAČNO	$T = .50$	NUMERICKI	TAČNO
$Z = .0$	.000000	.000000	$Z = .0$	.000000	.000000
$Z = .1$	.222040	.222022	$Z = .1$	.204463	.222241
$Z = .2$	.422346	.422311	$Z = .2$	.388912	.422728
$Z = .3$	.581309	.581261	$Z = .3$	.535292	.581835
$Z = .4$	.683370	.683314	$Z = .4$	.629273	.683989
$Z = .5$	.718537	.718479	$Z = .5$	.661657	.719188
$Z = .6$	.683370	.683314	$Z = .6$	.629273	.683989
$Z = .7$	.581309	.581261	$Z = .7$	.535292	.581835
$Z = .8$	.422346	.422311	$Z = .8$	.388912	.422728
$Z = .9$	.222040	.222022	$Z = .9$	.204463	.222241
$Z = 1.0$	.000000	.000000	$Z = 1.0$	.000000	.000000

```

PROGRAM e1djd
DIMENSION u(0:101) , v(0:101)
OPEN (1,FILE='ULAZ')
OPEN (2,FILE='IZLAZ')
pi = 4.*atan(1.)
READ (1,99004) n , vk
DO 100 i = 0 , n
    h = 1./n
    z = i*h
    v(i) = 100.*sin(pi*z)
100 CONTINUE
200 DO 300 i = 1 , n - 1
    u(i) = v(i) + vk*n*n*(v(i+1)-2*v(i)+v(i-1))
300 CONTINUE
t = t + vk
u(0) = 0.
u(n) = 0.
DO 400 i = 1 , n + 1
    v(i-1) = u(i-1)
400 CONTINUE
IF ( abs(0.5-t).GT.vk/2 ) GOTO 200
WRITE (2,99001) n , vk
WRITE (2,99002) t
DO 500 i = 0 , n
    h = 1./n
    z = i*h
    WRITE (2,99003) z , u(i) , 100.*exp(-pi*pi*t)*sin(pi*z)
500 CONTINUE
STOP
99001 FORMAT ('N=',I5,T18,'VK=',F10.6)

```

```

99002 FORMAT ('T=',F5.2,T18,'NUMERICKI',T35,'TACNO',/)
99003 FORMAT ('Z=',F4.1,T13,F13.6,T30,F13.6)
99004 FORMAT (I5,F15.6)
      END
    
```

### 1.1.7 Application of the finite difference method to hyperbolic partial differential equations – one-dimensional wavelength

Let us now consider a one-dimensional wavelength equation, as a typical representative of hyperbolic partial differential equations:

$$u_{tt} = c^2 u_{xx} \quad (1.48)$$

let  $(x_n, t_j) = (nh, jk)$  ( $n, j = 0, 1, 2, \dots$ ) and  $s = k/h$ . We will use the following equations:

$$\frac{U_{n,j+1} - 2U_{nj} + U_{n,j-1}}{k^2} = c^2 \frac{U_{n+1,j} - 2U_{nj} + U_{n-1,j}}{h^2} \quad (1.49)$$

i.e.

$$\delta_t^2 U_{nj} = c^2 s^2 \delta_x^2 U_{nj}$$

for explicit method:

$$\delta_t^2 U_{nj} = c^2 s^2 \frac{\delta_x^2 U_{n,j+1} + \delta_x^2 U_{n,j-1}}{2} \quad (1.50)$$

i.e.

$$-c^2 s^2 U_{n-1,j+1} + (2 + 2c^2 s^2) U_{n,j+1} - c^2 s^2 U_{n+1,j+1} = 4U_{nj} - 2U_{n,j-1} + c^2 s^2 \delta_x^2 U_{nj}$$

for the implicit method.

Herein local rounding errors in both cases are  $O(k^2 + h^2)$ ; the explicit method is conditionally stable iff  $c^2 s^2 \leq 1$ , and the implicit method is unconditionally stable. Attention should be paid to conditions  $u(x, 0) = f(x)$  and  $u_t(x, 0) = g(x)$ , which, unlike the previous ones, also include  $u_t(x, 0)$ , and are defined in accordance with the principle that the error introduced here must not be greater than the local rounding error, which in this case is  $O(k^2 + h^2)$ . It is obvious that this condition is fulfilled for  $U_{n0} = f(x_n)$ , since the error is equal to 0. As far as  $U_{n1}$  is concerned, let us assume that  $f$  is contained in  $C^2$  and that (1.49) also holds for  $t = 0$ , and then apply the Taylor theorem:

$$\begin{aligned}
 u(x_n, t_1) &= u(x_n, 0) + k u_t(x_n, 0) + \frac{k^2}{2} u_{tt}(x_n, 0) + O(k^3) = \\
 &= u(x_n, 0) + k g(x_n) + \frac{k^2}{2} c^2 f''(x_n) + O(k^3) = \\
 &= u(x_n, 0) + k g(x_n) + \frac{k^2 c^2}{2} [f(x_{n-1}) - 2f(x_n) + f(x_{n+1}))] + O(k^2 h^2 + k^3)
 \end{aligned} \quad (1.51)$$

Where  $f''(x_n)$  was approximated in the final step using a second-order finite difference, see (1.50). It can now be seen that the expression

$$g(x_n) = \frac{U_{n1} - U_{n0}}{k} - \frac{kc^2}{2h^2} [f(x_{n-1}) - 2f(x_n) + f(x_{n+1}))] \quad (1.52)$$

is satisfied by an exact solution  $u$  with an error of  $O(h^2 + k^2)$ ; in other words, if the expression (1.49) determines  $U_{n1}$ , we obtain an error whose order is higher than  $O(k^2 + h^2)$ .

### 1.1.8 Application of the finite difference method to elliptical differential equations

For a linear elliptical contour problem, if all derivatives are replaced by their finite differences, we obtain a system of linear algebraic equations, as is the case with the Dirichlet problem over a domain  $\Omega$  ( $0 < x < l; 0 < y < l$ ):

$$u_{xx} + u_{yy} = f(x, y) \quad (1.53)$$

with the following boundary conditions:

$$u = g(x, y) \quad \text{on the boundary } S. \quad (1.54)$$

If we chose a mesh step of  $h = l/4$ , we obtain the following mesh:

$$(x_m, y_m) = (mh, nh) \quad (m, n = 0, 1, 2, 3, 4). \quad (1.55)$$

By using the central differences, we obtain:

$$\frac{U_{m+1,n} - 2U_{mn} + U_{m-1,n}}{h^2} + \frac{U_{m,n+1} - 2U_{mn} + U_{m,n-1}}{h^2} = f_{mn}, \quad (1.56)$$

where  $f_{mn} = f(x_m, y_n)$ , a  $U_{mn} = g_{mn}$  za  $m, n = 0$  or  $4$ . In this case, the following is obtained:

$$\begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \\ U_9 \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \\ B_6 \\ B_7 \\ B_8 \\ B_9 \end{bmatrix} \quad (1.57)$$

where for  $g = 0$ ,  $B_j = -h^2 f_j$ . Based on this example, it can be seen that for a general 2D linear elliptical contour problem, we can obtain a system of linear algebraic equations:

$$\mathbf{AU} = \mathbf{B} \quad (1.58)$$

wherein the following holds:

- (1) Dimensions of vectors  $\mathbf{U}$  and  $\mathbf{B}$  correspond to the number of nodes for which the solution is sought.
- (2) Vector  $\mathbf{B}$  is defined by its contour conditions and terms in the partial differential equation that are not dependent from  $u$ .
- (3) Matrix  $\mathbf{A}$  is square and contains no more than 5 terms per row or column which are not 0. With a fixed  $\Omega$ , the order of matrix  $\mathbf{A}$  is a decreasing function of the number of mesh steps,  $h$  (higher step – coarser mesh, lower order of  $\mathbf{A}$ ). Hence the eigenvalues of matrix  $\mathbf{A}$  (necessary for the iterative solving of a system of linear algebraic equations) are dependent of  $h$ .

- (4) If the contour problem has a unique solution, and  $h$  is sufficiently small,  $\mathbf{A}$  is a non-singular matrix, thus the system of linear algebraic equations has a unique solution.

The rounding error in the case the central differences are used to approximate a Laplacian,  $u_{xx} + u_{yy}$  on a rectangular mesh,  $(x_m, y_n) = (mh, nk)$ , can be expressed as (assuming that  $u_{xxxx}$  and  $u_{yyyy}$ ):

$$-\frac{h^2}{12}U_{xxxx}(\tilde{x}, y_n) - \frac{k^2}{12}U_{yyyy}(x_m, \tilde{y}) = O(h^2 + k^2) \quad (1.59)$$

It follows from the above that if the solution of a contour problem for the Poisson equation has fourth-order derivatives equal to  $\equiv 0$ , as in the case of  $u = xy$ , then the solution obtained by finite differences is exact.

### 1.1.9 Neumann problem

Let us now also consider the Neumann problem, given as:

$$u_{xx} + u_{yy} = f(x, y) \quad \text{on the domain } \Omega \quad (1.60)$$

and

$$\frac{\partial u}{\partial n} = g(x, y) \quad \text{on the boundary } S, \quad (1.61)$$

where  $\Omega$  represents a rectangle  $0 < x < a$ ,  $0 < y < b$ . Mesh nodes will be selected so that  $Mh = a$  and  $Nh = b$ . Based on (1.61) it follows that:

$$-U_{m-1,n} - U_{m,n-1} + 4U_{mn} - U_{m+1,n} - U_{m,n+1} = -h^2 f_{mn}, \quad (1.62)$$

wherein the rounding error is  $O(h^2)$ ,  $m = 0, 1, \dots, M$  and  $n = 0, 1, \dots, N$ , and  $f(x, y)$  is defined in  $S$  as well. The main characteristic of Neumann problem solution is that the derivative  $\frac{\partial u}{\partial n}$  representation via central differences is achieved by introducing primary nodes (Figure 1.1), which are not located in the mesh nodes (domain):

$$U_{M+1,n} - U_{M-1,n} = 2hg_{Mn} \quad (n = 1, 2, \dots, N-1), \quad (1.63)$$

$$U_{m,N+1} - U_{m,N-1} = 2hg_{mN} \quad (m = 1, 2, \dots, M-1), \quad (1.64)$$

$$U_{-1,n} - U_{1n} = 2hg_{0n} \quad (n = 1, 2, \dots, N-1) \quad (1.65)$$

$$U_{m,-1} - U_{m1} = 2hg_{m0} \quad (m = 1, 2, \dots, M-1). \quad (1.66)$$

In mesh nodes, where the normal is not determined, we will adopt the mean value of the two nearest normal, which provides additional four boundary conditions:

$$U_{-1,0} + U_{0,-1} = U_{10} + U_{01} + 4hg_{00}, \quad (1.67)$$

$$U_{M,-1} + U_{M+1,0} = U_{M1} + U_{M-1,0} + 4hg_{M0}, \quad (1.68)$$

$$U_{M+1,N} + U_{M,N+1} = U_{M-1,N} + U_{M,N-1} + 4hg_{MN}, \quad (1.69)$$

$$U_{0,N+1} + U_{-1,N} = U_{0,N-1} + U_{1,N} + 4hg_{0N}. \quad (1.70)$$

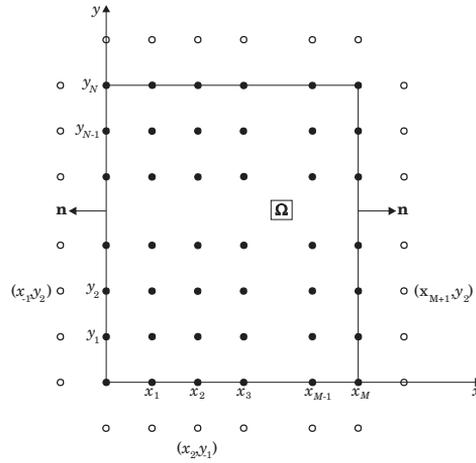


Figure 1.1: Finite difference mesh.

If we now assume that  $n = N = 2$  and  $g = 0$ , we can solve the system of linear algebraic equations  $\mathbf{AU} = \mathbf{B}$ , which includes equations (1.63)-(1.70) and is given below:

$$\begin{bmatrix} 4 & -2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -2 & 0 & 0 & 0 & 0 \\ 0 & -2 & 4 & 0 & 0 & -2 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -2 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -2 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -2 & 0 & 0 & 4 & -2 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & -2 & 4 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \\ U_9 \end{bmatrix} = -h^2 \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \end{bmatrix} \quad (1.71)$$

Since the sum of matrix terms in each row is 0, the unit vector column  $\mathbf{C} = [1, 1, \dots, 1]^T$  satisfies the condition that  $\mathbf{AC} = \mathbf{0}$ . Thus, the equation  $\mathbf{AU} = \mathbf{0}$  has a solution different from zero, which implies that  $\mathbf{A}$  is singular. Hence, **Neumann problem** might not have a solution, but could also have an infinite amount of them, in the form of  $\mathbf{U} = \mathbf{U}^* + \alpha\mathbf{C}$ . By dividing the I, III, VII and IX rows of matrices  $\mathbf{A}$  and  $\mathbf{B}$  by 2, and multiplying row V by 2, we obtain a new system of linear algebraic equations,  $\mathbf{A}'\mathbf{U} = \mathbf{B}'$ : as follows:

$$\begin{bmatrix} 2 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -2 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -2 & 0 & -1 & 0 & 0 \\ 0 & -2 & 0 & -2 & 8 & -2 & 0 & -2 & 0 \\ 0 & 0 & -1 & 0 & -2 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \\ U_9 \end{bmatrix} = -h^2 \begin{bmatrix} f_1/2 \\ f_2 \\ f_3 \\ f_4 \\ 2f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9/2 \end{bmatrix} \quad (1.72)$$

Since for this system of linear algebraic equations we have that  $\mathbf{A}' = \mathbf{A}'^T$  and  $\mathbf{A}'\mathbf{C} = 0$ , if the

solution  $U$  exists, then the following holds:

$$\mathbf{B}'\mathbf{C} = (\mathbf{A}'\mathbf{U})^T\mathbf{C} = \mathbf{U}^T(\mathbf{A}'\mathbf{C}) = 0, \tag{1.73}$$

which suggests that the sum of  $\mathbf{B}'$  terms is zero. This condition is equivalent to:

$$\int_{\Omega} f(x,y) \, dx \, dy = 0, \tag{1.74}$$

if the integral is calculated using the trapeze rule in nine introduced nodal points (fig. 1.1) Thus, this condition ( $\int_{\Omega} f \, d\Omega = \int_S g \, ds = 0$ ) is sufficient for the existence of a (non-unique) solution of the system  $\mathbf{A}'\mathbf{U} = \mathbf{B}'$ .

### 1.1.10 Curvilinear boundaries

We will now demonstrate how to apply finite differences to the following problem:

$$u_{xx} + u_{yy} = f(x,y) \quad \text{nad } \Omega \tag{1.75}$$

$$u = g(x,y) \quad \text{na } S \tag{1.76}$$

in the case that  $S$  represents a curvilinear boundary of domain  $\Omega$ . In each node of mesh defined in  $\Omega$ , which is surrounded by four nodes also within the same domain, usual expressions for finite difference method are applied.

Let us now consider nodes for which at least one adjacent node is not in  $\Omega$ , e.g.  $P = (x_m, y_n)$ , fig. 1.2.

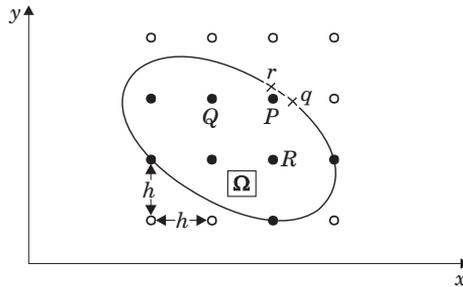


Figure 1.2: Finite difference mesh for curvilinear domain.

Coordinates of points  $r$  and  $q$  on contour  $S$ , obtained by drawing a horizontal and an orthogonal line through  $P$ , are  $(x_m + \alpha h, y_n)$  and  $(x_m, y_n + \beta h)$ , respectively, where  $0 < \alpha, \beta < 1$ . Since  $u$  is known at  $S$ ,  $u(r)$  and  $u(q)$  are also known. By deriving into a Taylor series, we obtain:

$$u(q) = u(P) + \alpha h u_x(P) + \frac{\alpha^2 h^2}{2} u_{xx}(P) + O(h^3), \tag{1.77}$$

$$u(Q) = u(P) - h u_x(P) + \frac{h^2}{2} u_{xx}(P) + O(h^3). \tag{1.78}$$

By summing we obtain:

$$u_{xx}(P) = \frac{\alpha u(Q) - (1 + \alpha)u(P) + u(q)}{h^2 \alpha (\alpha + 1)/2} + O(h). \tag{1.79}$$

In an analogous manner, we obtain

$$u_{yy}(P) = \frac{\beta u(R) - (1 + \beta)u(P) + u(r)}{h^2 \beta (\beta + 1)/2} + O(h). \quad (1.80)$$

By summing we obtain an approximation of  $O(h)$  order of Poisson's equations in  $P$ :

$$\frac{U(Q)}{\alpha + 1} + \frac{U(R)}{\beta + 1} - \left( \frac{1}{\alpha} + \frac{1}{\beta} \right) U(P) + \frac{U(q)}{\alpha(\alpha + 1)} + \frac{U(r)}{\beta(\beta + 1)} = \frac{h^2}{2} f(P). \quad (1.81)$$

Based on the expression derived above, the problem can be solved as follows:

$$u_{xx} + u_{yy} = 0 \quad \text{na} \quad x^2 + y^2 < 1 \quad y > 0, \quad (1.82)$$

$$u(x, y) = 100 \quad x^2 + y^2 = 1 \quad y > 0, \quad (1.83)$$

$$u(x, y) = 0 \quad y = 0 \quad -1 < x < 1, \quad (1.84)$$

By using a square mesh, with  $h = 0.5$ . Thanks to its  $y$ -axis symmetry, the number of unknowns in the problem is reduced from three to two, since the semi-circular domain can be reduced to a quarter-circle, Fig. 1.3.

Based on boundary conditions, we obtain:

$$U_{00} = 0, U_{10} = 0, U_{02} = 100, U(q) = 100, U(r) = 100, U_{11} = U_{-1,1}.$$

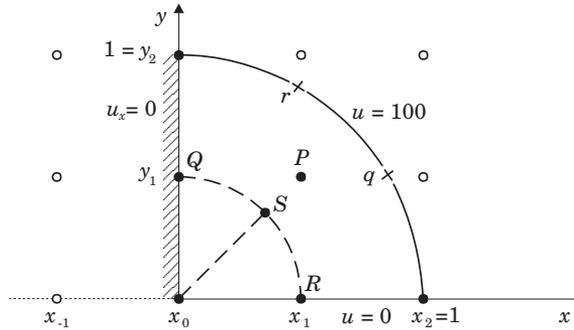


Figure 1.3: Half circle domain, reduced to circle quarter.

The only nodes in the mesh where the solution needs to be approximated are  $P(x_1, y_1)$  and  $Q(x_0, y_1)$ , which is achieved by using a finite difference method in  $Q$ :

$$U_{11} + 100 - 4U_{01} + U_{-1,1} + 0 = 0.$$

Taking into account the boundary conditions, we obtain

$$2U_{01} - U_{11} = 50.$$

Coordinates  $q$  and  $r$  are  $(\sqrt{3}h, h)$  and  $(h, h\sqrt{3}) \Rightarrow \alpha = \beta = \sqrt{3} - 1$ , thus the following is obtained:

$$\frac{U_{01}}{\sqrt{3}} + \frac{U_{10}}{\sqrt{3}} - \frac{2U_{11}}{\sqrt{3} - 1} + \frac{U(q)}{3 - \sqrt{3}} + \frac{U(r)}{3 - \sqrt{3}} = 0,$$

from which follows that

$$(1 - \sqrt{3})U_{01} + 2\sqrt{3}U_{11} = 200.$$

By solving the previous system of equations, we obtain:

$$U_{01} \equiv U(Q) = \frac{100(2 + \sqrt{3})}{1 + 3\sqrt{3}},$$

$$U_{11} \equiv U(P) = \frac{50(7 + \sqrt{3})}{1 + 3\sqrt{3}}.$$

## 2. Finite element method

Finite element method is one of the basic methods for obtaining approximate solutions for various types of boundary problems. It involves the discretization of the solution domain into subdomains called finite elements. Approximate solutions in these finite element sets is then obtained using variation principles. Thanks to this general approach, finite element method became a widely used tool in solving numerous problems in various fields, such as mathematical physics, aerospace engineering, structural analysis, etc.

The best way to understand the concepts of finite element method is to start with a one-dimensional problem, hence such problems will be the first to be analysed here.

### 2.1 Finite element application to solving of one-dimensional problems

In physics, most problems can be represented using two functions which need to be determined, the state variable  $u$  and the flux  $P$ . These two functions are related to each other via constitutive equations, which contain all necessary data about the material wherein the process takes place. These equations can be related to various laws of physics, but are always of the same mathematical shape, which can be represented in the following way, for the case of elastic stresses in a bar, where  $u$  is the displacement, and  $P$  is the stress:

$$P(z) = -k(z) \frac{du(z)}{dz}. \quad (2.1)$$

The **constitutive equation** shown above is known as **Hooke's law** modulus (material property related to its stiffness). This equation (2.1) indicates that the non-uniformity of a state variable (displacement) results in the occurrence of a flux (stress in this case), i.e. that this flux is proportional

to the rate of change of variable ( $u$ ). According to conservation laws, the total flux entering each subdomain is equal to zero, and the flux can enter the subdomain of a body in one of two ways:

- As an internal source, defined by function  $f$  and
- Through an external boundary.

Depending on the nature of the problem, internal sources of flux include volumetric forces  $f$  (our case), heat sources, fluid sources, etc. In order for the constitutive equations to hold, the state variable  $u$  (displacement), needs to be a continuous function of  $z$ , with defined values at both ends of its one-dimensional domain.

Now, let us explain the process of discretization into finite elements, using a one-dimensional body (bar), on a domain  $0 < z < \ell$  (along the length of the bar). The bar is made of two materials, one of which has a domain of  $0 < z < z_1$  and the other has  $z_1 < z < \ell$ . Material modulus  $k$  has a simple discontinuity in point  $z_1$ , but it remains continuous in all points left and right of it. Internal source distribution (function)  $f(z)$  also has a discontinuity at a point  $z_2$  where a concentrated force with magnitude  $\bar{f}$ , represented by a Dirac delta function is acting. In addition, let us assume there is a simple source distribution discontinuity at point  $z_3$ . Thus, in order to preserve the continuity of the domain, we can divide it into four subdomains, and 5 nodes connecting them. In this way, the quantities within the subdomain are continuous, since their corresponding discontinuities are located within the nodes between them.

Let us now consider some of the basic terms of mathematical physics:

(1) The flux must be maintained in every point within a body (bar).

Let  $\bar{z}$  be an arbitrary point within a subdomain in the interval  $a < \bar{z} < b$ , Fig. 2.1b. Boundary fluxes in this area are denoted by arrows in the figure, and this area also contains internal force sources  $f(z)$ . Thus, the conservation law is given by:

$$P(b) - P(a) = \int_a^b f(z) dz. \quad (2.2)$$

(2) The flux is continuous in all subdomain points.

The limit value of expression (2.2) when  $a \rightarrow \bar{z}^-$  and  $b \rightarrow \bar{z}^+$  (limits  $a$  and  $b$  are contracted into point  $\bar{z}$ ) is:

$$\lim_{b \rightarrow \bar{z}^+} P(b) - \lim_{a \rightarrow \bar{z}^-} P(a) = 0, \quad (2.3)$$

since  $f(z)$  is a limited function within the  $a < z < b$  interval. If the term jump function is introduced at point  $\bar{z}$

$$\|P(\bar{z})\| = \lim_{b \rightarrow \bar{z}^+} P(b) - \lim_{a \rightarrow \bar{z}^-} P(a) \quad (2.4)$$

we can write the following:

$$\|P(\bar{z})\| = 0, \quad (2.5)$$

which indicates that, in the case there are no jumps, the flux is continuous in all point within a subdomain  $\Omega_i$ ,  $i = 1, 2, 3, 4$ .

Table 2.1: Tabelle.

Fizički problem	Princip održanja	Promenljiva stanja $u$	Fluks $\sigma$	Materijalni moduli $k$	Izvori $f$	Konstitutivne jednačine $\sigma = -ku'$
Deformacija elastičnog štapa	Ravnoteža sila	Pomeranje	Napon	Jangov modul	Zapreminske sile	Hukov zakon
Toplotna provodljivost u štapu	Održanje energije	Temperatura	Toplotni fluks	Termo provodljivost	Toplotni izvor	Furijeov zakon
Tečenje fluida	Zakon količine kretanja	Brzina	Napon smicanja	Viskoznost	Zapreminske sile	Stoksov zakon
Elektrostatika	Održanje električnog fluksa	Električni potencijal	Električni fluks	Dielektrična propustljivost	Naelektrisanje	Kulonov zakon
Proticanje kroz poroznu sredinu	Održanje mase	Hidraulični pritisak	Brzina proticanja	Propustljivost	Izvor fluida	Darsijev zakon

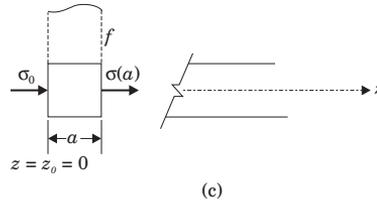
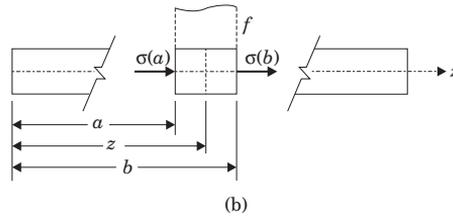
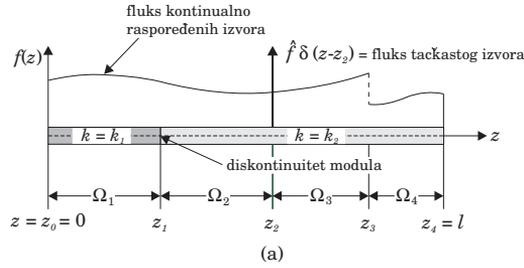


Figure 2.1: One-dimensional problem with 5 nodes and 4 elements.

(3) Equation describing this problem is a second-order ordinary differential equation.

Since  $f(z)$  is continuous, it follows in accordance with the mean integral values theorem that:

$$\int_a^b f(z) dz = (b-a)f(\xi), \text{ where } a < \xi < b \text{ and } f(\xi) \text{ is the mean value of } f(z) \text{ at the } (a, b) \text{ interval.}$$

It follows that:

$$P(b) - P(a) = (b-a)f(\xi), \quad (2.6)$$

i.e.

$$\lim_{\substack{a \rightarrow \bar{z}^- \\ b \rightarrow \bar{z}^+}} \frac{P(b) - P(a)}{b-a} = \lim_{\substack{a \rightarrow \bar{z}^- \\ b \rightarrow \bar{z}^+}} f(\xi).$$

As  $f(z)$  is continuous, we have:

$$\lim_{\substack{a \rightarrow \bar{z}^- \\ b \rightarrow \bar{z}^+}} f(\xi) = f(\bar{z}).$$

In addition:

$$\lim_{\substack{a \rightarrow \bar{z}^- \\ b \rightarrow \bar{z}^+}} \frac{P(b) - P(a)}{b-a} = \frac{dP(\bar{z})}{dz},$$

hence we obtain the balance equation:

$$\frac{dP(z)}{dz} = f(z). \quad (2.7)$$

By replacing the constitutive equation (2.1) into the balance equation (2.6), we obtain an ordinary differential equation for the observed one-dimensional elliptical problem:

$$-\frac{d}{dz} \left[ k(z) \frac{du(z)}{dz} \right] = f(z). \quad (2.8)$$

At the points where the material module  $k$  is continuous, equation (2.8) can be represented as a second-order linear differential equation in the following way:

$$-k(z) \frac{d^2u(z)}{dz^2} - \frac{dk(z)}{dz} \frac{du(z)}{dz} = f(z). \quad (2.9)$$

(4) Let us now consider nodes with discontinuities, starting with  $z = z_3$ .

We have previously shown that there are no jumps in the flux:

$$\llbracket P(z_3) \rrbracket = 0 \quad z = z_3. \quad (2.10)$$

Mean value theorem cannot be applied in this case, due to integrand discontinuity:  $ku'$  is continuous, but  $(ku)'$  is not defined. Thus, there is no differential equation for  $z = z_3$ !

At point (node)  $z = z_1$ , where  $f$  is continuous, but  $k$  is not, the flux zero jump condition,  $\llbracket P(z_1) \rrbracket = 0$ , also holds. Equation (2.7) is satisfied, however since  $k(z)$  is not differentiable at  $z_1$ , equation (2.8) cannot be transformed into (2.9).

(5) At point  $z = z_2$ , where a concentrated force  $f = \hat{f}\delta(z - z_2)$  is acting, flux balance equation for the area around  $z_2$  is given by:

$$P(b) - P(a) = \int_a^b f(z) dz = \int_a^b \bar{f}(z) dz = \int_a^b \hat{f}\delta(z - z_2) dz \quad (2.11)$$

where  $\bar{f}$  represents the smooth part of  $f$ . Same as in the previous case, we obtain a homogeneous jump condition (integral with respect to  $\bar{f}$  has a limit of 0):

$$\llbracket \llbracket \rrbracket = \hat{f} \quad \text{for } z = z_2 \quad (2.12)$$

where  $\hat{f}$  does not depend on  $a$  and  $b$ . This is the exact reason why we cannot obtain a differential equation at  $z = z_2$ .

(6) Finally, let us consider the boundary conditions, i.e. conditions at point  $z = z_0$  and  $z = z_4$ .

In reality, all physical systems must have some kind of interaction with their environment, which is usually included in the system equations by defining the flux or state variable values on body's boundaries. Hence, in a general case, approximate modelling of the body's is necessary in order to define the boundary conditions.

Shown in Figure 2.1c, is a small area of a bar which contains the left boundary, where the flux is defined as  $P_0$ . Flux in this area is conserved if the following holds:

$$P(a) - P(0) = \int_0^a f(z) dz, \quad (2.13)$$

since in the boundary case, when  $a \rightarrow 0$ ,  $P(0) = P_0$ . When defining the flux in the boundary points, we must take into account that it is necessary to know the direction of displacement change rate in

these points, in order to know whether the flux is entering or exiting the body. In other words, we define the following:

$$-k(0) \left( -\frac{du(0)}{dz} \right) = P_0, \quad (2.14)$$

$$-k(\ell) \frac{du(\ell)}{dz} = P_\ell, \quad (2.15)$$

where  $\frac{du(0)}{dz}$  and  $\frac{du(\ell)}{dz}$  is obviously derived from the same side.

In many cases, the boundary flux is known, thus equations (2.14) and (2.15) become boundary conditions themselves. For second-order equation (e.g. equation (2.9)), these boundary conditions, which contain the first derivative of the unknown  $u$  are called **natural boundary condition**. In some other cases, it is assumed that the boundary flux is proportional to the difference in values of state variables at the boundaries and its values at a certain distance in its vicinity, which requires a simplified constitutive equation for the environment. For example, for  $z = z_0 = 0$ , this conditions is given in the following form:

$$P_0 = p_0[u(0) - u_0], \quad (2.16)$$

where  $p_0$  is a known constant dependent on the material module of the environment, and  $u_0$  is the known value of environment state variable. By replacing equation (2.16) into (2.14), we obtain:

$$k(0) \frac{du(0)}{dz} = p_0 [u(0) - u_0].. \quad (2.17)$$

Since  $u'(0)$  once again appears here, the above equation is also a natural boundary condition for equation (2.8).

The selection of boundary conditions in a boundary problem is, of course, influenced by its physical nature. For elastic deformation, this selection involves:

- **Essential boundary conditions**, which define the boundary displacements.
- **Natural boundary conditions**, which define:
  - (a) boundary stresses, equations (2.14) and (2.15) or
  - (b) linear combination of stresses and displacement at the boundary, Eq. (2.17).

It should be noted that when the problem requires a stress-related boundary condition (flux) at both boundaries, the defined value of boundary stress must fulfil the global conservation condition:

$$P_l + P_0 = \int_0^l f(z) dz. \quad (2.18)$$

We can now formulate the general physical problem:

$$-\frac{d}{dz} \left[ k(z) \frac{du(z)}{dz} \right] = f(x), \quad z \in \Omega_i, \quad i = 1, 2, 3, 4, \quad (2.19)$$

with the following jump conditions in discontinuity points:

$$\|P(z_1)\| = 0, \quad \|P(z_2)\| = \hat{f}, \quad \|P(z_3)\| = 0 \quad (2.20)$$

and the following boundary conditions:

$$\begin{aligned} u(z) &= u_0 \quad \text{or} \quad u_l && \text{for } z = 0 \text{ or } z = l \\ P(z) &= -P_0 && \text{for } z = 0 \\ P(z) &= P_l && \text{for } z = l \\ P(z) - p_0 u(z) &= -p_0 u_0 && \text{for } z = 0 \\ P(z) - p_l u(z) &= -p_l u_l && \text{for } z = l. \end{aligned} \quad (2.21)$$

### 2.1.1 Variation formulation

Classic representation of boundary problems using equations (2.18)-(2.20) requires solution regularity which is too strong for most real problems. In addition, equation (2.18) itself is not satisfied in discontinuity points  $(z_1, z_2, z_3)$ , hence practical needs of numerical solving require nothing more than the use of weaker conditions for function  $u$  and its derivatives. Such approach to problem solving is referred to as weak or variation formulation, and is applicable to non-smooth data and solutions. Of course, if a smooth ("classical") solution to the problem exists, it also represents the solution of the weak problem. Hence, using this approach does not result in any losses, and allows the solving of most commonly encountered practical problems.

Variation formulation is based on the need to find a function  $u$ , such that differential equation (2.18) and boundary conditions (2.20) are satisfied in the sense of pondered mean values:

$$\begin{aligned} \int_0^l -(ku')'v \, dz &= \int_0^l f v \, dz \\ \int_0^l -(ku')'v \, dz &= \int_0^l -(ku'v)' \, dz + \int_0^l ku'v' \, dz = \\ &= -ku'v \Big|_0^l + \int_0^l ku'v' \, dz. \end{aligned} \tag{2.22}$$

In expression (2.21),  $v$  is the **weight function** of  $z$  which is sufficient to ensure that integrals in this expression are of finite value. The set of all weight functions will be denoted as  $H$ . With this, we can provide a compact problem formulation as follows:

$$\int_0^l [(ku')' + f]v \, dz = 0 \quad \text{za } v \in H, \tag{2.23}$$

$$u(0) = u_0 \quad u(l) = u_l. \tag{2.24}$$

If  $u$  and  $v$  functions are sufficiently smooth (at least in their subdomains), then partial integration provides the following:

$$\int_0^l -(ku')'v \, dz = \int_0^l ku'v' \, dz - ku'v \Big|_0^l. \tag{2.25}$$

Having in mind the constitutive equation (??) we can replace  $ku'$  with  $P$ , and by further replacing (2.25) into (2.23), we obtain:

$$\int_0^l ku'v' \, dz = Pv \Big|_0^l + \int_0^l f v \, dz. \tag{2.26}$$

Solution  $u$  belongs to a set  $\tilde{H}$ , called the **test function class**. This class's functions must fulfil the essential boundary conditions, whereas natural boundary conditions occur automatically during the variation formulation of this problem. It should be noted that weight functions in points where essential BCs are defined equal zero.

Taking into account that functions  $u$  and  $v$  appear in equation (2.26), they can be chosen from the same class of allowable functions

$$H = \tilde{H} = H^1. \tag{2.27}$$

Since  $v$  can represent any given function from the allowable function set, we will consider a variant where  $u = v (= w)$ . For simplicity's sake, we will consider a case where  $k = \text{const}$ . Obviously,  $H^1$  represents a class of functions  $w$ , where "1" denoted that all of its terms have first derivatives, whose squares are integrable at the interval of  $0 < z < l$ :

$$\int_0^l (w')^2 \, dz < \infty. \tag{2.28}$$

### 2.1.2 Galerkin's method

Assuming that weak (variation) formulation given by (2.26), which holds for all functions  $v \in H^1$ . Function  $H^1$ , in addition to its aforementioned properties, is a linear function space, with an infinite number of dimensions.

The term **linear space** means that any linear combination of functions from  $H^1$  is also a function from  $H^1$ . Thus, if  $v_1$  and  $v_2$  are weight functions, then  $\alpha v_1 + \beta v_2$  is also a weight function.

The term **infinitely dimensional** means that determining of function  $v$  in a space requires an infinite number of parameters.

Let us assume that an infinite set of functions  $\phi_1(z), \phi_2(z) \dots$  in  $H^1$ , wherein every weight function can be expressed as a linear combination of functions  $\phi_i(z)$ .

$$v(z) = \sum_{i=1}^{\infty} \beta_i \phi_i(z), \quad (2.29)$$

where  $\beta_i$  are constants, and series defined by (2.29) converges "in the sense of  $H^1$ ". Set of functions  $\phi_i$ , which satisfy this condition, provides a base for  $H^1$ , hence such functions are referred to as base functions. If we assume a finite number of series terms  $N$ , instead of an infinite amount, we obtain the approximation function of  $v$ , denoted as  $v_N$ :

$$v_N(z) = \sum_{i=1}^N \beta_i \phi_i(z). \quad (2.30)$$

In the case  $v_N$  is given by (2.30), then  $v_N$  converges towards  $v$  "in the sense of  $H^1$ ", if the following holds:

$$\lim_{N \rightarrow \infty} \int_0^l (v' - v'_N)^2 dz = 0. \quad (2.31)$$

A total of  $N$  base functions  $\{\phi_1, \phi_2 \dots \phi_N\}$  defines an  $N$ -dimensional subspace  $H^{(N)}$  of space  $H^1$ , since every function  $\phi_i$ ,  $i = 1, \dots, N$  is a member of  $H^1$ . Additionally, let us assume that these functions are linearly independent. In this case, we can formulate Galerkin's method as a means of determining the approximate solution of equation (2.26) in a finite-dimensional subspace  $H^{(N)}$  of space  $H^1$ , which contains allowable functions (rather than the entire  $H^1$  space). Thus, instead of observing an infinitely-dimensional problem, we seek an approximate solution in the following form:

$$u_N(z) = \sum_{i=1}^N \alpha_i \phi_i(z) \quad \phi_i(z) \in H^{(N)}, \quad (2.32)$$

the above form satisfies the conditions given by (2.26) in the case  $H^1$  is replaced by  $H^{(N)}$ , where  $\alpha_i$  represent approximation degrees of freedom. We can now formulate the variation base for the approximate problem: find a function  $u_N \in H^{(N)}$ , so that the following holds:

$$\int_0^l k u'_N v'_N dz = P v_N \Big|_0^l + \int_0^l f v_N dz \quad \text{for all } v_N \in H^{(N)}. \quad (2.33)$$

By replacing (2.30) and (2.32) into (2.33), we obtain:

$$\int_0^l k \left( \sum_{i=1}^N \beta_i \phi_i \right)' \left( \sum_{j=1}^N \alpha_j \phi_j \right)' dz = P \sum_{i=1}^N (\beta_i \phi_i) \Big|_0^l + \int_0^l f \sum_{i=1}^N \beta_i \phi_i dz, \quad (2.34)$$

In other words, we have:

$$\sum_{i=1}^N \beta_i \left( \sum_{j=1}^N \int_0^l k \phi'_i \phi'_j dz \alpha_j - P \phi_i \Big|_0^l - \int_0^l f \phi_i dz \right) = 0, \quad \forall \beta_i, \quad (2.35)$$

for arbitrary  $\beta_i$ .

This expression can also be represented in a simplified form

$$\sum_{i=1}^N \beta_i \left( \sum_{j=1}^N K_{ij} \alpha_j - F_i \right) = 0, \quad (2.36)$$

where

$$K_{ij} = \int_0^l k \phi_i' \phi_j' dz \quad (2.37)$$

$$F_i = P \phi_i \Big|_0^l + \int_0^l f \phi_i dz. \quad (2.38)$$

$K_{i,j}$  is the **stiffness matrix** of the problem for **base functions**  $\phi_i$ , of  $N \times N$ -th order, and  $F_i$  is the **load vector**, for the same base functions, of the  $N$ -th order. Since  $\beta_i$  values are arbitrary, the condition below needs to be fulfilled in order for equation (2.36) to hold:

$$\sum_{j=1}^N K_{ij} \alpha_j = F_i \quad i = 1, 2, \dots, N. \quad (2.39)$$

Since functions  $\phi_i$  were selected as independent, equations (2.39) are also independent, hence matrix  $K_{ij}$  is invertible, and coefficients  $\alpha_j$  now become:

$$\alpha_j = \sum_{i=1}^N (K)_{ij}^{-1} F_i. \quad (2.40)$$

Once these coefficients are obtained, the approximate solution  $u_N$  are also known, based on equation (2.32). Thus, it can be seen that Galerkin's method represents a very powerful tool, but only if a systematic technique for constructing of base functions is present. Hence, this method only became widely applied once finite element method was introduced, as will be shown in the following paragraphs.

### 2.1.3 Finite element base functions

There are significant difficulties in selecting of base functions, since apart from the fact that the members of space  $H_0^1$  are independent, nothing else is known about them. The situation is further complicated with the increase of the number of dimensions of the boundary problem, since this makes the boundary conditions considerably more complex. Additionally, taking into account the dependence of the solution and the finite element on the selected functions  $\phi_i$ , it is clear that the selection of base functions is of crucial importance in practical applications of the Galerkin's method. In order to demonstrate how these base functions are determined, we will start by considering a homogeneous differential equation:

$$(ku)' = 0 \quad z \in \Omega_i \quad \Omega_i = [-h/2, h/2], \quad (2.41)$$

The exact solution of this equation is given as:

$$ku + C_1 z + C_0 = 0. \quad (2.42)$$

If we introduce the following boundary conditions:

$$u = u_i \quad \text{for } z = -h/ \quad (2.43)$$

$$u = u_{i+1} \quad \text{for } z = h/2, \quad (2.44)$$

we obtain the following expressions for constants  $C_1$  and  $C_0$ :

$$C_1 = -\frac{k}{h}(u_{i+1} - u_i) \quad (2.45)$$

$$C_0 = -\frac{k}{2}(u_i + u_{i+1}), \quad (2.46)$$

hence the exact solution is:

$$u = \frac{1}{2} \left( 1 + \frac{2z}{h} \right) u_{i+1} + \frac{1}{2} \left( 1 - \frac{2z}{h} \right) u_i \quad (2.47)$$

If we now introduce the non-dimensional constant  $\xi = \frac{2z}{h}$  (for  $z = -h/2$ ,  $\xi = -1$ ;  $z = h/2$ ,  $\xi = 1$ ), it follows that:

$$u = \left[ \frac{1}{2}(1 - \xi) \quad \frac{1}{2}(1 + \xi) \right] \begin{bmatrix} u_i \\ u_{i+1} \end{bmatrix}. \quad (2.48)$$

If we introduce the so-called interpolation functions, defined as  $\psi_1 = \frac{1}{2}(1 - \xi)$  and  $\psi_2 = \frac{1}{2}(1 + \xi)$ , we can observe that they define a linear interpolation of displacement  $u$  (state variable) between nodes  $i$  and  $i + 1$  in a system, i.e. elements 1 and 2, Fig. 2.2. It is easily noticeable that these two functions have values of 1 on one end, and 0 on the other.

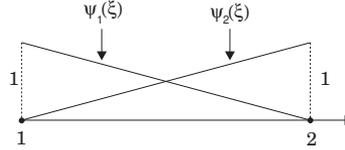


Figure 2.2: Linear interpolation function.

It is also evident that these two functions are linearly independent and continuous at the element boundaries. The second conclusion follows from the fact that two base functions, constructed from two adjacent segments of an interpolation function in a node, have the ordinate value of 1 in their own corresponding nodes.

Let us now observe the non-homogeneous differential equation with a constant term:

$$(ku')' = C_2. \quad (2.49)$$

This equation physically corresponds to the elastic stress state problem with a constant volumetric force.

The exact solution of equation (2.49-53) is:

$$ku + \frac{1}{2}C_2z^2 + C_1z + C_0 = 0. \quad (2.50)$$

Formally, we can consider  $C_2$  as the unknown, as long as the displacement value  $u$  is defined in one of the points within the interval, e.g. in its middle, in addition to the previously defined boundary conditions. Boundary conditions in this case can be written as:

$$u = u_{i-1} \quad \text{for } z = -h/2 \quad (2.51)$$

$$u = u_i \quad \text{for } z = 0 \quad (2.52)$$

$$u = u_{i+1} \quad \text{for } z = h/2. \quad (2.53)$$

By replacing the boundary conditions (2.51) - (2.53) into equation (2.50), and solving for  $C_0$ ,  $C_1$  and  $C_2$ , we obtain:

$$C_0 = -ku_i, \tag{2.54}$$

$$C_1 = -\frac{k}{h}(u_{i+1} - u_{i-1}), \tag{2.55}$$

$$C_2 = -\left(\frac{2}{h}\right)^2 k(u_{i+1} - 2u_i + u_{i-1}). \tag{2.56}$$

By replacing (2.54) - (2.56) into (2.50) and introducing  $\xi = 2z/h$  we obtain:

$$u = \left[ -\frac{1}{2}\xi(1-\xi) \quad (1-\xi)(1+\xi) \quad \frac{1}{2}\xi(1+\xi) \right] \cdot \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}. \tag{2.57}$$

We can now introduce  $\psi_1 = -\frac{1}{2}\xi(1-\xi)$ ,  $\psi_2 = (1-\xi)(1+\xi)$ , and  $\psi_3 = \frac{1}{2}\xi(1+\xi)$ , i.e. the square interpolation functions, Fig. 2.3.

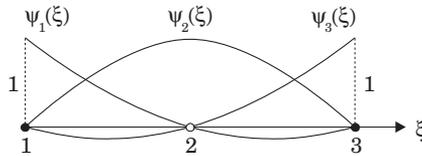


Figure 2.3: Quadratic interpolation function.

It is possible to introduce higher order interpolation functions in the same way, i.e. in the general case, **Lagrange interpolation** can be applied to an interval of  $[-1, 1]$ , wherein we adopted a non-dimensional coordinate system  $\xi = 2z/h$  in the case that all elements have the same length  $h$ . In this general case, the non-dimensional coordinate  $\xi$  is introduced using the following expression:

$$\xi = \frac{2z - (z_{k+1} + z_1)}{z_{k+1} - z_1}. \tag{2.58}$$

In accordance with designations shown in Fig. 2.4.

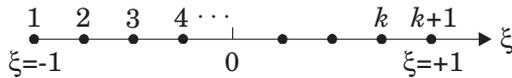


Figure 2.4: Non-dimensional coordinates  $\xi$ .

In the general case, the  $i$ -th Lagrange interpolation function of the  $k$ -th order is:

$$\psi_i(\xi) = \frac{(\xi - \xi_1)(\xi - \xi_2) \dots (\xi - \xi_{i-1})(\xi - \xi_{i+1}) \dots (\xi - \xi_{k+1})}{(\xi_i - \xi_1)(\xi_i - \xi_2) \dots (\xi_i - \xi_{i-1})(\xi_i - \xi_{i+1}) \dots (\xi_i - \xi_{k+1})}. \tag{2.59}$$

Apparently, the following holds:

$$\psi_i(\xi_j) = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}, \tag{2.60}$$

From which it follows that functions  $\psi_i$  are linearly independent. This total of  $k + 1$  functions define the base for the set of  $k$ -th order polynomials, as well as for those of lesser order, hence we can say that this base is complete, which means that any and all polynomials of the  $k$ -th order (or lower) can be uniquely represented using the Lagrange interpolation functions. In other words, Lagrange interpolation functions can be used as the base functions for the finite element.

### Example 2.2

For  $k = 1$  ( $\xi_1 = -1, \xi_2 = 1$ )

$$\psi_1 = \frac{\xi - \xi_2}{\xi_1 - \xi_2} = \frac{1}{2}(1 - \xi) \quad (2.61)$$

$$\psi_2 = \frac{\xi - \xi_1}{\xi_2 - \xi_1} = \frac{1}{2}(1 + \xi), \quad (2.62)$$

$$(2.63)$$

For  $k = 2$  ( $\xi_1 = -1, \xi_2 = 0, \xi_3 = 1$ )

$$\psi_1 = \frac{(\xi - \xi_2)(\xi - \xi_3)}{(\xi_1 - \xi_2)(\xi_1 - \xi_3)} = \frac{\xi(\xi - 1)}{-1(-2)} = \frac{1}{2}\xi(\xi - 1), \quad (2.64)$$

$$\psi_2 = \frac{(\xi - \xi_1)(\xi - \xi_3)}{(\xi_2 - \xi_1)(\xi_2 - \xi_3)} = \frac{(\xi + 1)(\xi - 1)}{1(-1)} = (1 - \xi^2), \quad (2.65)$$

$$\psi_3 = \frac{(\xi - \xi_1)(\xi - \xi_2)}{(\xi_3 - \xi_1)(\xi_3 - \xi_2)} = \frac{(\xi + 1)\xi}{2 \cdot 1} = \frac{1}{2}\xi(1 + \xi). \quad (2.66)$$

## 2.2.1 Interpolation error

Lagrange linear interpolation error can be determined as the difference in the values of the interpolated function  $g$  and the Lagrange interpolation function  $g_h$ :

$$E = g - g_h. \quad (2.67)$$

Let us observe an arbitrary element  $\Omega_e(z)$  defined within a mesh as  $z_i < z < z_{i+1}$  and derive a Taylor series from error  $E$ , within an element in the vicinity of an arbitrary point  $\bar{z}$  in the following way:

$$E(z) = E(\bar{z}) + E'(\bar{z})(z - \bar{z}) + \frac{1}{2}E''(\zeta)(z - \bar{z})^2, \quad (2.68)$$

where  $\zeta \in [z, \bar{z}]$ , and  $E(z_i) = E(z_{i+1}) = 0$ . Let us now select  $\bar{z}$  as a point where the error reaches its maximum value, such that  $E'(\bar{z}) = 0$

$$E(z) = E(\bar{z}) + \frac{1}{2}E''(\zeta)(z - \bar{z})^2. \quad (2.69)$$

Now, let us consider the boundary point  $z$ , which is closest to  $\bar{z}$ , e.g.  $z_i$ :

$$E(z_i) = E(\bar{z}) + \frac{1}{2}E''(\zeta)(z_i - \bar{z})^2 = 0, \quad (2.70)$$

It follows from the above expression that the absolute value of the error is equal to:

$$|E(\bar{z})| = \frac{1}{2}|E''(\zeta)|(z_i - \bar{z})^2. \quad (2.71)$$

Since  $z_{i+1} - z_i = h$ , it is clear that  $|z - \bar{z}| \leq h/2$ , hence we have the following:

$$|E(\bar{z})| \leq \frac{h^2}{8} |E''(\zeta)|. \quad (2.72)$$

Since a linear interpolation of the second derivative is considered, we obtain:

$$E'' = g'' - g_h'' = g'' \quad (2.73)$$

and by replacing into equation (2.72) and determining the maximum for all elements, we obtain:

$$\max |E(z)| \leq \frac{h^2}{8} \max |g''(z)|. \quad (2.74)$$

Since the second derivative of  $g$  is limited,  $g'' \leq c \leq \infty$ , we finally get that:

$$\|E\|_\infty = \max |E(z)| \leq Ch^2, \quad (2.75)$$

where  $C$  is a constant, independent of  $h$ . Using an analogue method for a Lagrange interpolation of the  $k$ -th order, the following is obtained:

$$\|E\|_\infty \leq Ch^{k+1}. \quad (2.76)$$

Since in the general case, Taylor series contains terms of every order up to  $k$ , it is necessary for the interpolant, as well as the interpolation functions for every element, to be able to represent each of these terms. For example, if interpolation functions contain independent terms proportional to  $z^0$  (const),  $z^2, z^3, \dots, z^k$ , but do not contain a term proportional to  $z^1$ , the error will be proportional to  $h$  instead of  $h^{k+1}$ . If the constants are missing from the interpolation functions, the convergence is non-existent. Hence, it is of great importance for the interpolation function set to contain complete polynomials.

During the determining of the error, we assumed that function  $g$  has (continuous) derivatives whose order is  $\leq k$ . However, if  $g$  only has derivatives up to order  $s$ , where  $0 < s < k$ , only the first  $s$  terms in the interpolation polynomial will participate in the approximate representation of  $g$ , regardless of  $k$ , hence instead of using expression (2.73) to define the error, we use the following one:

$$\max |g(z) - g_h(z)| \leq Ch^s. \quad (2.77)$$

It is clear that the error cannot be decrease by increasing the interpolation function order, but can still be decreased by reducing the step of mesh  $h$ .

Finally, we will give an example of finite element interpolation: for function  $g(h) = \sin(\pi x)$  at the interval  $0 \leq z \leq 1$ , using two quadratic elements, Fig. 2.5. Nodes are given by  $z = 0; 0,25; 0,5; 0,75; 1$ , whereas function values in these nodes are  $g(z) = 0; 0,707; 1; 0,707; 0$ , such that the interpolant  $g_h(z) = 0,707\phi_2(z) + \phi_3(z) + 0,707\phi_4(z)$ .

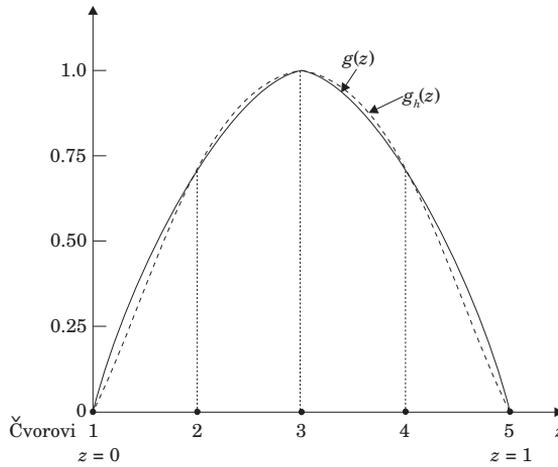


Figure 2.5: Quadratic interpolation function with two elements.

In order to evaluate the error, let us notice that,  $\max|g''(z)| \leq \pi^2$  hence:

$$|g(z) - g_h(z)| \leq Ch^3, \quad \text{gde je } C = \pi^2/48. \quad (2.78)$$

### 2.3 Finite element approximation

We can now divide the domain  $\Omega$  into a certain number of subdomains  $\Omega_e$ , with lengths  $h_e$  ( $\sum h_e = l$ ), which will be referred to as **finite elements**, as seen in Fig. 2.5, for example. Let us assume a concentrated source in  $f$ , shown in Fig. ??, located at  $z = \bar{z}$  ( $= z_2$  in Fig. ??). Then flux  $P = ku'$  has a jump of  $\|P\| = \hat{f}$  in  $\bar{z}$ . On the other hand, element shape functions have continuous derivatives and cannot include such jumps. Hence, the finite element mesh needs to be constructed in a way that ensures all discontinuities (including jumps) are located at the nodes. In this case, terms such as  $\hat{f}v_h(\bar{z})$ , which represents a jump, are not a part of local equations that describe the approximate behaviour within the elements.

Let us now consider the selection of shape functions  $\psi_i^e$ . Theoretical considerations enable the application of shape functions of any given order, with higher order functions providing more accurate solutions. However, due to practical reasons, only linear or quadratic elements are used, in order to avoid complications during finite element formulation.

### 2.4 Determining of finite element matrices and finite element system matrices

For any given finite element (subdomain)  $\Omega_i$  between nodes  $S_1$  and  $S_2$ , the following holds:

$$\int_{S_1}^{S_2} ku'v' dz = \int_{S_1}^{S_2} \bar{f}v dz + P(S_1)v(S_1) - P(S_2)v(S_2), \quad (2.79)$$

where  $P(S_i)$  is the flux in nodes  $S_1$  and  $S_2$ , and represents the natural boundary condition.

Consider a typical finite element  $\Omega_e$  with nodes  $S_1^e$  and  $S_2^e$ . Variation formulation of equation (2.79) for finite elements  $\Omega_e$ , regardless of boundary conditions in  $z = 0$  and  $z = l$  is:

$$\int_{S_1^e}^{S_2^e} k(u_h^e)'(v_h^e)' dz = \int_{S_1^e}^{S_2^e} \bar{f}v_h^e dz + P(S_1^e)v_h^e(S_1^e) - P(S_2^e)v_h^e(S_2^e), \quad (2.80)$$

where  $u_h^e$  and  $v_h^e$  are the limits for  $u_h$  and  $v_h$  in  $\Omega_e$ . Values of fluxes  $P(S_1^e)$  and  $P(S_2^e)$  are exact, rather than approximate, and represent the natural boundary conditions in their corresponding nodes.

It should be noted that equation (2.80) is defined for global coordinate  $z$ , and not the local  $\xi$  as usual, in order to make monitoring of the global stiffness matrix and load vector forming easier. Stiffness matrix and load vectors are typically determined in a local coordinate system, and then are translated into the global coordinate system using coordinate transformations, for all finite elements.

Let us observe the form of  $u_h^e$ :

$$u_h^e(z) = \sum_{j=1}^{N_e} u_j^e \psi_j^e(z), \quad (2.81)$$

where  $N_e$  is the number of nodes in the finite element  $\Omega_e$ ,  $\psi_j^e$  is the shape function, and  $u_j^e$  is the value of  $u_h^e$  and  $z = z_j^e$ .

$$u_j^e = u_h^e(z_j^e) \quad j = 1, \dots, N_e. \quad (2.82)$$

By replacing (2.81) into (2.80), and assuming that  $v_h^e = \psi_i^e$  we obtain a system of linear algebraic equations in the following form:

$$\sum_{j=1}^{N_e} k_{ij}^e u_j^e = f_i^e + P(S_1^e) \psi_i^e(S_1^e) - P(S_2^e) \psi_i^e(S_2^e), \quad (2.83)$$

where  $k_{ij}^e$  is the stiffness matrix, and  $f_i^e$  is the load vector for finite element  $\Omega_e$ :

$$k_{ij}^e = \int_{S_1^e}^{S_2^e} k(\psi_i^e)'(\psi_j^e)' dz \quad (2.84)$$

$$f_i^e = \int_{S_1^e}^{S_2^e} \bar{f} \psi_i^e dz. \quad (2.85)$$

In practice, integrals in (2.84-2.85) are not determined analytically, but numerically, with sufficient accuracy. The load vector  $f_i^e$  is usually determined using interpolants and not  $f$  itself; e.g. if  $\bar{f}$  is the continuous part of  $f$  (no concentrated forces) and if the following holds:

$$f_h^e(z) = \sum_{i=1}^{N_e} \bar{f}(z_i^e) \psi_i^e(z). \quad (2.86)$$

then instead (2.85), we use:

$$f_i^e = \int_{S_1^e}^{S_1^e} f_h^e \psi_i^e dz. \quad (2.87)$$

In this way, the load vector can be determined based on nodal values. Since we have determined the necessary matrices and finite element equations, it is now necessary to form global equation systems, which will be used to represent the body being considered here.

Let us adopt linear shape functions, i.e. two-node finite elements, defined by the following equations:

$$\begin{aligned} k_{11}^e u_1^e + k_{12}^e u_2^e &= f_1^e + P(S_1^e) \\ k_{21}^e u_1^e + k_{22}^e u_2^e &= f_2^e - P(S_2^e), \end{aligned} \quad (2.88)$$

where indices 1 and 2 are related to the nodes of a finite element, and  $P(S_1^e)$  and  $P(S_2^e)$  are the fluxes of stress  $P = ku'$  in these nodes. When applied to other types of finite elements, the indices in these equations need to be changed in accordance with the way in which the nodes are designated in that case. For example, if the finite element is introduced between nodes 6 and 7,  $u_1^e$  becomes  $u_6$ ,  $u_2^e$  becomes  $u_7$ ,  $P(S_1^e)$  is the value of  $ku'$  when approaching node 6 from the right and  $P(S_2^e)$  is its value when approaching node 7 from the left.

Let us now consider the four finite element mesh for a bar with two nodes, Fig. ??, i.e. for a total of 5 nodes. The global stiffness matrix will be a  $5 \times 5$  matrix, and the equations needed to form it are, in addition to the ones mentioned in (2.88):

$$\text{second finite element} \quad \begin{cases} k_{11}^2 u_2 + k_{12}^2 u_3 = f_1^2 + P(z_1^+) \\ k_{21}^2 u_2 + k_{22}^2 u_3 = f_2^2 - P(z_2^-) \end{cases} \quad (2.89)$$

$$\text{third finite element} \quad \begin{cases} k_{11}^3 u_3 + k_{12}^3 u_4 = f_1^3 + P(z_2^+) \\ k_{21}^3 u_3 + k_{22}^3 u_4 = f_2^3 - P(z_3^-) \end{cases} \quad (2.90)$$

$$\text{fourth finite element} \quad \begin{cases} k_{11}^4 u_4 + k_{12}^4 u_5 = f_1^4 + P(z_3^+) \\ k_{21}^4 u_4 + k_{22}^4 u_5 = f_2^4 - P(z_4^-) \end{cases}, \quad (2.91)$$

which results in the following matrix equation:

$$\begin{bmatrix} k_{11}^1 & k_{12}^1 & 0 & 0 & 0 \\ k_{21}^1 & k_{22}^1 + k_{11}^2 & k_{12}^2 & 0 & 0 \\ 0 & k_{21}^2 & k_{22}^2 + k_{11}^3 & k_{12}^3 & 0 \\ 0 & 0 & k_{21}^3 & k_{22}^3 + k_{11}^4 & k_{12}^4 \\ 0 & 0 & 0 & k_{21}^4 & k_{22}^4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} f_1^1 + P(0) \\ f_2^1 + f_1^2 + \|P(z_1)\| \\ f_2^2 + f_1^3 + \|P(z_2)\| \\ f_2^3 + f_1^4 + \|P(z_3)\| \\ f_2^4 - P(z_4 = l) \end{bmatrix} \quad (2.92)$$

What remains now is to incorporate displacement boundary conditions, since stress boundary conditions are already a part of the finite element equations. If, for example, we define that  $u(0) = u_0$  and  $u(l) = u_l$  then equation (2.92) has two less unknown displacements, and can be written as:

$$\begin{bmatrix} k_{21}^1 & k_{22}^1 + k_{11}^2 & k_{12}^2 & 0 & 0 \\ 0 & k_{21}^2 & k_{22}^2 + k_{11}^3 & k_{12}^3 & 0 \\ 0 & 0 & k_{21}^3 & k_{22}^3 + k_{11}^4 & k_{12}^4 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} f_2^1 + f_1^2 \\ f_2^2 + f_1^3 + \hat{f} \\ f_2^3 + f_1^4 \end{bmatrix} \quad (2.93)$$

with two auxiliary equations:

$$\begin{aligned} k_{11}^1 u_1 + k_{12}^1 u_2 = f_1^1 + P(0) &\Rightarrow P(0) = k_{11}^1 u_1 + k_{12}^1 u_2 - f_1^1 \\ k_{21}^4 u_4 + k_{22}^4 u_5 = f_2^4 - P(l) &\Rightarrow P(l) = -k_{21}^4 u_4 - k_{22}^4 u_5 + f_2^4, \end{aligned} \quad (2.94)$$

with unknown values of  $P(0)$  and  $P(l)$ .

### 2.4.1 Application of finite element method to parabolic and hyperbolic partial differential equations

Let us now consider time-dependent problems, i.e. parabolic and hyperbolic partial differential equation. For the sake of simplicity, we will once again observe only a single space variable. By introducing time-dependence for the state variable, we obtain the law of balance in the following form:

$$\frac{\partial P(z,t)}{\partial z} - f(z,t) + \frac{\partial G(z,t)}{\partial t} = 0 \quad 0 < z < l \quad t > 0, \quad (2.95)$$

where  $P$  is the flux,  $f$  is the density of distributed internal sources, and  $G$  is a quantity which is being "conserved" during the process. For example, if energy conservation (balance) is in question,  $\frac{\partial G}{\partial t}$ , it can represent the rate of entropy change per unit length and unit temperature, in which case it is related to the state variable (temperature) via the state equation:

$$\frac{\partial G(z,t)}{\partial t} = C(z,t) \frac{\partial u(z,t)}{\partial t}, \quad (2.96)$$

where  $C(z,t)$  is a material property – specific heat. If, for example, (2.95) represent the law of motion quantity balance in a deformable body, and  $u$  is the displacement field, then the motion quantity  $G$  in  $z$  at a time  $t$  is:

$$G(z,t) = \rho(z) \frac{\partial u(z,t)}{\partial t}; \quad \frac{\partial G(z,t)}{\partial t} = \rho(z) \frac{\partial^2 u(z,t)}{\partial t^2}, \quad (2.97)$$

where  $\rho(z)$  is the mass density in point  $z$ . The following state equation applies to flux  $P$ :

$$P(z,t) = -k(z) \frac{\partial u(z,t)}{\partial z} \quad (2.98)$$

where  $|k(z)| \geq k_0 = \text{const} > 0$  for  $\forall z \in [0, l]$ , wherein a convention is adopted according to which  $k$  is considered positive in the case of heat transfer, and negative for elasticity.

By using equations (2.96)-(2.98) in order to eliminate  $G$  from (2.95), we obtain two type of partial differential equations, one of which is parabolic:

$$C(z,t) \frac{\partial u(z,t)}{\partial t} - \frac{\partial}{\partial z} \left[ k(z) \frac{\partial u(z,t)}{\partial z} \right] - f(z,t) = 0 \quad 0 < z < l \quad t > 0, \quad (2.99)$$

which holds for state equation (2.96-98) and is related to heat conduction problems, and the second, hyperbolic equation:

$$\rho(z) \frac{\partial^2 u(z,t)}{\partial t^2} - \frac{\partial}{\partial z} \left[ k(z) \frac{\partial u(z,t)}{\partial z} \right] - f(z,t) = 0 \quad 0 < z < l \quad t > 0. \quad (2.100)$$

We will first analyse finite element method application to parabolic partial differential equation, (2.99). For the sake of simplicity, we will assume that  $C(z)$  is independent from  $t$ , and we then have that  $C(z) \geq c_0 > 0$  for  $\forall z$ , where  $c_0$  is a positive constant. In addition to boundary conditions, defined in the following simple form:

$$u(0,t) = u(l,t) = 0 \quad t \geq 0,$$

We also need to define the initial conditions:

$$u(z,0) = \tilde{u}(z) \quad 0 < z < l,$$

where  $\tilde{u}(z)$  is a smooth function of  $z$ . Finally, if time is limited by  $0 \leq t \leq T$ , we can define the boundary problem as:

$$\begin{aligned} C(z) \frac{\partial u(z,t)}{\partial t} - \text{pizz} \left[ k(z) \frac{\partial u(z,t)}{\partial z} \right] &= f(z,t) \quad 0 \leq z \leq l \quad 0 < t \leq T \\ u(0,t) = u(l,t) &= 0 \quad 0 \leq t \leq T \\ u(z,0) &= \tilde{u}(z) \quad 0 < z < l. \end{aligned} \quad (2.101)$$

Of many ways in which variation formulation can be applied to problem (2.101), we will focus on one of the simpler ones. For this purpose, we will choose a favourable smooth function  $v = v(z)$ , independent from  $t$ , and we will multiply equation (2.99) with it. Then, after partial integration (see (2.25)), we obtain:

$$\int_0^l \left[ C(z) \frac{\partial u(z,t)}{\partial t} v(z) + k(z) \frac{\partial U(z,t)}{\partial z} \frac{\partial v(z)}{\partial z} \right] dz - \int_0^l f(z,t) v(z) dz = 0. \quad (2.102)$$

Equation (2.102) holds for  $\forall v(z) \in H_0^1(0,l)$ , and its solution should be found for  $u = u(z,t) \in H_0^1(0,l)$  for  $0 \leq t \leq T$ .  $H_0^1(0,l)$  is the usual space of allowable functions with square-integral derivatives for  $0 < z < l$ , whose values are 0 for  $z = 0$  and  $z = l$ .

The method for solving of equation (2.102) is analogous to the previously described one, with one crucial difference: nodal values of an approximate solution  $u_h$  are unknown functions of time. Hence we can write:

$$u_h(z,t) = \sum_{j=1}^N u_j(t) \phi_j(z), \quad (2.103)$$

where  $\phi_j(z)$  are the base functions defining space  $H^h$ ,  $u_j(t)$  is the value of  $u_h$  in node  $z_j$  and at time  $t$

$$u_h(z_j,t) = \sum_{i=1}^N u_i(t) \phi_i(z_j) = u_j(t). \quad (2.104)$$

When applied to equation (2.102), Galerkin method provides a system of  $N$  ordinary differential equations with a total of  $N$  unknown functions  $u_j(t)$ :

$$\begin{aligned} \mathbf{C} \frac{d\mathbf{U}(t)}{dt} + \mathbf{K}\mathbf{U}(t) &= \mathbf{f}(t) \quad 0 < t \leq T \\ \mathbf{U}(0) &= \hat{\mathbf{U}}, \end{aligned} \quad (2.105)$$

where

$$\begin{aligned} C_{ij} &= \int_0^l C(z) \phi_i(z) \phi_j(z) dz \\ K_{ij} &= \int_0^l K(z) \frac{d\phi_i(z)}{dz} \frac{d\phi_j(z)}{dz} dz \\ f_i(t) &= \int_0^l f(z,t) \phi_i(z) dz. \end{aligned} \quad (2.106)$$

Vector  $\mathbf{U}$ , with an order of  $N \times 1$ , made of nodal values  $N_j(t)$  and  $\tilde{\mathbf{U}}$  are the interpolated values of  $\tilde{U}$  (i.e. the  $N \times 1$  vector of nodal values  $\tilde{U}(z)$ ).  $\mathbf{C}$  is the heat capacity matrix,  $\mathbf{K}$  is the typical conductivity matrix for a stationary problem and  $\mathbf{f}$  is the time-independent load vector. Matrices  $\mathbf{C}$  and  $\mathbf{K}$  are sparse, with a narrow strip, symmetrical and invertible, and are obtained in the previously described manner.

By applying finite element method, we obtained system of ordinary differential equations. Since  $u_h$  was no discretized with respect tot time, equation (2.103) is only discretized in space. In order to

obtain full discretization, it is necessary to assume how  $\mathbf{U}(t)$  behaves in time. Usual methods for this are implicit and explicit, as was the case with the finite difference method.

In the case of the explicit method, the time domain  $0 \leq t \leq T$  is divided into  $M$  equal intervals with length of  $\Delta t = T/M$ , and the following equation is used

$$\frac{d\mathbf{U}(n\Delta t)}{dt} \approx \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t}, \quad \mathbf{U}^n = \mathbf{U}(n\Delta t) \quad (2.107)$$

for the purpose of determining the first derivative of a state variable in the interval of  $t = n\Delta t$  to  $(n+1)\Delta t$ , and by replacing this into (2.105), we obtain:

$$\mathbf{U}^{(n+1)} = (\mathbf{I} - \Delta t \mathbf{C}^{-1} \mathbf{K}) \mathbf{U}^n + \Delta t \mathbf{C}^{-1} \mathbf{f}^n. \quad (2.108)$$

According to the above, based on known values of  $\mathbf{U}^0$ , we obtain  $\mathbf{U}^1$  followed by  $\mathbf{U}^2$ , and so on. As with the finite difference method, the explicit time integration is conditionally stable, which requires limited, smaller values of  $\Delta t$ , proportional to the mesh step  $h$ .

In the case of implicit methods, instead of (2.107), the following equations are used to determine the first derivative:

$$\frac{d\mathbf{U}[(n+1)\Delta t]}{dt} \approx \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t}, \quad \text{ili} \quad \frac{d\mathbf{U}[(n+1/2)\Delta t]}{dt} \approx \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} \quad (2.109)$$

which provides an unconditionally stable solution, when replaced into (2.105).

Hyperbolic partial differential equation (2.100) requires more complex initial conditions, including first derivatives, since this equation has a second derivative with respect to time:

$$u(z,0) = \tilde{u}(z) \quad \text{i} \quad \frac{\partial u(z,0)}{\partial t} = \hat{g}(z) \quad 0 < z < l. \quad (2.110)$$

We will once again adopt a simple-form boundary condition, as in the case of (2.101). By multiplying (2.99) with  $v(z)$  and integrating along  $z$ , we obtain:

$$\int_0^l \left[ \rho(z) \frac{\partial^2 u(z,t)}{\partial t^2} v(z) dz - \frac{\partial}{\partial z} \left[ k(z) \frac{\partial u(z,t)}{\partial z} \right] v(z) - f(z,t) v(z) \right] dz = 0 \quad (2.111)$$

from which follows that:

$$\begin{aligned} \int_0^l \rho(z) \frac{\partial^2 u(z,t)}{\partial t^2} v(z) dz + \int_0^l k(z) \frac{\partial u(z,t)}{\partial z} \frac{\partial v(z)}{\partial z} dz - \int_0^l f(z,t) v(z) dz = \\ -k(l) \frac{\partial u(l,t)}{\partial z} v(l) + k(0) \frac{\partial u(0,t)}{\partial z} v(0) \quad \text{za} \quad \forall v \in H_0^1(0,l). \end{aligned} \quad (2.112)$$

If we introduce  $u_h(z,t)$  into equations (2.103) and (2.104), in the same way, we obtain:

$$\mathbf{C} \frac{d^2 \mathbf{U}(t)}{dt^2} + \mathbf{K} \mathbf{U}(t) = \mathbf{f}(t) \quad (2.113)$$

$$\mathbf{U}(0) = \tilde{\mathbf{U}} \quad d\mathbf{U}(0)/dt = \tilde{\mathbf{g}}$$

where

$$\mathbf{C}_{ij} = \int_0^l \rho(z) \phi_i(z) \phi_j(z) dz \quad (2.114)$$

is the mass matrix, and stiffness matrix  $K_{ij}$  and force vector  $f_i$  have the same form as in (2.105). Equation (2.113) is also solved using explicit or implicit methods. In the case of explicit methods, the second derivative with respect to time is represented as:

$$\frac{d^2 \mathbf{U}(n\Delta t)}{dt^2} \approx \frac{\mathbf{U}^{n+1} - 2\mathbf{U}^n + \mathbf{U}^{n-1}}{\Delta t^2}, \quad (2.115)$$

which gives us:

$$\mathbf{C}(\mathbf{U}^{n+1} - 2\mathbf{U}^n + \mathbf{U}^{n-1}) = (-\mathbf{K}\mathbf{U}^n + \mathbf{f}^n)\Delta t^2. \quad (2.116)$$

It is necessary to know  $\mathbf{U}^0$  and  $\mathbf{U}^1 = \mathbf{U}^0 + \Delta t \frac{\partial \mathbf{U}^0}{\partial t}$  in order to calculate  $\mathbf{U}^2$  from equation (2.116), and then calculate  $\mathbf{U}^3$  base on  $\mathbf{U}^1$  and  $\mathbf{U}^2$ , ... Thus, the solution is conditionally stable.

### 3. Comparison of finite element method and finite difference method

As an example for comparing finite element method and finite difference method, we will consider an equation describing the free oscillations of a beam:

$$EI \frac{\partial^4 \Omega}{\partial x^4} + m \frac{\partial^2 \Omega}{\partial t^2} = 0, \quad (3.1)$$

which will be transformed by replacing  $\Omega(x, t) = W(x)e^{i\omega t}$ , for the purpose of determining its exact solution. We now have:

$$\frac{d^4 W}{dx^4} - \lambda W = 0 \quad (3.2)$$

where  $m$  is unit mass per unit length,  $\omega$  is the eigenfrequency,  $E$  is the elasticity module and  $I$  is the moment of inertia,

$$\lambda = \frac{m\omega^2}{EI} = \beta^4. \quad (3.3)$$

The general solution of (3.1) is:

$$W(x) = C_1 \sin(\beta x) + C_2 \cos(\beta x) + C_3 \sinh(\beta x) + C_4 \cosh(\beta x). \quad (3.4)$$

Constants  $C_1 - C_4$  are determined from the initial conditions:

$$\begin{aligned} W(0) &= W(L) = 0 \\ \frac{dW}{dx}(0) &= \frac{dW}{dx}(L) = 0 \end{aligned} \quad (3.5)$$

which results in four homogeneous equations with four unknowns. In order for a non-trivial solution to exist, the system matrix needs to be equal to zero, from which we obtain the frequency equation

as follows:

$$\cos(\beta L)\cosh(\beta L) = 1, \quad (3.6)$$

which has an infinite number of solutions. By replacing its solutions in terms of  $\beta$ , we obtain  $\omega_n^2 = \frac{\beta_n^4 EI}{m}$ , i.e.:

$$\omega_n = \beta_n^2 \sqrt{\frac{EI}{m}} = \frac{(\beta_n L)^2}{L^2} \sqrt{\frac{EI}{m}}, \quad (3.7)$$

where  $\beta_1 L = 4,73$  i  $\beta_2 L = 7,85$ , i.e.  $\omega_1 = \frac{21,4}{L^2} \sqrt{\frac{EI}{m}}$ ;  $\omega_2 = \frac{61,6}{L^2} \sqrt{\frac{EI}{m}}$ .

### 3.1 Approximate analytical solution

Let us assume the solution of (3.2) in the following form:

$$W(x) = \sum_{i=1}^n C_i f_i(x), \quad (3.8)$$

where  $c_i$  are constants, and  $f_i(x)$  are functions which satisfy the boundary conditions. Since this solution is not exact, they do not satisfy equation (3.2), hence replacing this solution into that equation will result in a remainder  $R$ . Values of constants  $c_i$  are obtained from the following condition:

$$\int_{x=0}^L f_i R dx = 0 \quad i = 1, 2, \dots, n \quad (3.9)$$

which provides a system of homogeneous linear algebraic equations along unknowns  $C_i$ , whose solutions can be used to determine the approximate solution. For example, if we assume that:

$$W(x) = C_1 f_1(x) + C_2 f_2(x) \quad (3.10)$$

where  $f_1(x) = \cos(\frac{2\pi x}{L}) - 1$  and  $f_2(x) = \cos(\frac{4\pi x}{L}) - 1$ , by replacing into equation (3.2), we obtain:

$$R = C_1 \left[ \left(\frac{2\pi}{L}\right)^4 - \beta^4 \right] \cos\left(\frac{2\pi x}{L}\right) + C_1 \beta^4 + C_2 \left[ \left(\frac{4\pi}{L}\right)^4 - \beta^4 \right] \cos\frac{4\pi x}{L} + C_2 \beta^4. \quad (3.11)$$

By applying the condition defined by (3.9), the following is obtained

$$\begin{aligned} & \int_{x=0}^L \left( \cos\frac{2\pi x}{L} - 1 \right) \left[ C_1 \left\{ \left(\frac{2\pi}{L}\right)^4 - \beta^4 \right\} \cos\frac{2\pi x}{L} + C_1 \beta^4 + \right. \\ & \left. + C_2 \left\{ \left(\frac{4\pi}{L}\right)^4 - \beta^4 \right\} \cos\frac{4\pi x}{L} + C_2 \beta^4 \right] dx = 0 \end{aligned} \quad (3.12)$$

and

$$\begin{aligned} & \int_{x=0}^L \left( \cos\frac{4\pi x}{L} - 1 \right) \left[ C_1 \left\{ \left(\frac{2\pi}{L}\right)^4 - \beta^4 \right\} \cos\frac{2\pi x}{L} + C_1 \beta^4 + \right. \\ & \left. + C_2 \left\{ \left(\frac{4\pi}{L}\right)^4 - \beta^4 \right\} \cos\frac{4\pi x}{L} + C_2 \beta^4 \right] dx = 0. \end{aligned} \quad (3.13)$$

i.e.

$$C_1 \left[ \frac{1}{2} \left\{ \left(\frac{2\pi}{L}\right)^4 - \beta^4 \right\} - \beta^4 \right] - C_2 \beta^4 = 0, \quad (3.14)$$

and

$$-C_1\beta^4 + C_2 \left[ \frac{1}{2} \left\{ \left( \frac{4\pi}{L} \right)^4 - \beta^4 \right\} - \beta^4 \right] = 0. \quad (3.15)$$

In order for the non-trivial solution of systems (3.14) and (3.15) to exist, their determinants must be equal to zero, from which it follows that:

$$\begin{aligned} \beta_1 L &= 4, 74 \\ \beta_2 L &= 11, 14 \end{aligned} \quad (3.16)$$

$$\begin{aligned} \omega_1 &= \frac{22,48}{L^2} \sqrt{\frac{EI}{m}} \\ \omega_2 &= \frac{124,1}{L^2} \sqrt{\frac{EI}{m}}. \end{aligned} \quad (3.17)$$

### 3.1.1 Finite difference method solution

Finite difference method requires approximate expressions for derivatives, which can be obtained via Taylor series:

$$f(x + \Delta x) = f(x) + \frac{df}{dx} \Delta x + \frac{d^2 f}{dx^2} \frac{(\Delta x)^2}{2} + \frac{d^3 f}{dx^3} \frac{(\Delta x)^3}{6} + \frac{d^4 f}{dx^4} \frac{(\Delta x)^4}{24} \quad (3.18)$$

$$f(x - \Delta x) = f(x) + \frac{df}{dx} \Delta x - \frac{d^2 f}{dx^2} \frac{(\Delta x)^2}{2} + \frac{d^3 f}{dx^3} \frac{(\Delta x)^3}{6} - \frac{d^4 f}{dx^4} \frac{(\Delta x)^4}{24}. \quad (3.19)$$

Taking into account the first two terms and subtracting of equations (3.18) and (3.19) results in:

$$\left. \frac{df}{dx} \right|_x \approx \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x}. \quad (3.20)$$

Taking into account the first three terms and subtracting of equations (3.18) and (3.19) results in:

$$\left. \frac{d^2 f}{dx^2} \right|_x = \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{(\Delta x)^2} \quad (3.21)$$

If  $f(x)$  in equation (3.21) is replaced with  $\left. \frac{d^2 f}{dx^2} \right|_x$ , the following is obtained:

$$\left. \frac{d^4 f}{dx^4} \right|_x = \frac{\left. \frac{d^2 f}{dx^2} \right|_{x+\Delta x} - 2 \left. \frac{d^2 f}{dx^2} \right|_x + \left. \frac{d^2 f}{dx^2} \right|_{x-\Delta x}}{(\Delta x)^2} \quad (3.22)$$

By replacing (3.21) into the right-hand side of equation (3.22), we obtain:

$$\begin{aligned} \left. \frac{d^4 f}{dx^4} \right|_x \approx \frac{1}{(\Delta x)^2} & \left[ \frac{f(x + 2\Delta x) - 2f(x + \Delta x) + f(x)}{(\Delta x)^2} - \right. \\ & - 2 \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{(\Delta x)^2} + \\ & \left. + \frac{f(x) - 2f(x - \Delta x) + f(x - 2\Delta x)}{(\Delta x)^2} \right] \end{aligned} \quad (3.23)$$

Now, we can apply the finite difference method in order to solve equation (204), by introducing nodes as seen in Figure 3.1, including hypothetical nodes (-1) and (4). For nodes (1) and (2), we obtain that:

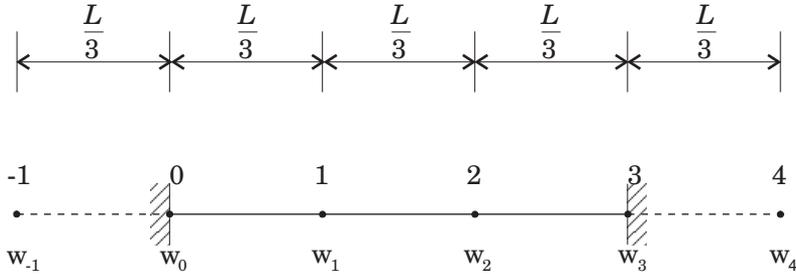


Figure 3.1: Finite difference mesh.

$$\begin{aligned} W_{-1} - 4W_0 + 6W_1 - 4W_2 + W_3 &= \beta_1^4 W_1, \\ W_0 - 4W_1 + 6W_2 - 4W_3 + W_4 &= \beta_1^4 W_2, \end{aligned} \quad (3.24)$$

where  $\beta_1 = \frac{l}{3}\beta$ .

Boundary conditions are  $W_0 = W_3 = 0$  and  $\frac{dW}{dx} = 0$  in nodes (0) and (3), i.e.  $W_{-1} = W_1$  and  $W_2 = W_4$ . By replacing boundary conditions in (3.24) we obtain a system of homogeneous linear algebraic equations with two unknowns,  $W_1$  and  $W_2$ :

$$\begin{aligned} 7W_1 - 4W_2 &= \beta_1^4 W_1, \\ -4W_2 + 7W_2 &= \beta_1^4 W_2, \end{aligned} \quad (3.25)$$

and by solving of the above, we obtain the following values:

$$\begin{aligned} \omega_1 &= \frac{15,59}{L^2} \sqrt{\frac{EI}{m}} & \begin{Bmatrix} W_1 \\ W_2 \end{Bmatrix} &= \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \\ \omega_2 &= \frac{29,85}{L^2} \sqrt{\frac{EI}{m}} & \begin{Bmatrix} W_1 \\ W_2 \end{Bmatrix} &= \begin{Bmatrix} 1 \\ -1 \end{Bmatrix} \end{aligned} \quad (3.26)$$

Solution accuracy can be improved by introducing additional nodes.

### 3.1.2 Finite element method solution

The beam is divided into an adequate number of subdomains (finite elements), in this case two, for which the following (cubic) interpolation is introduced:

$$\begin{aligned} \Omega(x) &= W_1^{(e)}(2\xi^3 - 3\xi^2 + 1) + W_3^{(e)}(3\xi^2 - 2\xi^3) + \\ &+ lW_2^{(e)}(\xi^3 - 2\xi^2 + \xi) + lW_4^{(e)}(\xi^3 - \xi^2), \end{aligned} \quad (3.27)$$

where  $\xi = x/l$ ,  $x$  is the longitudinal coordinate,  $l$  is the element length, Figure 3.2.

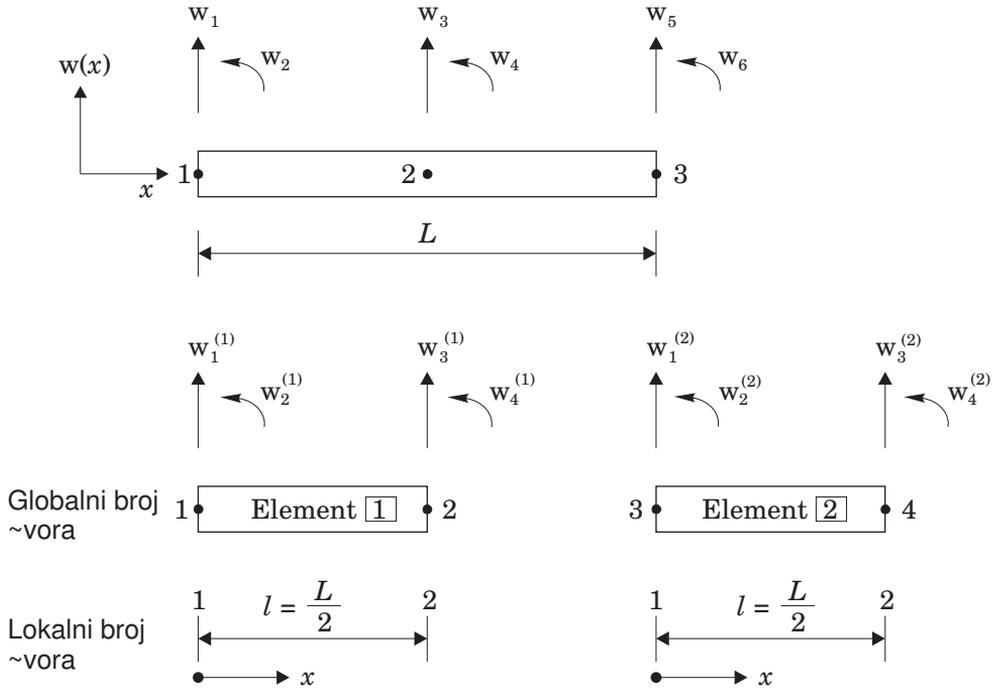


Figure 3.2: Finite elements.

For potential (static) and kinetic energy of the element, the following is obtained:

$$\Pi^{(e)} = \frac{1}{2} \int_0^l EI \left( \frac{\partial^2 \omega}{\partial x^2} \right)^2 dx = \frac{1}{2} \vec{W}^{(e)T} [K^{(e)}] \vec{W}^{(e)} \quad (3.28)$$

$$T^{(e)} = \frac{1}{2} \int_0^l \rho A \left( \frac{\partial \omega}{\partial x} \right)^2 dx = \frac{1}{2} \dot{\vec{W}}^{(e)T} [M^{(e)}] \dot{\vec{W}}^{(e)} \quad (3.29)$$

where  $\rho$  is the density,  $A$  is the cross-section area, and  $\dot{\omega}$  is the derivative with respect to time. By replacing (3.27) into (3.28) and (3.29), the stiffness matrix  $[K^{(e)}]$  is obtained, along with the mass matrix  $[M^{(e)}]$ :

$$[K^{(e)}] = \frac{2EI}{l^3} \begin{bmatrix} 6 & 3l & -6 & 3l \\ 3l & 2l^2 & -3l & l^2 \\ -6 & -3l & 6 & -3l \\ 3l & l^2 & -3l & 2l^2 \end{bmatrix} \quad (3.30)$$

$$[M^{(e)}] = \frac{\rho AI}{420} \begin{bmatrix} 156 & 22l & 54 & -13l \\ 22l & 4l^2 & 13l & -3l^2 \\ 54 & 13l & 156 & -22l \\ -13l & -3l^2 & -22l & 4l^2 \end{bmatrix} \quad (3.31)$$

If finite element equations are arranged according to Figure 3.2, global stiffness matrix is

obtained in the following form:

$$[K] = \frac{2EI}{l^3} \begin{bmatrix} W_1 & W_2 & W_3 & W_4 & W_5 & W_6 \\ 6 & 3l & -6 & 3l & 0 & 0 \\ 3l & 2l^2 & -3l & l^2 & 0 & 0 \\ -6 & -3l & 6+6 & -3l+3l & -6 & 3l \\ 3l & l^2 & -3l+3l & 2l^2+2l^2 & -3l & l^2 \\ 0 & 0 & -6 & -3l & 6 & -3l \\ 0 & 0 & 3l & l^2 & -3l & 2l^2 \end{bmatrix}$$

where

$$\begin{aligned} W_1 &= W_1^{(1)} \\ W_2 &= W_2^{(1)} \\ W_3 &= W_3^{(1)} = W_1^{(2)} \\ W_4 &= W_4^{(1)} = W_2^{(2)} \\ W_5 &= W_3^{(2)} \\ W_6 &= W_4^{(2)}, \end{aligned}$$

wherein we can remove rows and columns related to  $W_1, W_2, W_5$  i  $W_6$ , since they equal zero due to boundary conditions. Thus we obtain

$$[K] = \frac{2EI}{l^3} \begin{bmatrix} 12 & 0 \\ 0 & 4l^2 \end{bmatrix} = \frac{16EI}{l^3} \begin{bmatrix} 12 & 0 \\ 0 & l^2 \end{bmatrix} \quad (3.32)$$

Global mass matrix is obtained in the same way:

$$[M] = \frac{\rho Al}{420} \begin{bmatrix} 156 & 0 \\ 0 & l^2 \end{bmatrix} \quad (3.33)$$

If we solve the eigenvalue problem:

$$[K]\mathbf{W} = \lambda[M]\mathbf{W} \quad (3.34)$$

where  $\mathbf{W} = \begin{Bmatrix} W_3 \\ W_4 \end{Bmatrix}$  is the eigenvalue vector, and  $\lambda$  is the eigenvalue. We obtain two natural frequencies:

$$\omega_1 = \frac{22,7}{L^2} \sqrt{\frac{EI}{m}}, \quad \omega_2 = \frac{82,0}{L^2} \sqrt{\frac{EI}{m}}. \quad (3.35)$$

By comparing coefficients in the above expressions obtained by analytical and numerical methods (finite difference and finite element methods, Table 3.1) with exact results, it can be concluded that the finite element method is more efficient in the case of simple modeling.

Table 3.1: Result comparison.

Exact	Analytical	FDM	FEM
21,46	22,48	15,59	22,7
61,6	124,1	29,85	82,0

## 3.2 Finite element method – two-dimensional problem

### 3.2.1 Introduction

As was shown in the previous chapter, basic steps in finite method application are:

1. Variation formulation of the problem, with identifying of the adequate allowable function space  $H$ .
2. Defining of a finite element mesh and polynomials in regions which establish a finite dimensional space within  $H$ .
3. Defining an approximation of the variation problem of boundary values in the subspace of finite elements  $H^h$  within  $H$ , which involves calculating of element matrices and defining of a system of linear algebraic equations in terms of unknown node values of the approximate solution.
4. Solving of the linear algebraic equation system, wherein zero terms and symmetry are of great use.
5. Evaluating the properties of the solution, and if possible, determining its error.

These steps essentially apply to problems in two or three dimensions. Considering the significant application to such problems, this chapter will present all of the important finite element equations for two-dimensional boundary problems, i.e. differential equations which need to be satisfied on a two-dimensional domain  $\Omega$ , whose boundary is generally curvilinear. Instead of a line element, finite elements now have simple two-dimensional forms – triangular or rectangular, and the finite element mesh generally approximates the problem domain. Natural ability of these finite elements to represent a domain of any shape without additional efforts is a key advantage of practical application of finite element method.

In this chapter, we will analyse problems in which the unknown scalar function  $u$  is a function of position, e.g. heat conduction problems ( $u$  in this case is the temperature), flow through a porous medium ( $u$  is the pressure gradient), or transverse deflection of an elastic membrane ( $u$  is the deflection).

### 3.2.2 Physical base of the problem

Let the problem domain  $\bar{\Omega}$  consist of its interior  $\Omega$  and boundary  $\partial\Omega$ . We assume that this domain is finite and has a sufficiently smooth boundary (unit boundary normal  $\mathbf{n}$  is a continuous function of position along the boundary, with the exception of corner points). Generally, the boundary can be defined using parametric equations  $x = x(s)$  and  $y = y(s)$ , where  $s$  is the arc length at  $\partial\Omega$ , measured from a referent point. Values of an arbitrary function  $g$  on the boundary are denoted as  $g(s) \equiv g(x(s), y(s))$ ,  $s \in \partial\Omega$ .

For state variable  $u(x, y)$ , the basic requirement is that it is a smooth function in  $\Omega$ , in accordance with a specific problem and functions  $x(s)$  and  $y(s)$  which define  $\partial\Omega$ , i.e. it should be as smooth as necessary.

The physical base of the problem is based on the rate of change of the scalar field  $u$  in terms of its position within the  $\Omega$ , which is defined by the vector function  $\nabla u$ , which is referred to as a gradient:

$$\nabla u(x, y) = \frac{\partial u(x, y)}{\partial x} \mathbf{i} + \frac{\partial u(x, y)}{\partial y} \mathbf{j}, \quad (3.36)$$

where  $\mathbf{i}$ ,  $\mathbf{j}$  are the unit vectors for axes  $x$  and  $y$ , respectively. The gradient defines the total scalar field rate of change at point  $(x, y)$  in any given direction. If  $\mathbf{t}$  is the unit vector at angle  $\theta$  relative to the  $x$  axis, and  $\mathbf{t} = \cos \theta \mathbf{i} + \sin \theta \mathbf{j}$ , then the rate of change for a scalar field at  $(x, y)$  in direction  $\mathbf{t}$  can be defined as:

$$\frac{du}{dt} = \nabla u \cdot \mathbf{t} = \frac{\partial u}{\partial x} \cos \theta + \frac{\partial u}{\partial y} \sin \theta. \quad (3.37)$$

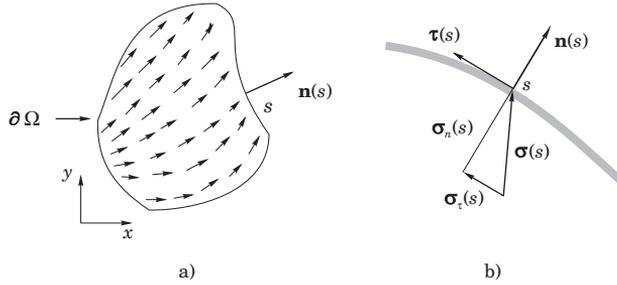


Figure 3.3: Flux in two-dimensional domain.

The second physical quantity we are interested in is the flux  $\sigma$ , which is a vector field. Flux  $\sigma$  is represented by arrows in  $\bar{\Omega}$ , i.e. by vector  $\sigma(s)$  at point  $s$  on the boundary  $\partial\Omega$ , Figure 3.3. The flux passing through this boundary at point  $s$  is defined by its component:

$$\sigma_n(s) = \sigma(s) \cdot \mathbf{n}(s), \quad (3.38)$$

where  $\mathbf{n}(s)$  is the normal at the boundary  $\partial\Omega$  at point  $s$ , Fig. 3.3b. Tangential component  $\tau(s)$  is given as  $\sigma(s) \cdot \boldsymbol{\tau}(s)$ , Fig. 3.3b.

Let us consider an arbitrary subregion  $\omega$ , containing a point  $P_0(x_0, y_0)$ . Shown in Fig. 3.4 is the distribution of  $\sigma_n(s)$  along the boundary  $\partial\omega$ , and the total flux passing through this boundary is given by:

$$\Sigma_\omega \equiv \int_{\partial\omega} \sigma_n(s) ds \quad (3.39)$$

If we divide  $\Sigma_\omega$  by the subregion  $A_\omega$  surface area, we obtain the mean value of flux  $\sigma$  which enters  $\omega$  per unit surface. The limit value of this quotient, when  $\omega$  is decreasing, while containing point  $P_0$  is referred to as the flux divergence in point  $P_0$ , and is denoted as  $\text{div}\sigma(x_0, y_0)$ . If we adopt that  $\omega$  is a square subregion with  $P_0$ , Fig. 3.4b, we obtain that  $\Sigma_\omega = \Delta\sigma_x\Delta y + \Delta\sigma_y\Delta x$ , hence the mean value theorem gives us:

$$\text{div}\sigma = \frac{\partial\sigma_x}{\partial x} + \frac{\partial\sigma_y}{\partial y} \quad (= \nabla \cdot \sigma). \quad (3.40)$$

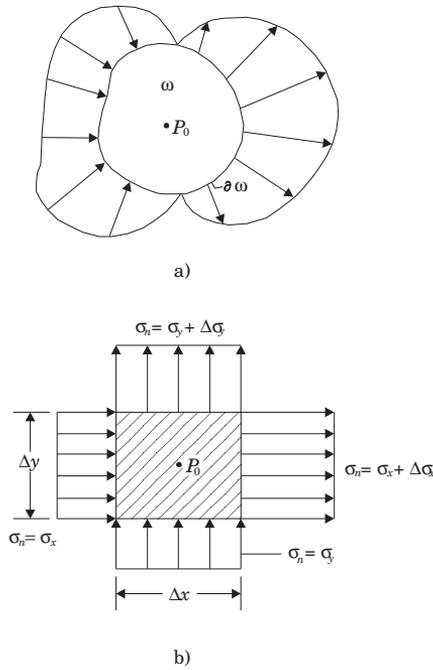


Figure 3.4: Flux distribution.

With his, we defined the net flux density in a point per unit area, hence the total flux in  $\Omega$  is given as:

$$\Sigma = \int_{\Omega} \nabla \cdot \sigma \, dx dy, \tag{3.41}$$

under the condition that  $\Omega$  and  $\sigma$  are sufficiently smooth. Based on (3.38), (3.39) and (3.41) we have that:

$$\int_{\Omega} \nabla \cdot \sigma \, dx dy = \int_{\partial\Omega} \sigma \cdot \mathbf{n} \, ds, \tag{3.42}$$

which represents the Gauss divergence theorem, which applies to any and all tensor fields.

### 3.2.3 Two-dimensional elliptical boundary problem

In order to formulate a two-dimensional elliptical boundary problem, we will use linear constitutive equations and conservation laws. In other words, the flux in every point is proportional to the state variable gradient, i.e. of the unknown  $u$ :

$$\sigma(x, y) = -k(x, y) \nabla u(x, y), \tag{3.43}$$

where  $k(x, y)$  is the material modulus (coefficient or a property) for which we assume that  $|k(x, y)| > k_0 (= \text{const.}) > 0$ .

Conservation (balance) aw suggests that for each part of the domain, the net flux through its boundary is equal to the total flux produced by the internal sources.

Let us now apply the conservation law to a material part  $\omega$ , located around point  $P_0$ , wherein all properties are smooth. If we donate a source per unit area as  $f$ , we obtain:

$$\int_{\partial\omega} \sigma \cdot \mathbf{n} \, ds = \int_{\omega} f \, dx dy. \quad (3.44)$$

By applying the divergence theorem, we obtain:

$$\int_{\omega} (\nabla \cdot \sigma - f) \, dx dy = 0 \quad (3.45)$$

for all subregions  $\omega$  in  $\Omega$ . Since  $\omega$  is an arbitrary region in which  $\nabla \cdot \sigma$  and  $f$  are smooth, subintegral function in (3.45) is equal to zero for all points within  $\omega$ .

$$\nabla \cdot \sigma(x, y) - f(x, y) = 0, \quad (3.46)$$

which represents a local conservation law.

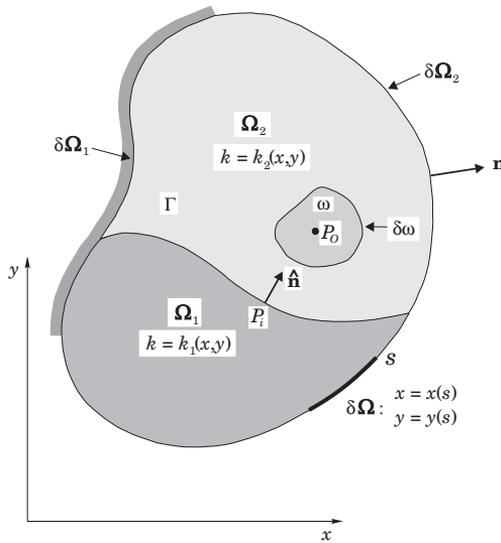


Figure 3.5: Two-dimensional domain with an interface.

Conservation law changes its form on inter-surfaces and boundaries, as can be shown by the following analysis. If we assume that body  $\bar{\Omega}$  is made of two different materials, one in subregion  $\Omega_1$  and the other in  $\Omega_2$ , Fig. 3.5, we can conclude that the material modulus  $k$  is defined via smooth function (i.e. by constants)  $k_1$  and  $k_2$ . The curve which defines the boundary between  $\Omega_1$  and  $\Omega_2$ , will be denoted by  $\Gamma$ , whereas  $\partial\Omega_1$  and  $\partial\Omega_2$  denote parts of  $\partial\Omega$ , i.e. boundaries for which boundary conditions are defined. These boundary conditions include  $u = \hat{u}$  at  $\partial\Omega_1$  (essential boundary condition), and the natural boundary conditions at  $\partial\Omega_2$ , which will be defined at a later stage.

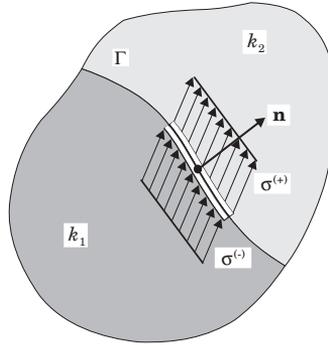


Figure 3.6: Flux change at the interface.

Let us consider point  $P_i$  on  $\Gamma$ , Fig. 3.5, i.e. a material strip which contains this point, Fig 3.6. Let us assume that this strip is narrow enough for the flux through its ends, as well as the source (proportional to the surface) can be neglected relative to the net flux through its edges. When the thickness of this layer tends to zero, the conservation law is as follows:

$$\Sigma = \int_{s_1}^{s_2} (-\sigma^{(-)} \cdot \mathbf{n} + \sigma^{(+)} \cdot \mathbf{n}) ds = 0, \quad (3.47)$$

where  $s_1$  and  $s_2$  are the ends of the strip. Since the integration area is arbitrary, local conservation law at points lying on boundary  $\Gamma$  is reduced to a jump  $\sigma \cdot \mathbf{n} = \sigma_n$  with respect to  $\Gamma$ :

$$\llbracket \sigma_n(s) \rrbracket = \sigma_n^{(+)}(s) - \sigma_n^{(-)}(s) = 0, \quad s \in \Gamma. \quad (3.48)$$

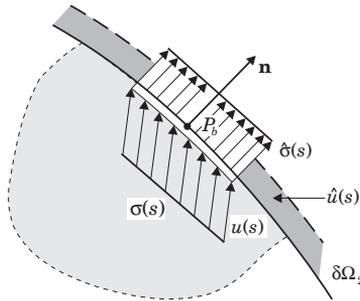


Figure 3.7: Boundary conditions.

Now let us consider boundary conditions at  $\partial\Omega_2$ , i.e. the area which contains a typical boundary point  $P_b$ , Fig. 3.7. We will assume that the normal flux  $\hat{\sigma}(s)$  through the surrounding material in the immediate vicinity of the boundary is proportional to the difference in value of  $u(s)$  at the boundary and the defined value of  $\hat{u}(s)$  in the external environment:

$$\hat{\sigma}(s) \equiv p(s)[u(s) - \hat{u}(s)].$$

Flux balance in the strip which contains  $P_b$  gives the following:

$$\sigma_n(s) \equiv \sigma(s) \cdot \mathbf{n}(s) = \hat{\sigma}(s), \quad s \in \partial\Omega_2. \quad (3.49)$$

Hence, the following holds:

$$\sigma_n(s) \equiv p(s)[u(s) - \hat{u}(s)]. \quad (3.50)$$

If we now use the constitutive equation (3.43) to eliminate  $\sigma$  and  $\sigma_n$  from (3.46) - (3.50), it is possible to mathematically formulate the boundary problem:

1. Boundaries  $\partial\Omega_1$ ,  $\partial\Omega_2$  and the inter-boundary  $\Gamma$  are defined by parametric equations:  $x = x(s)$ ,  $y = y(s)$ ,  $s \in \partial\Omega$  or  $s \in \Gamma$ ;
2. Source distribution is defined as  $f = f(x, y)$  in  $\Omega_i$ ,  $i = 1, 2$ ;
3. Material coefficients are defined as  $k_i = k_i(x, y)$  for  $(x, y) \in \Omega_i$ ,  $i = 1, 2$ ;
4. State variable  $\hat{u}(s)$  for  $s \in \partial\Omega_i$ , is given;
5. The value of boundary coefficients is  $p(s)$  and  $\hat{u}(s)$  on  $\partial\Omega_2$  or  $\hat{\sigma}(s)$  is given for  $s \in \partial\Omega_2$ .

For a problem formulated in this way, it is necessary to find a function  $u = u(x, y)$  which fulfills the following requirements:

1. The partial differential equation in internal points of smooth subdomains  $\Omega_1$  and  $\Omega_2$

$$-\nabla \cdot k(x, y) \nabla u(x, y) - f(x, y) = 0 \quad \text{za } (x, y) \in \Omega_i; \quad (3.51)$$

2. The jump condition at the inter-boundary  $\Gamma$

$$\|k \nabla u \cdot \mathbf{n}\| = 0 \quad s \in \Gamma; \quad (3.52)$$

3. Essential boundary conditions at  $\partial\Omega_1$

$$u(s) = \hat{u}(s) \quad s \in \partial\Omega_1; \quad (3.53)$$

4. Natural boundary conditions at  $\partial\Omega_2$

$$-k(s) \frac{\partial u(s)}{\partial n} = \hat{\sigma}(s) \quad s \in \partial\Omega_2. \quad (3.54)$$

### 3.3 Variation formulation of the boundary problem

As before, we will multiply equation (3.51) with a sufficiently smooth weight function  $v$  and determine the integral with respect to all domains with smooth output:

$$\int_{\Omega_1} [-\nabla \cdot (k \nabla u) - f] v \, dx dy + \int_{\Omega_2} [-\nabla \cdot (k \nabla u) - f] v \, dx dy = 0. \quad (3.55)$$

Two-dimensional partial integration is necessary in order to reduce the first terms in both integrals to their first derivatives. By applying the derivative product we obtain:

$$\begin{aligned} \nabla \cdot (v k \nabla u) &= k \nabla u \cdot \nabla v + v \nabla \cdot (k \nabla u) \\ v \nabla \cdot (k \nabla u) &= \nabla \cdot v k \nabla u - k \nabla u \cdot \nabla v, \end{aligned} \quad (3.56)$$

By replacing (3.56) into (3.55), we obtain:

$$\begin{aligned} \int_{\Omega_1} (k \nabla u \cdot \nabla v + b u v - f v) \, dx dy + \int_{\Omega_2} (k \nabla u \cdot \nabla v + b u v - f v) \, dx dy - \\ - \int_{\Omega_1} \nabla \cdot (v k \nabla u) \, dx dy - \int_{\Omega_2} \nabla \cdot (v k \nabla u) \, dx dy = 0. \end{aligned} \quad (3.57)$$

Two last integrals in (3.57) can be transformed into linear integrals by using the divergence theory, thus resulting in:

$$\begin{aligned}
 & - \int_{\Omega_1} \nabla \cdot (vk \nabla u) \, dx dy - \int_{\Omega_2} \nabla \cdot (vk \nabla u) \, dx dy = \\
 & = \int_{\partial(\Omega_1)} k \frac{\partial u}{\partial n} v \, ds - \int_{\partial(\Omega_2)} k \frac{\partial u}{\partial n} v \, ds,
 \end{aligned} \tag{3.58}$$

where  $\partial(\Omega_1)$  and  $\partial(\Omega_2)$  are the subdomain limits for domains  $\Omega_1$  and  $\Omega_2$ , the integration direction is a positive mathematical direction, and  $\partial u / \partial n = -\nabla u \cdot \mathbf{n}$ .

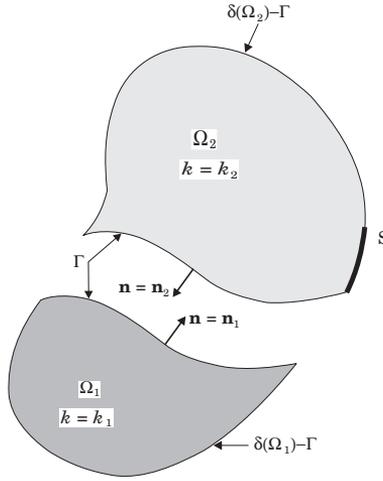


Figure 3.8: Domain boundaries.

Figure 3.8 shows that the limit of each domain is divided into two parts -  $\partial(\Omega_i)$ , which do not coincide with  $\Gamma$  are denoted  $\partial(\Omega_i) - \Gamma$ ,  $i = 1, 2$ . In accordance with this, line integrals in (3.58) are separated in the following way:

$$\begin{aligned}
 & - \int_{\partial(\Omega_1) - \Gamma} k \frac{\partial u}{\partial n} v \, ds - \int_{\partial(\Omega_2) - \Gamma} k \frac{\partial u}{\partial n} v \, ds + \\
 & + \int_{\Gamma} \left( -k \frac{\partial u}{\partial n} \right)_1 v \, ds + \int_{\Gamma} \left( -k \frac{\partial u}{\partial n} \right)_2 v \, ds,
 \end{aligned} \tag{3.59}$$

where  $\left( -k \frac{\partial u}{\partial n} \right)_i$  indicates that  $-k \frac{\partial u}{\partial n}$  need to be determined within the region  $\Omega_i$ . Taking into account the sign of the normal in the last two integrals in (3.59), we obtain:

$$\int_{\Gamma} \left[ -k^{(+)} \frac{\partial u^{(+)}}{\partial n} + k^{(-)} \frac{\partial u^{(-)}}{\partial n} \right] v \, ds. \tag{3.60}$$

Subintegral function in (3.60) is equal to  $v \|\sigma_n(s)\|$ , which is according to (3.46) equal to zero, hence integral in (3.60) is equal to zero.

First two integrals in (3.59) can be transformed into a single integral along the whole boundary  $\partial\Omega$ . In addition, subintegral functions in the first two integral in (3.57) contain highest first derivatives with respect to  $u$  and  $v$ , and thus can be transformed into a single integral along the whole domain  $\Omega$ , assuming that functions  $u$  and  $v$  are smooth enough:

$$\int_{\Omega} (k\nabla u \cdot \nabla v - fv) \, dx dy - \int_{\partial\Omega} k \frac{\partial u}{\partial n} v \, ds = 0. \quad (3.61)$$

Replacing of natural boundary conditions (3.54) into the linear integral gives:

$$\int_{\Omega} (k\nabla u \cdot \nabla v - fv) \, dx dy + \int_{\partial\Omega} \hat{\sigma} v \, ds = 0, \quad (3.62)$$

Which applies to all allowable weight functions  $v$ .

If this solution data is sufficiently smooth, the solution of equation (3.62) is also the solution of (3.54). Vice versa, every solution of (3.54) is automatically the solution of equation (3.62).

An important question still remains about determining of the adequate class of allowable functions for problem (3.62). Let us observe that the integrals in (3.62) are defined if functions  $u$  and  $v$  and their partial derivatives are smooth enough for square integration with respect to  $\Omega$ :

$$\int_{\Omega} \left[ \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + v^2 \right] \, dx dy < \infty. \quad (3.63)$$

Such class of functions will be denoted by  $H^1(\Omega)$ , where 1 refers to the „square integrability“ of the first derivative, and  $\Omega$  is the domain of the defined functions.

As in the case of one-dimensional problems, natural boundary conditions are the integral part of equation (3.62), and they appear in term  $\int_{\partial\Omega_2} \hat{\sigma} v \, ds$ . Essential boundary conditions are taken into account via allowable functions definition. For weight functions, we will select functions  $v$  from  $H^1(\Omega)$  which are equal to zero at  $\partial\Omega_1$ , and their solution must be a function within  $H^1(\Omega)$ , for which  $u = \hat{u}$  at  $\partial\Omega_1$ .

Variation formulation of the boundary condition is as follows: find a function  $u \in H^1(\Omega)$  such that  $u = \hat{u}$  at  $\partial\Omega_1$  and (3.63) holds for  $\forall v \in H^1(\Omega)$ , such that  $v = 0$  at  $\partial\Omega_1$ . Thus, in addition to the aforementioned **variation formulation** properties (see text after (3.62)), the following holds:

- Constraint imposed by the variation formulation are weaker than the constraints that apply to equation (3.54).
- The jump condition (3.52) does not require additional consideration in the case of variation formulation

### 3.4 Finite element interpolation

This chapter represents a direct generalisation of the corresponding one-dimensional problem analysis chapter, but with significant differences in certain important details. Above all, in the case of the one-dimensional problems, the finite element mesh is defined simply by dividing a linear domain into linear subdomains, by introducing nodes in all discontinuities. For two-dimensional problems, discretisation is not that simple. Essentially, we still tend to represent the approximate solution  $u_h$  and weight functions  $v_h$  via polynomials defined on subdomains with simple geometry, for a given area  $\Omega_h$ , which is located in the plane  $x, y$ . Additionally, discretization should be general enough to model irregular domains, while containing elements which are sufficiently simple to calculate. For this purpose, the most favourable shapes are the triangle and the rectangle, Fig. 3.9

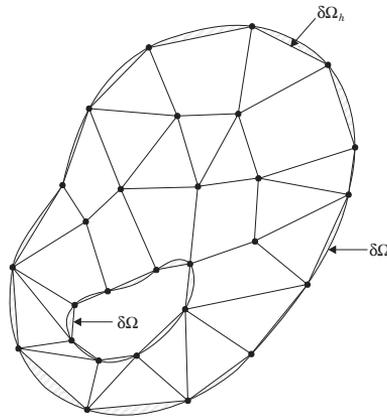


Figure 3.9: Discretization of two-dimensional problem.

It was shown that for one-dimensional problems, it is possible to represent the state variable of a unit length element in linear form:

$$u = u_0 + u_1 \xi,$$

Or, alternatively, in quadratic form:

$$u = u_0 + u_1 \xi + u_2 \xi^2.$$

In analogous manner, we can introduce the **canon triangle** for a two-dimensional domain.

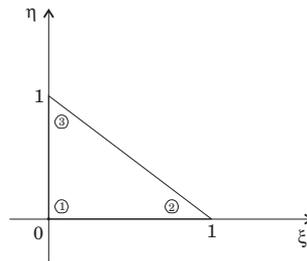


Figure 3.10: Triangular finite element.

Similar to the one-dimensional problem, wherein the unit canon length is projected into a real arbitrary length, the unit right triangle for the two-dimensional case is projected to a real triangular element of arbitrary shape and size.

In addition to being simple to calculate, the triangular finite element is introduced for the purpose of natural agreement between the number of nodes and the order of the polynomial which represents the approximate solution. Namely, a two-dimensional space linear equation has the following form:

$$v_h = a_1 + a_2 \xi + a_3 \eta \quad (3.64)$$

with three constants ( $a_i, i = 1, 2, 3$ ). In order to determine these constants, three independent values  $v_h$  are necessary, which implies that the finite element needs three nodes. Furthermore, for two

adjacent elements, the continuity of the function on their common side is achieved with the condition that the linear functions of these elements have the same values in common nodes.

Similar to this, the linear function given below:

$$v_h = a_1 + a_2\xi + a_3\eta + a_4\xi\eta \quad (3.65)$$

has four constants, and thus its corresponding finite element is rectangular, and its quadratic function is:

$$v_h = a_1 + a_2\xi + a_3\eta + a_4\xi^2 + a_5\xi\eta + a_6\eta^2 \quad (3.66)$$

which has 6 constants, and requires a six node triangle (with three nodes at its points, and three more at each side midpoint).

In accordance with the description given in the previous chapter, we can now define interpolation  $g_h$  for function  $g$  in the following form:

$$g_h(\xi, \eta) = g_i\phi_i(\xi, \eta) \quad (\xi, \eta) \in \Omega_h \quad (3.67)$$

where  $\phi_i, i = 1, \dots, N$  are the base functions defined at  $\Omega_h$  as:

$$\phi_i(\xi_j, \eta_j) = \begin{cases} 1, & \text{za } i = j \\ 0, & \text{za } i \neq j \end{cases} \quad (3.68)$$

where  $(\xi_j, \eta_j)$  are finite element nodes, which means that the following holds:

$$g_h(\xi_j, \eta_j) = g_j \quad j = 1, 2, \dots, N. \quad (3.69)$$

According to the above, by including  $g_j = g(\xi_j, \eta_j)$ ,  $g_j$  becomes equal to the value of  $g$  in nodes, i.e. it becomes its interpolation. The following two conditions also need to be fulfilled:

1. Definition of local interpolation functions  $\psi_i^e$  for each element must be such that its fitting in the finite element mesh results in base functions which fulfill (3.68);
2. Base functions  $\phi_i$  should be square-integrable, along with their partial derivatives:

$$\int_{\Omega_h} \left[ \left( \frac{\partial \phi_i}{\partial x} \right)^2 + \left( \frac{\partial \phi_i}{\partial y} \right)^2 + \phi_i^2 \right] dx dy < \infty. \quad (3.70)$$

This requirement is fulfilled if functions  $\phi_i$  are continual along the element boundaries.

### 3.4.1 Interpolation within triangles

Since the linear function:

$$v_h(\xi, \eta) = a_1 + a_2\xi + a_3\eta \quad (3.71)$$

defines a planar surface, linear interpolation within a triangle approximates a smooth function (surface)  $v$  with plane (3.71), inside an arbitrary element  $\Omega_e$ .

Let us assume that  $\Omega_h$  is made of a total of  $E$  triangular elements, and that the following linear interpolation holds:

$$v_h^e(\xi, \eta) = a_1 + a_2\xi + a_3\eta. \quad (3.72)$$

The following applies to triangle nodes (fig. 3.10):

$$\begin{aligned} v_1 &= v_h^e(\xi_1, \eta_1) = a_1 \\ v_2 &= v_h^e(\xi_2, \eta_2) = a_1 + a_2 \\ v_3 &= v_h^e(\xi_3, \eta_3) = a_1 + a_3 \end{aligned} \quad (3.73)$$

where  $(\xi_i, \eta_i)$  are node coordinates. By solving this system of linear algebraic equation in terms of  $a_1, a_2$  and  $a_3$ , we obtain:

$$\begin{aligned} a_1 &= v_1, \\ a_2 &= v_2 - v_1, \\ a_3 &= v_3 - v_1, \end{aligned} \quad (3.74)$$

By replacing (3.74) into (3.72), the following is obtained:

$$v_h^e(\xi, \eta) = v_1 \psi_1^e(\xi, \eta) + v_2 \psi_2^e(\xi, \eta) + v_3 \psi_3^e(\xi, \eta), \quad (3.75)$$

where  $\psi_i^e(\xi, \eta)$  are element shape functions:

$$\begin{aligned} \psi_1^e(\xi, \eta) &= 1 - \xi - \eta, \\ \psi_2^e(\xi, \eta) &= \xi, \\ \psi_3^e(\xi, \eta) &= \eta. \end{aligned} \quad (3.76)$$

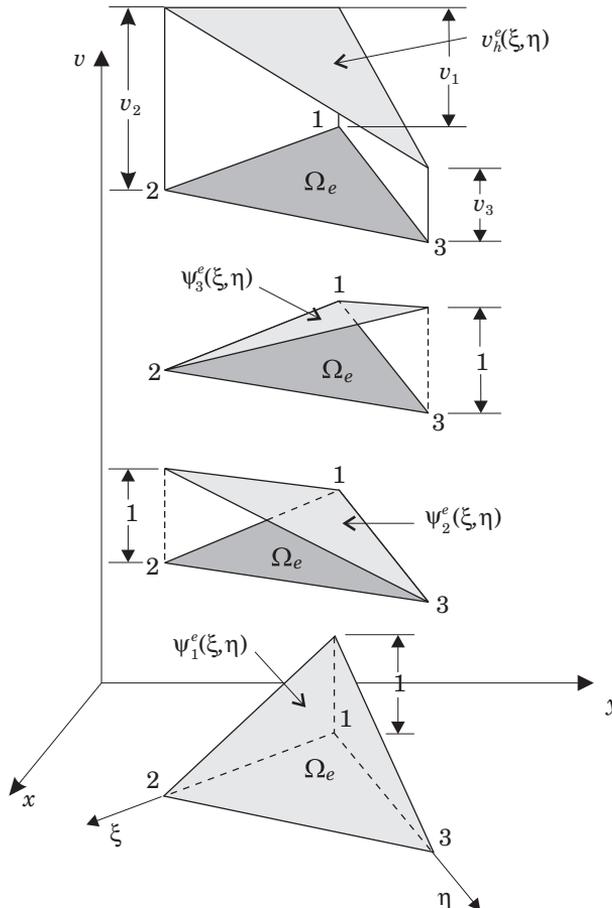


Figure 3.11: Interpolation functions of a triangular finite element.

These linear functions are represented in Figure 3.11, and the following applies to them:

$$\psi_i^e(\xi_j, \eta_j) = \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases} \quad (3.77)$$

In addition, we should mention that expression (3.75) can be written in condensed form as:

$$v_h^e(\xi, \eta) = v_i \psi_i^e(\xi, \eta), \quad (3.78)$$

by using the aforementioned adding conventions along the same (repeated) indices. Again, observe that the expression (3.78) is more general, since it can be used for elements with an arbitrary number of interpolation functions,  $\psi_i^e, i = 1, 2, \dots, n$ .

We will now determine “global”  $\phi_i(\xi, \eta)$ , which are obtained by “arranging” shape functions in a “pyramid”, fig. 3.11. Cartesian coordinates in this case is represented in the same way as the state variables  $v_i$  are represented using interpolation functions  $\psi_i$  (3.78):

$$\begin{aligned} x(\xi, \eta) &= x_i \psi_i(\xi, \eta), \\ y(\xi, \eta) &= y_i \psi_i(\xi, \eta), \end{aligned} \quad (3.79)$$

where  $x_i$  and  $y_i$  are the Cartesian coordinates for the  $i$ -th node.

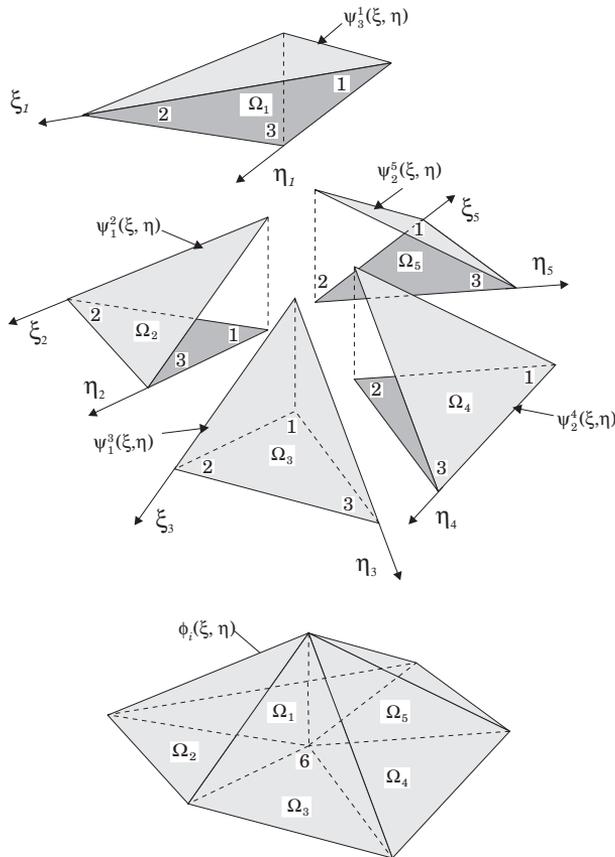


Figure 3.12: Global base functions.

It is clear that  $\phi_i$  fulfill the condition given by (3.77) and that they are continuous along the element boundaries, i.e. at  $\Omega_h$ ; their first partial derivatives are reduced to step functions, hence their squares are also integrable. Thus, such base functions represent an adequate choice for finite element approximation applications.

### 3.4.2 Interpolation error

Let us assume that a smooth function  $g$  is interpolated by function  $g_h$ , which contains complete polynomials of the  $k$ -th order. If partial derivatives of the  $k + 1$ -th order of function  $g$  are limited within  $\Omega_e$ , the following applies to the interpolation error:

$$\|g - g_h\|_{\infty, \Omega_e} = \max |g(x, y) - g_h(x, y)| < Ch_e^{k+1}, \quad (3.80)$$

where  $C$  is a positive constant, and  $h_e$  is the “diameter” of  $\Omega_e$ , i.e. the greatest distance between any two points in the element. Similar to this, we have that:

$$\left\| \frac{\partial g}{\partial x} - \frac{\partial g_h}{\partial x} \right\|_{\infty, \Omega_e} \leq C_1 h_e^k \quad \text{and} \quad \left\| \frac{\partial g}{\partial y} - \frac{\partial g_h}{\partial y} \right\|_{\infty, \Omega_e} \leq C_2 h_e^k. \quad (3.81)$$

$H^1$  is the two-dimensional norm defined by the following expression:

$$\|g\|_1^2 = \int_{\Omega} \left[ g^2 + \left( \frac{\partial g}{\partial x} \right)^2 + \left( \frac{\partial g}{\partial y} \right)^2 \right] dx dy. \quad (3.82)$$

If we assume that  $\Omega_h = \Omega$  and that  $h$  is the greatest “diameter” of all elements, we obtain the following (assuming that the mesh is sufficiently regular):

$$\|g - g_h\|_1 \leq C_3 h^k \quad (3.83)$$

for a sufficiently small  $h$ . In this case, it is also necessary for  $g_h$  to contain a complex polynomial of the  $k$ -th order.

## 3.5 Finite element approximations

Let  $H^1(\Omega)$  be a class of functions which satisfy expression  $H^1(\Omega)$ , and are defined for the whole domain  $\Omega$ . The problem here involves the determining of a function  $u$  from  $H^1(\Omega)$ , such that  $u = \hat{u}$  at boundary  $\partial\Omega_1$  and that the following condition is fulfilled:

$$\int_{\Omega} k \nabla u \cdot \nabla v dx dy + \int_{\partial\Omega_2} \hat{\sigma} v ds - \int_{\Omega} f v dx dy = 0 \quad (3.84)$$

for  $\forall v \in H^1(\Omega)$ , such that  $v = 0$  at  $\partial\Omega_1$  and  $\gamma = p\hat{u}$ .

The approximate solution of equation (3.84) requires replacing of domain  $\Omega$  with  $\Omega_h$ , which is actually the finite element mesh (with a total of  $N$  nodes and  $E$  elements), and defining of an  $N$ -dimensional subspace  $H^h$  in  $H^1(\Omega_h)$  by introducing global base functions  $\phi_i$ ,  $i = 1, \dots, N$ , along with finite elements. In order for these shape functions to remain continuous along each element, they cannot be used to model jumps, e.g. a jump in material properties. Thus the finite element mesh is generated in a way that ensures that element nodes and edges correspond to the lines where the material modulus jump occurs, Figure 3.13.

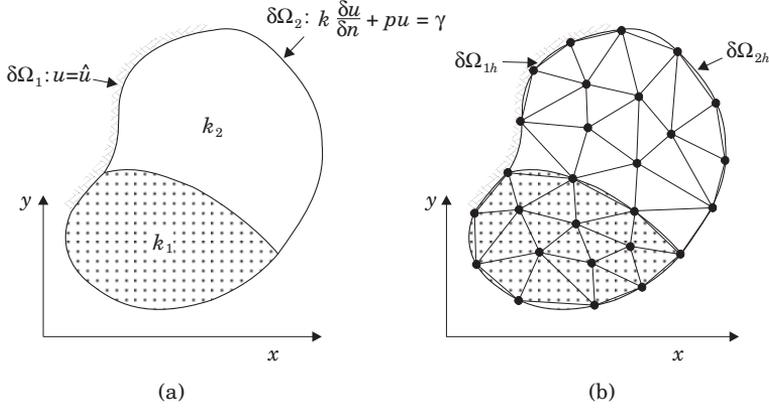


Figure 3.13: Two-dimensional domain (a) and its discretization (b).

The **weight function** can now be observed within each individual element:

$$v_h(\xi, \eta) = v_i \phi_i(\xi, \eta) \quad (3.85)$$

where  $v_i$  is the weight function value in the  $i$ -th node. In a general case, Dirichlet function  $\hat{u}$ , given on  $\partial\Omega_1$  is approximated by the interpolation  $\hat{u}_h(s) = \hat{u}_i \phi_i(s)$ , wherein the summing is performed along all nodes of the approximation  $\partial\Omega_{1h}$  (of  $\partial\Omega_1$ ). The approximate solution for function  $u_h$  in  $H^h$  is:

$$u_h(\xi, \eta) = u_i \phi_i(\xi, \eta) \quad (3.86)$$

hence,  $u_i = \hat{u}_i$  in nodes at  $\partial\Omega_{1h}$ , and the following holds:

$$\int_{\Omega_h} k \nabla u \cdot \nabla v \, dx dy + \int_{\partial\Omega_{2h}} \hat{\sigma} v_h \, ds - \int_{\Omega_h} f v_h \, dx dy = 0 \quad (3.87)$$

for  $\forall v_h \in H^h$ , such that  $v_h = 0$  at  $\partial\Omega_{1h}$ , i.e. for the approximation  $\partial\Omega_1$ .

By replacing (3.85) and (3.86) into (3.87), we obtain:

$$K_{ij} u_j = F_i \quad i, j = 1, 2, \dots, N \quad (3.88)$$

where  $K_{ij}$  are stiffness matrix terms:

$$K_{ij} = \int_{\Omega_h} k \nabla \phi_i \cdot \nabla \phi_j \, dx dy \quad (3.89)$$

while  $F_i$  are the components of force vector:

$$F_i = \int_{\Omega_h} f \phi_i \, dx dy - \int_{\partial\Omega_{2h}} \hat{\sigma} \phi_i \, ds. \quad (3.90)$$

Now we need to modify the equation (3.87) in order to include Dirichlet's conditions and solve the resulting system of unknown nodal values,  $u_j$ , thus determining the finite element approximation of the equation, (3.84) solution,  $u$ .

It is necessary to point out the similarities between this approach and the on-dimensional problem:

1. Stiffness matrix  $\mathbf{K}$  is sparse, since global functions  $\phi_i$  and  $\phi_j$  and their derivatives are not equal to zero only when there are elements that contain both nodes  $i$  and  $j$ .
2. In the case being analysed,  $\mathbf{K}$  is a symmetrical matrix (since the operator in (3.46) is self-adjointed) and has the form of a strip in the case that the nodes were numbered in order. All of this allows for efficient solving of linear algebraic equation system (3.88).
3. All integrals in (3.89) and (3.90) can be calculated as the sum of all mesh elements contributions, for which the exact solution of a boundary problem satisfies the condition given by the following equation:

$$\int_{\Omega_e} k \nabla u \cdot \nabla v \, dx dy - \int_{\Omega_e} f v \, dx dy + \int_{\partial \Omega_e} \hat{\sigma}_n v \, ds = 0 \quad (3.91)$$

for each allowed  $v$ , where  $\hat{\sigma}_n$  is the normal flux component at the element boundary. Let  $u_h^e$  and  $v_h^e$  represent the boundaries of approximations  $u_h$  and  $v_h$  on  $\Omega_e$ . Then, the local approximation of the variation boundary condition on  $\Omega_e$  has the following form:

$$\int_{\Omega_e} k \nabla u_h \cdot \nabla v_h \, dx dy - \int_{\Omega_e} f v_h \, dx dy + \int_{\partial \Omega_e} \hat{\sigma}_n v_h \, ds = 0, \quad (3.92)$$

where  $\hat{\sigma}_n$  represents the real (exact) flux through  $\partial \Omega_e$ , which, despite not being given as data in the initial problem, still occurs as a natural boundary condition at  $\Omega_e$ . Since  $v_h = 0$  at  $\partial \Omega_{1h}$ , there will be no contribution to the last integral in (3.91) by the elements whose sides coincide with  $\partial \Omega_{1h}$ . Since  $u_h^e$  and  $v_h^e$  have the following form:

$$u_h^e(\xi, \eta) = u_i^e \psi_i^e(\xi, \eta) \quad v_h^e(\xi, \eta) = v_i^e \psi_i^e(\xi, \eta), \quad i = 1, \dots, N_e \quad (3.93)$$

where  $\psi_i^e$  are local shape functions for  $\Omega_e$ , and  $N_e$  is the number of nodes in  $\Omega_e$ . Equation (3.91) now becomes a linear system:

$$k_{ij}^e u_j^e = f_i^e - \sigma_i^e, \quad i, j = 1, 2, \dots, N_e \quad (3.94)$$

where

$$k_{ij}^e = \int_{\Omega_e} k \nabla \psi_i^e \cdot \nabla \psi_j^e \, dx dy. \quad (3.95)$$

$$f_i^e = \int_{\Omega_e} f \psi_i^e \, dx dy. \quad (3.96)$$

Stiffness matrix and load vector components of element  $\Omega_e$  whereas the following expression:

$$\sigma_i^e = \int_{\Omega_e} \sigma_n \psi_i^e \, ds \quad (3.97)$$

Formally, global system of equations (3.88) is obtained by summing of (3.94) with respect to all elements in the mesh, wherein matrix and vector terms (3.95), (3.96) and (3.97) are placed in their corresponding locations in the global system matrix. As an example, stiffness matrix terms for element  $\Omega_e$  are placed in columns and rows which correspond to that element's nodes. Thus, first terms in expressions (3.89) and (3.90) are obtained as:

$$\sum_{e=1}^E \int_{\Omega_e} k \nabla \phi_i \cdot \nabla \phi_j \, dx dy = \sum_{e=1}^E K_{ij}^e \quad (3.98)$$

$$\sum_{e=1}^E \int_{\Omega_e} f \phi_i \, dx dy = \sum_{e=1}^E F_i^e \quad i, j = 1, 2, \dots, N, \quad (3.99)$$

where  $N$  is the global system matrix dimensions, i.e.:

$$\sum_{e=1}^E (K_{ij}^e u_j - F_i^e + \Sigma_i^e) = 0, \quad i = 1, \dots, N. \quad (3.100)$$

It should be noticed that contributions from boundary conditions to matrix  $K_{ij}$  and vector  $F_i$  originate from term  $\Sigma_i^e$ , which can be written in following form:

$$\sum_{e=1}^E \Sigma_i^e = S_i^{(0)} + S_i^{(1)} + S_i^{(2)} \quad (3.101)$$

where

$$S_i^{(0)} = \sum_{e=1}^E \int_{\partial\Omega_e - \partial\Omega_h} \sigma_n \phi_i \, ds, \quad (3.102)$$

$$S_i^{(1)} = \int_{\partial\Omega_{1h}} \sigma_n \phi_i \, ds \quad (3.103)$$

$$S_i^{(2)} = \int_{\partial\Omega_{2h}} \sigma_n \phi_i \, ds. \quad (3.104)$$

Symbol  $\partial\Omega_e - \partial\Omega_h$  denotes the segment of boundary  $\partial\Omega_e$ , which does not include  $\partial\Omega_h$  (i.e. the part of  $\partial\Omega_e$  which is related to common element boundaries), hence term  $S_i^{(0)}$  is only defined in internal nodes.

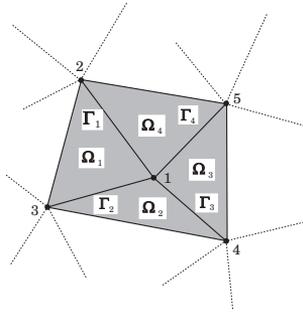


Figure 3.14: Partition into inner elements.

For the purpose of a more thorough interpretation, figure 3.14, we will analyse four internal elements with a common node 1, for which the following holds:

$$\begin{aligned} S_1^{(0)} &= \sum_{e=1}^4 \int_{\partial\Omega_e} \sigma_n \phi_i \, ds = \\ &= \int_{\Gamma_1} \|\sigma_n\| \phi_i \, ds + \int_{\Gamma_2} \|\sigma_n\| \phi_i \, ds + \int_{\Gamma_3} \|\sigma_n\| \phi_i \, ds + \int_{\Gamma_4} \|\sigma_n\| \phi_i \, ds. \end{aligned}$$

Based on conservation laws,  $\llbracket \sigma_n \rrbracket$  equals zero through the inter-surface where there are no point or line sources, hence for smooth  $f$  we have that:

$$S_1^{(0)} = 0. \quad (3.105)$$

On the other hand, if the source function  $f$  contains sources within a point or a line, then the jump  $\llbracket \sigma \rrbracket_n$  represents a source equation and is not zero. If the source is a point, then, strictly speaking, variation formulation is not applicable. Thus, it is assumed that  $f$  has the following form:

$$f(x, y) = \bar{f}(x, y) + \hat{f} \delta(x - x_i, y - y_i) \quad (3.106)$$

where  $\bar{f}$  is the smooth (integrable) part of  $f$ , and  $\hat{f} \delta(x - x_i, y - y_i)$  denotes the source in point  $(x_i, y_i) \in \Omega_h$ , with a magnitude  $f$ . As was the case with the one-dimensional problem, mesh nodes  $\Omega_h$  are placed in the source point, since then integrals (3.91) and (3.96) only contain the smooth part of  $f$ :

$$S_1^{(0)} = \sum_{e=1}^4 \int_{\partial\Omega_e} \sigma_n \phi_1 ds = \sum_{m=1}^4 \int_{\Gamma_m} \llbracket \sigma_n \rrbracket \phi_1 ds. \quad (3.107)$$

The presence of the base function  $\phi_1$  indicates that  $S_1^{(0)}$  represents the weight mean of these jumps in the internal node 1. If we (figuratively) balance these flux jumps which differ from zero, with a source in point  $\hat{f}$ , we obtain:

$$S_1^{(0)} = \hat{f} \quad (3.108)$$

when (3.106) holds. It should also be noted that the point source in the case of a two-dimensional problem provides a singular solution.

According to essential boundary conditions, values  $u_h$  are given in nodes at  $\partial\Omega_{1h}$ . Since  $\sigma_n$  is unknown at  $\partial\Omega_{1h}$ ,  $S_i^{(1)}$  cannot be defined here. However, if all nodal displacements,  $u_1, u_2, \dots, u_N$ , are known,  $S_i^{(1)}$  can be approximated directly from (3.103).

On the boundary  $\partial\Omega_{2h}$ , natural boundary conditions are defined, thus the following holds:

$$\sigma_n(s) = \hat{\sigma}(s), \quad (3.109)$$

thus, we have the following approximation:

$$S_i^{(2)} \cong \sum_{i=1}^N \int_{\partial\Omega_{2h}} \hat{\sigma} \phi_i ds. \quad (3.110)$$

Symbol  $\partial\Omega_{2h}^e$  refers to the part of  $\partial\Omega_e$  which intersects with  $\partial\Omega_{2h}$ . Now we have reached a linear algebraic system of equations as follows:

$$K_{ij} u_j = F_i - S_i^{(2)}, \quad i, j = 1, 2, \dots, N, \quad (3.111)$$

where

$$\begin{aligned} K_{ij} &= \sum_{e=1}^E K_{ij}^e \\ F_i &= \sum_{e=1}^E F_i^e. \end{aligned} \quad (3.112)$$

**An illustration of finite element method application to solving of two-dimensional problems**

Let's define the following problem:

$$\begin{aligned}
 -\Delta u(x,y) &= f(x,y) & \text{na } \Omega \\
 u &= 0 & \text{na } \Gamma_{41}, \\
 \frac{\partial u}{\partial n} &= 0 & \text{na } \Gamma_{12}, \Gamma_{25}, \Gamma_{67} \text{ i } \Gamma_{74} \\
 \frac{\partial u}{\partial n} + \beta u &= \gamma & \text{na } \Gamma_{56},
 \end{aligned}
 \tag{3.113}$$

where  $\Omega$  is the polygonal domain, figure 3.15, and  $\Gamma_{41}, \Gamma_{12} \dots \Gamma_{74}$  are boundary segments. In this case  $\partial\Omega_1 = \Gamma_{41}$  and  $\partial\Omega_2 = \Gamma_{12} \cup \Gamma_{25} \cup \Gamma_{56} \cup \Gamma_{67} \cup \Gamma_{74}$ . The analysis of this problem consists of:

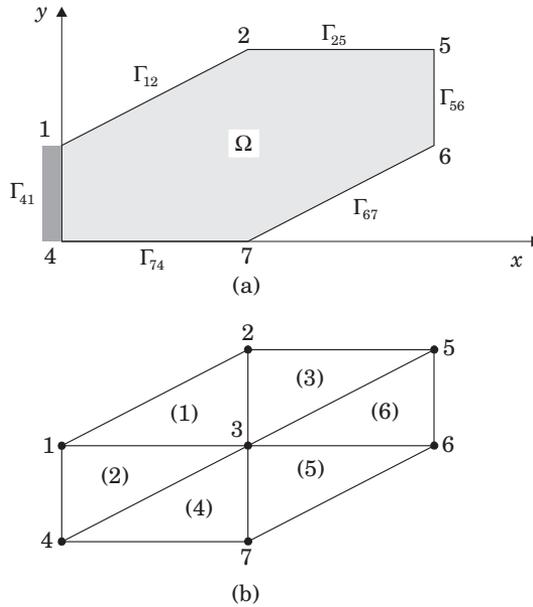


Figure 3.15: Two-dimensional problem (a) and division into finite element (b).

1. We divide the domain into six triangular elements with seven nodes, fig. 3.15, within which we define a linear approximation  $u_h$  for solution  $u$  of equation (3.112). Here  $\partial\Omega_1 = \partial\Omega_{1h}$  and  $\partial\Omega_2 = \partial\Omega_{2h}$ , since the domain boundary is linear and thus identical to the finite element boundary.
2. Next, by using (3.95) and (3.96), we calculate  $K^e$  and  $f^e$ ,  $e = 1, 2, \dots, 6$ :

$$K^1 = \begin{bmatrix} K_{11}^1 & K_{12}^1 & K_{13}^1 & 0 & 0 & 0 & 0 \\ K_{21}^1 & K_{22}^1 & K_{23}^1 & 0 & 0 & 0 & 0 \\ K_{31}^1 & K_{32}^1 & K_{33}^1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad F^1 = \begin{bmatrix} f_1^1 \\ f_2^1 \\ f_3^1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$K^2 = \begin{bmatrix} K_{11}^2 & 0 & K_{12}^2 & K_{13}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ K_{21}^2 & 0 & K_{22}^2 & K_{23}^2 & 0 & 0 & 0 \\ K_{31}^2 & 0 & K_{32}^2 & K_{33}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad F^2 = \begin{bmatrix} f_1^2 \\ 0 \\ f_2^2 \\ f_3^2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.114)$$

$$K^6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_{11}^6 & 0 & K_{12}^6 & K_{13}^6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_{21}^6 & 0 & K_{22}^6 & K_{23}^6 & 0 \\ 0 & 0 & K_{31}^6 & 0 & K_{32}^6 & K_{33}^6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad F^6 = \begin{bmatrix} 0 \\ 0 \\ f_1^6 \\ 0 \\ f_2^6 \\ f_3^6 \\ 0 \end{bmatrix}$$

3. We establish the global equation system:

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & K_{25} & 0 & 0 \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} & K_{36} & K_{37} \\ K_{41} & 0 & K_{43} & K_{44} & 0 & 0 & K_{47} \\ 0 & K_{52} & K_{53} & 0 & \tilde{K}_{55} & \tilde{K}_{56} & 0 \\ 0 & 0 & K_{63} & 0 & \tilde{K}_{65} & \tilde{K}_{66} & K_{67} \\ 0 & 0 & K_{73} & K_{74} & 0 & K_{76} & K_{77} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ \tilde{F}_5 \\ \tilde{F}_6 \\ F_7 \end{bmatrix} - \begin{bmatrix} \Sigma_1 \\ 0 \\ 0 \\ \Sigma_4 \\ \Sigma_5 \\ \Sigma_6 \\ 0 \end{bmatrix} \quad (3.115)$$

where  $F_1 = f_1^1 + f_1^2$ ;  $F_2 = f_2^1 + f_2^3 \dots$ , and  $\Sigma_i$  is defined based on (3.101). Terms denoted by  $\sim$  will be modified when taking into account boundary conditions at  $\Gamma_{56}$ .

4. Since non-homogeneous boundary conditions are defined only on the segment which connects nodes 5 and 6,  $\gamma$  and  $P$  have the following form:

$$\gamma = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \gamma_5 \\ \gamma_6 \\ 0 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{55} & P_{56} & 0 \\ 0 & 0 & 0 & 0 & P_{65} & P_{66} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.116)$$

hence the linear algebraic equation system (3.115) becomes:

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & K_{25} & 0 & 0 \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} & K_{36} & K_{37} \\ K_{41} & 0 & K_{43} & K_{44} & 0 & 0 & K_{47} \\ 0 & K_{52} & K_{53} & 0 & K_{55} & K_{56} & 0 \\ 0 & 0 & K_{63} & 0 & K_{65} & K_{66} & K_{67} \\ 0 & 0 & K_{73} & K_{74} & 0 & K_{76} & K_{77} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = \begin{bmatrix} F_1 - \Sigma_1 \\ F_2 \\ F_3 \\ F_4 - \Sigma_4 \\ F_5 \\ F_6 \\ F_7 \end{bmatrix} \quad (3.117)$$

where  $K_{55} = \tilde{K}_{55} + P_{55}$ ,  $K_{56} = \tilde{K}_{56} + P_{56}$ ,  $F_5 = \tilde{K}_5 + \gamma_5$ .

5. We can now introduce essential boundary conditions  $u_1 = u_4 = 0$ , which reduces the linear algebraic equation system (3.117) to a system of equations with five unknowns:

$$\begin{bmatrix} K_{22} & K_{23} & K_{25} & 0 & 0 \\ K_{32} & K_{33} & K_{34} & K_{36} & K_{37} \\ K_{52} & K_{53} & K_{55} & K_{56} & 0 \\ 0 & K_{63} & K_{65} & K_{66} & K_{67} \\ 0 & K_{73} & 0 & K_{76} & K_{77} \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = \begin{bmatrix} F_2 \\ F_3 \\ F_5 \\ F_6 \\ F_7 \end{bmatrix} \quad (3.118)$$

which is solved for unknown displacements  $u_2, u_3, u_5, u_6$  and  $u_7$ . Remaining two equations  $\Sigma_1$  and  $\Sigma_4$ :

$$\begin{aligned} -\Sigma_1 &= K_{12}u_2 + K_{13}u_3 + K_{14}u_4 - F_1, \\ -\Sigma_4 &= K_{43}u_3 + K_{47}u_7 - F_4. \end{aligned} \quad (3.119)$$

Other solution properties can be determined based on the completely known  $u_h$ , equation (3.86).

### 3.5.1 Determining of the finite element matrix

The first step involves the defining of a finite element mesh, followed by determining of necessary matrices. It should be taken into account that necessary matrices and vectors are very complex to determined when using a global coordinate system  $(x, y)$ , fig. 3.16, and that each finite element would then require individual determining of interpolation boundaries. Thus, matrices and vectors are determined in the local coordinate system  $(\xi, \eta)$ , and are then transformed into the global system. This allows for a unique procedure for all finite elements, making this task significantly easier.

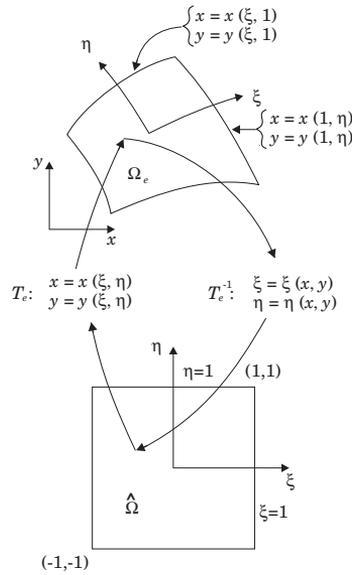


Figure 3.16: Finite element mapping.

### Local to global transformations

Let us introduce the “master” element  $\hat{\Omega}$ , of simple form, whose coordinates are defined as  $-1 \leq \xi \leq 1$  and  $-1 \leq \eta \leq 1$ . In this case, coordinate transformation can be written as:

$$\begin{aligned} x &= x(\xi, \eta) \\ y &= y(\xi, \eta). \end{aligned} \tag{3.120}$$

The basic idea behind introducing the “master” element is the ability to interpret a finite element mesh generation achieved by a series of transformations,  $[T_e]$ , 3.120), shown in fig. 3.17.

To this end, we will assume that functions  $x$  and  $y$  are continuously differentiable with respect to  $\xi$  and  $\eta$ , hence we have that:

$$dx = \frac{\partial x}{\partial \xi} d\xi + \frac{\partial x}{\partial \eta} d\eta \quad dy = \frac{\partial y}{\partial \xi} d\xi + \frac{\partial y}{\partial \eta} d\eta \tag{3.121}$$

i.e. in matrix form

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{bmatrix} d\xi \\ d\eta \end{bmatrix} \tag{3.122}$$

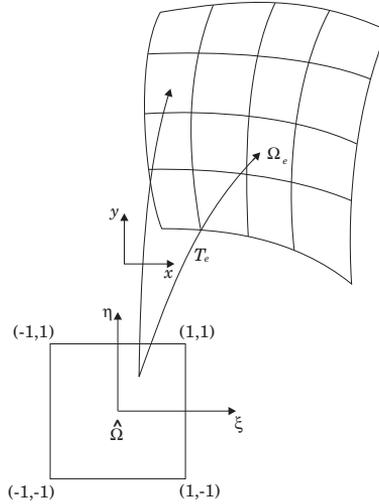


Figure 3.17: Local to global transformation.

Partial derivative matrix of the  $2 \times 2$  order is called the transformation Jacobian and is denoted by  $\mathbf{J}$ . In order for the inverse transformation to hold, it is obvious that  $|\mathbf{J}| \neq 0$ . In this case, we have that:

$$\begin{bmatrix} d\xi \\ d\eta \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} dx \\ dy \end{bmatrix} = \frac{1}{|\mathbf{J}|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & \frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix}, \quad (3.123)$$

thus

$$\begin{aligned} \xi &= \xi(x, y) \\ \eta &= \eta(x, y) \end{aligned} \quad (3.124)$$

Defines the inverse transformation. Considering the below expression, analogous to (3.122)

$$\begin{bmatrix} d\xi \\ d\eta \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix} \quad (3.125)$$

it is clear that the following holds:

$$\begin{aligned} \frac{\partial \xi}{\partial x} &= \frac{1}{|\mathbf{J}|} \frac{\partial y}{\partial \eta}, & \frac{\partial \xi}{\partial y} &= -\frac{1}{|\mathbf{J}|} \frac{\partial x}{\partial \eta}, \\ \frac{\partial \eta}{\partial x} &= -\frac{1}{|\mathbf{J}|} \frac{\partial y}{\partial \xi}, & \frac{\partial \eta}{\partial y} &= \frac{1}{|\mathbf{J}|} \frac{\partial x}{\partial \xi}. \end{aligned} \quad (3.126)$$

Now we can define the conditions necessary for a function in order to be used as a transformation function:

1. Within each element, functions  $\xi = \xi(x, y)$  and  $\eta = \eta(x, y)$  must be invertible and continuously differentiable in order for the relation (3.126) to be applicable to them.
2. Mesh generated using a series of transformations  $[T_e]$  must not contain overlapping elements and empty spaces between them.

3. Each transformation must be easy to perform, based on element geometry.
4. Functions  $x(\xi, \eta)$  and  $y(\xi, \eta)$  should be simple in the mathematical sense.

These conditions are met by shape functions of finite elements, hence it is natural to use them as the transformation functions:

$$\begin{aligned} x &= x_i \psi_i(\xi, \eta) \\ y &= y_i \psi_i(\xi, \eta), \end{aligned} \quad (3.127)$$

where  $(x_i, y_i)$  are the nodal coordinates of point  $i$  and  $M$  is the number of elements. The following holds:

$$\frac{\partial \xi}{\partial x} = \frac{1}{|\mathbf{J}|} y_j \frac{\partial \psi_j}{\partial \eta}, \quad \frac{\partial \xi}{\partial y} = -\frac{1}{|\mathbf{J}|} x_j \frac{\partial \psi_j}{\partial \eta}, \quad (3.128)$$

$$\frac{\partial \eta}{\partial x} = -\frac{1}{|\mathbf{J}|} y_j \frac{\partial \psi_j}{\partial \xi}, \quad \frac{\partial \eta}{\partial y} = \frac{1}{|\mathbf{J}|} x_j \frac{\partial \psi_j}{\partial \xi}, \quad (3.129)$$

$$|\mathbf{J}| = \left\{ x_i \frac{\partial \psi_i}{\partial \xi} \right\} \left\{ y_j \frac{\partial \psi_j}{\partial \eta} \right\} - \left\{ x_i \frac{\partial \psi_i}{\partial \eta} \right\} \left\{ y_j \frac{\partial \psi_j}{\partial \xi} \right\}. \quad (3.130)$$

As an example, we will illustrate this using a four-node “master” element  $\hat{\Omega}$  and four finite elements,  $e = 1, 2, 3, 4$ , obtained by transformation given by (3.127). Shape functions are:

$$\begin{aligned} \psi_1(\xi, \eta) &= \frac{1}{4}(1-\xi)(1-\eta) & \psi_2(\xi, \eta) &= \frac{1}{4}(1+\xi)(1-\eta) \\ \psi_3(\xi, \eta) &= \frac{1}{4}(1+\xi)(1+\eta) & \psi_4(\xi, \eta) &= \frac{1}{4}(1-\xi)(1+\eta). \end{aligned} \quad (3.131)$$

By transforming into element  $\Omega_1$ ; fig. 3.18

$$(x_i, y_i) = \{(3, 0), (3, 1), (0, 1), (0, 0)\} \Rightarrow$$

$$\begin{aligned} x &= 3\psi_1 + 3\psi_2 = \frac{3}{2}(1-\eta) \\ y &= \psi_2 + \psi_3 = \frac{1}{2}(1+\xi). \end{aligned} \quad (3.132)$$

Since  $|\mathbf{J}| > 0$ , the transformation is invertible. This actually involves simple (linear) expansion and contraction of corresponding element sides, fig. 3.18.

By transforming into element number  $\Omega_2$  in the same way like before, results in  $|\mathbf{J}| = -\frac{3}{4}$ . It should be noticed that the only difference compared to element  $\Omega_1$  is in node numeration – here it has a negative mathematical direction, which should be generally avoided.

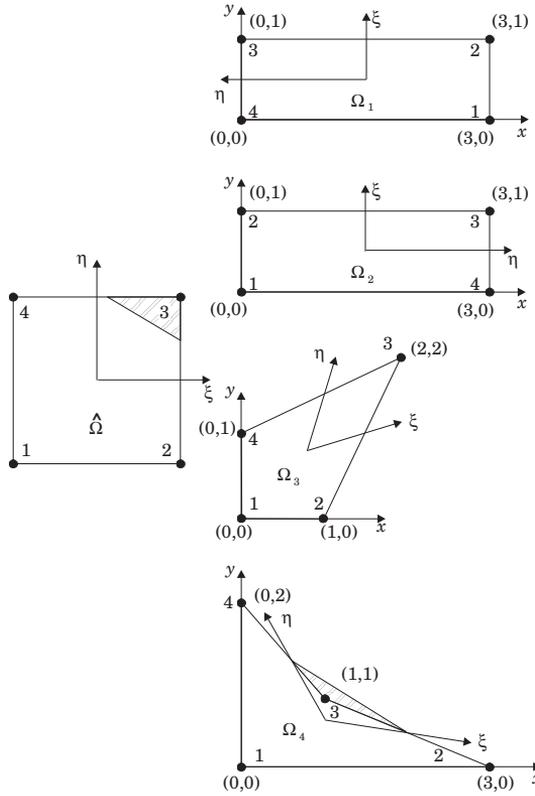


Figure 3.18: Finite element transformation.

By transforming into element  $\Omega_3$ , fig. 3.18.

$$(x_i, y_i) = \{(0, 0), (1, 0), (2, 2), (0, 1)\} \Rightarrow$$

$$x = \psi_2 + 2\psi_3 = \frac{1}{4}(3 + 3\xi + \eta + \xi\eta)$$

$$y = 2\psi_3 + \psi_4 = \frac{1}{4}(3 + \xi + 3\eta + \xi\eta)$$

$$|\mathbf{J}| = \frac{1}{2} + \frac{1}{8}\xi + \frac{1}{8}\eta.$$

Considering that  $|\mathbf{J}| > 0$  for  $\xi, \eta \in (-1, 1)$ , this transformation is invertible. Value of  $|\mathbf{J}|$  is lowest at node 1 and highest at node 3, indicating relative elongation of different parts of  $\hat{\Omega}$  via transformation  $T_3$ .

By transforming element  $\Omega_4$ , fig. 3.18, we obtain:

$$(x_i, y_i) = \{(0, 0), (3, 0), (1, 1), (0, 2)\} \Rightarrow$$

$$x = 3\psi_2 + \psi_3$$

$$y = \psi_3 + 2\psi_4$$

(3.133)

$$|\mathbf{J}| = \frac{1}{8}(5 - 3\xi - 4\eta)$$

which is not greater than zero for  $\forall \xi, \eta \in (-1, 1)$  - for example for  $3\xi = 5 - 4\eta$   $|\mathbf{J}| = 0$ , beneath the greater-than-zero line, and above the less-than-zero line! The area of line  $3\xi = 5 - 4\eta$  is transformed outside of  $\Omega_4$ , hence this element is obviously unacceptable. In addition, there is a problem in the node 3 corner, where it is greater than  $\pi$ . It can be shown that the transformation function (3.127) is integrable if the angles of the rectangle are  $< \pi$ .

For square shape functions (e.g. the eight-node finite elements), constraints, apart from those previously mentioned, are related to nodes which are not in element corners (fig. 3.19). Namely, in order to preserve the invertibility of transformation functions, it is necessary for these nodes to be located at midpoints of sides, i.e. elements (in the case internal nodes exist).

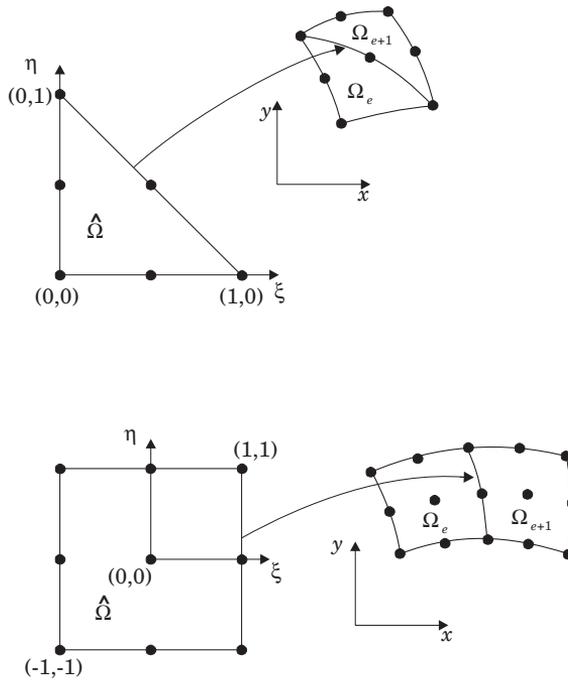


Figure 3.19: Quadratic finite elements.

### 3.5.2 Calculating of finite element matrices

We can now calculate all of the necessary finite element matrices and vectors:

$$k_{ij}^e = \int_{\Omega^e} \nabla \psi_i^e \cdot \nabla \psi_j^e \, dx dy, \tag{3.134}$$

$$f_i^e = \int_{\Omega^e} f \psi_i^e \, dx dy, \tag{3.135}$$

$$s_i^e = \int_{\Omega^e} \hat{\sigma} \psi_i^e \, dx dy. \tag{3.136}$$

Interpolation function gradient can be written as:

$$\nabla \psi_i(\xi, \eta) = u_i \frac{\partial \psi_i}{\partial \xi^\alpha} \mathbf{g}^\alpha, \quad \alpha = 1, 2, \quad (3.137)$$

where  $\xi^1 \equiv \xi$  and  $\xi^2 \equiv \eta$ , and  $\mathbf{g}^\alpha$ , is the **contravariant base vector** of coordinate systems for element  $\xi^\alpha$ . Hereinafter we will denote partial derivatives of interpolation functions in their shortened form:

$$\psi_{i\alpha} = \frac{\partial \psi_i}{\partial \xi^\alpha}. \quad (3.138)$$

By replacing interpolation function gradients obtained in this manner into (3.134)-(3.136), and taking into account that the element surface is:

$$dx dy = J d\xi d\eta, \quad (3.139)$$

we obtain:

$$k_{ij}^e = \int_{\Omega^e} \psi_{i\alpha}^e k g^{\alpha\beta} \psi_{j\beta}^e J d\xi d\eta. \quad (3.140)$$

where the following applies, formally:

$$g^{\alpha\beta} = \mathbf{g}^\alpha \cdot \mathbf{g}^\beta. \quad (3.141)$$

In a practical sense,  $g^{\alpha\beta}$  is calculated by inverting the expression given below:

$$g_{\alpha\beta} = \delta_{kl} z_\alpha^k z_\beta^l, \quad k, l = 1, 2, \quad (3.142)$$

where

$$z_\alpha^k = \frac{\partial z^k}{\partial \xi^\alpha}, \quad (3.143)$$

while

$$z^1 = x, \quad z^2 = y \quad (3.144)$$

are the Cartesian coordinates.

Taking into account (3.127), it is obvious that  $z_\alpha^k = z_i^k \xi_{i\alpha}$ . For the observed two-dimensional case, the matrix of expressions given as  $g_{\alpha\beta}$  is:

$$[g_{\alpha\beta}] = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \quad (3.145)$$

where

$$\begin{aligned} g_{11} &= (z_1^1)^2 + (z_1^2)^2 \\ g_{12} &= g_{21} = z_1^1 z_1^2 + z_1^2 z_2^2 \\ g_{22} &= (z_2^1)^2 + (z_2^2)^2. \end{aligned} \quad (3.146)$$

The inverse matrix is by definition:

$$[g^{\alpha\beta}] = [g_{\alpha\beta}]^{-1} = \frac{1}{|g_{\alpha\beta}|} \begin{bmatrix} g_{11} & -g_{12} \\ -g_{21} & g_{22} \end{bmatrix}. \quad (3.147)$$

It is easy to show that the determinant of matrix  $g_{\alpha\beta}$  is:

$$|g_{\alpha\beta}| = g_{11} g_{22} - g_{12}^2 = (z_1^2 z_2^2 - z_2^1 z_1^1)^2 = J^2. \quad (3.148)$$

!!

Sada članovi matrice krutosti možemo da napišemo u obliku

$$k_{ij}^e = \int_{\Omega^e} \frac{k}{J} [\psi_{i1}^e \ \psi_{i2}^e] \begin{bmatrix} g_{11}^e & -g_{12}^e \\ -g_{21}^e & g_{22}^e \end{bmatrix} \begin{bmatrix} \psi_{j1}^e \\ \psi_{j2}^e \end{bmatrix} d\xi d\eta. \quad (3.149)$$

The basic problem with the integration of the above expression is the occurrence of a relatively complex function – the Jacobian in the denominator, and due to this, numerical squaring is typically used.

In a general case, the expression for numerical integration is defined by determining of coordinates  $(\xi_t, \eta_t)$  of a specific number of  $N_t$  integration points within the domain for which the integral is being calculated, along with the quantity  $w_t$  which is referred to as the **integration weight function** ( $t = 1, 2, \dots, N_t$ ). Thus, if expression  $G(\xi, \eta)$  needs to be integrated along the area  $\Omega_e$ , we will use the following relation:

$$\int_{\Omega_e} G(\xi, \eta) d\xi d\eta = \sum_{t=1}^{N_t} \hat{G}(\xi, \eta) w_t + E,$$

where  $E$  represents the integration error.

If the integrand  $G(\xi, \eta)$  is a polynomial, the integration order should be sufficiently high for the integration to be accurate. For example if we use the Gaussian square method to integrate a linear function in the canonical triangle, this integration will be exact when using the single point rule  $(\xi_1, \eta_1)$ , in the centre, with  $w_1 = \frac{1}{2}$ .

As for expression (3.135), it is obvious that it is easier to calculate than (3.134), hence we will leave it for the practice examples.

Expression (3.136) is a boundary integral and its calculation is slightly more complicated. We calculate it for contours of elements  $\partial\Omega_{2h}^e$  of domain  $\Omega^e$ , along which the natural boundary conditions are defined. We will demonstrate this calculation for the edge where  $\xi = \text{const.}$ , for the canonical element, which needs to be projected onto  $\partial\Omega_{2h}^e$ .

In the observed case  $g_{\alpha\beta}$  is reduced to  $g_{22}$ :

$$g_{22} = \left( \frac{\partial z^1}{\partial \xi^2} \right)^2 + \left( \frac{\partial z^2}{\partial \xi^2} \right)^2,$$

which, at the same time, represents the “determinant” for expression  $g_{\alpha\beta}$ . As can be seen, the Jacobian is equal to the square root of the matrix:

$$J = \left[ \left( \frac{\partial x}{\partial \eta} \right)^2 + \left( \frac{\partial y}{\partial \eta} \right)^2 \right]^{1/2},$$

Hence the integral is:

$$s_i^e = \int_{\partial\Omega_{2h}^e} \hat{\sigma} \psi_i^e \left[ \left( \frac{\partial x}{\partial \eta} \right)^2 + \left( \frac{\partial y}{\partial \eta} \right)^2 \right]^{1/2} d\eta.$$

It is common practice to calculate these integrals using numerical integration.

### 3.6 Program for solving of elliptical two-dimensional problems using finite element method

#### 3.6.1 Uvodne napomene

A program which uses finite element method will be shown in this section. It consists of complete sub-programs (with insignificant changes) and solution to a number of problems given in reference [5].

Let us first define the problem that needs to be solved:

A real function  $u(x, y)$  should be determined in the points within region  $\Omega \subset R^2$ , such that it satisfies the **linear partial differential equation**:

$$-\nabla \cdot (k\nabla u) + bu = f. \quad (3.150)$$

Let us mention that this differential equation is a bit more general than (3.51). The difference lies in term  $bu$ . By introducing this term, it is possible to analyse problems such as continuously supported membrane, or chemical reactions which depend on the temperature.

At points within a part of boundary  $\Omega (\partial\Omega_1)$ , function  $u$  can have known values (essential boundary conditions):

$$u = \hat{u},$$

whereas in the case of points on the other part of the boundary  $(\partial\Omega_2)$ , this function can have known values of derivatives in the direction perpendicular to the boundary (natural boundary conditions):

$$-k \frac{\partial u}{\partial n} = pu - \gamma.$$

This form of natural boundary conditions is also somewhat broader than (3.54), and allows the analysis of problems where the derivative in the direction of the normal depends on the value of function  $u$  at the boundaries.

Region  $\Omega$  can be divided into subregions such that  $k$ ,  $b$  and  $f$  can have different values in each of them.  $k$  and  $b$  are constant in these subregions. As for  $f$ , we can assume that it has the following structure

$$f(x, y) = \bar{f} + \hat{f}\delta(x - x_i, y - y_i)$$

where  $\bar{f}$  and  $\hat{f}$  are constants.

Even though we allow the region  $\Omega$  to be divided into subregions, the jump in quantity  $\nabla u \cdot n$  has to be equal to zero in all points at the boundaries between individual subregions.

A problem defined in this way can be translated into variation form in a way shown in the previous chapters. After the finite element approximation, it is transformed into a system of linear algebraic equations in terms of unknown values of function  $u$  in nodal points.

A triangular isoparametric element with six nodes and rectangular isoparametric element with nine nodes will be used here. Let us describe these elements and conventions related to them, which are necessary to follow in order to ensure that the program will function properly.

The following figure shows these elements in parametric  $(\xi, \eta)$  plane, with standard node and side numbering for each of them.

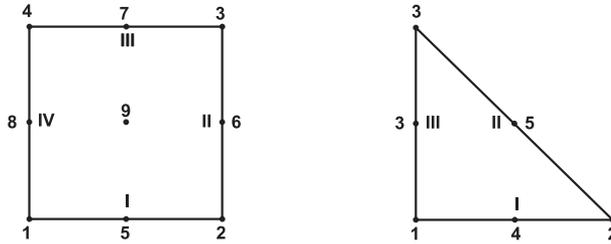


Figure 3.20: Isoparametric finite elements.

Base function for a rectangular element in parametric plane are:

$$\begin{aligned}
 \psi_1 &= \frac{1}{4}(\xi^2 - \xi)(\eta^2 - \eta) & \psi_5 &= \frac{1}{2}(1 - \xi^2)(\eta^2 - \eta) \\
 \psi_2 &= \frac{1}{4}(\xi^2 + \xi)(\eta^2 - \eta) & \psi_6 &= \frac{1}{2}(\xi^2 + \xi)(1 - \eta^2) \\
 \psi_3 &= \frac{1}{4}(\xi^2 + \xi)(\eta^2 + \eta) & \psi_7 &= \frac{1}{2}(1 - \xi^2)(\eta^2 + \eta) \\
 \psi_4 &= \frac{1}{4}(\xi^2 - \xi)(\eta^2 + \eta) & \psi_8 &= \frac{1}{2}(\xi^2 - \xi)(1 - \eta^2) \\
 \psi_9 &= (1 - \xi^2)(1 - \eta^2)
 \end{aligned}$$

Base function for a triangular element in parametric plane are:

$$\begin{aligned}
 \psi_1 &= 2\zeta(\zeta - 0.5) & \psi_4 &= 4\zeta\xi \\
 \psi_2 &= 2\xi(\xi - 0.5) & \psi_5 &= 4\xi\eta \\
 \psi_3 &= 2\eta(\eta - 0.5) & \psi_6 &= 4\eta\zeta \\
 \zeta &= 1 - \xi - \eta
 \end{aligned}$$

### 3.6.2 Subprogram structure

The program consists of three basic parts: the preprocessor, processor and postprocessor. Subprograms which make up the preprocessor read input data (from an input file named “fem.in”), which define the problem and are used as the base for generating of all of the necessary quantities. The processor then forms a system matrix and system vector and finally solves the obtained system of equations. The postprocessor displays the obtained results. All relevant input and output data are written in the output database, “fem.out”.

Let us now list some of the most important variables in the program:

- MAXN – maximum number of nodes allowed by the program.
- MAXE – maximum number of elements.
- MAXM – Maximum number of subregions into which region  $\Omega$  can be divided into.
- MAXBCE – maximum number of points in which function  $\hat{u}$  is defined.
- MAXBCN – maximum number of element sides at the boundary of region  $\Omega$  where function  $u$  variation is defined in the normal direction.
- MAXPT – Maximum number of points at which  $f$  has a  $\delta$  function type singularity.
- MAXBAND - maksimalna poluširina trake oko dijagonale matrice sistema u kojoj su skoncentrisane vrednosti različite od nule. ???????????????????

All of these quantities need to be defined in the program (depending on the available working memory of the computer), before compiling the program.

Let us briefly describe all subprograms which are a part of this program.

- SETINT defines all quantities necessary for numerical integration via Gauss method along a surface or a line.
- PREP is the preprocessor. Every series of input data for a given problem starts with a short text than identified it. If this text only contains the word “end”, the program stops there.

- RNODE reads data from the input file (“fem.in”), which are related to nodes used to “cover” the region  $\Omega$ . Based on data containing the initial and final point coordinates and the number of nodes, the subprogram generates data about all of the nodes located between them. In all preprocessor subprograms, generated data are written in the output database “fem.out”.
- RELEM reads the data based on which series of elements that cover the approximation of region  $\Omega$  are automatically generated. Since the elements used are isoparametric and the base functions are bi-quadratic, these elements can be both curvilinear rectangular and triangular.
- RMAT reads the values  $k$ ,  $b$  and  $f$  for each subregion within  $\Omega$ .
- RBC reads the data about function  $f$  singularities and about essential and natural boundary conditions.
- PROC is the processor subprogram.
- FORMKF generates terms of matrix  $K$  and vector  $F$  without taking boundary conditions into account.
- ELEM calculates elements of matrix  $K_e$  (with dimensions of  $9 \times 9$  or  $6 \times 6$ ) and vector  $F_e$  which are obtained from element  $e$ .
- SHAPE2 calculates the values of all base functions in the parametric plane and their partial derivatives with respect to  $\xi$  and  $\eta$ .
- GETMAT returns the value of constants  $k$ ,  $b$  and  $\bar{f}$  based on the number of the subregion within  $\Omega$ .
- ASSMB sums the elements of matrix  $K_e$  and vector  $F_e$  with corresponding elements in  $K$  and  $F$ .
- APLYBC changes the elements of matrix  $K$  and vector  $F$  in accordance with singularity data for function  $f$ , as well as essential and natural boundary conditions. Let us mention that essential boundary conditions are imposed not by changing the matrix  $K$  dimensions, but by using the “penalising” method.
- BCINT calculates integrals along the boundary where natural boundary conditions are defined.
- SHAPE1 gives the value of one-dimensional base functions and their derivatives.
- SOLVE solves the equation system  $Ku = F$ .
- TRIB triangulates the symmetrical square matrix defined its non-zero elements which are located at the diagonal and above it.
- RHSB determines the element of the unknown vector  $u$ , based on the triangulated matrix  $K$  and vector  $F$ .
- POST writes data about the nodes and obtained approximate values of function  $u$  in them into the output database.

We will now describe the format of the input database “fem.in”, In order to better distinct between groups of data, each of them **must start** with a single line filled with random signs (such as ‘-’ or ‘\*’). Apart from the first, all data groups start with a line that contains a single integer which denotes the number of data rows that the group contains. All data are defined in free format. The input database can contain data for several different problems.

1. A data group is a text (maximum length of 80 signs) which briefly identifies a problem that needs to be solved, in one way or the other. The final problem in the database always contains the “end” or “END” in it.
2. A data group consists of a number of lines equal to the number of nodes which we would like to generate. Each line contains the following seven data:
  - The serial number of the first node (integer)
  - Number of nodes (integer)
  - Difference between each subsequent and previous node in the series (integer)
  - $x$  coordinate of the first node (float)

- y coordinate of the first node (float)
  - x coordinate of the last node (float)
  - y coordinate of the last node (float)
3. A data group contains data based on which series of elements can be generated, by using data for previously generated nodes.
    - Number of elements in a series (integer)
    - Difference between serial numbers of corresponding nodes which belong to adjacent elements in the series (integer)
    - Number of nodes in each element (6 or 9) (integer)
    - Serial number of subregion which the series elements belong to (integer)
    - 6 or 9 serial numbers (integers) of the first element in the series (pay attention the standard distribution of nodes in the element).
  4. grupa su podaci o vrednostima konstanti  $k$ ,  $b$  i  $\hat{f}$ 
    - Values of constant  $k$  (float)
    - Values of constant  $b$  (float)
    - Values of constant  $\hat{f}$  (float)
  5. grupa su podaci o eventualnim singularitetima funkcije  $f$ .
    - Number of nodes which contain singularities (integer)
    - Value of  $\hat{f}$  (float).
  6. A data group contains essential boundary conditions. The same value of function  $\hat{u}$  can be defined along the entire line of nodes.
    - Serial number of the first element in the series (integer)
    - Number of nodes in the series (integer)
    - Difference between adjacent nodes in the series (integer)
    - Values of  $\hat{u}$  (float).
  7. A data group defines natural boundary conditions. These conditions are also possible to define on the entire series of elements along the boundary.
    - Serial number of the first element in the series (integer)
    - Number of elements in the series (integer)
    - Difference in serial numbers of adjacent elements (integer)
    - Serial number of the side located at the region boundary (pay attention to the standard numbering of sides within an element) (integer)
    - Value of  $p$  (float)
    - Value of  $\gamma$  (float).

### 3.6.3 Examples

1. Example. We will solve the equation  $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 4$  for a  $2 \times 2$  square. All boundary conditions will be natural, except in  $(0,0)$ , wherein the unknown function  $u$  is equal to zero. Natural boundary conditions are chosen so that the solution of the problem is  $u = x^2 + y^2$ . Input database is as follows:

```
-----
Test primer br. 1: Jednacina Lap(u)=4. Resenje: u=x^2+y^2.
-----
```

```
3
```

```
1 3 1 0. 0.    2. 0.
4 3 1 0. 1.    2. 1.
7 3 1 0. 2.    2. 2.
```

```

-----
1
1 0 9 1 1 3 9 7 2 6 8 4 5
-----
1
-1. 0. 4.
-----
0
-----
1
1 1 0 0.
-----
4
1 1 0 1 0. 0.
1 1 0 2 0. -4.
1 1 0 3 0. -4.
1 1 0 4 0. 0.
-----
end
-----

```

After running the program, the "fem.out" database is formed:

Test primer br. 1: Jednacina Lap(u)=4. Resenje:  $u=x^2+y^2$ .

node no	x-coordinate	y-coordinate
1	.0000	.0000
2	1.0000	.0000
3	2.0000	.0000
4	.0000	1.0000
5	1.0000	1.0000
6	2.0000	1.0000
7	.0000	2.0000
8	1.0000	2.0000
9	2.0000	2.0000

elem no	ne	mat	node numbers
1	9	1	1 3 9 7 2 6 8 4 5

mat no	k	b	f
1	-1.0000	.0000	4.0000

node no	essent. value
1	.0000

element no	side	p	gamma
------------	------	---	-------

1	1	.0000	.0000
1	2	.0000	-4.0000
1	3	.0000	-4.0000
1	4	.0000	.0000

node	x	y	u
1	.000	.000	0.000
2	1.000	.000	1.000
3	2.000	.000	4.000
4	.000	1.000	1.000
5	1.000	1.000	2.000
6	2.000	1.000	5.000
7	.000	2.000	4.000
8	1.000	2.000	5.000
9	2.000	2.000	8.000

We can see that obtained values of function  $u$  are exact in all points which are covered with a single element. This is, of course, due to the fact that base functions are bi-square, and the exact solution is a square function along  $x$  and  $y$ . Figures 3.21 and 3.22 are showing the distribution of nodes in the area and lines of constant values of function  $u$ .

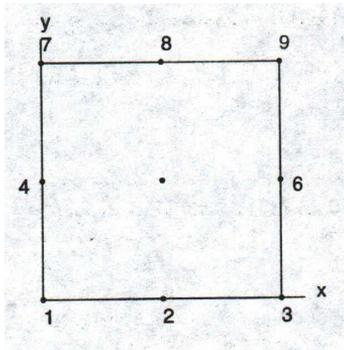


Figure 3.21: Disposition of nodes.

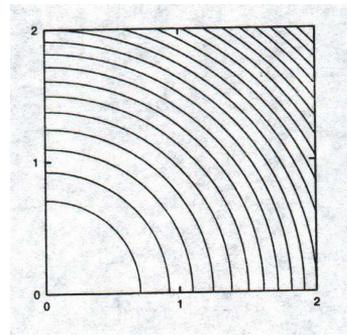


Figure 3.22: Constant function  $u$  values isolines.

- Example. Torsion of a straight cantilever rod with an arbitrary cross-section is determined by solving the equation

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = -2G\theta.$$

$\Phi$  is the Prandtl<sup>1</sup>,  $G$  is the shear modulus, and  $\theta$  is the torsion angle per unit length of the

<sup>1</sup>Prandtl Ludwig (1875-1953), a German scientist. He also studied fluid mechanics. Known for his extensive research involving fluid flow around obstacles and determining of supersonic, stationary and non-turbulent currents.

rod.  $\Phi(x,y)$  is the function of position of a point in the cross-section and in the points of its edges, these values must be equal to zero (essential boundary condition). We will solve the rod torsion problem for a cross-section shaped like an equilateral triangle, with a height of 9 units. We will select  $G$  and  $\theta$  such that  $G\theta = 1$ . Due to the symmetrical nature of the problem, only one half of region  $\Omega$  of the cross-section can be selected. On the boundary of such region, which coincides with the height of the equilateral triangle, the derivative of the Prandtl function along the normal direction must be equal to zero.

Input database:

```
-----
Test primer br. 2: Torzija stapa trougaonog popreznog preseka.
-----
```

```
7
 1  9  5  .00000 -3.00000  0.00000  5.00000
 2  9  5  1.29904 -3.00000  0.14434  5.00000
 3  9  5  2.59808 -3.00000  0.28868  5.00000
 4  9  5  3.89711 -3.00000  0.43301  5.00000
 5  9  5  5.19615 -3.00000  0.57735  5.00000
46  3  1  0.00000  5.50000  0.28867  5.50000
49  1  0  0.00000  6.00000  0.00000  6.00000
-----
```

```
4
4  10  9  1  1  3 13 11  2  8 12  6  7
4  10  9  1  3  5 15 13  4 10 14  8  9
1  0  6  1  41 43 49 42 47 46
1  0  6  1  43 45 49 44 48 47
-----
```

```
1
1.0  0.0  2.0
-----
```

```
0
-----
```

```
4
 1  4  1  0.0
 5  9  5  0.0
48  1  0  0.0
49  1  0  0.0
-----
```

```
2
 1  4  1  4  0.0 0.0
 9  1  0  3  0.0 0.0
-----
```

```
end
-----
```

Figure 3.23 shows the discretization of the region into finite elements. Figure 3.24 shows the lines of constant function  $\Phi$  values.

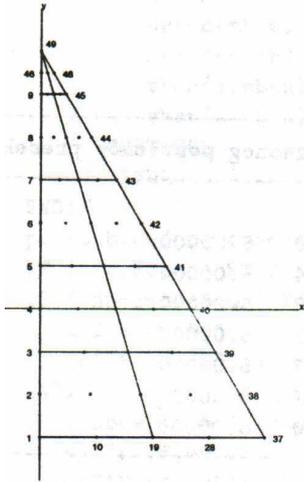


Figure 3.23: Finite element mesh.

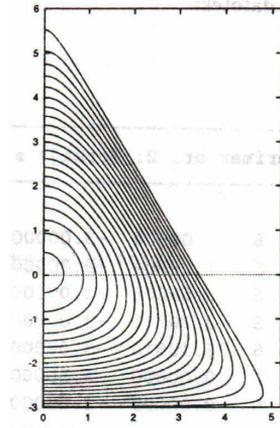


Figure 3.24: Constant function  $\Phi$  values isolines.

3. Example. Plane flow of an incompressible fluid will be shown using an example involving a cylinder with a radius of  $8\text{ cm}$ , between two parallel plates at a distance of  $32\text{ cm}$ . We will solve the Laplace differential equation:

$$\frac{\partial^2 \Psi}{\partial x^2} + \frac{\partial^2 \Psi}{\partial y^2} = 0$$

where  $\Psi(x, y)$  is the flow function. Due to the symmetry of the region around both  $x$  and  $y$  axes, a square with a side of  $16\text{ cm}$ , with a quarter of a circle ( $8\text{ cm}$  radius) removed from it can be selected as the region  $\Omega$ . The line between nodes 1 and 5 (see Figure 3.25) and the quarter-circle (where nodes 21 and 41 lie) will be considered as the zero streamline, i.e. a line along which function  $\Psi$  is equal to zero. The line containing elements 25, 35 and 45 is also the streamline, and function  $\Psi$  will be given value of 10 on this line. Figure 3.26 shows the obtained streamlines.

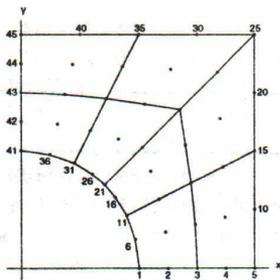


Figure 3.25: Finite element mesh.

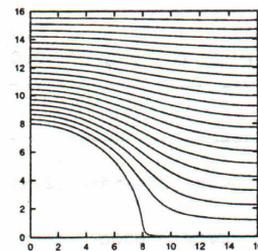


Figure 3.26: Constant function  $\Psi$  values isolines.

Input database for this example is:

```
-----
Test primer br. 3: Odredjivanje strujne funkcije.
-----
```

```
9
1 5 1 8.      0.      16.      0.
6 5 1 7.761   1.941   16.      4.
11 5 1 7.155  3.578   16.      8.
16 5 1 6.4    4.8     16.     12.
21 5 1 5.656  5.656   16.     16.
26 5 1 4.8    6.4     12.     16.
31 5 1 3.578  7.155   8.      16.
36 5 1 1.941  7.761   4.      16.
41 5 1 0.     8.      0.      16.
```

```
-----
2
4 10 9 1      1 3 13 11 2 8 12 6 7
4 10 9 1      3 5 15 13 4 10 14 8 9
-----
```

```
1
-1. 0. 0.
```

```
-----
0
-----
```

```
3
1 5 1 0.
6 8 5 0.
25 5 5 10.
```

```
-----
2
5 2 1 2 0. 0.
4 2 4 4 0. 0.
-----
```

```
end
-----
```

Finally, let us show the Fortran text for the whole FEM program.

```
PROGRAM fem
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
CHARACTER title*80
COMMON /ctitl / title
COMMON /ccon  / nnode , nelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
COMMON /celem / ne(maxe) , mat(maxe) , nodes(9,maxe)
COMMON /cmatl / prop(3,maxm)
COMMON /cbc   / nodbc(maxbce) , vbce(maxbce) , nelbc(maxbcn) ,
&          nside(maxbcn) , vbcn(2,maxbcn) , npt(maxpt) ,
&          vpt(maxpt)
```

```

COMMON /cmatrix/ gk(maxn,maxband) , gf(maxn)
COMMON /cint1 / etaq(3) , ww(3)
COMMON /cint2 / xiq(2,9) , xit(2,7) , wq(9) , wt(7)

c
c
OPEN (UNIT=5,FILE='fem.in')
OPEN (UNIT=6,FILE='fem.out')
CALL setint
100 CALL prep
CALL proc
CALL post
GOTO 100
END

SUBROUTINE setint
COMMON /cint1 / etaq(3) , ww(3)
COMMON /cint2 / xiq(2,9) , xit(2,7) , wq(9) , wt(7)
c nine point quadrature rule for quadrilaterals
sq35 = sqrt(3./5.)
wq(1) = 25./81.
wq(2) = 40./81.
wq(3) = 25./81.
wq(4) = 40./81.
wq(5) = 64./81.
wq(6) = 40./81.
wq(7) = 25./81.
wq(8) = 40./81.
wq(9) = 25./81.
xiq(1,1) = -sq35
xiq(2,1) = -sq35
xiq(1,2) = 0.
xiq(2,2) = -sq35
xiq(1,3) = sq35
xiq(2,3) = -sq35
xiq(1,4) = -sq35
xiq(2,4) = 0.
xiq(1,5) = 0.
xiq(2,5) = 0.
xiq(1,6) = sq35
xiq(2,6) = 0.
xiq(1,7) = -sq35
xiq(2,7) = sq35
xiq(1,8) = 0.
xiq(2,8) = sq35
xiq(1,9) = sq35
xiq(2,9) = sq35
c seven point rule for triangulars
sq15 = sqrt(15.0)
a = (155.0+sq15)/2400.0

```

```

b = (155.0-sq15)/2400.0
wt(1) = 9.0/80.0
wt(2) = a
wt(3) = a
wt(4) = a
wt(5) = b
wt(6) = b
wt(7) = b
a = (6.0+sq15)/21.0
b = (6.0-sq15)/21.0
c = (9.0+2.0*sq15)/21.0
d = (9.0-2.0*sq15)/21.0
xit(1,1) = 1.0/3.0
xit(2,1) = 1.0/3.0
xit(1,2) = a
xit(2,2) = d
xit(1,3) = d
xit(2,3) = a
xit(1,4) = a
xit(2,4) = a
xit(1,5) = b
xit(2,5) = c
xit(1,6) = c
xit(2,6) = b
xit(1,7) = b
xit(2,7) = b
c  tri point rule for line segment [-1,+1]
   etaq(1) = -sq35
   etaq(2) = 0.
   etaq(3) = sq35
   ww(1) = 5./9.
   ww(2) = 8./9.
   ww(3) = 5./9.
   RETURN
   END

SUBROUTINE prep
CHARACTER title*80
COMMON /ctitl / title
READ (5,*)
READ (5,99001) title
IF ( title.EQ.'end' .OR. title.EQ.'END' ) THEN
    CLOSE (6)
    CLOSE (5)
    STOP
ENDIF
PRINT *
PRINT * , 'reading problem data ...'
WRITE (6,*)

```

```

WRITE (6,99001) title
CALL rnode
CALL relem
CALL rmat
CALL rbc
RETURN
99001 FORMAT (A80)
END

SUBROUTINE rnode
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /ccon / nnode , nelelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
READ (5,*)
READ (5,*) nrec
IF ( nrec.LT.1 ) STOP 'number of node records less than 1'
nnode = 0
DO 100 irec = 1 , nrec
  READ (5,*) n1 , num , inc , x1 , y1 , xn , yn
  num1 = num - 1
  xnum = float(num1)
  x(1,n1) = x1
  x(2,n1) = y1
  nnode = nnode + 1
  IF ( num.NE.1 ) THEN
    dx = (xn-x1)/xnum
    dy = (yn-y1)/xnum
    DO 20 n = 1 , num1
      xn = float(n)
      in = n1 + n*inc
      IF ( in.GT.maxn ) STOP 'number of nodes grater than maxn'
      x(1,in) = x1 + xn*dx
      x(2,in) = y1 + xn*dy
      nnode = nnode + 1
20    CONTINUE
  ENDIF
100 CONTINUE
c  print nodal point coordinates
  WRITE (6,99001)
  DO 200 n = 1 , nnode
    WRITE (6,99002) n , x(1,n) , x(2,n)
200 CONTINUE
  RETURN
99001 FORMAT ( ' node no   x-coordinate   y-coordinate   '/')
99002 FORMAT (I7,2F16.4)
END

```

### Chapter 3. Comparison of finite element method and finite difference method

94

```
SUBROUTINE relelem
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /ccon / nnode , nelelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /celem / ne(maxe) , mat(maxe) , nodes(9,maxe)
DIMENSION node(9)
c read element data
READ (5,*)
READ (5,*) nrec
IF ( nrec.LT.1 ) STOP 'number of element records less than 1'
nel = 0
nband = 0
DO 200 irec = 1 , nrec
  READ (5,*) num , inc , nee , mate , (node(i),i=1,nee)
  nmin = maxn + 1
  nmax = 0
  DO 50 i = 1 , nee
    nmin = min(nmin,node(i))
    nmax = max(nmax,node(i))
50 CONTINUE
  nband = max(nband,nmax-nmin) + 1
  IF ( nband.GT.maxband ) STOP 'band width too big'
  n1 = nel + 1
  nel = nel + num
  IF ( nel.GT.maxe ) STOP 'number of elements greater than nelelem'
  DO 100 n = n1 , nel
    ninc = (n-n1)*inc
    DO 60 m = 1 , nee
      nodes(m,n) = node(m) + ninc
60 CONTINUE
    ne(n) = nee
    mat(n) = mate
100 CONTINUE
200 CONTINUE
  nelelem = nel
c print element definitions and find band width
WRITE (6,99001)
DO 300 n = 1 , nelelem
  nen = ne(n)
  WRITE (6,99002) n , nen , mat(n) , (nodes(i,n),i=1,nen)
300 CONTINUE
RETURN
99001 FORMAT (/ ' elem no ne mat',10X,'node numbers'/)
99002 FORMAT (I5,I7,I6,5X,9I4)
END

SUBROUTINE rmat
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
```

```

&          maxband=25)
COMMON /ccon / nnode , nelelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cmat1 / prop(3,maxm)
WRITE (6,99001)
READ (5,*)
READ (5,*) nmat
IF ( nmat.LT.1 .OR. nmat.GT.maxm ) THEN
  PRINT * , 'error in rmat, nmat=' , nmat
  STOP
ENDIF
DO 100 i = 1 , nmat
  READ (5,*) (prop(j,i),j=1,3)
  WRITE (6,99002) i , (prop(j,i),j=1,3)
100 CONTINUE
RETURN
99001 FORMAT (/ ' mat no          k          b          f' /)
99002 FORMAT (I4,5X,3F15.4)
END

SUBROUTINE rbc
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /ccon / nnode , nelelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cbc / nodbc(maxbce) , vbce(maxbce) , nelbc(maxbcn) ,
&          nside(maxbcn) , vbcn(2,maxbcn) , npt(maxpt) ,
&          vpt(maxpt)
c  read point loads
  ir = 0
  READ (5,*)
  READ (5,*) nrec
  npoint = nrec
  IF ( nrec.NE.0 ) THEN
    ir = ir + 1
    DO 50 i = 1 , nrec
      READ (5,*) n , v
      npt(i) = n
      vpt(i) = v
50  CONTINUE
  ENDF
c  read essential boundary condition data
  READ (5,*)
  READ (5,*) nrec
  nbce = 0
  IF ( nrec.NE.0 ) THEN
    ir = ir + 1
    DO 100 j = 1 , nrec
      READ (5,*) n1 , num , inc , v

```

### Chapter 3. Comparison of finite element method and finite difference method

96

```

        DO 60 i = 1 , num
            nbce = nbce + 1
            n = n1 + (i-1)*inc
            nodbc(nbce) = n
            vbce(nbce) = v
60      CONTINUE
100     CONTINUE
        ENDIF
c      read natural boundary condition data
        READ (5,*)
        READ (5,*) nrec
        nbcn = 0
        IF ( nrec.NE.0 ) THEN
            ir = ir + 1
            DO 150 j = 1 , nrec
                READ (5,*) n1 , num , inc , ns , p , v
                DO 120 i = 1 , num
                    nbcn = nbcn + 1
                    n = n1 + (i-1)*inc
                    nelbc(nbcn) = n
                    nside(nbcn) = ns
                    vbcn(1,nbcn) = p
                    vbcn(2,nbcn) = v
120             CONTINUE
150             CONTINUE
            ENDIF
c      print boundary condition data
        IF ( ir.EQ.0 ) STOP 'no boundary value data'
        IF ( npoint.NE.0 ) THEN
            WRITE (6,99001)
            WRITE (6,99002) (npt(i),vpt(i),i=1,npoint)
        ENDIF
        IF ( nbce.NE.0 ) THEN
            WRITE (6,99003)
            WRITE (6,99004) (nodbc(i),vbce(i),i=1,nbce)
        ENDIF
        IF ( nbcn.NE.0 ) THEN
            WRITE (6,99005)
            WRITE (6,99006) (nelbc(i),nside(i),vbcn(1,i),vbcn(2,i),i=1,
&                          nbcn)
        ENDIF
        RETURN
99001 FORMAT (/ ' node no           point load'/)
99002 FORMAT (I5,10X,F15.4)
99003 FORMAT (/ ' node no           essent. value'/)
99004 FORMAT (I5,10X,F15.4)
99005 FORMAT (/ ' element no side      p           gamma'/)
99006 FORMAT (I7,7X,I1,5X,2F10.4)
        END

```

```

SUBROUTINE proc
CALL formkf
CALL aplybc
CALL solve
RETURN
END

SUBROUTINE formkf
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /ccon / nnode , nelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
COMMON /celem / ne(maxe) , mat(maxe) , nodes(9,maxe)
COMMON /cmatrix/ gk(maxn,maxband) , gf(maxn)
COMMON /cint2 / xiq(2,9) , xit(2,7) , wq(9) , wt(7)
DIMENSION ek(9,9) , ef(9) , xx(2,9) , nod(9)
PRINT * , 'forming elements...'
DO 100 i = 1 , maxn
    gf(i) = 0.
    DO 50 j = 1 , maxband
        gk(i,j) = 0.
50    CONTINUE
100  CONTINUE
DO 200 i = 1 , nelem
    mati = mat(i)
    nei = ne(i)
    DO 150 k = 1 , nei
        k1 = nodes(k,i)
        nod(k) = k1
        DO 120 j = 1 , 2
            xx(j,k) = x(j,k1)
120    CONTINUE
150  CONTINUE
    PRINT * , 'element : ' , i
c    nine point integration rule for 9-node quadrilaterals
    IF ( nei.EQ.9 ) CALL elem(xx,nei,ek,ef,9,xiq,wq,mati)
c    seven point integration rule for 6-node triangulars
    IF ( nei.EQ.6 ) CALL elem(xx,nei,ek,ef,7,xit,wt,mati)
    CALL assmb(ek,ef,nei,nod)
200  CONTINUE
WRITE (6,*)
RETURN
END

SUBROUTINE elem(x,n,ek,ef,nl,xi,w,mat)
DIMENSION x(2,n) , ek(n,n) , ef(n) , xi(2,nl) , w(nl)
DIMENSION dpsix(9) , dpsiy(9) , dxds(2,2) , dsdx(2,2)

```

```

        DIMENSION psi(9) , dps(9,2)
c      initilaze element arrays
        DO 100 i = 1 , n
            ef(i) = 0.
            DO 50 j = 1 , n
                ek(i,j) = 0.
50      CONTINUE
100     CONTINUE
        CALL getmat(xk,xb,xf,mat)
c      begin integration point loop
        DO 300 l = 1 , nl
            CALL shape2(xi(1,l),xi(2,l),n,psi,dpsi)
c      calculate dxds
            DO 150 i = 1 , 2
                DO 120 j = 1 , 2
                    dxds(i,j) = 0.
                    DO 110 k = 1 , n
                        dxds(i,j) = dxds(i,j) + dpsi(k,j)*x(i,k)
110      CONTINUE
120      CONTINUE
150      CONTINUE
c      calculate dsdx
            detj = dxds(1,1)*dxds(2,2) - dxds(1,2)*dxds(2,1)
            IF ( detj.LE.0. ) GOTO 500
            dsdx(1,1) = dxds(2,2)/detj
            dsdx(2,2) = dxds(1,1)/detj
            dsdx(1,2) = -dxds(1,2)/detj
            dsdx(2,1) = -dxds(2,1)/detj
c      calculate d(psi)/dx
            DO 200 i = 1 , n
                dpsix(i) = dpsi(i,1)*dsdx(1,1) + dpsi(i,2)*dsdx(2,1)
                dpsiy(i) = dpsi(i,1)*dsdx(1,2) + dpsi(i,2)*dsdx(2,2)
200      CONTINUE
c      accumulate integration point value of integrals
            fac = detj*w(l)
            DO 250 i = 1 , n
                ef(i) = ef(i) + xf*psi(i)*fac
                DO 220 j = i , n
                    ek(i,j) = ek(i,j)
                    &          + fac*(xk*(dpsix(i)*dpsix(j)+dpsiy(i)*dpsiy(j)
                    &          )+xb*psi(i)*psi(j))
220      CONTINUE
250      CONTINUE
300     CONTINUE
c      calculate lower symmetric part of ek
        DO 400 i = 1 , n
            DO 350 j = 1 , i
                ek(i,j) = ek(j,i)
350      CONTINUE

```

```

400 CONTINUE
c   print '(9f8.4)',((ek(i,j),j=1,n),i=1,n)
      RETURN
500 WRITE (6,99001) detj , x
      STOP
99001 FORMAT (' bad jacobian ',E10.3/9E10.3/9E10.3)
      END

```

```

SUBROUTINE shape2(ski,eta,n,psi,dpsi)
DIMENSION psi(9) , dpsi(9,2)
IF ( n.EQ.6 ) THEN
  zet = 1. - ski - eta
  psi(1) = 2.*zet*(zet-0.5)
  psi(2) = 2.*ski*(ski-0.5)
  psi(3) = 2.*eta*(eta-0.5)
  psi(4) = 4.*zet*ski
  psi(5) = 4.*ski*eta
  psi(6) = 4.*eta*zet
  dpsi(1,1) = 1. - 4.*zet
  dpsi(1,2) = 1. - 4.*zet
  dpsi(2,1) = 4.*ski - 1.
  dpsi(2,2) = 0.
  dpsi(3,1) = 0.
  dpsi(3,2) = 4.*eta - 1.
  dpsi(4,1) = 4.*(zet-ski)
  dpsi(4,2) = -4.*ski
  dpsi(5,1) = 4.*eta
  dpsi(5,2) = 4.*ski
  dpsi(6,1) = -4.*eta
  dpsi(6,2) = 4.*(zet-eta)
  RETURN
ENDIF
ski2 = ski*ski
eta2 = eta*eta
ski2m = ski2 - ski
eta2m = eta2 - eta
ski2p = ski2 + ski
eta2p = eta2 + eta
psi(1) = ski2m*eta2m/4.
psi(2) = ski2p*eta2m/4.
psi(3) = ski2p*eta2p/4.
psi(4) = ski2m*eta2p/4.
psi(5) = (1.-ski2)*eta2m/2.
psi(6) = ski2p*(1.-eta2)/2.
psi(7) = (1.-ski2)*eta2p/2.
psi(8) = ski2m*(1.-eta2)/2.
psi(9) = (1.-ski2)*(1.-eta2)
dpsi(1,1) = (2.*ski-1.)*eta2m/4.
dpsi(1,2) = ski2m*(2.*eta-1.)/4.

```

```

dpsi(2,1) = (2.*ski+1.)*eta2m/4.
dpsi(2,2) = ski2p*(2.*eta-1.)/4.
dpsi(3,1) = (2.*ski+1.)*eta2p/4.
dpsi(3,2) = ski2p*(2.*eta+1.)/4.
dpsi(4,1) = (2.*ski-1.)*eta2p/4.
dpsi(4,2) = ski2m*(2.*eta+1.)/4.
dpsi(5,1) = -ski*eta2m
dpsi(5,2) = (1.-ski2)*(2.*eta-1.)/2.
dpsi(6,1) = (2.*ski+1.)*(1.-eta2)/2.
dpsi(6,2) = -ski2p*eta
dpsi(7,1) = -ski*eta2p
dpsi(7,2) = (1-ski2)*(2.*eta+1.)/2.
dpsi(8,1) = (2.*ski-1.)*(1.-eta2)/2.
dpsi(8,2) = -ski2m*eta
dpsi(9,1) = -2.*ski*(1.-eta2)
dpsi(9,2) = -2.*(1.-ski2)*eta
RETURN
END

SUBROUTINE getmat(xk,xb,xf,mat)
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /cmatl / prop(3,maxm)
xk = prop(1,mat)
xb = prop(2,mat)
xf = prop(3,mat)
RETURN
END

SUBROUTINE assmb(ek,ef,n,node)
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /cmatrix/ gk(maxn,maxband) , gf(maxn)
DIMENSION ek(n,n) , ef(n) , node(n)
DO 100 i = 1 , n
  ig = node(i)
c   assemble global vector gf
  gf(ig) = gf(ig) + ef(i)
  DO 50 j = 1 , n
c   assemble global stiffnes matrix gk
    jg = node(j) - ig + 1
    IF ( jg.GT.0 ) gk(ig,jg) = gk(ig,jg) + ek(i,j)
50  CONTINUE
100 CONTINUE
RETURN
END

SUBROUTINE aplybc
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,

```

```

&          maxband=25)
COMMON /ccon / nnode , nele , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
COMMON /celem / ne(maxe) , mat(maxe) , nodes(9,maxe)
COMMON /cbc / nodbc(maxbce) , vbce(maxbce) , nelbc(maxbcn) ,
&          nside(maxbcn) , vbcn(2,maxbcn) , npt(maxpt) ,
&          vpt(maxpt)
COMMON /cmatrix/ gk(maxn,maxband) , gf(maxn)
DIMENSION nod(3) , xbc(2,3) , pe(3,3) , game(3)
PRINT * , 'applying boundary conditions ...'
c  apply point loads
  IF ( npoint.NE.0 ) THEN
    DO 50 i = 1 , npoint
      n = npt(i)
      gf(n) = gf(n) + vpt(i)
50    CONTINUE
  ENDIF
c  apply essential boundary conditions
  IF ( nbce.NE.0 ) THEN
    big = 1.E30
    DO 100 i = 1 , nbce
      n = nodbc(i)
      gk(n,1) = big
      gf(n) = big*vbce(i)
100  CONTINUE
  ENDIF
c  apply natural boundary condition
  IF ( nbcn.NE.0 ) THEN
    DO 150 i = 1 , nbcn
c    pick out nodes on side of element
      nel = nelbc(i)
      ns = nside(i)
      nc = 4
      IF ( ne(nel).EQ.6 ) nc = 3
      nod(1) = ns
      nod(2) = ns + nc
      nod(3) = ns + 1
      IF ( ns.EQ.nc ) nod(3) = 1
c    pick out nodal coordinates
      DO 120 j = 1 , 3
        nj = nod(j)
        nod(j) = nodes(nj,nel)
        nj = nod(j)
        xbc(1,j) = x(1,nj)
        xbc(2,j) = x(2,nj)
120    CONTINUE
c    call bcint to calculate boundary integrals pe and game
      CALL bcint(vbcn(1,i),vbcn(2,i),xbc,pe,game)

```

```

c      call assmb to add pe to gk and game to gf
      CALL assmb(pe,game,3,nod)
150   CONTINUE
      ENDIF
      RETURN
      END

      SUBROUTINE bcint(v1,v2,x,pe,game)
      COMMON /cint1 / etaq(3) , ww(3)
      DIMENSION theta(3) , dtheta(3)
      DIMENSION x(2,3) , pe(3,3) , game(3)
      DO 100 i = 1 , 3
         game(i) = 0.
         DO 50 j = 1 , 3
            pe(i,j) = 0.
50      CONTINUE
100   CONTINUE
      DO 200 k = 1 , 3
         CALL shape1(etaq(k),theta,dtheta)
         x1 = x(1,1)*dtheta(1) + x(1,2)*dtheta(2) + x(1,3)*dtheta(3)
         y1 = x(2,1)*dtheta(1) + x(2,2)*dtheta(2) + x(2,3)*dtheta(3)
         sjac = sqrt(x1*x1+y1*y1)
         fac = ww(k)*sjac
         DO 150 i = 1 , 3
            game(i) = game(i) + v2*theta(i)*fac
            DO 120 j = 1 , 3
               pe(i,j) = pe(i,j) + v1*theta(i)*theta(j)*fac
120      CONTINUE
150   CONTINUE
200   CONTINUE
      RETURN
      END

      SUBROUTINE shape1(eta,theta,dtheta)
      DIMENSION theta(3) , dtheta(3)
      eta2 = eta*eta
      theta(1) = (eta2-eta)/2.
      theta(2) = 1. - eta2
      theta(3) = (eta2+eta)/2.
      dtheta(1) = eta - 0.5
      dtheta(2) = -2.*eta
      dtheta(3) = eta + 0.5
      RETURN
      END

      SUBROUTINE solve
      PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&              maxband=25)
      COMMON /ccon / nnode , nelem , nmat , nbce , nbcn , npoint ,

```

```

&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
COMMON /cmatrix/ gk(maxn,maxband) , gf(maxn)
PRINT * , 'solving equations ...'
CALL trib(gk,nnode,nband,maxn,maxband)
CALL rhsb(gk,u,gf,nnode,nband,maxn,maxband)
RETURN
END

SUBROUTINE trib(a,n,ib,l1,l2)
DIMENSION a(l1,l2)
c this subroutine triangularizes a banded and symmetric matrix a
c only the upper half-band of matrix is stored
c storage is in the form of a rectangular array l1 x l2
c the half-band width is ib
c the number of equations is n
DO 100 i = 2 , n
    m1 = min0(ib-1,n-i+1)
    DO 50 j = 1 , m1
        sum = 0.0
        k1 = min0(i-1,ib-j)
        DO 20 k = 1 , k1
            sum = sum + a(i-k,k+1)*a(i-k,j+k)/a(i-k,1)
20        CONTINUE
        a(i,j) = a(i,j) - sum
50    CONTINUE
100 CONTINUE
RETURN
END

SUBROUTINE rhsb(a,x,b,n,ib,l1,l2)
DIMENSION a(l1,l2) , x(1) , b(1)
c for the linear sytem a*x=b with the matrix a
c triangularized by routine trib
c this routine performs the forward substitution
c into b and back substitution into x
c the half-band width of a is ib
c the number of equations is n
npl = n + 1
DO 100 i = 2 , n
    sum = 0.0
    k1 = min0(ib-1,i-1)
    DO 50 k = 1 , k1
        sum = sum + a(i-k,k+1)/a(i-k,1)*b(i-k)
50    CONTINUE
    b(i) = b(i) - sum
100 CONTINUE
c begin back-substitution
x(n) = b(n)/a(n,1)

```

```

DO 200 k = 2 , n
  i = np1 - k
  j1 = i + 1
  j2 = min0(n,i+ib-1)
  sum = 0.0
  DO 150 j = j1 , j2
    mm = j - j1 + 2
    sum = sum + x(j)*a(i,mm)
150  CONTINUE
  x(i) = (b(i)-sum)/a(i,1)
200  CONTINUE
  RETURN
  END

SUBROUTINE post
PARAMETER (maxn=400,maxe=100,maxm=5,maxbce=70,maxbcn=40,maxpt=10,
&          maxband=25)
COMMON /ccon / nnode , nelelem , nmat , nbce , nbcn , npoint ,
&          nband
COMMON /cnode / x(2,maxn) , u(maxn)
PRINT * , 'writing solutions ... '
WRITE (6,*)
WRITE (6,*) '   node           x           y           u '
WRITE (6,*)
&          ' -----'
DO 100 i = 1 , nnode
  WRITE (6,99001) i , x(1,i) , x(2,i) , u(i)
100  CONTINUE
99001 FORMAT (2X,I4,1X,3F15.3)
END

```



## Bibliography

- [1]
- [2]
- [3]
- [4]
- [5]



# FVM

<b>4</b>	<b>Finite volume method . . . . .</b>	<b>109</b>
4.1	Introduction	
4.2	Solution of Euler equations	
4.3	Solution of Navier–Stokes equations	

	<b>Bibliography . . . . .</b>	<b>155</b>
--	-------------------------------	------------

	<b>III</b>	<b>Numerical method -</b>
<b>159</b>		<b>SPH</b>



## 4. Finite volume method

### 4.1 Introduction

The finite volume method was introduced as clearly recognizable procedure of numerical fluid mechanics at the beginning of the eighth decade of twentieth century, independently by McDonald [26] and MacCormack and Paullay [25], who applied the mentioned method in solving non-stationary two-dimensional flow of non-viscous fluid. Applying it to the calculation of a three-dimensional flow, the method was later generalized by Rizzi [39], when finally the path to more universal application was open. The method is based on the integral formulation of the basic laws of fluid mechanics and discrete solution in physical space.

The method can be applied to all types of computational grids, with great flexibility in terms of the efficiency and accuracy of flux calculations through the appropriate control surfaces. Direct discretization of integral equation forms in fluid mechanics allows a very simple formulation of discrete changes of basic quantities such as *mass*, *momentum* and *energy*, which describe fluid flow.

The integral form of continuity equation, momentum and energy equation for arbitrary control volume  $\Omega$ , bounded by surface  $\partial\Omega$ , reads:

$$\frac{\partial}{\partial t} \iiint_{\Omega} W \, d\Omega + \iint_{\partial\Omega} \mathbf{F} \cdot d\mathbf{S} = \iiint_{\Omega} Q \, d\Omega. \quad (\text{I})$$

By applying equation (I) to the control volume  $\Omega_J$ , in which required quantity is  $W_J$ , and  $Q_J$  is volume source, its discrete form is obtained:

$$\frac{\partial}{\partial t} (W_J \Omega_J) + \sum_I (\mathbf{F}_I \cdot \mathbf{S}_I) = Q_J \Omega_J, \quad (\text{II})$$

whereby the summation of fluxes is performed on all surfaces that surround the control volume  $\Omega_J$ . The system of discrete equations (II) is *conservative* if the sum of all fluxes through common surfaces between adjacent cells is equal to zero. When this condition is not met, system (II) is *nonconservative* and internal fluxes play the role of *internal numerical volume sources*.

After the geometric discretization of the flow domain, i.e. construction of the computational grid, conservative form of equation (II) requires the fulfillment of the following conditions [14], [7]:

- control volumes  $\Omega_J$  must occupy the entire flow domain  $\Omega$ ,
- adjacent cells  $\Omega_J$  can overlap if internal computational surface  $\Gamma_J$  is common to the mentioned volumes and
- fluxes through the surfaces of the computational cell must be calculated in the manner *independent* from the observed computational cell  $\Omega_J$ .

Although it can be considered as one form of the finite difference method, its importance and the possibility of wide application in the fluid flow justifies the name of the finite volume method and its separate treatment. Properties that clearly distinguish the finite volume method approach from procedures related to finite difference method and finite element method can be systematized through the properties:

- the coordinate of the point  $J$ , which determines the variable  $W$  in control volume  $\Omega_J$ , does not appear explicit in the formulation of the problem. That actually means that the variable  $W_J$  is not adjoined to particular point inside the control volume and can be viewed as *mean value* of the variable  $W_J$  in the computational cell,
- grid point coordinates figure only in determining the volume of computational cells and the area values of the surfaces of their sides,
- in the absence of a volume source member in the appropriate conservative equations, formulation method defines the change of the mean value of the variable  $W$  in time interval  $\Delta t$ , which is equal to the sum of the fluxes through the sides of the adjacent computational cell and
- method allows the natural introduction of boundary conditions.

## 4.2 Solution of Euler equations

### 4.2.1 Explicit numerical scheme

Previous research in the transonic fluid flow computations has confirmed the justification for the application of potential theory in numerical flow field analysis only for moderate Mach numbers of undisturbed flow. Increasing velocity of undisturbed flow in front of the immersed body increases the intensity of the shock waves, in the general case curved, so the abrupt changes of flow variable magnitudes are accompanied by a significant change in entropy, hence the assumption about isentropic, potential flow becomes practically unsustainable. In this chapter a numerical approach, based on the finite volume method, originally introduced in potential theory, and later to the exact methods in transonic flow field calculation [19], [27] and [18], will be presented.

#### Central difference scheme

In the integral form Euler equations can be written on the following:

$$\frac{\partial}{\partial t} \iiint_{\Omega} W \, d\Omega + \iint_{\partial\Omega} \mathbf{F} \cdot d\mathbf{S} = 0, \quad (4.1)$$

where  $\Omega$  is the control volume of the fluid with the surface  $\partial\Omega$ . In the equation (4.1)  $W$  represents observed flow variable, while  $\mathbf{F} \cdot d\mathbf{S}$  denotes the corresponding elementary flux through the boundary

surface of the control volume  $\Omega$ . The following applies to the continuity equation:

$$W = \rho \quad \text{and} \quad \mathbf{F} = (\rho u, \rho v, \rho w)^T. \quad (4.2)$$

In the momentum equation  $W$  and  $\mathbf{F}$  become:

$$W = \begin{cases} \rho u \\ \rho v \\ \rho w \end{cases} \quad \text{and} \quad \mathbf{F} = \begin{cases} (\rho u^2 + p, \rho uv, \rho uw)^T, \\ (\rho uv, \rho v^2 + p, \rho vw)^T, \\ (\rho uw, \rho vw, \rho w^2 + p)^T, \end{cases} \quad (4.3)$$

while in the energy equation

$$W = \rho e_t \quad \text{and} \quad \mathbf{F} = (\rho H u, \rho H v, \rho H w)^T. \quad (4.4)$$

In the expressions (4.2)-(4.4) variables  $p, \rho, u, v, w, e_t$  and  $H$  represent pressure, density, velocity vector projections on three Cartesian coordinate axes, energy and enthalpy, respectively. For the perfect gas system of equations (4.2)-(4.4) is extended by defining the energy of the fluid

$$e_t = \frac{1}{\gamma - 1} \frac{p}{\rho} + \frac{1}{2} (u^2 + v^2 + w^2), \quad (4.5)$$

having in mind the relation between enthalpy  $H$  and energy  $e_t$

$$H = e_t + \frac{p}{\rho}, \quad (4.6)$$

whereby the system of Euler equations is finally closed.

In the numerical approach in solving Euler equations by the finite volume method the computational domain is discretized to the appropriate number of hexahedron-shaped cells, and for each mentioned cell, a system of equations that approximates system (4.2)-(4.6) is set up. If inside each cell  $(i, j, k)$  value of the dependent variable  $W$  is known, an approximate form of system of equations (4.1) is obtained:

$$\frac{\partial}{\partial t} (h_{i,j,k} W_{i,j,k}) + Q(W)_{i,j,k} = 0, \quad (4.7)$$

where  $Q(W)_{i,j,k}$  represents the flux of flow quantities through the boundaries of observed cell. Cell volume, denoted by  $h_{i,j,k}$  in the expression (4.7), can be considered constant in time.

Flux of flow quantities through the boundaries of the observed cell is calculated as follows:

$$Q(W)_{i,j,k} = \mathbf{F}(W) \cdot \mathbf{S}|_{i+1/2,j,k} - \mathbf{F}(W) \cdot \mathbf{S}|_{i-1/2,j,k} + \mathbf{F}(W) \cdot \mathbf{S}|_{i,j+1/2,k} - \mathbf{F}(W) \cdot \mathbf{S}|_{i,j-1/2,k} + \mathbf{F}(W) \cdot \mathbf{S}|_{i,j,k+1/2} - \mathbf{F}(W) \cdot \mathbf{S}|_{i,j,k-1/2}. \quad (4.8)$$

The quantity  $\mathbf{S}|_{i+1/2,j,k}$  in the relation (4.8) represents a surface vector of the computational cell between the grid points  $(i, j, k)$  and  $(i + 1, j, k)$ , at whereby the flux through the observed cell surface is calculated by averaging the quantity  $\mathbf{F}(W)$ , calculated in points  $(i, j, k)$  and  $(i + 1, j, k)$ . Fluxes through the remaining five surfaces of observed hexahedron are determined by an analogous procedure.

The use of a central difference scheme requires introduction of additional terms in the approximate equation (4.7) in order to obtain stationary solutions of dynamic equations [45]. Supplementary dissipative terms, known in the literature as *artificial viscosity* terms, have a role to prevent even-odd decoupling of the numerical solution in equations (4.7), on the one hand, and the occurrence of high-frequency oscillations of the solution in the shock wave zone, on the other hand. Presence of dissipative terms also provides the uniqueness of the numerical solution, which is shown in the

literature [24]. Dissipative terms are usually represented by a combination of the second and fourth order terms, ie.

$$D(W)_{i,j,k} = (D_x^2 + D_y^2 + D_z^2 - D_x^4 - D_y^4 - D_z^4)W_{i,j,k}. \quad (4.9)$$

In the expression (4.9) the term  $D_x^2 W_{i,j,k}$  is determined by the relation

$$D_x^2 W_{i,j,k} = d_{i+1/2,j,k}^{(2)} - d_{i-1/2,j,k}^{(2)}, \quad (4.9.1)$$

where is

$$d_{i+1/2,j,k}^{(2)} = \varepsilon_{i+1/2,j,k}^{(2)} \frac{h_{i+1/2,j,k}}{\Delta t} \Delta_x W_{i,j,k}, \quad (4.9.2)$$

while the term  $D_x^4 W_{i,j,k}$  is calculated on the basis of the expression

$$D_x^4 W_{i,j,k} = d_{i+1/2,j,k}^{(4)} - d_{i-1/2,j,k}^{(4)}, \quad (4.9.3)$$

in which is

$$d_{i+1/2,j,k}^{(4)} = \varepsilon_{i+1/2,j,k}^{(4)} \frac{h_{i+1/2,j,k}}{\Delta t} \Delta_x^3 W_{i,j,k}. \quad (4.9.4)$$

In the relations (4.9.2) and (4.9.4) terms  $\Delta_x W_{i,j,k}$  and  $\Delta_x^3 W_{i,j,k}$  represent the difference operators “forward” oriented

$$\Delta_x W_{i,j,k} = W_{i+1,j,k} - W_{i,j,k} \quad (4.9.5)$$

and

$$\Delta_x^3 W_{i,j,k} = W_{i+2,j,k} - 3W_{i+1,j,k} + 3W_{i,j,k} - W_{i-1,j,k}. \quad (4.9.6)$$

The values of the remaining dissipative terms will be determined in a similar way in the expression (4.9). Quantities  $\varepsilon_{i+1/2,j,k}^{(2)}$  and  $\varepsilon_{i+1/2,j,k}^{(4)}$ , present in relations (4.9.2) and (4.9.4), are calculated as follows:

$$\varepsilon_{i+1/2,j,k}^{(2)} = k^{(2)} \max(v_{i+1,j,k}, v_{i,j,k}), \quad (4.9.7)$$

ie.

$$\varepsilon_{i+1/2,j,k}^{(4)} = \max(0, k^{(4)} - \varepsilon_{i+1/2,j,k}^{(2)}), \quad (4.9.8)$$

where the quantity  $v_{i,j,k}$  is defined by the expression

$$v_{i,j,k} = \frac{|p_{i+1,j,k} - 2p_{i,j,k} + p_{i-1,j,k}|}{|p_{i+1,j,k} + 2p_{i,j,k} + p_{i-1,j,k}|}. \quad (4.9.9)$$

In the relations (4.9.7) and (4.9.8) constants  $k^{(2)} = \mathcal{O}(1)$  and  $k^{(4)}$  determine the amount of artificial viscosity added and allow the transition from a fourth-order difference scheme to the scheme of the second order accuracy in the zone of abrupt change of pressure, ie. in the immediate vicinity of the shock wave, which can be deduced from expression (4.9.8). Their size also determines stability and accuracy of numerical schemes [46]. Higher value of constants  $k^{(2)}$  and  $k^{(4)}$  increases the stability of the numerical scheme making the solution smooth, reducing at the same time its accuracy in the zone of abrupt changes of flow variables.

Special attention has to be paid to the analysis of dissipative terms behavior at the boundary of the physical domain. Taking into account relations (4.9.5) and (4.9.6), introduction of second order terms requires information of the value of flow variables in neighboring grid cells, while in the case of fourth-order terms influence domain extends to two adjacent cells in each coordinate direction. For cells at the boundary of the physical domain defining of the artificial viscosity terms is achieved by combining physical and numerical boundary conditions. In this way the lack of necessary information is compensated, and at the same time the stability of the scheme at the boundaries is ensured.

After the introduction of dissipative terms the system of equations (4.7) receives the final form

$$\frac{\partial}{\partial t} (h_{i,j,k} W_{i,j,k}) + Q(W)_{i,j,k} - D(W)_{i,j,k} = 0. \quad (4.10)$$

Since the volume of the hexahedral cell is invariant in time, application of Runge-Kutta method for solving system of ordinary differential equations leads to relations

$$\begin{aligned} W_{i,j,k}^{(0)} &= W_{i,j,k}^n, \\ W_{i,j,k}^{(r)} &= W_{i,j,k}^{(0)} - \alpha_r h_{i,j,k}^{-1} \Delta t [Q(W)_{i,j,k}^{(r-1)} - D(W)_{i,j,k}^{(r-1)}], \quad r = 1, 2, \dots, r_{max} \\ W_{i,j,k}^{n+1} &= W_{i,j,k}^{(r_{max})}, \end{aligned} \quad (4.11)$$

where  $r_{max}$  determines the order of the adopted Runge-Kutta scheme, while  $\alpha_r$  are the corresponding coefficients of the scheme. In equations (4.11) the indices  $n$  and  $n+1$  indicate the values of the variable  $W_{i,j,k}$  at the beginning and the end of an iterative cycle. To achieve significant savings in computer time, artificial viscosity terms are most often "frozen" on the first, ie. the second step within an iterative cycle, bearing in mind that the stability of such a modified equation is almost unchanged. If the calculation of dissipative terms is performed only within the first step of each iterative cycle, the system of equations (4.11) receives form

$$\begin{aligned} W_{i,j,k}^{(0)} &= W_{i,j,k}^n, \\ W_{i,j,k}^{(r)} &= W_{i,j,k}^{(0)} - \alpha_r h_{i,j,k}^{-1} \Delta t [Q(W)_{i,j,k}^{(r-1)} - D(W)_{i,j,k}^{(0)}], \quad r = 1, 2, \dots, r_{max} \\ W_{i,j,k}^{n+1} &= W_{i,j,k}^{(r_{max})}. \end{aligned} \quad (4.11.1)$$

When the values inside the brackets on the right hand of equation (4.11) and equation (4.11.1), become zero, a steady state is reached and the iterative process is completed. The steady state does not depend on the value of the adopted integration step  $\Delta t$ . In this case, the value of the variable  $W_{i,j,k}^{(r_{max})}$  at the end of iterative cycle is equal to the value of  $W_{i,j,k}^{(0)}$  at the beginning of the iteration, ie  $W_{i,j,k}^{n+1} = W_{i,j,k}^n$ .

A very important detail in this procedure is determination of the integration step  $\Delta t$ , taking into account that it is the largest value determined by the time interval required to perturbation transfers from one cell boundary to another. Integration step  $\Delta t$  for the cell  $(i, j, k)$  is determined by the relation

$$\Delta t_{i,j,k} = \left[ \frac{1}{(\Delta t_{\xi})_{i,j,k}} + \frac{1}{(\Delta t_{\eta})_{i,j,k}} + \frac{1}{(\Delta t_{\zeta})_{i,j,k}} \right]^{-1}, \quad (4.12)$$

in which  $(\Delta t_{\xi})_{i,j,k}$ ,  $(\Delta t_{\eta})_{i,j,k}$  and  $(\Delta t_{\zeta})_{i,j,k}$  are time intervals of disturbance propagation in the directions of the corresponding coordinate axes within the boundary of the observed grid cell. By mapping to a curvilinear coordinate system  $(\xi, \eta, \zeta)$ , in which constant values of curvilinear coordinates

determine the grid surfaces in physical space, time intervals  $(\Delta t_\xi)_{i,j,k}$ ,  $(\Delta t_\eta)_{i,j,k}$  and  $(\Delta t_\zeta)_{i,j,k}$  can be calculated:

$$\begin{aligned} (\Delta t_\xi)_{i,j,k} &= \left( \frac{1}{|U| + c\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}} \right)_{i,j,k}, \\ (\Delta t_\eta)_{i,j,k} &= \left( \frac{1}{|V| + c\sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}} \right)_{i,j,k}, \\ (\Delta t_\zeta)_{i,j,k} &= \left( \frac{1}{|W| + c\sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2}} \right)_{i,j,k}. \end{aligned} \quad (4.12.1)$$

In the expressions (4.12.1)  $U$ ,  $V$  and  $W$  denote contravariant velocity vector coordinates,<sup>1</sup> while  $c$  represents the local speed of sound.

Integration step  $\Delta t$ , determined by the expression (4.12), can be scaled by a constant, known in literature as the Courant number [**aga**], which in the case of the fourth order Runge-Kutta method equals to  $2\sqrt{2}$ . Taking into account that grid cell dimensions can vary drastically, depending on the type of chosen grid, application of the variable integration steps leads to a significant acceleration of solution convergence. As the values of the flow variables do not change significantly within of an iterative cycle, that the time interval  $\Delta t$  calculation does not need to be performed after each iteration, resulting in significant savings in computational (CPU) time.

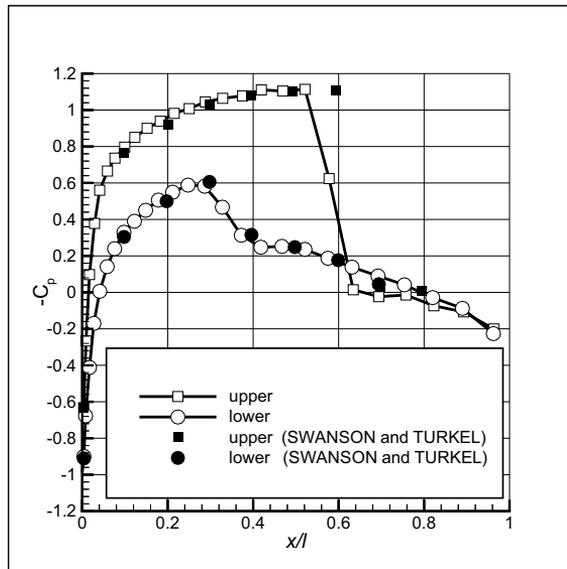


Figure 4.1: Pressure distribution on a rectangular wing – plane of symmetry.

<sup>1</sup>Numeric determination of partial derivatives  $\xi_x$ ,  $\xi_y$ ,  $\xi_z$ ,  $\eta_x$ ,  $\eta_y$ ,  $\eta_z$ ,  $\zeta_x$ ,  $\zeta_y$  and  $\zeta_z$  are explained in detail in literature [**hirsch**] (Hirsch, C. *Numerical Computation of Internal and External Flows*, Vol. 1, pp. 253–260).

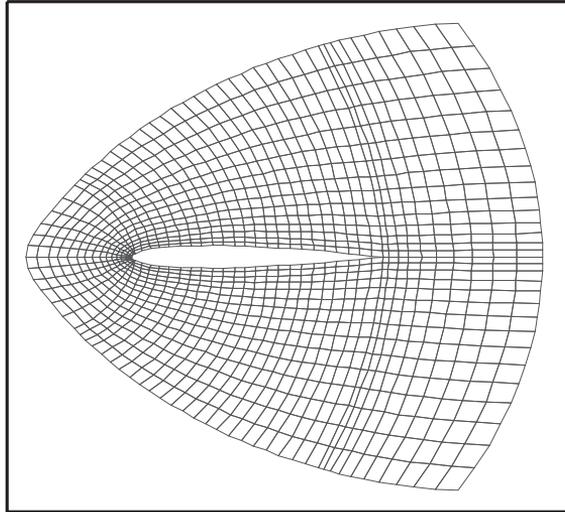


Figure 4.2: The algebraic “C-H” computational grid.

Pressure distribution on a rectangular wing with an airfoil NACA 0012, constant along the wing span, is shown for cross section in the plane of symmetry of the wing at Figure 4.1. In numerical calculation an algebraic non-orthogonal “C-H” grid was used in three-dimensional space, whose appearance in the wing symmetry plane is illustrated in Figure 4.2. At Mach number of undisturbed flow  $M_\infty = 0.8$  and angle of attack  $\alpha = 1.25^\circ$  the results obtained show very good agreement with the results [45], despite coarse grid discretization ( $65 \times 12 \times 15$ ).

Described procedure in the analysis of non-viscous flow in transonic speed range made possible the calculation of the aerodynamic load in presence of strong shock waves, when the application of potential theory becomes unsustainable. Numerical stability and fast convergence of system of differential equations solution by the Runge-Kutta method make exposed procedure very acceptable in the analysis of three-dimensional flow. The great flexibility of the applied approach is reflected in the fact that it allows very easy introduction of real viscous terms in the Euler equations, thus leading to the Navier-Stokes equations, which represents a significant improvement in the aerodynamic load analysis in the cases of large wing attack angles with pronounced effects of flow separation.

#### 4.2.2 Implicit numerical scheme

Application of an explicit scheme in Euler equations solving, explained in chapter 4.2.1 of this book and literature [18], [27] and [19], enables a very precise analysis of the transonic three-dimensional flows, eliminating irregularities related to results of potential theory. The main disadvantage of the explicit scheme is reflected, however, in limited value of the Courant number (CFL), related to value of integration step. The introduction of an implicit scheme will have the task of increasing the allowable value of the Courant number and thus the calculation of the transonic flows makes more acceptable, even in solving problem related to non-stationary flow. The procedure of two-factor LU implicit factorization, based on finite volume method with the application of “flux decomposition” approach, is described in this chapter. Special attention will be paid to the correct implementation of boundary conditions related to features of “flux decomposition”.

### LU implicit factorization method

In Cartesian coordinate system of three-dimensional non-stationary Euler equations can be written in conservative form, as follows:

$$\partial_t \mathbf{q} + \partial_x \mathbf{F} + \partial_y \mathbf{G} + \partial_z \mathbf{H} = 0, \quad (4.13)$$

where  $\mathbf{q}$  is a variable flow quantities vector, ie.

$$\mathbf{q} = (\rho, \rho u, \rho v, \rho w, e)^T, \quad (4.14)$$

while  $\mathbf{F}$ ,  $\mathbf{G}$  and  $\mathbf{H}$  are flux vector projections on three coordinate axes defined by expressions:

$$\mathbf{F} = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(e+p) \end{Bmatrix}, \quad \mathbf{G} = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(e+p) \end{Bmatrix} \quad \text{and} \quad \mathbf{H} = \begin{Bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(e+p) \end{Bmatrix}. \quad (4.15)$$

In the expressions (4.14) and (4.15) the quantities  $\rho$ ,  $u$ ,  $v$ ,  $w$  and  $p$  represent the density, velocity vector projections on three coordinate axes and pressure, respectively. The variable  $e$  is defined by the expression  $e = \rho e_t$ , where  $e_t$  is the total fluid energy determined by the relation (4.5).

After introducing of curvilinear coordinate system  $(\xi, \eta, \zeta)$ , computational grid in physical space, whose surfaces correspond to constant values of curvilinear coordinates, is mapped into a rectangle grid in computational domain. In the transformed space, the equation (4.13) becomes:

$$\partial_\tau \bar{\mathbf{q}} + \partial_\xi \bar{\mathbf{F}} + \partial_\eta \bar{\mathbf{G}} + \partial_\zeta \bar{\mathbf{H}} = 0, \quad (4.13.1)$$

where the quantities  $\bar{\mathbf{q}}$ ,  $\bar{\mathbf{F}}$ ,  $\bar{\mathbf{G}}$  and  $\bar{\mathbf{H}}$  are defined by the following relations:

$$\bar{\mathbf{q}} = J \mathbf{q} \quad (4.14.1)$$

and

$$\begin{aligned} \bar{\mathbf{F}} &= J(\xi_\tau \mathbf{q} + \xi_x \mathbf{F} + \xi_y \mathbf{G} + \xi_z \mathbf{H}), \\ \bar{\mathbf{G}} &= J(\eta_\tau \mathbf{q} + \eta_x \mathbf{F} + \eta_y \mathbf{G} + \eta_z \mathbf{H}), \\ \bar{\mathbf{H}} &= J(\zeta_\tau \mathbf{q} + \zeta_x \mathbf{F} + \zeta_y \mathbf{G} + \zeta_z \mathbf{H}). \end{aligned} \quad (4.15.1)$$

After arranging the expression (4.15.1) quantities  $\bar{\mathbf{F}}$ ,  $\bar{\mathbf{G}}$  and  $\bar{\mathbf{H}}$  become

$$\begin{aligned} \bar{\mathbf{F}} &= J \begin{Bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ \rho w U + \xi_z p \\ U(e+p) - \xi_t p \end{Bmatrix}, \quad \bar{\mathbf{G}} = J \begin{Bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ \rho w V + \eta_z p \\ V(e+p) - \eta_t p \end{Bmatrix} \quad \text{and} \\ \bar{\mathbf{H}} &= J \begin{Bmatrix} \rho W \\ \rho u W + \zeta_x p \\ \rho v W + \zeta_y p \\ \rho w W + \zeta_z p \\ W(e+p) - \zeta_t p \end{Bmatrix}, \end{aligned} \quad (4.15.2)$$

where  $U$ ,  $V$  and  $W$  denote the contravariant coordinates of the velocity vector, defined by transformations:

$$\begin{aligned} U &= \xi_t + \xi_x u + \xi_y v + \xi_z w, \\ V &= \eta_t + \eta_x u + \eta_y v + \eta_z w, \\ W &= \zeta_t + \zeta_x u + \zeta_y v + \zeta_z w, \end{aligned} \quad (4.15.3)$$

while  $\xi_t$ ,  $\eta_t$  and  $\zeta_t$  are determined in the following way:

$$\begin{aligned}\xi_t &= -x_\tau \xi_x - y_\tau \xi_y - z_\tau \xi_z, \\ \eta_t &= -x_\tau \eta_x - y_\tau \eta_y - z_\tau \eta_z, \\ \zeta_t &= -x_\tau \zeta_x - y_\tau \zeta_y - z_\tau \zeta_z,\end{aligned}\tag{4.15.4}$$

bearing in mind the well-known law of coordinate transformation

$$\xi = \xi(x, y, z, t), \quad \eta = \eta(x, y, z, t), \quad \zeta = \zeta(x, y, z, t) \quad \text{and} \quad \tau = t.\tag{4.15.5}$$

Time derivatives  $\xi_t$ ,  $\eta_t$  and  $\zeta_t$  in the expressions (4.15.4) are equal to zero for the case of a stationary, fixed grid.

*Transformation Jacobian*  $J = \partial(x, y, z) / \partial(\xi, \eta, \zeta)$  in relations (4.14.1), (4.15.1) and (4.15.2) is calculated from the expression

$$J = x_\xi (y_\eta z_\zeta - z_\eta y_\zeta) - y_\xi (x_\eta z_\zeta - z_\eta x_\zeta) + z_\xi (x_\eta y_\zeta - y_\eta x_\zeta).\tag{4.15.6}$$

In the system of equations (4.13), ie. (4.13.1) unknown quantity to be determined is a vector of flow variables  $\bar{\mathbf{q}}$ . Because the quantities  $\bar{\mathbf{F}}$ ,  $\bar{\mathbf{G}}$  and  $\bar{\mathbf{H}}$  are nonlinear functions of variable  $\bar{\mathbf{q}}$ , defining their values at time  $n + 1$  is done by local linearization with respect to the previous moment determined by time index  $n$

$$\begin{aligned}\bar{\mathbf{F}}^{n+1} &= \bar{\mathbf{F}}^n + \left[ \frac{D\bar{\mathbf{F}}}{D\bar{\mathbf{q}}} \right]^n \Delta \bar{\mathbf{q}}^n, \\ \bar{\mathbf{G}}^{n+1} &= \bar{\mathbf{G}}^n + \left[ \frac{D\bar{\mathbf{G}}}{D\bar{\mathbf{q}}} \right]^n \Delta \bar{\mathbf{q}}^n, \\ \bar{\mathbf{H}}^{n+1} &= \bar{\mathbf{H}}^n + \left[ \frac{D\bar{\mathbf{H}}}{D\bar{\mathbf{q}}} \right]^n \Delta \bar{\mathbf{q}}^n,\end{aligned}\tag{4.16}$$

where is

$$\Delta \bar{\mathbf{q}}^n = \bar{\mathbf{q}}^{n+1} - \bar{\mathbf{q}}^n.\tag{4.17}$$

In the equations (4.16) matrices  $[D\bar{\mathbf{F}}/D\bar{\mathbf{q}}]^n$ ,  $[D\bar{\mathbf{G}}/D\bar{\mathbf{q}}]^n$  and  $[D\bar{\mathbf{H}}/D\bar{\mathbf{q}}]^n$ , marked with  $\bar{\mathbf{A}}$ ,  $\bar{\mathbf{B}}$  and  $\bar{\mathbf{C}}$ , respectively, are defined as follows:

$$\bar{\mathbf{A}}, \quad \bar{\mathbf{B}}, \quad \bar{\mathbf{C}} =$$

$$\begin{bmatrix} k_t & k_x & k_y & k_z & 0 \\ k_x \phi^2 - u\theta & k_t + \theta - k_x(\gamma - 2)u & k_y u - k_x(\gamma - 1)v & k_z u - k_x(\gamma - 1)w & k_x(\gamma - 1) \\ k_y \phi^2 - v\theta & k_x v - k_y(\gamma - 1)u & k_t + \theta - k_y(\gamma - 2)v & k_z v - k_y(\gamma - 1)w & k_y(\gamma - 1) \\ k_z \phi^2 - w\theta & k_x w - k_z(\gamma - 1)u & k_y w - k_z(\gamma - 1)v & k_t + \theta - k_z(\gamma - 2)w & k_z(\gamma - 1) \\ \theta(\phi^2 - \omega) & k_x \omega - (\gamma - 1)u\theta & k_y \omega - (\gamma - 1)v\theta & k_z \omega - (\gamma - 1)w\theta & k_t + \gamma\theta \end{bmatrix},$$

where  $k = (\xi, \eta, \zeta)$  for matrices  $\bar{\mathbf{A}}$ ,  $\bar{\mathbf{B}}$  and  $\bar{\mathbf{C}}$ , respectively, while the quantities  $\phi^2$ ,  $\theta$  and  $\omega$  are defined by expressions:

$$\begin{aligned}\phi^2 &= \frac{1}{2}(\gamma - 1)(u^2 + v^2 + w^2), \\ \theta &= k_x u + k_y v + k_z w, \\ \omega &= \gamma e / \rho - \phi^2.\end{aligned}\tag{4.18}$$

Finite volume method, applied in solution of Euler equations, implies discretization of the computational domain on an appropriate number of cells, which in this approach have a hexahedron form. After the discretization was performed, for each cell a system of equations, that approximate

the system (4.13)-(4.15), supplemented by the relation (4.5) is set up. With known values of the dependent variable  $\bar{q}^n$  inside each cell  $(i, j, k)$ , approximate form of the system of equations (4.13.1) is obtained by

$$[\mathbf{I} + \beta \Delta t (\delta_\xi \bar{\mathbf{A}}^n + \delta_\eta \bar{\mathbf{B}}^n + \delta_\zeta \bar{\mathbf{C}}^n)] \Delta \bar{\mathbf{q}}^n + \Delta t \bar{\mathbf{R}}^n = 0, \quad (4.19)$$

where the residual  $\bar{\mathbf{R}}^n$  is

$$\bar{\mathbf{R}}^n = \delta_\xi \bar{\mathbf{F}}(\bar{\mathbf{q}}^n) + \delta_\eta \bar{\mathbf{G}}(\bar{\mathbf{q}}^n) + \delta_\zeta \bar{\mathbf{H}}(\bar{\mathbf{q}}^n). \quad (4.19.1)$$

In the equations (4.19) and (4.19.1)  $\delta_\xi$ ,  $\delta_\eta$  and  $\delta_\zeta$  represent central difference operators  $\partial/\partial\xi$ ,  $\partial/\partial\eta$  and  $\partial/\partial\zeta$ .

The parameter  $\beta$  in the expression (4.19) determines time accuracy of applied scheme. For the value  $\beta = 0.5$  the scheme is second order accurate, while accuracy decreases to the first order for  $\beta = 1$ . Unfactored implicit scheme, based on the equation (4.19), requires large available computer memory space for solving huge band block matrices, which is very difficult to provide for the case of three-dimensional flows. Unconditionally stable implicit scheme, with the largest possible error up to the order  $(\Delta t)^2$ , regardless of the number of spatial dimensions, is derived by the so-called LU factorization [20], [21] and [6]

$$\begin{aligned} & [\mathbf{I} + \beta \Delta t (\delta_\xi^- \bar{\mathbf{A}}^+ + \delta_\eta^- \bar{\mathbf{B}}^+ + \delta_\zeta^- \bar{\mathbf{C}}^+)]^n * \\ & * [\mathbf{I} + \beta \Delta t (\delta_\xi^+ \bar{\mathbf{A}}^- + \delta_\eta^+ \bar{\mathbf{B}}^- + \delta_\zeta^+ \bar{\mathbf{C}}^-)]^n \Delta \bar{\mathbf{q}}^n + \Delta t \bar{\mathbf{R}}^n = 0, \end{aligned} \quad (4.20)$$

where  $\delta_\xi^-$ ,  $\delta_\eta^-$  and  $\delta_\zeta^-$  are difference operators “backward” oriented, while  $\delta_\xi^+$ ,  $\delta_\eta^+$  and  $\delta_\zeta^+$  are “forward” oriented:

$$\begin{aligned} \delta_\xi^- (\bar{\mathbf{A}}^+ \Delta \bar{\mathbf{q}}^n)_{i,j,k} &= \bar{\mathbf{A}}_{i+1/2,j,k}^+ \Delta \bar{\mathbf{q}}_{i,j,k}^n - \bar{\mathbf{A}}_{i-1/2,j,k}^+ \Delta \bar{\mathbf{q}}_{i-1,j,k}^n, \\ \delta_\xi^+ (\bar{\mathbf{A}}^- \Delta \bar{\mathbf{q}}^n)_{i,j,k} &= \bar{\mathbf{A}}_{i+1/2,j,k}^- \Delta \bar{\mathbf{q}}_{i+1,j,k}^n - \bar{\mathbf{A}}_{i-1/2,j,k}^- \Delta \bar{\mathbf{q}}_{i,j,k}^n, \end{aligned} \quad (4.20.1)$$

where the terms of the matrices with indices  $(i+1/2, j, k)$  and  $(i-1/2, j, k)$  are computed by averaging flow variables between grid cells defined by indices  $(i, j, k)$  and  $(i+1, j, k)$ , ie.  $(i-1, j, k)$  and  $(i, j, k)$ . Difference operators related to the remaining two spatial directions can be determined on the basis of similar relations.

Matrix decomposition in factorized scheme (4.20) provides diagonal dominance of terms within the parentheses, while implicitly introduces an artificial viscosity, necessary to stabilize the applied scheme based on central difference operators. Flux matrices  $\bar{\mathbf{A}}^+$ ,  $\bar{\mathbf{B}}^+$ ,  $\bar{\mathbf{C}}^+$ ,  $\bar{\mathbf{A}}^-$ ,  $\bar{\mathbf{B}}^-$  and  $\bar{\mathbf{C}}^-$  in the equation (4.20) are formed in way providing that eigenvalues of the “+” matrix are non-negative, and the eigenvalues of “-” matrix are non-positive

$$\begin{aligned} \bar{\mathbf{A}}^+ &= \frac{1}{2} (\bar{\mathbf{A}} + r_A \mathbf{I}), & \bar{\mathbf{A}}^- &= \frac{1}{2} (\bar{\mathbf{A}} - r_A \mathbf{I}), \\ \bar{\mathbf{B}}^+ &= \frac{1}{2} (\bar{\mathbf{B}} + r_B \mathbf{I}), & \bar{\mathbf{B}}^- &= \frac{1}{2} (\bar{\mathbf{B}} - r_B \mathbf{I}), \\ \bar{\mathbf{C}}^+ &= \frac{1}{2} (\bar{\mathbf{C}} + r_C \mathbf{I}), & \bar{\mathbf{C}}^- &= \frac{1}{2} (\bar{\mathbf{C}} - r_C \mathbf{I}), \end{aligned} \quad (4.20.2)$$

where  $\mathbf{I}$  is the unit matrix, while the factors  $r_A$ ,  $r_B$  and  $r_C$  are defined as follows:

$$r_A \geq \max(|\lambda_A|), \quad r_B \geq \max(|\lambda_B|) \quad \text{and} \quad r_C \geq \max(|\lambda_C|), \quad (4.20.3)$$

bearing in mind that  $\lambda_A$ ,  $\lambda_B$  and  $\lambda_C$  represent eigenvalues of flux matrices  $\bar{\mathbf{A}}$ ,  $\bar{\mathbf{B}}$  and  $\bar{\mathbf{C}}$ , respectively,

ie.

$$\begin{aligned}\lambda_A &= (U, U, U, U + c(\xi_x^2 + \xi_y^2 + \xi_z^2)^{\frac{1}{2}}, U - c(\xi_x^2 + \xi_y^2 + \xi_z^2)^{\frac{1}{2}}), \\ \lambda_B &= (V, V, V, V + c(\eta_x^2 + \eta_y^2 + \eta_z^2)^{\frac{1}{2}}, V - c(\eta_x^2 + \eta_y^2 + \eta_z^2)^{\frac{1}{2}}), \\ \lambda_C &= (W, W, W, W + c(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}, W - c(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}).\end{aligned}\quad (4.20.4)$$

In the expressions (4.20.4)  $U$ ,  $V$  and  $W$  are contravariant coordinates of velocity vector, previously determined by transformations (4.15.3), while  $c$  denotes local speed of sound.

Solutions of the system of equations (4.20) can be determined in two passes:

$$\begin{aligned}[\mathbf{I} + \beta \Delta t (\delta_\xi^- \bar{\mathbf{A}}^+ + \delta_\eta^- \bar{\mathbf{B}}^+ + \delta_\zeta^- \bar{\mathbf{C}}^+)]^n \Delta \bar{\mathbf{q}}^{*n} &= -\Delta t \bar{\mathbf{R}}^n, \\ [\mathbf{I} + \beta \Delta t (\delta_\xi^+ \bar{\mathbf{A}}^- + \delta_\eta^+ \bar{\mathbf{B}}^- + \delta_\zeta^+ \bar{\mathbf{C}}^-)]^n \Delta \bar{\mathbf{q}}^n &= \Delta \bar{\mathbf{q}}^{*n}.\end{aligned}\quad (4.20.5)$$

The process of solving of the first system of equation (4.20.5) takes place in the direction of increasing index  $(i, j, k)$  by replacing the known, previously calculated values of unknown quantities  $\Delta \bar{\mathbf{q}}^{*n}$ , while the second one is solved in the direction of decreasing index  $(i, j, k)$ .

The two-factor **LU** implicit scheme requires correct implementation of boundary conditions in the implicit term on the left hand of equation (4.20.5). One way comes down to use of boundary conditions based on determination values of *characteristic variables* [50], which on the body surface are determined by the following expressions:<sup>2</sup>

$$\begin{aligned}p_b &= p_1 + \rho_1 c_1 (\bar{\zeta}_t + \bar{\zeta}_x u_1 + \bar{\zeta}_y v_1 + \bar{\zeta}_z w_1), \\ \rho_b &= \rho_1 + (p_b - p_1)/c_1^2, \\ u_b &= u_1 - \bar{\zeta}_x (\bar{\zeta}_t + \bar{\zeta}_x u_1 + \bar{\zeta}_y v_1 + \bar{\zeta}_z w_1), \\ v_b &= v_1 - \bar{\zeta}_y (\bar{\zeta}_t + \bar{\zeta}_x u_1 + \bar{\zeta}_y v_1 + \bar{\zeta}_z w_1), \\ w_b &= w_1 - \bar{\zeta}_z (\bar{\zeta}_t + \bar{\zeta}_x u_1 + \bar{\zeta}_y v_1 + \bar{\zeta}_z w_1),\end{aligned}\quad (4.20.6)$$

where  $\bar{\zeta}_t$ ,  $\bar{\zeta}_x$ ,  $\bar{\zeta}_y$  and  $\bar{\zeta}_z$  are:

$$\begin{aligned}\bar{\zeta}_t &= \frac{\zeta_t}{(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}}, & \bar{\zeta}_y &= \frac{\zeta_y}{(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}}, \\ \bar{\zeta}_x &= \frac{\zeta_x}{(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}}, & \bar{\zeta}_z &= \frac{\zeta_z}{(\zeta_x^2 + \zeta_y^2 + \zeta_z^2)^{\frac{1}{2}}}.\end{aligned}\quad (4.20.7)$$

In the relations (4.20.6) the index 1 denotes values of flow quantities inside cells in direct contact with the body surface, while index  $b$  correspond to values of the mentioned quantities exactly on the body surface. At the outer boundaries of the physical domain, flow quantities are determined by the far field values, for the case of subsonic flow, when the Mach number of undisturbed flow  $M_\infty$  does not exceed 1. In supersonic flow ( $M_\infty \geq 1$ ), values at the outer boundary are determined by extrapolation of flow variables values known within cells on "outflow" boundary.

By analyzing the equations (4.20) and (4.20.1) it can be concluded that on the body surface information on the values of flow quantities within the body are needed, thus coming into conflict with the physical flow pattern. In order to overcome the mentioned problem, it is necessary to perform a modification of the scheme for all grid cells with index  $k = 1$

$$\bar{\mathbf{C}}_{i,j,k-1/2}^+ \Delta \bar{\mathbf{q}}_{i,j,k-1} = \mathbf{E} \bar{\mathbf{C}}_{i,j,k-1/2}^- \Delta \bar{\mathbf{q}}_{i,j,k}, \quad (4.20.8)$$

<sup>2</sup>On the meaning of introduction *characteristic variables* will be said in chapter 4.2.3.

where the matrix  $\mathbf{E}$  is defined as follows:

$$\mathbf{E} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}. \quad (4.20.9)$$

The introduced correction eliminates the flow through the boundary surface of the body, and in the case of symmetric flow it is similarly possible to perform with cells in the plane of symmetry of the aircraft. At the outer boundaries of the physical domain Whitfield [5] proved the assumption  $\Delta \bar{\mathbf{q}}^n = 0$ . With mentioned corrections, solving algebraic systems equation (4.20.5) is possible without the use of huge memory space, which in addition with reduced factorization error, due to the presence of only two passes, represents a great advantage of this method over classic **ADI** schemes. If the process of solving the system of equation (4.20.5) in the computational domain occurs in “diagonal” planes, defined by relation  $i + j + k = \text{const}$ , program code can be completely vectorized for efficient usage on supercomputers.

Application of central differential scheme in calculation of the residual term  $\bar{\mathbf{R}}^n$  in relation (4.19.1) requires the introduction of additional terms of artificial viscosity to obtain stationary solutions of differential equations [45]. Method of stabilization of the factorized equation (4.20), based on introduction of dissipative terms of the second and fourth order modifying a fourth-order accurate scheme to a scheme of second order accuracy in the immediate vicinity of discontinuous changes, identical is to the procedure described in chapter 4.2.1. By switching to second-order accuracy it is possible to precisely determine the abrupt changes in the small number of calculation cells, thus avoiding inaccuracy in the vicinity of shock waves. A combination of dissipative terms is described by relation

$$\mathbf{R}_1^n = (D_\xi^2 + D_\eta^2 + D_\zeta^2 - D_\xi^4 - D_\eta^4 - D_\zeta^4) \mathbf{q}_{i,j,k}^n. \quad (4.21)$$

In the expression (4.21) the term  $D_\xi^2 \mathbf{q}_{i,j,k}^n$  is determined by relation

$$D_\xi^2 \mathbf{q}_{i,j,k}^n = \mathbf{d}_{i+1/2,j,k}^{(2)} - \mathbf{d}_{i-1/2,j,k}^{(2)}, \quad (4.21.1)$$

where is

$$\mathbf{d}_{i+1/2,j,k}^{(2)} = \varepsilon_{i+1/2,j,k}^{(2)} \frac{J_{i+1/2,j,k}}{\Delta t} \delta_\xi^+ \mathbf{q}_{i,j,k}^n, \quad (4.21.2)$$

where the term  $D_\xi^4 \mathbf{q}_{i,j,k}^n$  is calculated by expression

$$D_\xi^4 \mathbf{q}_{i,j,k}^n = \mathbf{d}_{i+1/2,j,k}^{(4)} - \mathbf{d}_{i-1/2,j,k}^{(4)}, \quad (4.21.3)$$

in which is

$$\mathbf{d}_{i+1/2,j,k}^{(4)} = \varepsilon_{i+1/2,j,k}^{(4)} \frac{J_{i+1/2,j,k}}{\Delta t} \delta_\xi^{3+} \mathbf{q}_{i,j,k}^n. \quad (4.21.4)$$

In relations (4.21.2) and (4.21.4) terms  $\delta_\xi^+ \mathbf{q}_{i,j,k}^n$  and  $\delta_\xi^{3+} \mathbf{q}_{i,j,k}^n$  represent forward-oriented difference operators

$$\delta_\xi^+ \mathbf{q}_{i,j,k}^n = \mathbf{q}_{i+1,j,k}^n - \mathbf{q}_{i,j,k}^n \quad (4.21.5)$$

and

$$\delta_\xi^{3+} \mathbf{q}_{i,j,k}^n = \mathbf{q}_{i+2,j,k}^n - 3\mathbf{q}_{i+1,j,k}^n + 3\mathbf{q}_{i,j,k}^n - \mathbf{q}_{i-1,j,k}^n. \quad (4.21.6)$$

The remaining dissipative terms, present in the expression (4.21), are determined in a similar way. Quantities  $\varepsilon_{i+1/2,j,k}^{(2)}$  and  $\varepsilon_{i+1/2,j,k}^{(4)}$  in relations (4.21.2) and (4.21.4) are defined by the expressions (4.9.7) and (4.9.8).

After the dissipative terms introduction, the system the equation (4.20) receives the final form

$$\begin{aligned}
 & [\mathbf{I} + \beta \Delta t (\delta_{\xi}^- \bar{\mathbf{A}}^+ + \delta_{\eta}^- \bar{\mathbf{B}}^+ + \delta_{\zeta}^- \bar{\mathbf{C}}^+)]^n * \\
 & * [\mathbf{I} + \beta \Delta t (\delta_{\xi}^+ \bar{\mathbf{A}}^- + \delta_{\eta}^+ \bar{\mathbf{B}}^- + \delta_{\zeta}^+ \bar{\mathbf{C}}^-)]^n \Delta \bar{\mathbf{q}}^n + \Delta t [\bar{\mathbf{R}}^n - \mathbf{R}_1^n] = 0.
 \end{aligned}
 \tag{4.22}$$

The integration step  $\Delta t$  for cell  $(i, j, k)$  is determined by the relation (4.12), in the same way as in the case of explicit numerical scheme.

Stability analysis of exposed two-factor implicit LU scheme [2] showed the insensitivity of the scheme to relatively large values of the Courant number. Also, in the described implicit approach the application of the variable integration steps lead to a significant acceleration of solution convergence, so the conclusions from the chapter 4.2.1 can be applied without any restriction.

The mentioned discussion certainly relates to stationary flow computation, while in the non-stationary flow field analysis a constant integration step  $\Delta t = \min(\Delta t_{i,j,k})$  has to be used.

Figures 4.3 and 4.4 show results of numerical calculation by the described procedure. Stationary transonic flow around rectangular wing and non-stationary flow around the rectangular wing in oscillatory translational motion are analyzed. In both cases three-dimensional algebraic non-orthogonal ‘‘C-H’’ computational grid is used. The presented results refer to a convergent solution that is realized at the moment when the value of the residual term is reduced to the size of the fourth order relative to the initial.

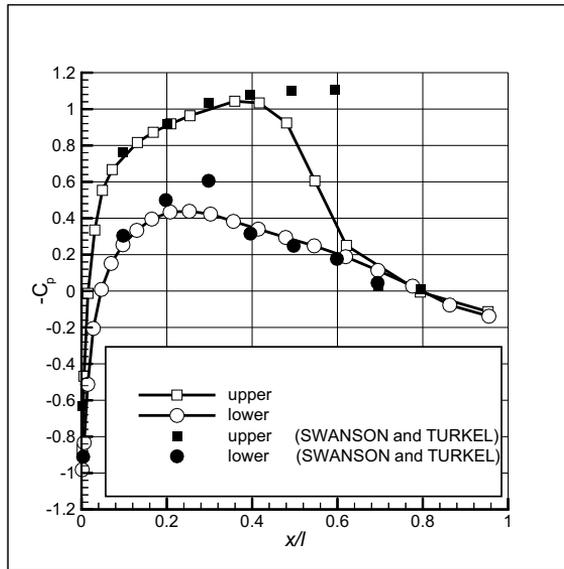


Figure 4.3: Pressure distribution on a rectangular wing – plane of symmetry.

The wing in non-stationary flow field, at Mach number  $M_{\infty} = 0.8$  of undisturbed flow and angle of attack  $\alpha = 1.25^\circ$  was observed. Pressure distribution on the airfoil in the plane of symmetry is

calculated for very coarse grid ( $65 \times 11 \times 15$ ) and shown on Figure 4.3 together with Swanson and Turkel results [45], calculated using an explicit numeric scheme.

As can be seen from Figure 4.3, good agreement of the values of the pressure coefficients is present near the leading edge of the airfoil, with minor deviations in the immediate vicinity of the shock wave. The reason for the mentioned disagreement lies in the application of a very coarse grid. Using a finer grid would certainly lead to a qualitative improvement in the zone of very strong pressure variations.

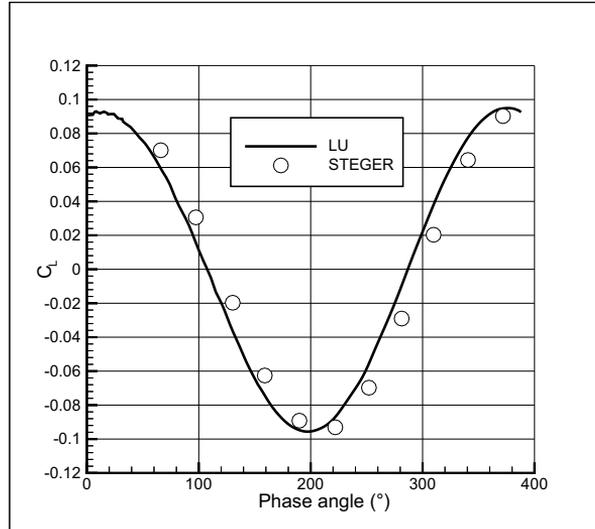


Figure 4.4: Dependence of lift coefficient on phase angle.

Example of non-stationary flow, shown in Figure 4.4, refers to the oscillatory translational wing motion with airfoil NACA 65A010 along the span. Change of the lift coefficient  $C_L$  of the airfoil in the plane of symmetry of the wing, depending on the phase angle of the given oscillatory wing motion, is compared with the corresponding Steger results [44]. A good match of the lift coefficient followed by small phase angle deviation is quite obvious.

Numerical stability and fast convergence of system of differential equations solution represent a significant improvement in relation to the application of the explicit method. **LU** implicit factorization by applying flux “decomposition” allow the use of very large integration steps thanks to its stability, even for Courant number  $CFL \geq 20$ . Increased scheme stability made the exposed procedure acceptable in the analysis of non-stationary flows in three-dimensional space, when the integration step has crucial role. On the other hand, the presence of only two factors in the three-dimensional flow case reduces the factorization error. Relative to application of the classical **ADI** scheme, the presented procedure also showed advantage in terms of CPU time savings. The way of solving systems of equations does not require a large memory space and it consists of the inversion of fifth-order square matrices. Correct program coding allows easy vectorization and application on supercomputers. The implemented modification of the scheme at the boundaries of the physical domain provided precise definition of boundary conditions, and thus high accuracy of the solution.

Results obtained by applying this numerical procedure in the analysis of three-dimensional fluid flow [31] showed a good agreement with the available literature data, based on different

computational methods in Euler equations solving, as well as the results obtained experimentally. Exceptional accuracy of the solution in the vicinity of shock waves, in the case of high Mach numbers of undisturbed flow, represents a significant advantage of the described approach in aerodynamic load calculation over methods based on potential theory.

### 4.2.3 Godunov scheme

Tendency in the formation of such a mathematical flow model which would show universality in solving a wide spectrum of problems is present in the modern numerical fluid mechanics. This universality consists in achieving satisfactory accuracy of the numerical solution, stability and rapid convergence of solutions in all flow problems, without introducing subsequent adjustment of mathematical model to a specific calculation task. Most recently developed methods of mathematical flow problems modeling are based on the introduction of appropriate adjustment constants, often empirically derived. The presence of the mentioned constants has for aim to obtain results by numerical simulation of physical problems that are nearly identical with experimentally determined ones.

In this chapter, starting from known gasdynamic laws and some approaches performed so far in the field of numerical fluid mechanics [33], [31], a universal modeling procedure will be recommended in modern computational fluid dynamics. The main goal will be forming a simple scheme that would be used, due to its own structure, in very complex computation of three-dimensional (3D) flow without request for powerful supercomputer usage.

The name “*Godunov scheme*” will refer to the whole a class of methods based on the wave properties of hyperbolic partial differential equations. Analysis of wave components propagation, caused by decomposition of complex nonlinear waves, led to the emergence of this numerical schemes class [52].

### Godunov scheme – one-dimensional Euler equations

Euler equations in the case of one-dimensional (1D), non-stationary flow can be written in integral form in the following way:

$$\begin{aligned}\oint [\rho dx - \rho u dt] &= 0, \\ \oint [\rho u dx - (p + \rho u^2) dt] &= 0, \\ \oint \left[ \rho \left( e_u + \frac{1}{2} u^2 \right) dx - \rho u \left( e_u + \frac{p}{\rho} + \frac{1}{2} u^2 \right) dt \right] &= 0,\end{aligned}\tag{4.23}$$

where the first equation (4.23) represents continuity equation, while the second and third represent momentum and energy equation, respectively. Integrals in expressions (4.23) are calculated on a closed contour in the  $(x, t)$  plane, where the  $x$  coordinate is in the flow direction, and  $t$  is time. Closed contours can also contain lines (surfaces) of discontinuity of flow quantities  $\rho$ ,  $u$ ,  $p$  and  $e_u$  which denote density, velocity, pressure and internal energy per unit fluid mass, respectively.

With the adopted assumption that the fluid moves parallel to the axis  $x$  at the moment  $t = t_0$  the area occupied by that gas will be divided on segments of equal lengths  $h$ . The mentioned segment boundaries will be marked with  $x_m, x_{m+1}, \dots$ , while the values of the flow variables in the segment with the boundaries  $x_m$  and  $x_{m+1}$  will be denoted by the lower index  $m + 1/2$ . Calculated values of the flow quantities at time  $t = t_0 + \tau$  in the mentioned segment will be denoted by the upper index  $m + 1/2$ . With the known flow variables values at time  $t = t_0$ , an approximation that the quantities  $\rho_{m+1/2}$ ,  $u_{m+1/2}$ ,  $p_{m+1/2}$  and  $e_{u_{m+1/2}}$  are constant within each cell is introduced, bearing in mind that the cell dimensions are small enough.

To calculate the values  $\rho^{m+1/2}$ ,  $u^{m+1/2}$ ,  $p^{m+1/2}$  and  $e_u^{m+1/2}$  inside cell with the boundaries  $x_m$  and  $x_{m+1}$ , it is assumed that the boundaries of the segment disappear, applying the equations (4.23) on cell bounded with  $x = x_m$ ,  $x = x_{m+1}$ ,  $t = t_0$  and  $t = t_0 + \tau$ . In calculation process it is considered that the flow variables after the disappearance of the boundaries, due to small time interval  $\tau$ , are constant on imaginary cell boundaries. Introducing the notations  $R$ ,  $U$ ,  $P$  and  $E_u$  for flow variables at the boundaries of the segments, the relation [25] is reached

$$\begin{aligned} \rho^{m+1/2} &= \rho_{m+1/2} - \frac{\tau}{h} [(RU)_{m+1} - (RU)_m], \\ (\rho u)^{m+1/2} &= (\rho u)_{m+1/2} - \frac{\tau}{h} [(P + RU^2)_{m+1} - (P + RU^2)_m], \\ \left[ \rho \left( e_u + \frac{1}{2} u^2 \right) \right]^{m+1/2} &= \left[ \rho \left( e_u + \frac{1}{2} u^2 \right) \right]_{m+1/2} - \\ &\quad - \frac{\tau}{h} \left\{ \left[ RU \left( E_u + \frac{P}{R} + \frac{U^2}{2} \right) \right]_{m+1} - \left[ RU \left( E_u + \frac{P}{R} + \frac{U^2}{2} \right) \right]_m \right\}, \end{aligned} \quad (4.24)$$

which are supplemented by the state equation of the ideal gas  $p = p(\rho, e_u)$ .

The quantities  $R$ ,  $U$ ,  $P$  and  $E_u$  are determined from breakdown formulae, described in detail in the literature [22], whose resolving leads to values of the mentioned quantities at the cell boundaries, which at the same time represent breakdown surfaces (lines). Despite the fact that breakdown formulae are relatively complicated [22], [8], flow variables can be considered constant at time boundaries  $t = t_0$  to  $t = t_0 + \tau$ , provided that the choice of sufficient small time interval  $\tau$  avoids wave interaction of disturbances emitted from the boundaries  $x = x_m$  and  $x = x_{m+1}$ .

Thanks to Godunov, Zabrodin and Prokopov [10], compact expressions for calculating the quantities on contact discontinuities  $p_{c.d.}$  and  $u_{c.d.}$ , after the disappearance of the imagined membrane on boundary  $x = x_m$  are obtained. Derived relations cover at the same time all possible breakdown cases arising from fictitious boundary removal

$$\begin{aligned} p_{c.d.} &= \frac{b_m p_{m-1/2} + a_m p_{m+1/2} + a_m b_m (u_{m-1/2} - u_{m+1/2})}{a_m + b_m}, \\ u_{c.d.} &= \frac{a_m u_{m-1/2} + b_m u_{m+1/2} + p_{m-1/2} - p_{m+1/2}}{a_m + b_m}, \end{aligned} \quad (4.25)$$

where is

$$a_m = \begin{cases} \left\{ \frac{1}{2} [(\gamma + 1)p_{c.d.} + (\gamma - 1)p_{m-1/2}] \rho_{m-1/2} \right\}^{1/2}, & \text{for } p_{c.d.} \geq p_{m-1/2}, \\ \frac{\gamma - 1}{2\gamma} (\gamma p_{m-1/2} \rho_{m-1/2})^{1/2} \frac{1 - p_{c.d.}/p_{m-1/2}}{1 - (p_{c.d.}/p_{m-1/2})^{(\gamma-1)/2\gamma}}, & \text{for } p_{c.d.} < p_{m-1/2}, \end{cases} \quad (4.26)$$

where  $\gamma$  is the adiabatic fluid constant. Quantity  $b_m$  in equations (4.25) is calculated by replacing the indices  $m - 1/2$  by the indices  $m + 1/2$  in relations (4.26).

The equations (4.25) and (4.26) are exact for finite discontinuities at the boundary  $x = x_m$  and are resolved by an iterative procedure. The iteration process is performed by adopting the initial assumption for  $p_{c.d.}$  and by determination the quantities  $a_m$  and  $b_m$  from the relation (4.26). The calculated values of  $a_m$  and  $b_m$  are then used in the first of the

equations (4.25), thus obtaining the new value of the contact discontinuity pressure  $p_{c.d.}$ . The process is repeated to complete convergence of pressure  $p_{c.d.}$ .

The mentioned iterative scheme need only be applied in the discontinuity zone (break-down), when the cell boundary coincides with the shock wave. In general case, when the changes in dependent flow variables across the cell boundary are small compared to cell length, approximate form of the equation (4.25) and (4.26) can be used successfully

$$\begin{aligned} a_m &= b_m = \left[ \frac{\gamma}{4} (\rho_{m-1/2} + \rho_{m+1/2}) (\rho_{m-1/2} + \rho_{m+1/2}) \right]^{1/2}, \\ p_{c.d.} &= \frac{\rho_{m+1/2} + \rho_{m-1/2}}{2} + a_m \frac{u_{m-1/2} - u_{m+1/2}}{2}, \\ u_{c.d.} &= \frac{u_{m+1/2} + u_{m-1/2}}{2} + \frac{\rho_{m-1/2} - \rho_{m+1/2}}{2a_m}. \end{aligned} \quad (4.27)$$

To determine the quantities  $R$ ,  $U$ ,  $P$  and  $E_u$  at the cell boundaries it is required to define three characteristic wave propagation speeds. Propagations speeds of the left and right acoustic waves are calculated by relations

$$D_L = u_{m-1/2} - \frac{a_m}{\rho_{m-1/2}} \quad \text{and} \quad D_D = u_{m+1/2} + \frac{a_m}{\rho_{m+1/2}}, \quad (4.28)$$

respectively, while the third characteristic speed is precisely the contact discontinuity speed  $u_{c.d.}$ . For time  $t > t_0$  the  $(x, t)$  plane near the boundary  $x = x_m$  is divided into four zones, shown in Figure 4.5.

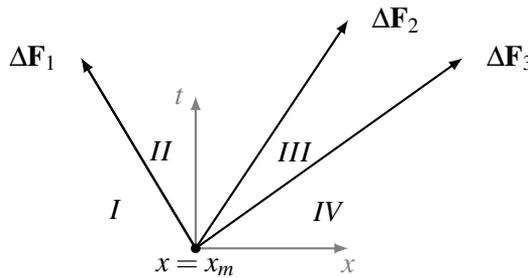


Figure 4.5: Wave propagation near the cell boundary.

Values of flow variables in zones **I** and **IV** are equal to values corresponding to the indices  $m - 1/2$  and  $m + 1/2$ , respectively. In the zones **II** and **III** values of contact discontinuity pressure  $p_{c.d.}$  and velocity  $u_{c.d.}$  are calculated on the basis of equation (4.25), or (4.27). The density on the left side is determined from

$$\frac{\rho_L}{\rho_{m-1/2}} = \frac{(\gamma + 1)p_{c.d.} + (\gamma - 1)\rho_{m-1/2}}{(\gamma - 1)p_{c.d.} + (\gamma + 1)\rho_{m-1/2}}, \quad (4.29)$$

while the corresponding value on the right side  $\rho_D$  is obtained by substitution index  $m - 1/2$  by index  $m + 1/2$  in the equation (4.29).

Depending on the sign of the wave propagation velocities  $D_L$  and  $D_D$  the values of the variables  $R$ ,  $U$  and  $P$  will be determined. In case where  $D_L$  and  $D_D$  are of the same sign,  $R$ ,  $U$  and  $P$  take the values  $(\rho_L, u_L, p_L)$  from zone **I** if  $D_L$  and  $D_D$  are positive, and values  $(\rho_D, u_D, p_D)$  from zone **IV** if  $D_L$  and  $D_D$  are negative. If the velocities  $D_L$  and  $D_D$  are of opposite signs, for  $P$  and  $U$  the values  $p_{c.d.}$  and  $u_{c.d.}$  are taken, respectively, while  $R = \rho_L$  if  $u_{c.d.} > 0$ , or  $R = \rho_D$  in case  $u_{c.d.} < 0$ .

The problem analyzed in this chapter is known in the literature as the *Riemann problem*.

### Roe scheme – approximation of Riemann problem

Unlike Godunov model, which represents an approximate solution of the exact physical problem, in this chapter will be presented procedure for solving a system of linear hyperbolic equations, which approximate the exact Euler system of equation [50], [40]

$$\begin{aligned} \frac{\partial \mathbf{q}}{\partial t} + \bar{\mathbf{A}}(\mathbf{q}_L, \mathbf{q}_D) \frac{\partial \mathbf{q}}{\partial x} &= 0, \\ \mathbf{q}(x, 0) &= \mathbf{q}_L, \quad \text{for } x < x_m \\ \mathbf{q}(x, 0) &= \mathbf{q}_D, \quad \text{for } x > x_m \end{aligned} \quad (4.30)$$

where  $\mathbf{q}$  is a vector of variable flow quantities

$$\mathbf{q} = (\rho, \rho u, \rho e_t)^T, \quad (4.31)$$

while  $\rho$ ,  $u$  and  $e_t$  are density, flow velocity and specific total fluid energy, respectively. In the first equation of system (4.30)  $\bar{\mathbf{A}}$  represents the locally constant Jacobi matrix  $\partial \mathbf{F} / \partial \mathbf{q}$ , where  $\mathbf{F}$  is the flux vector

$$\mathbf{F} = \left\{ \begin{array}{c} \rho u \\ \rho u^2 + p \\ \rho u(e_t + p/\rho) \end{array} \right\}, \quad (4.32)$$

bearing in mind that  $e_t = e_u + u^2/2$ .

Jump of the flow variables vector  $\mathbf{q}$  through each *characteristic line*, shown on Figure 4.5, corresponding to eigenvalue  $\lambda^{(j)}$  of matrix  $\bar{\mathbf{A}}$ , is proportional to the right eigenvector  $\mathbf{r}^{(j)}$  associated to eigenvalue  $\lambda^{(j)}$ , ie.

$$\mathbf{q}_D - \mathbf{q}_L = \sum_{j=1}^3 \alpha_j \mathbf{r}^{(j)}, \quad (4.33)$$

where  $\alpha_j$  is “strength” of the  $j$ th wave, while the jump the flux vector  $\mathbf{F}$  can be represented by relation

$$\mathbf{F}_D - \mathbf{F}_L = \sum_{j=1}^3 \Delta \mathbf{F}_j = \sum_{j=1}^3 \alpha_j \lambda^{(j)} \mathbf{r}^{(j)}. \quad (4.34)$$

To solve the hyperbolic system equation (4.30) the flux vectors of flow variables at each boundary cells have to be known. At the cell boundary  $x = x_m$  (Figure 4.5) flow variables flux, denoted by  $\mathbf{F}_m$ , is calculated on the basis of expression

$$\mathbf{F}_m = \mathbf{F}_L + \sum_{j=1}^{k_1} \alpha_j \lambda^{-(j)} \mathbf{r}^{(j)}, \quad (4.35)$$

where the index  $(-)$  of eigenvalue  $\lambda^{(j)}$  shows that summation refers to characteristic lines with a negative slopes, associated to negative eigenvalues. Also, a relation can be obtained:

$$\mathbf{F}_m = \mathbf{F}_D - \sum_{j=1}^{k_2} \alpha_j \lambda^{+(j)} \mathbf{r}^{(j)}, \quad (4.36)$$

in which the index  $(+)$  denotes the sum of the terms associated only with positive eigenvalues. By combination of equation (4.35) and (4.36) a new form of flow variables flux is obtained

$$\mathbf{F}_m = \frac{1}{2}(\mathbf{F}_L + \mathbf{F}_D) - \frac{1}{2} \sum_{j=1}^3 \alpha_j |\lambda^{(j)}| \mathbf{r}^{(j)}. \quad (4.37)$$

To prevent "even-odd" decoupling of numerical solution and the presence of unwanted high-frequency oscillations near the shock wave, and also to accelerate convergence, it is necessary to introduce terms of artificial viscosity into appropriate discrete operators. One way, based on Jameson research [18], mentioned in the chapter 4.2.1, refers to the central difference schemes. In the expression for the numerical flux (4.37) artificial viscosity is implicitly included by the second term on the right hand of the mentioned equation, and avoids additional explicit introduction of stabilizing terms. In the case of a one-dimensional, non-stationary flow of the ideal gas quantities  $\alpha_j$ ,  $\lambda^{(j)}$  and  $\mathbf{r}^{(j)}$ , present in the equations (4.34)-(4.37), are defined by relations

$$\begin{aligned} \alpha_1 &= \frac{1}{\tilde{c}^2}(\hat{\Delta}p - \tilde{\rho}\tilde{c}\hat{\Delta}u), & \alpha_2 &= \frac{1}{\tilde{c}^2}(\tilde{c}^2\hat{\Delta}\rho - \hat{\Delta}p), & \alpha_3 &= \frac{1}{\tilde{c}^2}(\hat{\Delta}p + \tilde{\rho}\tilde{c}\hat{\Delta}u), \\ \lambda^{(1)} &= \tilde{u} - \tilde{c}, & \lambda^{(2)} &= \tilde{u}, & \lambda^{(3)} &= \tilde{u} + \tilde{c}, \\ \mathbf{r}^{(1)} &= \begin{Bmatrix} 1 \\ \tilde{u} - \tilde{c} \\ \tilde{H} - \tilde{u}\tilde{c} \end{Bmatrix}, & \mathbf{r}^{(2)} &= \begin{Bmatrix} 1 \\ \tilde{u} \\ \frac{1}{2}\tilde{u}^2 \end{Bmatrix}, & \mathbf{r}^{(3)} &= \begin{Bmatrix} 1 \\ \tilde{u} + \tilde{c} \\ \tilde{H} + \tilde{u}\tilde{c} \end{Bmatrix}, \end{aligned} \quad (4.38)$$

in which quantities labeled by  $(\sim)$  are defined in [roe] by expressions:

$$\begin{aligned} \tilde{\rho}^2 &= \rho_L \rho_D, & \tilde{H} &= \frac{\rho_L^{1/2} H_L + \rho_D^{1/2} H_D}{\rho_L^{1/2} + \rho_D^{1/2}}, \\ \tilde{u} &= \frac{\rho_L^{1/2} u_L + \rho_D^{1/2} u_D}{\rho_L^{1/2} + \rho_D^{1/2}}, & \tilde{c}^2 &= (\gamma - 1) \left( \tilde{H} - \frac{1}{2} \tilde{u}^2 \right), \end{aligned} \quad (4.39)$$

where quantity  $\tilde{H}$  corresponds to the "averaged" enthalpy of the fluid. The jump of flow variables through cell boundaries is indicated by operator  $\hat{\Delta}$  in relations (4.38).

The problem of applying described scheme is reflected in the possibility of a discontinuous expansion wave formation in the vicinity of the sonic point where Mach number is  $M = 1$ . In order to solve this problem, it is necessary to make a numerical flux correction in the environment of sonic point [41]. Harten and Hyman [13] propose a modification of eigenvalue module  $|\lambda^{(j)}|$  in equation (4.37) in the following manner:

$$|\lambda^{(j)}|_{\text{mod}} = \begin{cases} |\lambda^{(j)}|, & \text{if } |\lambda^{(j)}| \geq \varepsilon, \\ \varepsilon, & \text{if } |\lambda^{(j)}| < \varepsilon, \end{cases} \quad (4.40)$$

where the quantity  $\varepsilon$  is determined by the relation

$$\varepsilon = \max \left[ 0, (\lambda^{(j)} - \lambda_L^{(j)}), (\lambda_D^{(j)} - \lambda^{(j)}) \right]. \quad (4.41)$$

Zero, as the value of the function in the expression (4.41), eliminates introduced correction in case of shock wave. The alternative procedure [13] is reduced to relations

$$|\lambda^{(j)}|_{\text{mod}} = \begin{cases} |\lambda^{(j)}|, & \text{if } |\lambda^{(j)}| \geq \varepsilon, \\ \frac{1}{2}(\lambda^{(j)})^2 / \varepsilon + \varepsilon, & \text{if } |\lambda^{(j)}| < \varepsilon, \end{cases} \quad (4.42)$$

thus ensuring the continuity of the first derivative of quantity  $|\lambda^{(j)}|$ .

When fluid flows through a pipe with variable cross area section  $S(x)$ , when cross area section change is small compared to the pipe length, the flow can be considered quasi one-dimensional and the first of the equations (4.30) takes the form

$$\frac{\partial}{\partial t}(S\mathbf{q}) + \frac{\partial}{\partial x}(S\mathbf{F}) - \mathbf{H} = 0, \quad (4.43)$$

bearing in mind that vectors  $\mathbf{q}$  and  $\mathbf{F}$  are defined by the relations (4.31) and (4.32), respectively. The vector  $\mathbf{H}$ , present in equation (4.43), introduces the variation of the cross section area  $S(x)$  and is determined by the relation

$$\mathbf{H} = \frac{dS}{dx} \begin{Bmatrix} 0 \\ p \\ 0 \end{Bmatrix}. \quad (4.44)$$

Implicit difference scheme applicable to Euler equations (4.43) is defined by an algorithm

$$S \frac{\mathbf{q}^{n+1} - \mathbf{q}^n}{\Delta t} + \left[ \frac{\partial}{\partial x}(S\mathbf{F}) \right]^{n+1} - \mathbf{H}^{n+1} = 0. \quad (4.45)$$

In the equation (4.43), bearing in mind implicit formulation, the second and the third term depend on the flow variables in the following moment, defined by index  $n+1$ . If the flow variables vector increment in the time interval determined by the indices  $n$  and  $n+1$  is

$$\Delta \mathbf{q} = \mathbf{q}^{n+1} - \mathbf{q}^n, \quad (4.46)$$

applying local linearization with respect to known values of flow variables at time level  $n$ , it can be written for the flux vector  $\mathbf{F}$  at time  $n+1$

$$\mathbf{F}^{n+1} = \mathbf{F}^n + \frac{\partial \mathbf{F}^n}{\partial \mathbf{q}} \Delta \mathbf{q} + \mathcal{O}(\Delta t^2). \quad (4.47)$$

From the equation (4.35), numerical flux at the boundary  $m$  is obtained by the first order accuracy flux vector linearization in the cell  $m-1/2$ , ie.

$$\mathbf{F}_m^{n+1} = \mathbf{F}_{m-1/2}^{n+1} + \mathbf{A}_m^-(\mathbf{q}_{m+1/2}^{n+1} - \mathbf{q}_{m-1/2}^{n+1}), \quad (4.48)$$

while for the boundary  $m - 1$  the flux is determined from the equation (4.36)

$$\mathbf{F}_{m-1}^{n+1} = \mathbf{F}_{m-1/2}^{n+1} - \mathbf{A}_{m-1}^+ (\mathbf{q}_{m-1/2}^{n+1} - \mathbf{q}_{m-3/2}^{n+1}). \quad (4.49)$$

Based on the equations (4.48) and (4.49) it follows

$$\begin{aligned} \mathbf{F}_m^{n+1} - \mathbf{F}_{m-1}^{n+1} &= \mathbf{F}_m^n - \mathbf{F}_{m-1}^n + \mathbf{A}_m^- \Delta \mathbf{q}_{m+1/2}^n - \mathbf{A}_m^- \Delta \mathbf{q}_{m-1/2}^n + \\ &+ \mathbf{A}_{m-1}^+ \Delta \mathbf{q}_{m-1/2}^n - \mathbf{A}_{m-1}^+ \Delta \mathbf{q}_{m-3/2}^n. \end{aligned} \quad (4.50)$$

With the known flux vector  $\mathbf{F}$ , Jacobi matrix  $\bar{\mathbf{A}} = \partial \mathbf{F} / \partial \mathbf{q}$  becomes

$$\bar{\mathbf{A}} = \frac{\partial \mathbf{F}}{\partial \mathbf{q}} = \begin{bmatrix} 0 & 1 & 0 \\ (\frac{\gamma-3}{2})u^2 & -(\gamma-3)u & (\gamma-1) \\ -\gamma u e_t + (\gamma-1)u^3 & \gamma e_t - \frac{3(\gamma-1)}{2}u^2 & \gamma u \end{bmatrix}, \quad (4.51)$$

while Jacobi matrix  $\partial \mathbf{H} / \partial \mathbf{q}$  is defined by relation

$$\bar{\mathbf{B}} = \frac{\partial \mathbf{H}}{\partial \mathbf{q}} = (\gamma-1) \frac{dS}{dx} \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2}u^2 & -u & 1 \\ 0 & 0 & 0 \end{bmatrix}. \quad (4.52)$$

Averaged variables<sup>3</sup> in the expressions (4.38) are introduced so that the equation

$$\mathbf{F}_D - \mathbf{F}_L = \bar{\bar{\mathbf{A}}}(\mathbf{q}_D, \mathbf{q}_L) (\mathbf{q}_D - \mathbf{q}_L) \quad (4.53)$$

is satisfied if the elements of the matrix  $\bar{\bar{\mathbf{A}}}$  are obtained by substitution of the independent flow variables in the corresponding elements of matrix  $\bar{\mathbf{A}}$  with such averaged quantities.

Matrix  $\bar{\mathbf{A}}$  in the expression (4.51) can be diagonalized into form

$$\bar{\mathbf{A}} = \mathbf{T} \Lambda \mathbf{T}^{-1}, \quad (4.54)$$

where  $\Lambda$  is a diagonal matrix whose diagonal elements are eigenvalues of the matrix  $\bar{\mathbf{A}}$ .  $\mathbf{T}$  is a matrix whose rows are right eigenvectors of the matrix  $\bar{\mathbf{A}}$ , ie.

$$\mathbf{T} = \begin{bmatrix} 1 & \alpha & \alpha \\ u & \alpha(u+c) & \alpha(u-c) \\ \frac{1}{2}u^2 & \alpha(\frac{1}{2}u^2 + uc + \frac{c^2}{\gamma-1}) & \alpha(\frac{1}{2}u^2 - uc + \frac{c^2}{\gamma-1}) \end{bmatrix}, \quad (4.55)$$

while  $\mathbf{T}^{-1}$  is the matrix whose rows are left eigenvectors of the matrix  $\bar{\mathbf{A}}$

$$\mathbf{T}^{-1} = \begin{bmatrix} 1 - \frac{1}{2}u^2 \frac{\gamma-1}{c^2} & (\gamma-1) \frac{u}{c^2} & -(\gamma-1)c^2 \\ \beta[(\gamma-1)\frac{u^2}{2} - uc] & \beta[c - (\gamma-1)u] & \beta(\gamma-1) \\ \beta[(\gamma-1)\frac{u^2}{2} + uc] & -\beta[c + (\gamma-1)u] & \beta(\gamma-1) \end{bmatrix}. \quad (4.56)$$

<sup>3</sup>For a detailed determination of averaged variables procedure it is necessary to consult the literature [hirsch] (Hirsch, C. *Numerical Computation of Internal and External Flows*, Vol. 2, pp. 463–467).

The quantities  $\alpha$  and  $\beta$ , present in matrices  $\mathbf{T}$  and  $\mathbf{T}^{-1}$ , are calculated from relations

$$\alpha = \rho/(c\sqrt{2}) \quad \text{and} \quad \beta = 1/(\rho c\sqrt{2}),$$

in which  $c$  is the local speed of sound propagation.

By grouping non-positive eigenvalues of matrix  $\bar{\mathbf{A}}$  in  $\Lambda^-$  and positive ones in  $\Lambda^+$  Jacobi matrix  $\bar{\mathbf{A}}$  can be decomposed into matrix

$$\bar{\mathbf{A}}^+ = \mathbf{T}\Lambda^+\mathbf{T}^{-1}, \quad (4.57)$$

and matrix

$$\bar{\mathbf{A}}^- = \mathbf{T}\Lambda^-\mathbf{T}^{-1}, \quad (4.58)$$

where is

$$\bar{\mathbf{A}} = \bar{\mathbf{A}}^+ + \bar{\mathbf{A}}^-. \quad (4.59)$$

The equation (4.45) can now be written in the form of an implicit difference scheme

$$\mathbf{A}\mathbf{M}_{m-3/2}\Delta\mathbf{q}_{m-3/2} + \mathbf{A}\mathbf{A}_{m-1/2}\Delta\mathbf{q}_{m-1/2} + \mathbf{A}\mathbf{P}_{m+1/2}\Delta\mathbf{q}_{m+1/2} = \mathbf{D}\mathbf{S}_{m-1/2}, \quad (4.60)$$

where are

$$\begin{aligned} \mathbf{A}\mathbf{M}_{m-3/2} &= -\frac{\Delta t}{\Delta x}\bar{\mathbf{A}}_{m-1}^+, \\ \mathbf{A}\mathbf{A}_{m-1/2} &= [S_{m-1/2}\mathbf{I} + \frac{\Delta t}{\Delta x}(S_m\bar{\mathbf{A}}_m^+ - S_{m-1}\bar{\mathbf{A}}_{m-1}^-) - \bar{\mathbf{B}}_{m-1/2}\Delta t], \\ \mathbf{A}\mathbf{P}_{m+1/2} &= \frac{\Delta t}{\Delta x}\bar{\mathbf{A}}_m^-, \\ \mathbf{D}\mathbf{S}_{m-1/2} &= -\frac{\Delta t}{\Delta x}(S_m\mathbf{F}_m - S_{m-1}\mathbf{F}_{m-1}) + \Delta t\mathbf{H}_{m-1/2}, \end{aligned} \quad (4.61)$$

and  $\mathbf{I}$  is unit matrix.

Euler system of equations, represented by difference implicit scheme, also supplemented by appropriate boundary conditions [13] and [15], is solved by applying a standard block-diagonal procedure based on the inversion of third order matrices in the case of one-dimensional flows. The problem of flow in several dimensions comes down to application the so-called **ADI** scheme [35], [35], which, unfortunately, in case of the **3D** flow field requires a large amount of CPU time.

In order to increase the efficiency of the exposed procedure in analysis of **2D** and **3D** flows, **ADI** scheme with block-diagonal inversion will be replaced by the **LU** method of implicit factorization, presented in the chapter 4.2.2 and literature [20], [21] and [31]. Decomposition of flux matrices in a factorized scheme provides diagonal terms dominance and implicitly introduces artificial viscosity, necessary for stabilization of the central difference schemes. In the case of one-dimensional flow, matrices  $\bar{\mathbf{A}}^+$  and  $\bar{\mathbf{A}}^-$  previously defined by the relations (4.57) and (4.58), will be formed so that eigenvalues of matrix  $\bar{\mathbf{A}}^+$  are non-negative and eigenvalues of matrix  $\bar{\mathbf{A}}^-$  are negative, ie.

$$\bar{\mathbf{A}}^+ = \frac{1}{2}(\bar{\mathbf{A}} + r_A\mathbf{I}) \quad \text{and} \quad \bar{\mathbf{A}}^- = \frac{1}{2}(\bar{\mathbf{A}} - r_A\mathbf{I}), \quad (4.62)$$

where  $r_A \geq \max(|\lambda^{(j)}|)$ . This approach of flux matrix decomposition, based on the spectral radius  $\rho(\mathbf{A}) = \max(|\lambda^{(j)}|)$  of matrix  $\mathbf{A}$ , contribute to easy realization of program code. Quantities averaged by the expressions (4.39) are used in calculation of matrix  $\mathbf{A}$  elements. Decomposition of the flux matrix by applying the relation (4.57) and (4.58) requires far more complex coding, but gives increased numerical stability of solutions with accelerated convergence. Numerical results, using this method of matrix decomposition, will be presented in detail in the next chapter.

The process of solving the system of equations (4.60) using **LU** implicit factorization method consists of two one-way passes, which is explained in detail in the chapter 4.2.2.

To approve the accuracy of the presented numerical procedures, a comparison of such obtained results with the exact one from shock wave theory based on Rankine-Hugoniot equations [22], [8], will be performed.

The first example relates to the one-dimensional problem of ideal gas flow in a pipe of constant cross section, caused by the sudden removal of the diaphragm that separates the two areas gases of different thermodynamic quantities. The diagram (Figure 4.6) shows the pressure and density distribution along the axis of the pipe after a certain time interval  $\Delta t$ .

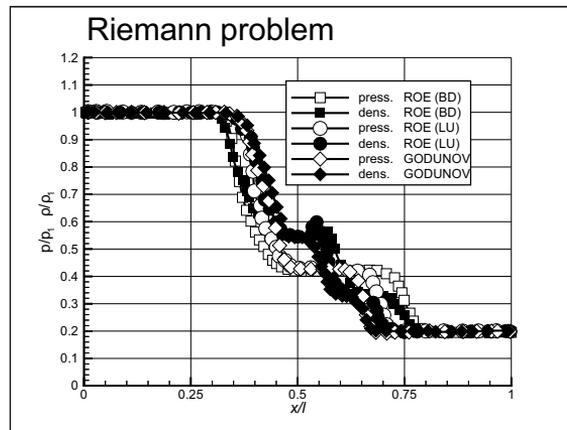


Figure 4.6: Pressure and density distribution in the pipe.

The diagram in Figure 4.6 shows a remarkable match of pressure and density distribution obtained by Godunov method application with the theoretical, in the shock wave zone and in zones of expansion wave and contact discontinuity. Despite of certain deviations in the zones of shock wave and contact discontinuity the results of the implicit block-diagonal procedure (**BD**) applied to the system of equations (4.39), obtained by the Roe scheme, as well as the methods of implicit **LU** factorization are quite acceptable. The main disadvantage of Godunov model is reflected in its complexity that is a major obstacle in the field of two-dimensional and three-dimensional flow field analysis. Explicit nature of this scheme prevents the use of larger integration steps, which greatly increases the computational time. Considering almost identical quality of the solution of the block-diagonal procedure and the procedure of implicit **LU** factorizations, taking into account given simplicity of **LU** factorization, this procedure is practically the best choice in the process of computer

analysis of the mentioned problem due to the essential computer time saving.

Flow analysis in a divergent nozzle, whose cross-section area varies along the  $x$  axis of the nozzle according to the law [15]

$$S(x) = 1.398 + 0.347 \tanh(0.8x + 0.4),$$

is the second task in which the results obtained by applying the block-diagonal procedure and the implicit **LU** factorization will be compared. The diagram in Figure 4.7 shows the pressure and density distribution in case of supersonic flow in entire flow domain, while the diagram in Figure 4.8 illustrates the mentioned distribution in the presence of a strong shock wave.

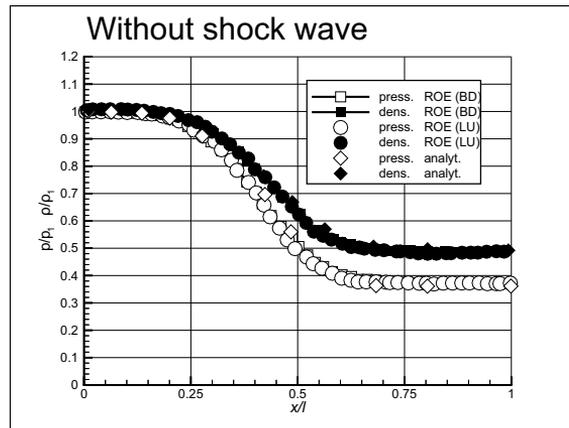


Figure 4.7: Flow in divergent nozzle – supersonic flow.

Qualitative and quantitative matching of numerical solutions with theoretical ones is remarkable in the case of an entirely supersonic flow, which can be concluded by analyzing the diagram on Figure 4.7. When a strong shock wave is formed in the nozzle, as shown on diagram in Figure 4.8, the numerical solutions are almost identical with theoretical in the zone of continuous change, except in the zone of abrupt discontinuity of flow quantities through the shock wave, where the block-diagonal procedure shows some advantage. It is obvious that this quality is paid with the longer computational time.

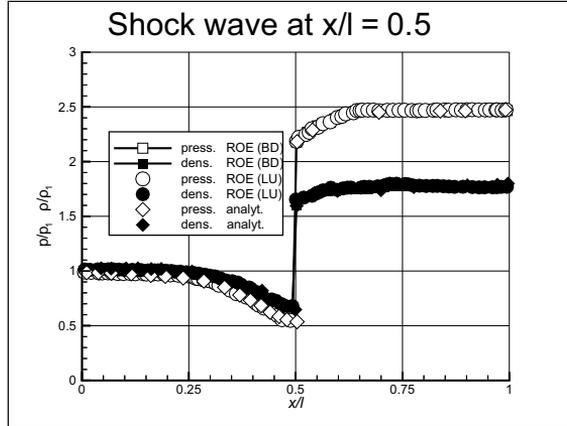


Figure 4.8: Flow in divergent nozzle – presence of a shock wave.

The most significant advantage of the Roe scheme over the procedure exposed in [15] is reflected in the possibility of determination very sharp shock wave without the presence of unwanted oscillation.

### Modification of Godunov scheme

In this chapter, special attention will be paid to the correct implementation of boundary conditions, based on *characteristic variables*, in order to increase accuracy, stability and the convergence rate of the numerical solution.

Three-dimensional Euler equations in non-stationary flow field can be written in Cartesian coordinate system in a conservative form, similar to the system of equations (4.13)

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} + \frac{\partial \mathbf{h}}{\partial z} = 0, \quad (4.63)$$

where  $\mathbf{U}$  is the vector of flow variables

$$\mathbf{U} = (\rho, \rho u, \rho v, \rho w, \rho e_t)^T, \quad (4.64)$$

while  $\mathbf{f}$ ,  $\mathbf{g}$  and  $\mathbf{h}$  are components of the flux vector

$$\mathbf{f} = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho uH \end{Bmatrix}, \quad \mathbf{g} = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ \rho vH \end{Bmatrix} \quad \text{and} \quad \mathbf{h} = \begin{Bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ \rho wH \end{Bmatrix}. \quad (4.65)$$

In the relations (4.64) and (4.65) the quantities  $\rho$ ,  $u$ ,  $v$ ,  $w$ ,  $p$ ,  $e_t$  and  $H$  denote fluid density, velocity vector projections on three coordinate axes, pressure, total energy and stagnation enthalpy per unit fluid mass, respectively.

Quasi-linear form of the equation (4.63) can be written as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{U}}{\partial x} + \mathbf{B} \frac{\partial \mathbf{U}}{\partial y} + \mathbf{C} \frac{\partial \mathbf{U}}{\partial z} = 0, \quad (4.66)$$

where  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$  are three Jacobi flux matrices, determined by derivatives

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{U}}, \quad \mathbf{B} = \frac{\partial \mathbf{g}}{\partial \mathbf{U}} \quad \text{i} \quad \mathbf{C} = \frac{\partial \mathbf{h}}{\partial \mathbf{U}}. \quad (4.67)$$

The Jacobi matrix  $\mathbf{A}$  can be explicitly expressed in the form

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -u^2 + \frac{\gamma-1}{2}q^2 & (3-\gamma)u & -(\gamma-1)v & -(\gamma-1)w & (\gamma-1) \\ -uv & v & u & 0 & 0 \\ -uw & w & 0 & u & 0 \\ -u[\gamma e_t - (\gamma-1)q^2] & \gamma e_t - \frac{\gamma-1}{2}(q^2 + 2u^2) & -(\gamma-1)uv & -(\gamma-1)uw & \gamma u \end{bmatrix}, \quad (4.68)$$

while the matrix  $\mathbf{B}$  is

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ -uv & v & u & 0 & 0 \\ -v^2 + \frac{\gamma-1}{2}q^2 & -(\gamma-1)u & (3-\gamma)v & -(\gamma-1)w & (\gamma-1) \\ -vw & 0 & w & v & 0 \\ -v[\gamma e_t - (\gamma-1)q^2] & -(\gamma-1)uv & \gamma e_t - \frac{\gamma-1}{2}(q^2 + 2v^2) & -(\gamma-1)vw & \gamma v \end{bmatrix}, \quad (4.69)$$

where the matrix  $\mathbf{C}$  is

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ -uw & w & 0 & u & 0 \\ -vw & 0 & w & v & 0 \\ -w^2 + \frac{\gamma-1}{2}q^2 & -(\gamma-1)u & -(\gamma-1)v & (3-\gamma)w & (\gamma-1) \\ -w[\gamma e_t - (\gamma-1)q^2] & -(\gamma-1)uw & -(\gamma-1)vw & \gamma e_t - \frac{\gamma-1}{2}(q^2 + 2w^2) & \gamma w \end{bmatrix}. \quad (4.70)$$

In the expressions (4.68)-(4.70) the variable  $q$  represents the intensity of the fluid velocity.

By introducing the vector of *basic variables*  $\mathbf{V} = (\rho, u, v, w, p)^T$  a system of Euler equations receives the form

$$\frac{\partial \mathbf{V}}{\partial t} + \tilde{\mathbf{A}} \frac{\partial \mathbf{V}}{\partial x} + \tilde{\mathbf{B}} \frac{\partial \mathbf{V}}{\partial y} + \tilde{\mathbf{C}} \frac{\partial \mathbf{V}}{\partial z} = 0, \quad (4.71)$$

where  $\tilde{\mathbf{A}}$ ,  $\tilde{\mathbf{B}}$  and  $\tilde{\mathbf{C}}$  are the corresponding Jacobi matrices. Matrices  $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$  can be

explicitly written in form

$$\tilde{\mathbf{A}} = \begin{bmatrix} u & \rho & \cdot & \cdot & \cdot \\ \cdot & u & \cdot & \cdot & 1/\rho \\ \cdot & \cdot & u & \cdot & \cdot \\ \cdot & \cdot & \cdot & u & \cdot \\ \cdot & \rho c^2 & \cdot & \cdot & u \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} v & \cdot & \rho & \cdot & \cdot \\ \cdot & v & \cdot & \cdot & \cdot \\ \cdot & \cdot & v & \cdot & 1/\rho \\ \cdot & \cdot & \cdot & v & \cdot \\ \cdot & \cdot & \rho c^2 & \cdot & v \end{bmatrix} \quad \text{and} \quad (4.72)$$

$$\tilde{\mathbf{C}} = \begin{bmatrix} w & \cdot & \cdot & \rho & \cdot \\ \cdot & w & \cdot & \cdot & \cdot \\ \cdot & \cdot & w & \cdot & \cdot \\ \cdot & \cdot & \cdot & w & 1/\rho \\ \cdot & \cdot & \cdot & \rho c^2 & w \end{bmatrix},$$

whereby it is obvious that their structure is far away simpler than the structure of Jacobi matrices  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$ , specified by relations (4.68)-(4.70).

### Characteristic variables in one-dimensional flow

One-dimensional flows play a very important role in calculation and analysis of solutions of Euler equations. They are simple enough to allow detailed analysis of nonlinear effects of wave propagation, allowing in many cases local application of one-dimensional flow properties in multiple spatial dimensions flows. Local application of a one-dimensional concept in defining boundary conditions is a very important consequence of introduction of one-dimensional characteristics.

The most general case is described as quasi one-dimensional flow in a nozzle of variable cross section area  $S(x)$ , where  $x$  is the distance measured along the nozzle axis. Conservative form of Euler equations becomes [42]

$$\begin{aligned} \frac{\partial}{\partial t}(\rho S) + \frac{\partial}{\partial x}(\rho u S) &= 0, \\ \frac{\partial}{\partial t}(\rho u S) + \frac{\partial}{\partial x}(\rho u^2 + p)S &= p \frac{dS}{dx}, \\ \frac{\partial}{\partial t}(\rho e_t S) + \frac{\partial}{\partial x}(\rho u H S) &= 0. \end{aligned} \quad (4.73)$$

By introducing *basic (nonconservative) variables*  $\rho$ ,  $u$  and  $p$ , the system (4.73) is transformed into

$$\begin{aligned} \frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + \rho \frac{\partial u}{\partial x} &= -\frac{\rho u}{S} \frac{dS}{dx}, \\ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{1}{\rho} \frac{\partial p}{\partial x} &= 0, \\ \frac{\partial p}{\partial t} + u \frac{\partial p}{\partial x} + \rho c^2 \frac{\partial u}{\partial x} &= -\frac{\rho u c^2}{S} \frac{dS}{dx}, \end{aligned} \quad (4.74)$$

where  $c$  is the local speed of sound. By defining the source vector  $\tilde{\mathbf{Q}}$ ,

$$\tilde{\mathbf{Q}} = \left\{ \begin{array}{c} -\rho u \\ 0 \\ -\rho c^2 u \end{array} \right\} \frac{1}{S} \frac{dS}{dx}, \quad (4.75)$$

equations (4.74) can be written in a more compact form by introducing the vector of *nonconservative variables*  $\mathbf{V} = (\rho, u, p)^T$

$$\frac{\partial \mathbf{V}}{\partial t} + \tilde{\mathbf{A}} \frac{\partial \mathbf{V}}{\partial x} = \tilde{\mathbf{Q}}. \quad (4.76)$$

Jacobi transformation matrix from *conservative* into *nonconservative variables* is defined by the derivative

$$\mathbf{M} = \frac{\partial \mathbf{U}}{\partial \mathbf{V}}. \quad (4.77)$$

Relation between Jacobi matrix  $\mathbf{A}$  of *conservative variables* and matrix  $\tilde{\mathbf{A}}$ , associated with *nonconservative variables*, can be established using the matrix  $\mathbf{M}$  and its inverse matrix  $\mathbf{M}^{-1}$ . Introduction of Jacobi matrix  $\mathbf{M}$  into the equation (4.76) leads to the relation

$$\mathbf{M} \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{M} \tilde{\mathbf{A}} \mathbf{M}^{-1}) \mathbf{M} \frac{\partial \mathbf{V}}{\partial x} = \mathbf{M} \tilde{\mathbf{Q}}. \quad (4.78)$$

Comparing the equation (4.78) with the corresponding equation

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{U}}{\partial x} = \mathbf{Q} \quad (4.79)$$

for *conservative variables* leads to relationships

$$\tilde{\mathbf{A}} = \mathbf{M}^{-1} \mathbf{A} \mathbf{M} \quad \text{and} \quad \mathbf{A} = \mathbf{M} \tilde{\mathbf{A}} \mathbf{M}^{-1}, \quad (4.80)$$

whereby a source member can be written

$$\tilde{\mathbf{Q}} = \mathbf{M}^{-1} \mathbf{Q}. \quad (4.81)$$

In the case of one-dimensional flow transformation matrix  $\mathbf{M}$  has the form

$$\mathbf{M} = \begin{bmatrix} 1 & \cdot & \cdot \\ u & \rho & \cdot \\ u^2/2 & \rho u & 1/(\gamma-1) \end{bmatrix}, \quad (4.82)$$

while Jacobi matrices  $\mathbf{A}$  and  $\tilde{\mathbf{A}}$  can be expressed by relations

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ -(3-\gamma) \frac{u^2}{2} & (3-\gamma)u & (\gamma-1) \\ (\gamma-1)u^3 - \gamma e_t u & \gamma e_t - 3(\gamma-1) \frac{u^2}{2} & \gamma u \end{bmatrix} \quad \text{and} \quad (4.83)$$

$$\tilde{\mathbf{A}} = \begin{bmatrix} u & \rho & \cdot \\ \cdot & u & 1/\rho \\ \cdot & \rho c^2 & u \end{bmatrix}.$$

If  $\lambda^{(j)}$  is the eigenvalue of the matrix  $\tilde{\mathbf{A}}$ , ie. root of the equation

$$\det|\lambda\mathbf{I}-\tilde{\mathbf{A}}| = 0, \quad (4.84)$$

then the left eigenvectors  $\tilde{\ell}^{(j)}$ , defined as row vectors in the three-dimensional space of the vector  $\mathbf{V}$ , are the solution of the equation

$$\tilde{\ell}^{(j)}\tilde{\mathbf{A}} = \lambda^{(j)}\tilde{\ell}^{(j)}. \quad (4.85)$$

From the equation (4.85) can be concluded that there exists a matrix  $\mathbf{L}^{-1}$  that diagonalizes matrix  $\tilde{\mathbf{A}}$ . By defining the matrix  $\mathbf{L}^{-1}$ , whose rows are left eigenvectors  $\tilde{\ell}^{(j)}$ , ie.  $j$ th row of matrix  $\mathbf{L}^{-1}$  is the left eigenvector  $\tilde{\ell}^{(j)}$ , the equation (4.85), after grouping all its eigenvectors, can be written in the form

$$\mathbf{L}^{-1}\tilde{\mathbf{A}} = \Lambda\mathbf{L}^{-1}, \quad (4.86)$$

where  $\Lambda$  is the diagonal matrix of all eigenvalues of matrix  $\tilde{\mathbf{A}}$

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}. \quad (4.87)$$

Based on the equations (4.86) and (4.87) the relation between the Jacobi matrix  $\tilde{\mathbf{A}}$  and diagonal matrix  $\Lambda$  can be determined

$$\tilde{\mathbf{A}} = \mathbf{L}\Lambda\mathbf{L}^{-1} \quad \text{and} \quad \Lambda = \mathbf{L}^{-1}\tilde{\mathbf{A}}\mathbf{L}. \quad (4.88)$$

By introducing the matrix  $\mathbf{L}^{-1}$  and its inverse  $\mathbf{L}$  it is possible to write the compatibility equation in a more compact form after multiplication of the equation (4.76) by the matrix  $\mathbf{L}^{-1}$

$$(\mathbf{L}^{-1}\partial_t + \mathbf{L}^{-1}\tilde{\mathbf{A}}\partial_x)\mathbf{V} = \mathbf{L}^{-1}\tilde{\mathbf{Q}}. \quad (4.89)$$

Equation (4.89) introduces a new set of *characteristic variables*. These variables are defined as a column vector  $\mathbf{W}$  by the relation that is valid for an arbitrary variations  $\delta$

$$\delta\mathbf{W} = \mathbf{L}^{-1}\delta\mathbf{V}. \quad (4.90)$$

The roots of the equation (4.84) determine eigenvalues of the matrix  $\tilde{\mathbf{A}}$

$$\lambda_1 = u, \quad \lambda_2 = u + c \quad \text{and} \quad \lambda_3 = u - c, \quad (4.91)$$

while the three eigenvectors of the matrix  $\tilde{\mathbf{A}}$  form diagonalizing matrix, which after normalization becomes

$$\mathbf{L}^{-1} = \begin{bmatrix} 1 & 0 & -1/c^2 \\ 0 & 1 & 1/\rho c \\ 0 & 1 & -1/\rho c \end{bmatrix}, \quad \text{ie.} \quad \mathbf{L} = \begin{bmatrix} 1 & \rho/2c & -\rho/2c \\ 0 & 1/2 & 1/2 \\ 0 & \rho c/2 & -\rho c/2 \end{bmatrix}. \quad (4.92)$$

Characteristic form of Euler equations in a one-dimensional flow field, written in variables  $\mathbf{W}$ , implies an uncoupled system of equations

$$\frac{\partial \mathbf{W}}{\partial t} + \Lambda \frac{\partial \mathbf{W}}{\partial x} = \mathbf{L}^{-1} \tilde{\mathbf{Q}}, \tag{4.93}$$

bearing in mind that variations of *characteristic variables*

$$\delta w_1 = \delta \rho - \frac{1}{c^2} \delta p, \quad \delta w_2 = \delta u + \frac{1}{\rho c} \delta p \quad \text{and} \quad \delta w_3 = \delta u - \frac{1}{\rho c} \delta p \tag{4.94}$$

extend along the corresponding characteristics at velocities  $u, u + c$  and  $u - c$ , respectively.

In the case of a three-dimensional flow variations of characteristic variables become:

$$\begin{aligned} \delta w_1 &= \delta \rho - \frac{1}{c^2} \delta p, & \delta w_2 &= \hat{k}_x \delta w - \hat{k}_z \delta u, & \delta w_3 &= \hat{k}_y \delta u - \hat{k}_x \delta v, \\ \delta w_4 &= \mathbf{l}_k \delta \mathbf{v} + \frac{1}{\rho c} \delta p, & \delta w_5 &= -\mathbf{l}_k \delta \mathbf{v} + \frac{1}{\rho c} \delta p, \end{aligned} \tag{4.95}$$

where  $u, v$  and  $w$  are projections of the velocity vector  $\mathbf{v}$  on the axes of the Cartesian coordinate system. Quantities  $\hat{k}_x, \hat{k}_y$  and  $\hat{k}_z$  in expressions (4.95) denote the projections of the unit vector  $\mathbf{l}_k$  on corresponding coordinate axes, where the vector  $\mathbf{l}_k$  determines the direction of wave propagation.

**Boundary conditions**

The considerations in the chapter 4.2.3 have a direct influence on the number of boundary conditions imposed on the physical boundaries in the one-dimensional non-viscous flow of the ideal fluid. The number of boundary conditions directly depends on the direction of wave propagation in a fluid along the characteristics at the boundaries of the flow domain. The problem will be well posed if complete information on characteristics directed towards the flow domain or oppositely oriented characteristics can be obtained by the appropriate combination of known *conservative* or *basic (nonconservative) variables*.

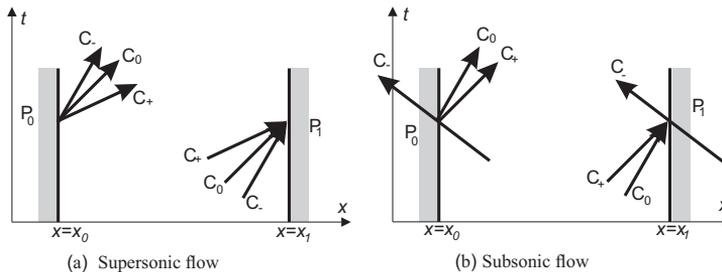


Figure 4.9: Boundary conditions in one-dimensional flow.

Number of physical boundary conditions, ie. variables, to be imposed at the boundaries of the flow domain depends on direction of propagating waves, ie. on the way the

information is transported from the boundary to the interior of the flow domain. At the entry point  $P_0$  the characteristics  $C_0$  and  $C_+$  have slopes  $u$  and  $u + c$  which are always positive for flow in positive direction of the axis  $x$ . The third characteristic  $C_-$  has the slope  $u - c$  which sign depends on the Mach number on the inflow domain boundary. Characteristics in supersonic flow are shown in Figure 4.9.a for the case of one-dimensional flow in the nozzle, while their disposition in the subsonic flow field is illustrated in Figure 4.9.b.

If the transformation matrix  $\mathbf{L}^{-1}$  of variables  $\mathbf{W}$  and  $\mathbf{V}$ , defined by the first relation (4.92), is written so that the set of variables that match physical boundary conditions is separated from the remaining variables that correspond to numerical boundary conditions, as described in [30] and [48], there must be enough information about the wave propagation along the characteristics to determine the required quantities inside the domain. The relation between the variables  $\mathbf{W}$  and  $\mathbf{V}$  can be written in the form

$$\Delta \mathbf{W} = \begin{Bmatrix} \Delta w_3 \\ \Delta w_1 \\ \Delta w_2 \end{Bmatrix} = \begin{bmatrix} -1/\rho c & 0 & 1 \\ -1/c^2 & 1 & 0 \\ 1/\rho c & 0 & 1 \end{bmatrix} \begin{Bmatrix} \Delta p \\ \Delta \rho \\ \Delta u \end{Bmatrix}. \quad (4.96)$$

Denoting by  $\Delta \mathbf{W}^F$  characteristic variables associated with physical boundary conditions and by  $\Delta \mathbf{W}^N$  remaining variables associated with numeric boundary conditions that provide information from the interior of the domain toward to boundaries, the equation (4.96) becomes

$$\Delta \mathbf{W} = \begin{Bmatrix} \Delta \mathbf{W}^F \\ \Delta \mathbf{W}^N \end{Bmatrix} = \begin{bmatrix} (\mathbf{L}^{-1})_I^F & (\mathbf{L}^{-1})_II^F \\ (\mathbf{L}^{-1})_I^N & (\mathbf{L}^{-1})_II^N \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{V}^I \\ \Delta \mathbf{V}^{II} \end{Bmatrix}. \quad (4.97)$$

The variables  $\mathbf{V}^I$  in the equation (4.97) represent imposed physical boundary conditions, while a group of variables  $\mathbf{V}^{II}$  denotes variables determined by numerical boundary conditions. In case of subsonic outlet  $\mathbf{W}^F = w_3$ , while at subsonic inflow  $\mathbf{W}^F$  consists of the variables  $w_1$  and  $w_2$ . All combinations of *conservative* and *nonconservative variables* may determine physical boundary conditions, except couple  $(u, p)$  in the subsonic inflow [14]. In the case of supersonic boundaries, for the problem to be well posed, it is necessary to define all variables at the inflow, while at the outflow it is not allowed to specify any variable, as shown in Figure 4.9.a.

Numeric variables  $\Delta \mathbf{W}^N$  at outflow are determined by extrapolating the values from interior points. For the zero order extrapolation, the following relation is valid:

$$\Delta \mathbf{W}^N|_M = \Delta \mathbf{W}^N|_{M-1}, \quad (4.98)$$

where  $M$  is total number of grid points. At the inflow boundary conditions are determined by similar relations

$$\Delta \mathbf{W}^N|_1 = \Delta \mathbf{W}^N|_2. \quad (4.99)$$

In the expressions (4.98) and (4.99) the operator  $\Delta$  indicates the corresponding increment in time.

Implicit difference scheme, represented by the system of equations (4.60), can now be supplemented by boundary conditions based on equation (4.98), which for the subsonic

outflow and inflow become

$$\begin{aligned}
 \Delta \mathbf{U}_{M-\frac{1}{2}} &= \mathbf{M}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta \rho \\ \Delta u \\ 0 \end{Bmatrix}_{M-\frac{1}{2}} = \mathbf{M}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta w_1 \\ \Delta w_2 \\ 0 \end{Bmatrix}_{M-\frac{3}{2}} = \\
 &= \mathbf{M}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta p/c^2 + \Delta \rho \\ \Delta u + \Delta p/\rho c u \\ 0 \end{Bmatrix}_{M-\frac{3}{2}}, \quad (4.100) \\
 \Delta \mathbf{U}_{\frac{1}{2}} &= \mathbf{M}_{\frac{1}{2}} \begin{Bmatrix} 0 \\ \Delta u \\ 0 \end{Bmatrix}_{\frac{1}{2}} = \mathbf{M}_{\frac{1}{2}} \begin{Bmatrix} 0 \\ \Delta w_3 \\ 0 \end{Bmatrix}_{\frac{3}{2}} = \mathbf{M}_{\frac{1}{2}} \begin{Bmatrix} 0 \\ \Delta u - \Delta p/\rho c \\ 0 \end{Bmatrix}_{\frac{3}{2}},
 \end{aligned}$$

while in the case of a supersonic outflow all three numerical boundary conditions are extrapolated from known values at interior points, which is illustrated in Figure 4.9.a, ie.

$$\begin{aligned}
 \Delta \mathbf{U}_{M-\frac{1}{2}} &= \mathbf{M}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta \rho \\ \Delta u \\ \Delta p \end{Bmatrix}_{M-\frac{1}{2}} = \mathbf{M}_{M-\frac{1}{2}} \mathbf{L}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta w_1 \\ \Delta w_2 \\ \Delta w_3 \end{Bmatrix}_{M-\frac{3}{2}} = \\
 &= \mathbf{M}_{M-\frac{1}{2}} \mathbf{L}_{M-\frac{1}{2}} \begin{Bmatrix} \Delta p/c^2 + \Delta \rho \\ \Delta u + \Delta p/\rho c \\ \Delta u - \Delta p/\rho c \end{Bmatrix}_{M-\frac{3}{2}}. \quad (4.101)
 \end{aligned}$$

The system of equations (4.60) can be solved using a standard block-diagonal procedure, based on the inversion of third-order matrices in case of one-dimensional flow. Two-dimensional problems or the three-dimensional flow of a non-viscous fluid is solved using a **ADI** scheme, which is unfortunately difficult to achieve, especially in three-dimensional flow.

To increase efficiency of **2D** and **3D** flow analysis **ADI** scheme, based on block-diagonal matrix inversion, is replaced by **LU** implicit factorization, explained in the chapter 4.2.2 and literature [20], [21], [31] and [32].

Verification of the presented procedure will be performed by numerical analysis of stationary non-viscous air flow through a nozzle of length  $\ell = 10\text{m}$ , whose cross section area varies according to the law

$$S(x) = 1.398 + 0.347 \tanh(0.8x - 4), \quad (4.102)$$

with supersonic inflow and supersonic/subsonic outflow boundary. The conditions at the nozzle inlet are defined as follows:

$$\rho_1 = 1.22112 \text{ kg/m}^3, \quad u_1 = 351.4351 \text{ m/s} \quad i \quad p_1 = 47.87837 \text{ KPa}. \quad (4.103)$$

In the case of a subsonic outflow boundary it is necessary to define only one physical boundary condition, ie.

$$p_i = 117.7617 \text{ KPa}. \quad (4.104)$$

In the presented calculation results, convergence is considered to be achieved if the residual term is reduced in size of fourth order relative to the initial.

Courant number **CFL**, used in numeric analysis, is determined by relation

$$\mathbf{CFL} = (u + c)_{\max} \frac{\Delta t}{\Delta x}, \quad (4.105)$$

in which  $\Delta t$  is the time step of integration, while  $\Delta x$  determines the spatial dimension of the computational grid.

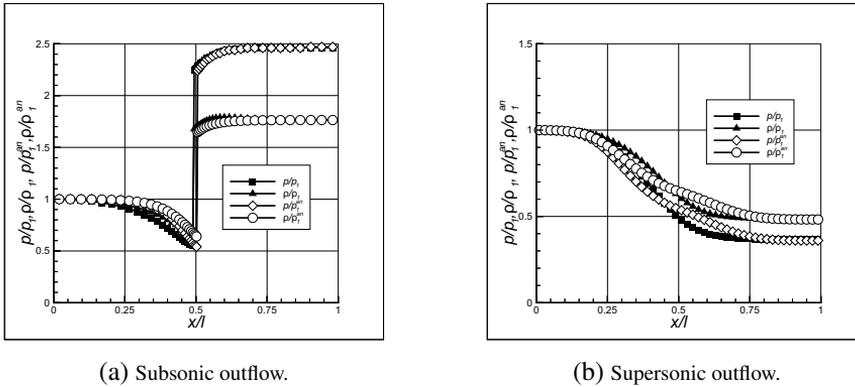


Figure 4.10: Pressure and density distribution in the nozzle.

In the Figure 4.10 the results of numerical analysis of one-dimensional flow in divergent nozzle are shown. Described method, based on zero order extrapolation of *characteristic variables*, increases accuracy of results compared to results achieved by applying the numerical scheme [15]. Very strong shock wave is present at a distance of  $x/\ell = 0.5$ , measured along nozzle axes, in the case of a subsonic outflow boundary, which is shown in Figure 4.10.a. In the case of supersonic outflow boundary numerical solution very well matches the analytical one [17] (Figure 4.10.b).

In addition to the improvements shown, modified Roe “averaged” scheme, based on finite volume method and boundary conditions applied to *characteristic variables*, increases the stability of the numerical scheme with significant increasing of the convergence rate compared to the original method [29], described in the chapter 4.2.3, where the boundary conditions are based on the extrapolation of *nonconservative variables*. Presented procedure allows the introduction of significantly larger values of Courant number (**CFL**) into a numeric procedure.

In order to properly estimate the influence of integration steps on convergence rate, different combinations of  $\Delta x$  and  $\Delta t$  were used, which determine the value of the Courant number, according to the equation (4.105). Intensive changes are noticed after the first few iteration, followed by a gradually convergence of numerical solution to a stationary one. Convergence rate and the size of the residual term for the subsonic and supersonic outflow boundary are shown in Figures 4.11, 4.12, 4.13 and 4.14, respectively.

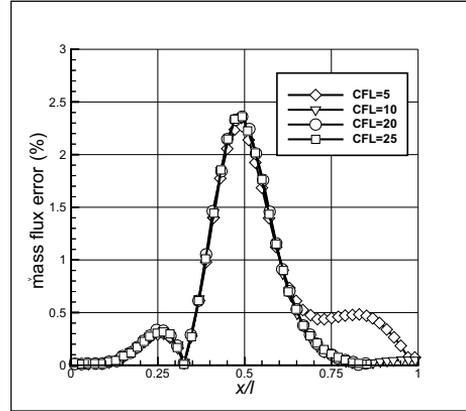
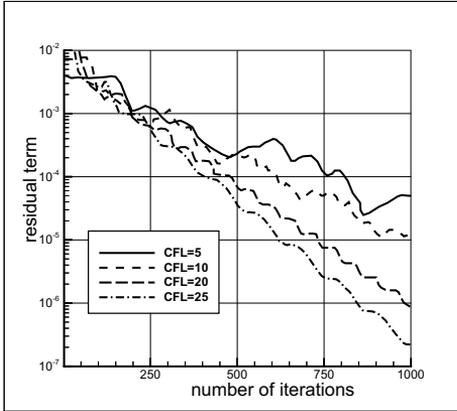


Figure 4.11: Convergence rate - supersonic outflow. Figure 4.12: Mass flux error – supersonic outflow.

In the case of a supersonic outflow, a shock wave does not occur in the flow field inside the nozzle and nonlinear effects do not appear significantly. Stability analysis of system of linear equations does not introduce a step integration constraints in the application of implicit numerical schemes, and allows the implementation of relatively large integration steps and corresponding numbers **CFL** to obtain convergent solution. The above conclusions are confirmed by the Figure 4.11. Distribution of relative mass flux error in the continuity equation is given along the nozzle axis in the Figure 4.12. Absolute value of the relative error increases slightly in the zones of more intensive variations of flow quantities, while close to inflow and outflow boundaries is almost negligible, since the change of the nozzle cross section area in that zone is less evident.

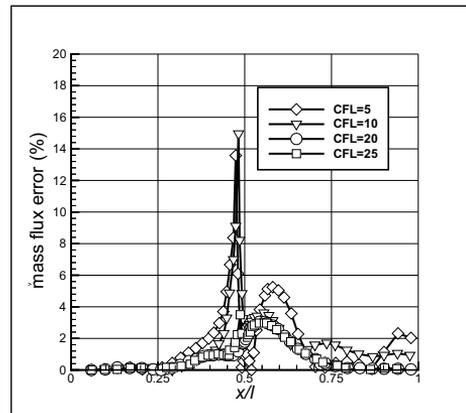
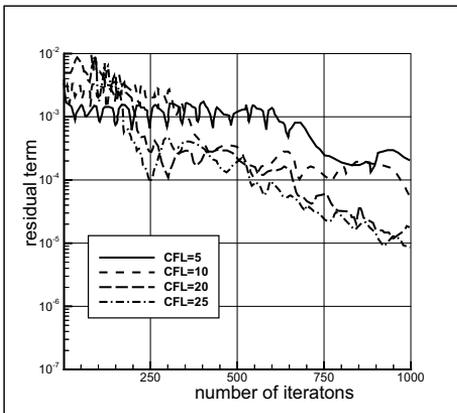


Figure 4.13: Convergence rate – subsonic outflow. Figure 4.14: Mass flux error – subsonic outflow.

As in the case of a smooth solution in a supersonic nozzle flow without the presence of a shock wave, number of iterations, required for the convergence of the discontinuous solution, decreases with increasing Courant number **CFL**.

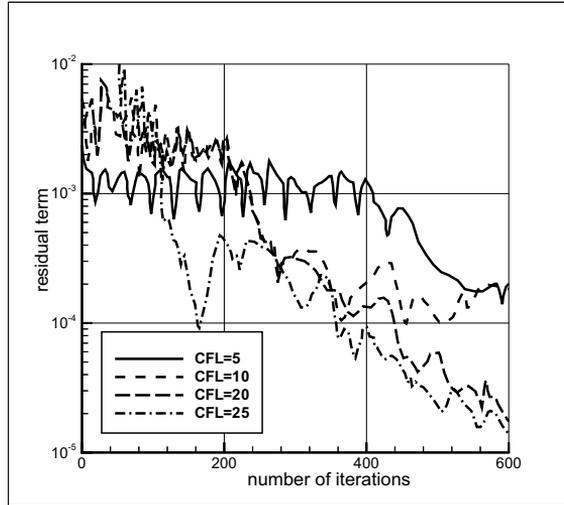


Figure 4.15: Convergence rate – subsonic outflow.

For transonic flow with shock wave arising (Figure 4.13), number of the required iterations needed to obtain a stationary solution is greater than in the case of completely supersonic flow without the presence of discontinuous changes of flow quantities. Residual member representing the mass flux error in the continuity equation after 1000 iterations for each Courant number is significantly larger compared to the corresponding value at entirely supersonic flow, which is also shown on the Figure 4.14. Numerical scheme error especially increases in the immediate vicinity of shock wave. Numerical solution error reduction is achieved by increasing the Courant number, whereby it should be noted that large values of the number **CFL** can reduce the stability of the numerical scheme.

Formation method of Jacobi matrices  $\mathbf{A}^+$  and  $\mathbf{A}^-$  in the relations (4.57) and (4.58) also has a significant influence on the convergence of the numerical solution of the system of equations (4.60). Jacobi matrices obtained by flux vector decomposition increase the convergence rate of the numerical solution, making it at the same time more robust than the solution based on matrix formation, used in the Roe scheme. The cause of this lies in the fact that Jacobi matrices are determined by derivatives of the positive and negative flux vectors and in the general differ from “decomposed” Jacobi matrices. Having in mind the equation (4.59), it can be written

$$\mathbf{f} = \mathbf{A}\mathbf{U} = \mathbf{A}^+\mathbf{U} + \mathbf{A}^-\mathbf{U} = \mathbf{f}^+ + \mathbf{f}^- \quad (4.106)$$

and

$$\frac{\partial \mathbf{f}}{\partial \mathbf{U}} = \mathbf{A} = \frac{\partial \mathbf{f}^+}{\partial \mathbf{U}} + \frac{\partial \mathbf{f}^-}{\partial \mathbf{U}} = \mathbf{A}^+ + \mathbf{A}^-, \quad (4.107)$$

taking into account at the same time that the mentioned Jacobi matrices differ, ie.

$$\bar{\mathbf{A}}^+ = \frac{\partial \mathbf{f}^+}{\partial \mathbf{U}} \neq \mathbf{A}^+ \quad \text{and} \quad \bar{\mathbf{A}}^- = \frac{\partial \mathbf{f}^-}{\partial \mathbf{U}} \neq \mathbf{A}^-. \quad (4.108)$$

“Decomposed” flux vectors  $\mathbf{f}^+$  and  $\mathbf{f}^-$ , present in equations (4.106) and (4.107), are determined by the expressions:

$$\mathbf{f}^+ = \frac{\rho}{2\gamma} \left\{ \begin{array}{l} 2\gamma u + c - u \\ 2(\gamma - 1)u^2 + (u + c)^2 \\ (\gamma - 1)u^3 + \frac{1}{2}(u + c)^3 + \frac{1}{2} \frac{(3 - \gamma)(u + c)c^2}{(\gamma - 1)} \end{array} \right\} \quad \text{and} \quad (4.109)$$

$$\mathbf{f}^- = \frac{\rho}{2\gamma} \left\{ \begin{array}{l} 2\gamma u - c \\ (u - c)^2 \\ \frac{1}{2}(u - c)^3 + \frac{1}{2} \frac{(3 - \gamma)(u - c)c^2}{(\gamma - 1)} \end{array} \right\}.$$

It is useful to note that, in addition to the the flux vector decomposition approach, determined by the equations (4.109), alternative methods exist [47].

Jacobi flux matrices  $\bar{\mathbf{A}}^+$  and  $\bar{\mathbf{A}}^-$ , as derivatives of the flux vectors  $\mathbf{f}^+$  and  $\mathbf{f}^-$ , are determined by application **MATHEMATICA**<sup>TM</sup> [53].

The Jacobi matrix  $\bar{\mathbf{A}}^+$  thus becomes

$$\frac{\partial \mathbf{f}^+}{\partial \mathbf{U}} = \begin{bmatrix} \frac{2c^2 - \gamma u^2 + \gamma^2 u^2}{8c\gamma} & \frac{-2c + 4c\gamma + \gamma u - \gamma^2 u}{4c\gamma} & \frac{-1 + \gamma}{4c} \\ \dots & \dots & \dots \\ \frac{(2 - 5\gamma + \gamma^2)u^2}{4\gamma} + \frac{(-1 + \gamma)u^3}{4c} - \frac{cu}{2\gamma} & \frac{(-2 + 5\gamma - \gamma^2)u}{2\gamma} + \frac{(1 - \gamma)u^2}{2c} + \frac{c}{\gamma} & \frac{-1 + \gamma}{2} + \frac{(-1 + \gamma)u}{2c} \\ \dots & \dots & \dots \\ \frac{c^3}{4(1 - \gamma)\gamma} + \frac{c^2 u}{2(1 - \gamma)} + \frac{3c(-3 + \gamma)u^2}{8\gamma} & \frac{c^2}{2(-1 + \gamma)} + \frac{3c(2 - \gamma)u}{4\gamma} & \frac{3c}{4} + \frac{\gamma u}{2} \\ \frac{(2 - 4\gamma + \gamma^2)u^3}{4\gamma} + \frac{3(-1 + \gamma)u^4}{16c} & \frac{(-3 + 6\gamma - 2\gamma^2)u^2}{4\gamma} + \frac{3(1 - \gamma)u^3}{8c} & \frac{3(-1 + \gamma)u^2}{8c} \end{bmatrix}, \quad (4.110)$$

while the matrix  $\overline{\mathbf{A}}^-$  is:

$$\frac{\partial \mathbf{f}^-}{\partial \mathbf{U}} = \begin{bmatrix} \frac{-2c^2 + \gamma u^2 - \gamma^2 u^2}{8c\gamma} & \frac{2c - \gamma u + \gamma^2 u}{4c\gamma} & \frac{1 - \gamma}{4c} \\ \dots & \dots & \dots \\ \frac{(c - u)u(2c - \gamma u + \gamma^2 u)}{4c\gamma} & \frac{(-c + u)(2c - \gamma u + \gamma^2 u)}{2c\gamma} & \frac{(1 - \gamma)(-c + u)}{2c} \\ \dots & \dots & \dots \\ \frac{c^3}{4(-1 + \gamma)\gamma} + \frac{c^2 u}{2(1 - \gamma)} + \frac{3c(3 - \gamma)u^2}{8\gamma} + \frac{(-2 + \gamma^2)u^3}{4\gamma} + \frac{3(1 - \gamma)u^4}{16c} & \frac{c^2}{2(-1 + \gamma)} + \frac{3c(-2 + \gamma)u}{4\gamma} + \frac{(3 - 2\gamma^2)u^2}{4\gamma} + \frac{3(-1 + \gamma)u^3}{8c} & \frac{-3c}{4\gamma u} + \frac{2}{3(1 - \gamma)u^2} \end{bmatrix} \quad (4.111)$$

It is now possible to code an alternative computer program, based on Jacobi matrices (4.110) and (4.111). When the flow field is entirely supersonic, the matrix  $\mathbf{A}^-$  vanishes, so there is no increase in the convergence rate compared to the method when decomposed flux matrices were used. In the presence of a sharp shock wave that separates the zones of subsonic and supersonic flow in the nozzle a significant increase in the rate of convergence is observed, which is shown in Figure 4.15, with almost unchanged pressure distribution along the nozzle axis.

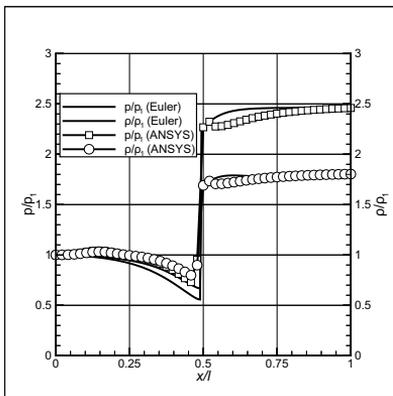


Figure 4.16: Pressure and density distribution.

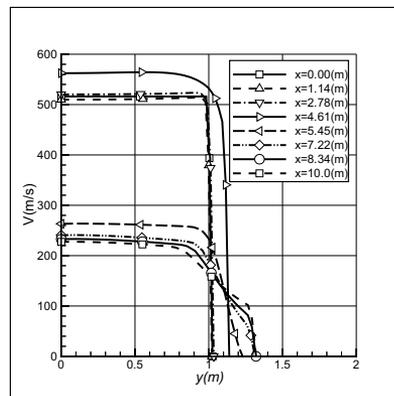


Figure 4.17: Velocity profiles.

In order to estimate influence of viscosity on the nozzle flow, the flow calculation in the axisymmetric nozzle whose cross-sectional area is defined by law (4.102) was performed, applying the Prandtl turbulent model [43] and [23]. The results for the same flow conditions are obtained using commercial ANSYS package and are illustrated in Figures 4.16 and 4.17. Pressure distribution on the nozzle axis (Figure 4.16) shows very well coincidence with the analytical solution of Euler equations, thus with the method presented in this chapter. At the nozzle inlet a uniform pressure and velocity distribution is given, which is held almost to the location of the shock wave. In the downstream flow, due to present viscous effects, there is a redistribution, which is shown in Figure 4.17.

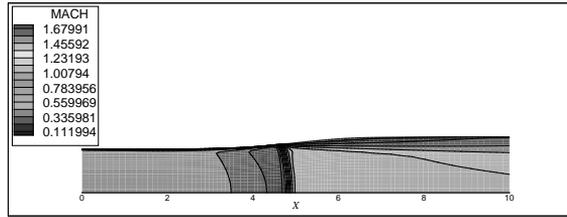


Figure 4.18: Mach number at viscous flow.

After crossing the shock wave the fluid flow is subsonic, and due to the increase of pressure in the flow direction, thickness of the boundary layer increases and flow separates from nozzle walls. The distribution of Mach number within the nozzle is given in Figure 4.18.

To generalize the procedure for three-dimensional flow, it is necessary to form matrices **A**, **B** and **C**, defined by expressions (4.68)-(4.70), whose elements will be calculated as follows:

$$\begin{aligned} \tilde{\rho}^2 &= \rho_L \rho_D, \quad \tilde{u} = \frac{\rho_L^{1/2} u_L + \rho_D^{1/2} u_D}{\rho_L^{1/2} + \rho_D^{1/2}}, \quad \tilde{v} = \frac{\rho_L^{1/2} v_L + \rho_D^{1/2} v_D}{\rho_L^{1/2} + \rho_D^{1/2}}, \\ \tilde{w} &= \frac{\rho_L^{1/2} w_L + \rho_D^{1/2} w_D}{\rho_L^{1/2} + \rho_D^{1/2}}, \quad \tilde{H} = \frac{\rho_L^{1/2} H_L + \rho_D^{1/2} H_D}{\rho_L^{1/2} + \rho_D^{1/2}}, \end{aligned} \quad (4.112)$$

where  $\tilde{u}$ ,  $\tilde{v}$  and  $\tilde{w}$  are “averaged” velocity projections, while

$$\tilde{c}^2 = (\gamma - 1) \left[ \tilde{H} - \frac{1}{2} (\tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2) \right]. \quad (4.113)$$

Numerical flux (4.35)-(4.37) in the case of multidimensional flow is calculated on the basis of such “averaged” flow quantities [41].

### Second order Godunov scheme

Direct replacement of terms in a first order numerical scheme with the corresponding terms of the second order accuracy leads to the difficulties associated with the occurrence of oscillations in vicinity of discontinuous changes of flow quantities.

In this chapter the procedure of second-order accuracy numerical scheme [29] formation, without oscillations present, capable of very accurate determination of flow quantities changes in the shock wave vicinity and contact discontinuity, will be outlined. Requirements that the scheme should fulfill in the mentioned cases are based on Godunov concept of *monotonicity*.

Constant value of quantities in numerical solution within each computational cell, which was applied in the original Roe scheme [40], described in chapter 4.2.3, implies the first order accuracy in space discretization. Linear distribution of the numerical solution in the computational cell increases accuracy of spatial discretization to the second order, while quadratic distribution leads to the spatial discretization of the third order accuracy. In numerical solution of Euler equations, based on finite volume method, *mean values* of flow variables are used at the observed time. In the general case distribution of flow variables in the computation cell “*i*” is given by the expression

$$\mathbf{U}(x) = \mathbf{U}_i + \frac{1}{\Delta x}(x - x_i)\delta_i\mathbf{U} + \frac{3k}{2\Delta x^2}\left[(x - x_i)^2 - \frac{\Delta x^2}{12}\right]\delta_i^2\mathbf{U}, \quad (4.114)$$

for  $x_{i-1/2} < x < x_{i+1/2}$ , where  $\mathbf{U}_i$  is *mean value* of quantity  $\mathbf{U}$  in computational cell *i*, ie.

$$\mathbf{U}_i = \frac{1}{\Delta x} \int_{i-1/2}^{i+1/2} \mathbf{U}(x) dx, \quad (4.115)$$

while  $\delta_i\mathbf{U}$  and  $\delta_i^2\mathbf{U}$  are approximations of the first and second derivative of quantity  $\mathbf{U}$ . The constant  $k$  in the expression (4.114) can affect the error of the chosen approximation. For the values  $x = x_i \pm \Delta x/2$  values of flow variables at the boundary of the calculation cell are obtained

$$\mathbf{U}_{i+1/2}^L = \mathbf{U}_i + \left[ \frac{(1-k)}{4}(\mathbf{U}_i - \mathbf{U}_{i-1}) + \frac{(1+k)}{4}(\mathbf{U}_{i+1} - \mathbf{U}_i) \right], \quad (4.116)$$

ie.

$$\mathbf{U}_{i+1/2}^D = \mathbf{U}_{i+1} - \left[ \frac{(1+k)}{4}(\mathbf{U}_{i+1} - \mathbf{U}_i) + \frac{(1-k)}{4}(\mathbf{U}_{i+2} - \mathbf{U}_{i+1}) \right]. \quad (4.117)$$

Analogous to the expressions (4.116) and (4.117) extrapolation of the numerical flux can be performed at the boundary of the calculation cell, which leads to relations

$$\mathbf{f}_{i+1/2}^{+b} = \mathbf{f}_i^+ + \left[ \frac{(1-k)}{4}(\mathbf{f}_i^+ - \mathbf{f}_{i-1}^+) + \frac{(1+k)}{4}(\mathbf{f}_{i+1}^+ - \mathbf{f}_i^+) \right], \quad (4.118)$$

$$\mathbf{f}_{i+1/2}^{-f} = \mathbf{f}_{i+1}^- - \left[ \frac{(1+k)}{4}(\mathbf{f}_{i+1}^- - \mathbf{f}_i^-) + \frac{(1-k)}{4}(\mathbf{f}_{i+2}^- - \mathbf{f}_{i+1}^-) \right]. \quad (4.119)$$

Numerical flux of the second order accuracy, on the boundary of the computational cell, is determined by the expression

$$\begin{aligned} \mathbf{f}_{i+1/2}^{(2)} = & \mathbf{f}_{i+1/2} + \left[ \frac{1-k}{4}(\mathbf{f}_i - \mathbf{f}_{i-1/2}) + \frac{1+k}{4}(\mathbf{f}_{i+1} - \mathbf{f}_{i+1/2}) \right] + \\ & + \left[ \frac{1+k}{4}(\mathbf{f}_i - \mathbf{f}_{i+1/2}) + \frac{1-k}{4}(\mathbf{f}_{i+1} - \mathbf{f}_{i+3/2}) \right], \end{aligned} \quad (4.120)$$

having in mind the relation  $\mathbf{f}_i = \mathbf{f}_i^+ + \mathbf{f}_i^-$ , ie.  $\mathbf{f}_{i+1/2} = \mathbf{f}_i^+ + \mathbf{f}_{i+1}^-$ . The first term on right hand of the equation (4.120) corresponds to the numerical flux of the first order accuracy, while the remaining terms make a correction of a higher order of accuracy.

Solutions of the system of Euler equations can be discontinuous functions, which in mathematical sense satisfy integral form of these equation. However, in addition to discontinuous solutions accompanied by entropy growth in the case of shock waves, there may be such, which are in contradiction with the second principle of thermodynamics. Mentioned solutions, followed by negative entropy increment, must be discarded, ie. such a mechanism that would prevent their emergence and enable obtaining acceptable solutions that satisfy the condition, in literature known as the *entropy condition* [34], has to be introduced. Godunov [11] defined the *monotonic numerical scheme* that prevents oscillations of numerical solution, which is of great importance for obtaining an acceptable solution in the vicinity of the shock wave. In the paper [12] it is shown that the numerical solution of *monotonic scheme* satisfies the entropy condition. Unfortunately, such a solution is of first-order accuracy, which represents a significant limitation in practical application.

A condition weaker than the *monotonicity* is provided by limitation of the *total variation* TV of numerical solution, ie.

$$\text{TV}(\mathbf{U}) \equiv \sum_i |\mathbf{U}_{i+1} - \mathbf{U}_i|. \quad (4.121)$$

The numerical scheme is said to be *total variation diminishing* if the relation is satisfied

$$\text{TV}(\mathbf{U}^{n+1}) \leq \text{TV}(\mathbf{U}^n), \quad (4.122)$$

where the indices  $n + 1$  and  $n$  refer to the two consecutive moments of integration. The condition (4.122) is sufficient to provide the convergence of the numerical solution in the case of higher order accuracy schemes, but in contrast to the condition of *monotonicity* does not provide satisfaction of *entropy condition*. Mentioned condition (4.122) prevents the arising of new local extremes while reducing values of existing local maximum and increasing local minimum of the numerical solution.

Fulfillment of conditions (4.122) is provided by introduction of *nonlinear limiters*, whose basic purpose is to limit abrupt changes of additional terms in numerical flux (4.120). One of the possible choices of *nonlinear limiters*  $\Psi$  leads to the expression for the numerical flux

$$\begin{aligned} \mathbf{f}_{i+1/2}^{(2)} = & \mathbf{f}_{i+1/2} + \sum_{j=1}^m \left[ \frac{1-k}{4} \Psi_j^+(-1, 1) \mathbf{r}_{i-1/2}^{(j)} + \frac{1+k}{4} \Psi_j^+(1, -1) \mathbf{r}_{i+1/2}^{(j)} \right] - \\ & - \sum_{j=1}^m \left[ \frac{1-k}{4} \Psi_j^-(3, 1) \mathbf{r}_{i+3/2}^{(j)} + \frac{1+k}{4} \Psi_j^-(1, 3) \mathbf{r}_{i+1/2}^{(j)} \right], \end{aligned} \quad (4.123)$$

where *limiter*  $\Psi_j^\pm(\ell, n)$  is defined by the expression

$$\Psi_j^\pm(\ell, n) = \text{minmod}(\sigma_{j,i+\ell/2}^\pm, b\sigma_{j,i+n/2}^\pm), \quad (4.124)$$

where is

$$\text{minmod}(x, y) = \text{sign}(x) \max\{0, \min[|x|, y \text{sign}(x)]\}, \quad (4.125)$$

while  $\sigma_{j,i+1/2}^{\pm} = \lambda_{i+1/2}^{\pm(j)} \alpha_{j,i+1/2}$ , bearing in mind that the quantities  $\alpha$ ,  $\lambda$  and  $\mathbf{r}$  were previously defined by the relations (4.38). *Compression parameter*  $b$ , present in the relation (4.124), is defined by the expression  $b = (3 - k)/(1 - k)$ , as in [51].

## 4.3 Solution of Navier–Stokes equations

### 4.3.1 Implicit numerical scheme

The basic limitation in the application of Euler equations is reflected in flow calculation around the wing at large angles of attack, when the viscous effects of the flow cannot be ignored. In cases where occurs flow separation it is necessary to add viscous terms to Euler equations, ie. to model the flow by Navier-Stokes equations. The use of "full" Navier-Stokes equations is a problem due to high memory space requirements, on the one hand, and large CPU time, on the other hand. In this chapter is presented model based on Navier-Stokes approximate equations, in the literature known as the *thin layer* equations. Numeric solution of the equations is based on the application of the **LU** implicit factorization approach in finite volume method with "flux decomposition", described in chapter 4.2.2.

#### LU implicit factorization method

Three-dimensional non-stationary Navier-Stokes equations can be write in Cartesian coordinate system in conservative form

$$\partial_t \mathbf{q} + \partial_x (\mathbf{F} - \mathbf{F}_v) + \partial_y (\mathbf{G} - \mathbf{G}_v) + \partial_z (\mathbf{H} - \mathbf{H}_v) = 0, \quad (4.126)$$

where  $\mathbf{q}$  is a vector of flow variables, defined by the relation (4.14), while  $\mathbf{F}$ ,  $\mathbf{G}$  and  $\mathbf{H}$  are flux vector projections on the three coordinate axes determined by the expressions (4.15). Viscous terms  $\mathbf{F}_v$ ,  $\mathbf{G}_v$  and  $\mathbf{H}_v$ , present in the equation (4.126), are determined as follows:

$$\mathbf{F}_v = \left\{ \begin{array}{c} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u \tau_{xx} + v \tau_{xy} + w \tau_{xz} - Q_x \end{array} \right\},$$

$$\mathbf{G}_v = \left\{ \begin{array}{c} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u \tau_{yx} + v \tau_{yy} + w \tau_{yz} - Q_y \end{array} \right\} \quad \text{and} \quad (4.127)$$

$$\mathbf{H}_v = \left\{ \begin{array}{c} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u \tau_{zx} + v \tau_{zy} + w \tau_{zz} - Q_z \end{array} \right\}.$$

The viscosity stress tensor is determined by the expressions:

$$\begin{aligned}\tau_{xx} &= 2\mu u_x - \frac{2}{3}\mu(u_x + v_y + w_z), & \tau_{xy} &= \tau_{yx} = \mu(u_y + v_x), \\ \tau_{yy} &= 2\mu v_y - \frac{2}{3}\mu(u_x + v_y + w_z), & \tau_{xz} &= \tau_{zx} = \mu(u_z + w_x), \\ \tau_{zz} &= 2\mu w_z - \frac{2}{3}\mu(u_x + v_y + w_z), & \tau_{yz} &= \tau_{zy} = \mu(v_z + w_y),\end{aligned}\quad (4.127.1)$$

while the vector of the thermal flux is:

$$Q_x = -k \frac{\partial T}{\partial x}, \quad Q_y = -k \frac{\partial T}{\partial y} \quad \text{i} \quad Q_z = -k \frac{\partial T}{\partial z}, \quad (4.127.2)$$

where the viscosity coefficient  $\mu$  is a function of temperature  $T$

$$\mu = \mu_\infty \left(\frac{T}{T_\infty}\right)^{0.67}. \quad (4.127.3)$$

The coefficient of thermal conductivity  $k$  is defined by the relation

$$k = \frac{\gamma}{\gamma - 1} \frac{\mu}{\text{Pr}}. \quad (4.127.4)$$

Prandtl number  $\text{Pr}$  in the expression (4.127.4) can be considered constant, ie. value  $\text{Pr} = 0.72$  is accepted, while  $\gamma = 1.4$  is the value of the air adiabatic constant. In the case of turbulent flow the value of the turbulent coefficient  $\mu_t$  is added to the laminar viscosity coefficient  $\mu$  while in the expression for the thermal conductivity coefficient  $k$  the ratio  $\mu/\text{Pr}$  is replaced by sum  $\mu/\text{Pr} + \mu_t/\text{Pr}_t$ , where  $\text{Pr}_t = 0.9$  is Prandtl number for turbulent flow. Coefficient of turbulent viscosity  $\mu_t$  is calculated on the basis of generally accepted Baldwin-Lomax model [4], [2].

After the introduction of the curvilinear coordinate system  $(\xi, \eta, \zeta)$ , the generated grid in physical space is mapped to a rectangle computational grid. If only partial derivatives of flow quantities in viscous terms in the direction normal to body surface are retained, the approximate Navier-Stokes equations are reached, ie. *thin layer* equation are obtained. For that matter, equation (4.126) in the transformed space becomes

$$\partial_\tau \bar{\mathbf{q}} + \partial_\xi \bar{\mathbf{F}} + \partial_\eta \bar{\mathbf{G}} + \partial_\zeta \bar{\mathbf{H}} = \partial_\zeta \bar{\mathbf{H}}_v, \quad (4.126.1)$$

where the quantities  $\bar{\mathbf{q}}$ ,  $\bar{\mathbf{F}}$ ,  $\bar{\mathbf{G}}$  and  $\bar{\mathbf{H}}$  are defined by relations (4.14.1) and (4.15.2),

Viscous term on the right hand of equation (4.126.1) after coordinate transformation takes the form

$$\bar{\mathbf{H}}_v = J \left\{ \begin{array}{l} 0 \\ \mu (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) u_\zeta + \mu/3 (\zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta) \zeta_x \\ \mu (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) v_\zeta + \mu/3 (\zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta) \zeta_y \\ \mu (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) w_\zeta + \mu/3 (\zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta) \zeta_z \\ [\mu/2 (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) (u^2 + v^2 + w^2) \zeta + \\ + \mu/3 (\zeta_x u + \zeta_y v + \zeta_z w) (\zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta) + \\ + k (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) T_\zeta] \end{array} \right\}, \quad (4.127.5)$$

where Jacobi transformation matrix  $J = \partial(x, y, z)/\partial(\xi, \eta, \zeta)$  is calculated from the expression (4.15.6). For the perfect gas system of equations (4.126) is completed by defining fluid energy (4.5), thus finally closing the Navier-Stokes system of equations.

In the system of equations (4.126), ie. (4.126.1) the unknown to be determined is a vector of variable flow quantities  $\bar{\mathbf{q}}$ .

After the physical flow domain is discretized, as described in the chapter 4.2.2, a system of equations that approximates partial differential system of equation (4.126), can be set up. If inside each computational cell  $(i, j, k)$  the value of the dependent variable  $\bar{\mathbf{q}}^n$  is known, approximate form of the system of equations (4.126.1) can be obtained

$$\left\{ \mathbf{I} + \beta \Delta t \left[ \delta_\xi \bar{\mathbf{A}}^n + \delta_\eta \bar{\mathbf{B}}^n + \delta_\zeta (\bar{\mathbf{C}}^n - \bar{\mathbf{C}}_v^n) \right] \right\} \Delta \bar{\mathbf{q}}^n + \Delta t \bar{\mathbf{R}}^n = 0, \quad (4.128)$$

where the matrices  $\bar{\mathbf{A}}$ ,  $\bar{\mathbf{B}}$  and  $\bar{\mathbf{C}}$  are previously defined by expressions (4.2.2), while the residual term is  $\bar{\mathbf{R}}^n$

$$\bar{\mathbf{R}}^n = \delta_\xi \bar{\mathbf{F}}(\bar{\mathbf{q}}^n) + \delta_\eta \bar{\mathbf{G}}(\bar{\mathbf{q}}^n) + \delta_\zeta (\bar{\mathbf{H}}(\bar{\mathbf{q}}^n) - \bar{\mathbf{H}}_v(\bar{\mathbf{q}}^n)). \quad (4.128.1)$$

In the equations (4.128) and (4.128.1)  $\delta_\xi$ ,  $\delta_\eta$  and  $\delta_\zeta$  represent central difference operators  $\partial/\partial\xi$ ,  $\partial/\partial\eta$  and  $\partial/\partial\zeta$ .

The parameter  $\beta$  in the expression (4.128) determines time accuracy of the applied scheme, which is analyzed in chapter 4.2.2. Unconditionally stable implicit scheme is derived by LU factorization [20], [21] and [6]

$$\begin{aligned} & \left\{ \mathbf{I} + \beta \Delta t \left[ \delta_\xi^- \bar{\mathbf{A}}^+ + \delta_\eta^- \bar{\mathbf{B}}^+ + \delta_\zeta^- (\bar{\mathbf{C}}^+ - \bar{\mathbf{C}}_v^+) \right] \right\}^n * \\ & * \left\{ \mathbf{I} + \beta \Delta t \left[ \delta_\xi^+ \bar{\mathbf{A}}^- + \delta_\eta^+ \bar{\mathbf{B}}^- + \delta_\zeta^+ (\bar{\mathbf{C}}^- - \bar{\mathbf{C}}_v^-) \right] \right\}^n \Delta \bar{\mathbf{q}}^n + \Delta t \bar{\mathbf{R}}^n = 0, \end{aligned} \quad (4.129)$$

where  $\delta_\xi^-$ ,  $\delta_\eta^-$ ,  $\delta_\zeta^-$ ,  $\delta_\xi^+$ ,  $\delta_\eta^+$  and  $\delta_\zeta^+$  are difference operators determined by relations (4.20.1), where the elements of the matrices with indices  $(i + 1/2, j, k)$  and  $(i - 1/2, j, k)$  are calculated by averaging flow variables between computational cells  $(i, j, k)$  and  $(i + 1, j, k)$ , ie.  $(i - 1, j, k)$  and  $(i, j, k)$ , respectively. Difference operators related to the remaining two coordinate directions can be determined on the basis of similar relations.

Solutions of the system of equations (4.129) is possible to determine in two passes:

$$\begin{aligned} & \left\{ \mathbf{I} + \beta \Delta t \left[ \delta_\xi^- \bar{\mathbf{A}}^+ + \delta_\eta^- \bar{\mathbf{B}}^+ + \delta_\zeta^- (\bar{\mathbf{C}}^+ - \bar{\mathbf{C}}_v^+) \right] \right\}^n \Delta \bar{\mathbf{q}}^{*n} = -\Delta t \bar{\mathbf{R}}^n, \\ & \left\{ \mathbf{I} + \beta \Delta t \left[ \delta_\xi^+ \bar{\mathbf{A}}^- + \delta_\eta^+ \bar{\mathbf{B}}^- + \delta_\zeta^+ (\bar{\mathbf{C}}^- - \bar{\mathbf{C}}_v^-) \right] \right\}^n \Delta \bar{\mathbf{q}}^n = \Delta \bar{\mathbf{q}}^{*n}, \end{aligned} \quad (4.129.1)$$

where the matrices of “viscous” flux  $\bar{\mathbf{C}}_v^+$  and  $\bar{\mathbf{C}}_v^-$ , present in the implicit term of the equation (4.129), are approximated according to the work of Pulliam [37]

$$\bar{\mathbf{C}}_v^+ = \lambda_{C_v} \mathbf{I} \quad \text{i} \quad \bar{\mathbf{C}}_v^- = -\lambda_{C_v} \mathbf{I}, \quad (4.129.2)$$

where  $\lambda_{C_v}$  is the eigenvalue of the viscous flux matrix  $\bar{\mathbf{C}}_v$

$$\lambda_{C_v} = \mu (\zeta_x^2 + \zeta_y^2 + \zeta_z^2) \partial_\zeta \left( \frac{1}{\rho} \right), \quad (4.129.3)$$

while the matrices  $\bar{\mathbf{A}}^+$ ,  $\bar{\mathbf{B}}^+$ ,  $\bar{\mathbf{C}}^+$ ,  $\bar{\mathbf{A}}^-$ ,  $\bar{\mathbf{B}}^-$ , and  $\bar{\mathbf{C}}^-$  are determined by expressions (4.20.2). The process of solving the first system of equations (4.129.1) takes place in the direction of increasing index  $(i, j, k)$  by replacing the known, previously calculated flow variable vector  $\Delta \bar{\mathbf{q}}^{*n}$ , while the second system of equations is solved in the direction of decreasing index  $(i, j, k)$ .

On the body surface boundary conditions are defined by zero flow velocity vector with the adopted assumption of adiabatic flow and zero pressure gradient [1] and [38]

$$u = v = w = 0, \quad \frac{\partial H}{\partial n} = 0 \quad \text{and} \quad \frac{\partial p}{\partial n} = 0. \quad (4.129.4)$$

where  $H$  is the enthalpy of the fluid. Defining boundary conditions at the outer boundaries of the physical domain has already been explained in chapter 4.2.2.

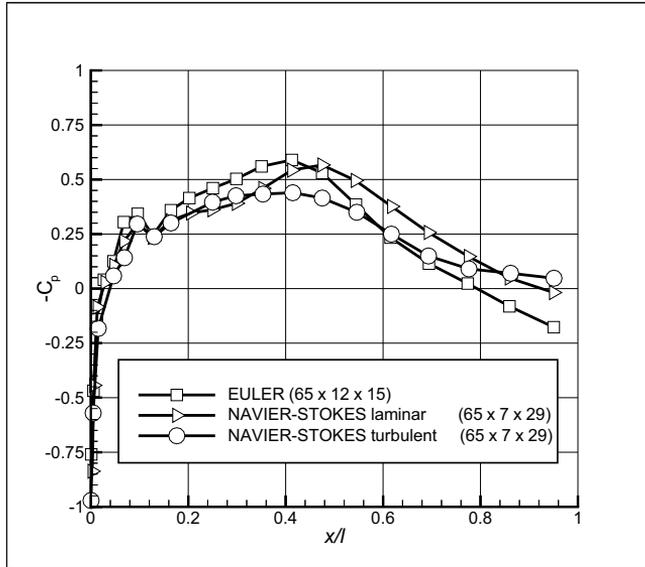


Figure 4.19: Pressure distribution on a rectangular wing – plane of symmetry.

Application of central difference scheme in calculation of the residual term  $\bar{\mathbf{R}}^n$  in the equation (4.128.1) requires the introduction of additional terms in order to obtain stationary solutions of dynamic equations [45]. Additional dissipative terms are introduced in the same way as in the chapter 4.2.2.

After the introduction of dissipative terms, the system the equations (4.129) receives the final form

$$\left\{ \mathbf{I} + \beta \Delta t \left[ \delta_{\xi}^- \bar{\mathbf{A}}^+ + \delta_{\eta}^- \bar{\mathbf{B}}^+ + \delta_{\zeta}^- (\bar{\mathbf{C}}^+ - \bar{\mathbf{C}}_v^+) \right] \right\}^n * \left\{ \mathbf{I} + \beta \Delta t \left[ \delta_{\xi}^+ \bar{\mathbf{A}}^- + \delta_{\eta}^+ \bar{\mathbf{B}}^- + \delta_{\zeta}^+ (\bar{\mathbf{C}}^- - \bar{\mathbf{C}}_v^-) \right] \right\}^n \Delta \bar{\mathbf{q}}^n + \Delta t [\bar{\mathbf{R}}^n - \mathbf{R}_1^n] = 0, \quad (4.130)$$

noting that the integration step has been determined previously by relations (4.12) and (4.12.1).

Figure 4.19 shows the results of numerical analysis of stationary non-viscous and viscous flow around rectangular wing with airfoil NACA 65A010, constant along the span. An algebraic three-dimensional non-orthogonal “C-H” computational grid was used, while from the chapter 4.2.2 the convergence criterion is retained.

The calculation was performed for Mach number of undisturbed flow  $M_\infty = 0.8$  at zero wing angle of attack and Reynolds number  $Re_\infty = 50000$  for laminar flow. In the case of turbulent flow  $Re_\infty = 1.2 \times 10^7$ . Very coarse grid ( $65 \times 7 \times 29$ ) was used, and the results are given for cross section in the plane of wing symmetry.

As can be seen from Figure 4.19, the flow separation occurs at the position of the greatest profile thickness, and with an increase of Reynolds number in turbulent flow, position of *separation bubble* moves towards the trailing edge of the airfoil. Unfortunately, the application of such a coarse grid does not allow qualitative determination of the pressure coefficient  $C_p$  and position of the separation point. Fine mesh would increase the accuracy of the numerical solution, but at the same time it would make a demand for the more powerful computer systems.

Exposed procedure in viscous flow analysis in transonic speed range made calculation of the aerodynamic load possible in cases of strong shock waves with boundary layer effects, when application of potential theory and Euler equations is practically impossible.





## Bibliography

- [1] Agarwal, R.K., Deese, J.E., *Computation of Transonic Viscous Airfoil, Inlet, and Wing Flowfields*, AI, 84–1551, 1984.
- [2] Anderson, D.A., Tannehill, J.C., Pletcher, R.H., *Computational Fluid Mechanics and Heat Transfer*, HP, New York. 1984.
- [3] Anderson, W.K., Thomas, J.L., Whitfield, D.L., *Multigrid Acceleration of the Flux Split Euler Equations*, AI, number =86–0274, 1986.
- [4] Baldwin, B., Lomax, H., *Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows*, AI, (78–257), 1978.
- [5] Belk, D., Whitfield, D., *Time-Accurate Euler Equations Solutions on Dynamic Blocked Grids*, AI, 87–1127, 1987.
- [6] Choo, Y., Yoon, S., *Composite Grid and Finite–Volume LU Implicit Scheme for Turbine Flow Analysis*, AI, 87–1129, 1987 .
- [7] Dick, E., *Introduction to Finite Volume Techniques in Computational Fluid Dynamics*, U Wendt, 1996.
- [8] Emanuel, G., *Gasdynamics: Theory and Applications*, AIAA Education Series, New York, 1986.
- [9] Glowinski, R., Lions, J.L., editor. *Computing Methods in Applied Sciences and Engineering*, North Holland, Amsterdam, 1984.

- [10] Godunov, S.K., Zabrodin, A.V., Prokopov, G.P., *Multigrid Acceleration of the Flux Split Euler Equations*, USSR Comp. Math. Phys. 6:1020-50, 1961
- [11] Годунов, С.К., Разностный метод численного расчета разрывных решений уравнений гидродинамики, Математический сборник, 47:271-306, 1959.
- [12] Harten, A., Hyman, J.M., Lax, P.D., *On Finite Difference Approximations and Entropy Conditions for Shocks*, Comm. Pure and Applied Mathematics, 29: 297-322, 1976.
- [13] Harten, A., Hyman, J.M., *Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws*, Journal of Computational Physics, 50:235-69, 1983.
- [14] Hirsch, C., *Numerical Computation of Internal and External Flows*, John Wiley & Sons, Chichester, 1992.
- [15] Hoffmann, K.A., *Computational Fluid Dynamics for Engineers*, A Publication of Engineering Education System, Austin, 1989.
- [16] Holt, M., *Numerical Methods in Fluid Dynamics*, Springer-Verlag, Berlin, 1977.
- [17] Houghton, E.L., Brock, A.E., *Tables for the Compressible Flow of Dry Air*, Edward Arnold, Bungay, Suffolk, 1975.
- [18] Jameson, A., Schmidt, W., Turkel, E., *Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes*, AIAA Paper, (81-1259), June 1981.
- [19] Jameson, A., Baker, T.J., *Solution of the Euler Equations for Complex Configurations*, AIAA Paper, (83-1929), July 1983.
- [20] Jameson, A., Yoon, S., *Multigrid Solution of the Euler Equations Using Implicit Schemes*, AIAA Journal, 248(11), November, 1986.
- [21] Jameson, A., Yoon, S., *LU Implicit Schemes with Multiple Grids for the Euler Equations*, AIAA Paper, (86-0105), January 1986.
- [22] Landau, L.D., Lifšic, E.M., *Mehanika neprekidnih sredina*, Građevinska knjiga, Beograd, 1965.
- [23] Launder, B.E., Spalding, D.B., *Mathematical Models of Turbulence*, Academic Press, London, 1972.
- [24] Lax, P.D., *Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*, SIAM region. Ser. Appl. Math. Vol. II, Philadelphia, 1973.
- [25] McCormack, R.W., Paullay, A.J., *Computational Efficiency Achieved by Time Splitting of Finite Difference Operators*, AIAA Paper, (72-154), 1972
- [26] McDonald, P.W., *The Computation of Transonic Flow through Two-Dimensional Gas Turbine Cascades*, ASME Paper, (71-GT-89), 1971.

- [27] Mladenović, N., *rimena Eulerovih jednačina u proračunu transoničnog trodimenzionalnog strujanja*, Tehnika, 7-8: 503-505, 1991.
- [28] Mladenović, N., *Primena modifikovane implicitne sheme Roe u proračunu transoničnog unutrašnjeg strujanja*, Tehnika, 4:M15-M19, 1993.
- [29] Mladenović, N., *Numeričko rešavanje Eulerovih jednačina modifikovanom šemom Godunova drugog reda*, XXII Jugoslovenski kongres Teorijske i Primenjene Mehanika, Vrnjačka Banja, 1997.
- [30] Mladenović, N., *Numerical Solution of the Euler Equations in Quasi One-Dimensional Flow*, Facta Universitatis, 2(6), 1996.
- [31] Mladenović, N., *Transonic Flow Solution of the Euler Equations Using Implicit LU factorization*, Theoretical and Applied Mechanics, (17), 1991.
- [32] Mladenović, N., *Transonic Flow Solution of the Thin Layer Navier–Stokes Equations Using Implicit LU Factorization*, Theoretical and Applied Mechanics, (18), 1992.
- [33] Mladenović, N., *Solution of the Euler Equations Using Implicit LU Scheme with ROE Averaging*, CTAM, (1), 1992.
- [34] Osher, S., *Riemann Solvers, the Entropy Condition, and Difference Approximations*, SIAM Journal Numerical Analysis, 21: 217-35, 1984.
- [35] Petrović Z., Stupar S., *Projektovanje računarom*, Mašinski fakultet, Beograd, 1992.
- [36] Petrović, Z., Stupar, S., *Fundamental Equations of Aerodynamics*, Mechanical Engineering Faculty, Belgrade, 1997.
- [37] Pulliam, T., Steger, J., *Recent Improvement in Efficiency, Accuracy, and Convergence for Implicit Approximate Factorization Algorithm*, AIAA Paper, (85-0360), January 1985.
- [38] Radespiel, R., Rossow, C., Swanson, R., *An Efficient Cell–Vertex Multigrid Scheme for the Three–Dimensional Navier–Stokes Equations*, AIAA Paper, (89-1953), June 1989.
- [39] Rizzi, A.W., Inouye, M., *Time Split Finite Volume Method for Three-Dimensional Blunt-Body Flows*, AIAA Journal, 11:1478-85, 1973
- [40] Roe, P.L., *Characteristic-Based Schemes for the Euler Equations*, Ann. Rev. Fluid Mech., 18:337-65, 1986
- [41] Roe, P.L., Pike, J., *Efficient Construction and Utilization of Approximate Riemann Solutions*, U Glowinski, R., Lions, J.L. [32], 1984.
- [42] Shapiro, A.H., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Roland Press, New York, 1953.
- [43] Shetz, J.A., *Boundary Layer Analysis*, Prentice Hall, Englewood Cliffs, 1993.

- 
- [44] Steger, J.L., Warming, R.F., *Flux Vector Splitting of the Inviscid Gasdynamics Equations with Application to Finite Difference Methods*, Journal of Computational Physics, 40(2), April 1981.
- [45] Swanson, R.C., Turkel, E., *Artificial Dissipation and Central Difference Schemes for the Euler and Navier–Stokes Equations*, NASA, (CR-178296), April 1987.
- [46] Turkel, E., *Accuracy Versus Convergence Rates for a Three Dimensional Multistage Euler Code*, NASA, (CR-181665), May 1988.
- [47] Van Leer, B., *Flux Splitting for the Euler Equations*, Eight International Conference on Numerical Methods in Fluid Dynamics, 1982.
- [48] Yee, H.C., *Numerical Approximations of Boundary Conditions with Applications to Inviscid Gas Dynamics*, NASA, (TM-81265), 1981.
- [49] Wendt, J.F., editor. *Computational Fluid Dynamics*, Springer-Verlag, Berlin, 1996.
- [50] Whitfield, D., Janus, M., *Three Dimensional Unsteady Euler Equations Solution Using Flux Vector Splitting*, AIAA Paper, (84-1552), June 1984.
- [51] Whitfield, D.L., Janus, J.M., Simpson, B.L., *Implicit Finite Volume High Resolution Wave-Split Scheme for Solving the Unsteady Three-Dimensional Euler and Navier-Stokes Equations on Stationary or Dynamic Grids*, NASA Lewis Research Center, (MSSU-EIRS-ASE-88-2): 1020-50, 1988.
- [52] Whitman, G.B., *Linear and Nonlinear Waves*, John Wiley & Sons, New York, 1074.
- [53] Wolfram, S., *Mathematica – A system for Doing Mathematics by Computer*, Addison Wesley Publishing Company, New York, 1988.

# Numerical method - SPH

## **5** Review of Development of the Smooth Particle Hydrodynamics (SPH) Method ..... 161

- 5.1 Introduction
- 5.2 Basic Formulation
- 5.3 Conservation Equations
- 5.4 Kernel Function
- 5.5 Variable Smoothing Length
- 5.6 Neighbour Search
- 5.7 SPH Shortcomings
- 5.8 Derivation of Normalised Corrected Gradient SPH formula
- 5.9 Tensile Instability
- 5.10 Zero-Energy Modes
- 5.11 Non-Local properties of SPH
- 5.12 Summary

## **Bibliography** ..... 203

## **IV** **Numerical method -** **CMD** 213



# 5. Review of Development of the Smooth Particle Hydrodynamics (SPH) Method

## 5.1 Introduction

This paper discusses the development of the Smooth Particle Hydrodynamics (SPH) method in its original form, which is based on the updated Lagrangian formalism. SPH is a relatively new numerical technique for the approximate integration of partial differential equations.

It is a **meshless Lagrangian method** that uses a pseudo-particle interpolation method to compute smooth field variables. Each pseudo-particle has a mass, Lagrangian position, Lagrangian velocity, and internal energy; other quantities are derived by interpolation or from constitutive relations. The pseudo-particles move with the velocity of the continuum, but not associated with a grid and consequently do not have fixed connectivity.

The advantage of the meshless approach is its ability to solve problems that cannot be effectively solved using other numerical techniques. It does not suffer from the mesh distortion problems that limit Lagrangian approaches based on a structured mesh when simulating large deformations. As it is a Lagrangian method it naturally tracks material history information, such as damage, without the diffusion that typically occurs in Eulerian approaches due to advection.

Gingold and Monaghan [28] and Lucy [55] initially developed SPH in 1977 for the simulation of astrophysics problems. Their breakthrough was a method for the calculation of derivatives that did not require a structured computational mesh. Review papers by

Benz [16] and Monaghan [58] cover the early development of SPH.

Right from the early days of SPH the importance of the smoothing kernel function as the essential feature of the SPH scheme was recognised. The Gaussian and the cubic B spline kernel functions are the most widely used in SPH, see Monaghan and Lattanzio [60]. However, most practical work relies on the monotone splines which, when used with small supports, allow for more accurate numerical solutions and higher numerical efficiency according to Balsara [7]. Liu et al. [50] among a number of other researchers demonstrated that in general, regardless of the choice of kernel function, the SPH method is not even zero order consistent. This is a consequence of the fact that the accuracy of the kernel interpolation depends on the distribution of the interpolation points within the kernel support. This effect is especially pronounced in the vicinity of boundaries, where the kernel support extends beyond the domain of the problem considered and consequently becomes incomplete Liu et al. [52].

Libersky and Petchek [47] extended SPH to work with the full stress tensor in 2D. This addition allowed SPH to be used in problems where material strength is important. The development SPH with strength of materials continued with its extension to 3D by Libersky [48]. Applications of SPH to model solids, i.e. material with strength, further highlighted shortcomings in the basic method: consistency, tensile instability, zero energy modes, treatment of contact and artificial viscosity. These shortcomings were discussed in detail in the first comprehensive analyses of the SPH method by Swegle [87], Wen [105]. The problems of consistency and accuracy of the SPH method, identified by Belytschko [11], were addressed by Randles and Libersky [74], Vignjevic and Campbell [97] and a number of other researchers. This resulted in a normalised first order consistent version of the SPH method with improved accuracy. The attempts to ensure first order consistency in SPH resulted in emergence of a number of variants of the SPH method, such as Element Free Galerkin Method (EFGM) Belytschko [13], Kongauz [44], Reproducing Kernel Particle Method (RKPM) Liu [52][54], Moving Least Square Particle Hydrodynamics (MLSPH) Dilts [22], Meshless Local Petrov Galerkin Method (MLPG) Atluri and Zhu [1]. These methods allow the restoration of consistency of any order by means of a correction function. It has been shown by Dilts [22] and independently by Atluri et al. [5] that the approximations based on corrected kernels are identical to moving least square approximations.

A comprehensive stability analysis of particle methods in general by Belytschko [14], Xiao and Belytschko [106], and independently by Randles [74] who worked specifically on the SPH method provided improved understanding of the methods analysed and confirmed the conclusions from Swegle's initial study. Randles' unique analysis, which included space and time discretisation, showed that SPH can be stabilised by precise choice of time step size and predictor corrector type of time integration. Rabezuk et al. [73] demonstrated that if used within a total Lagrangian framework SPH does not exhibit the tensile instability.

Tensile instability in SPH has, as a consequence, non-physical motion of particles which form clusters. This was first observed in materials loaded in tension (negative stress), however the instability can develop under compressive loading, see Swegle [87]. In simulations of solids the instability may result in non-physical numerical fragmentation.

Another unconventional solution to the SPH tensile instability and zero energy mode instability problems was first proposed by Dyka [23] in which the stresses are calculated at

the locations other than the SPH particles (non-collocational SPH). The results achieved in 1D were encouraging but a rigorous stability analysis was not performed. A 2D version of this approach was investigated by Vignjevic and Campbell [97], based on the normalised version of SPH. This investigation showed that extension to 2D was possible, although general boundary condition treatment and simulation of large deformations would require further research.

Monaghan [64] showed how the instability can be removed by using an artificial stress which, in the case of fluids, is an artificial pressure. When used in simulation of solids this artificial, in other words non-physical stress, may result in an unrealistic material strength and therefore has to be used with caution.

In spite of these developments, the crucial issue of convergence in a rigorous mathematical sense and the links with conservation have not been well understood. Encouraging preliminary steps in this direction have already been made by Moussa and Vila [67], who proved convergence of their meshless scheme for non-linear scalar conservation laws; see also Moussa [66]. This theoretical result appears to be the first of its kind in the context of meshless methods. Furthermore, Moussa and Vila, proposed an interesting new way to stabilise normalised SPH and allow for treatment of boundary conditions by using approximate Riemann solvers and up-winding, an approach usually associated with finite volume shock-capturing schemes of the Godunov type. This work developed a strong following which include: Parshikov et al. [68] also uses the Riemann solver to calculate the numerical flux between pair of interacting particles; Cha and Whitworth [20], who have applied the Riemann solver of van Leer [93][94] to isothermal hydrodynamics; and Inutsuka [36] who proposed an interesting but fairly complex approach to obtain second-order accurate SPH in 1D.

The improvements in accuracy and stability achieved by kernel re-normalisation or correction do not come for free; now it is necessary to treat the essential boundary conditions in a rigorous way. The approximations in SPH do not have the property of strict interpolants, so that in general they are not equal to the particle value of the dependent variable, i.e.  $u^h(\mathbf{x}_j) = \sum_I \phi_I(\mathbf{x}_j) u_I \neq u_j$ . Consequently it does not suffice to impose zero values for at the boundary positions to enforce homogeneous boundary conditions. Another issue with this approach is that in conventional SPH the boundary is diffuse. In the case of normalised SPH particles do lie on the domain boundary which is in this case precisely defined.

The treatment of boundary conditions and contact could be and was neglected in the conventional SPH method. If the imposition of the free surface boundary condition (stress free condition) is simply ignored, then conventional SPH behaves in an approximately correct manner, giving zero pressure for fluids and zero surface stresses for solids, because of the deficiency of particles at the boundary. This is the reason why conventional SPH gives physically reasonable results at free surfaces. Contact between bodies, in conventional SPH, is treated by smoothing over all particles neighbouring the contact interface, regardless of material types in contact (for instance contact between a solid body and a fluid). Although simple this approach can give physically incorrect results, such as tensile forces between the bodies in contact.

Campbell et al. [19] made an early attempt to introduce a more systematic treatment of boundary condition by re-considering the original kernel integral estimates and taking

into account the boundary conditions through residual terms in the integral by parts. Very interesting work on boundary conditions in SPH is due to Takeda et al. [89], who applied SPH to a variety of viscous flows. A similar approach was also used by Randles [74] with the ghost particles added to enforce reflected and symmetry surface boundary conditions. Belytschko, Lu and Gu [13] imposed the essential boundary conditions by the use of Lagrange multipliers leading to an awkward structure of the linear algebraic equations, which are not positive definite. Krongauz and Belytschko [43] proposed a simpler technique for the treatment of the essential boundary conditions in meshless methods, by employing a string of finite elements along the essential boundaries. This allowed for the boundary conditions to be treated accurately, but reintroduced the shortcomings inherent to structured meshes.

Randles et al. [74][78] were first to propose a more general treatment of boundary conditions based on an extension of the ghost particle method. In their approach the boundary is considered to be a surface one half of the local smoothing length away from the so-called boundary particles. A boundary condition is applied to a field variable by assigning the same boundary value of the variable to all ghost particles. A constraint is imposed on the boundary by interpolating it smoothly between the specified boundary particle value and the calculated values on the interior particles. This serves to communicate to the interior particles the effect of the specific boundary condition. There are two main difficulties in this:

- Definition of the boundary (surface normal at the vertices).
- Communication of the boundary value of a dependent variable from the boundary to internal particles.

A penalty contact algorithm for SPH was developed by Campbell and Vignjevic [18]. This algorithm was tested on normalised SPH in combination with the Randles' approach for treatment of free surfaces. The contact algorithm considered only particle-particle interactions, and allowed contact and separation to be correctly simulated. However, tests showed that when this approach is used zero-energy modes are often excited. Further development of this contact algorithm for the treatment of contact problems involving frictionless sliding and separation under large deformations was achieved by the contact conditions through the use a contact potential for particles in contact, see Vignjevic et al. [99]. Inter-penetration is checked as a part of the neighbourhood search. In the case of conventional SPH contact conditions are enforced on the boundary layer  $2h$  thick while in the case of the normalized SPH contact conditions are enforced for the particles lying on the contact surface.

In a number of engineering applications it is beneficial to discretise only certain parts of the domain with particles and the rest with finite elements. The main reasons for this are to take advantage of the strengths of both methods, which include significantly better numerical efficiency of the finite element (FE) method, and in SPH modelling arbitrary crack propagation, large deformations and adaptive refinement of the discretisation.

One of the first coupling procedures for FE-SPH coupling was proposed by Attaway et al. [6]. They developed a penalty-based approach for modelling of fluid-structure interactions where the fluid was discretised with particles and the structure was modelled with finite elements. A similar approach was proposed by Johnson [38] and Johnson et al. [39]. In addition to the contact algorithm, they developed a tied interface where SPH

particles are fixed to FE nodes. This allows for a continuous coupling of the SPH and the FE domains. Sauer [82] proposed an SPH–FE coupling by extending the SPH domain onto the FE mesh. Different possibilities for exchanging forces between FE nodes and particles were shown, and the approach was used for adaptive conversion of elements into particles. The main difference with most other coupling methods is the use of a strong-form coupling. This approach was successfully applied to a number of impact problems, see Sauer et al. [83][84] and Hiermaier et al. [31][32]. Using a variation of the contact algorithm they developed for SPH, De Vuyst and Vignjevic [21] coupled Cranfield University SPH code with Lawrence Livermore National Laboratory DYNA3D.

Coupling algorithms developed for other meshless techniques can be applied for use with SPH. Among many recently proposed techniques a selected few are mentioned below. A mixed hierarchical approximation based on meshless methods and FE, proposed by Huerta et al. [33][34], remove the discontinuities in the derivative across the interior boundaries when coupling FE and the element-free Galerkin method (EFG). Belytschko and Xiao [15] proposed the ‘bridging domain coupling method’ which uses Lagrange multipliers over a domain where FE and particle discretisations overlap. They applied this approach to multi-scale simulations for coupling continua with molecular dynamics. Another method for atomic and continua scale bridging was proposed by Wagner and Liu [103] and Kadowaki and Liu [41]. By matching dynamic impedances of different discretisation domains spurious wave reflection is prevented in this approach.

A comprehensive overview of techniques for coupling of a range of meshless methods with FE with examples is given in Li and Liu [49].

## 5.2 Basic Formulation

The spatial discretisation of the state variables is provided by a set of points. Instead of a grid, SPH uses kernel interpolation to approximate the field variables at any point in a domain. For instance, an estimate of the value of a function  $f(\mathbf{x})$  at the location  $\mathbf{x}$  is given in a continuous form by an integral of the product of the function and a kernel (weighting) function  $W(|\mathbf{x} - \mathbf{x}'|, h)$ :

$$\langle f(\mathbf{x}) \rangle = \int_{\Omega} f(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}'. \quad (5.1)$$

Where: the angle brackets  $\langle \cdot \rangle$  denote a kernel approximation  $h$  is a parameter that defines size of the kernel support known as the smoothing length  $\mathbf{x}'$  is the new independent variable.

The kernel function usually has the following properties:

- Compact support, which means that it's zero everywhere but on a finite domain, in conventional SPH this domain is taken to be all points within twice the smoothing length,  $h$ , of the centre:

$$W(|\mathbf{x} - \mathbf{x}'|, h) = 0 \quad \text{for} \quad |\mathbf{x} - \mathbf{x}'| \geq 2h. \quad (5.2)$$

- Normalised

$$\int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}' = 1. \quad (5.3)$$

These requirements, formulated by Lucy [43], ensure that the kernel function reduces to the Dirac delta function when  $h$  tends to zero:

$$\lim_{h \rightarrow 0} W(|\mathbf{x} - \mathbf{x}'|, h) = \delta(|\mathbf{x} - \mathbf{x}'|, h). \quad (5.4)$$

And therefore, it follows that:

$$\lim_{h \rightarrow 0} \langle f(\mathbf{x}) \rangle = f(\mathbf{x}). \quad (5.5)$$

If the function  $f(\mathbf{x})$  is only known at  $N$  discrete points, the integral of equation (5.1) can be approximated by a summation:

$$\begin{aligned} f_I = f(\mathbf{x}_I) &\approx \langle f(\mathbf{x}_I) \rangle = \int_{\Omega} f(\mathbf{x}') W(\mathbf{x} - \mathbf{x}') d\Omega \approx \\ &\approx \sum_{J=1}^N \frac{m_J}{\rho_J} f(\mathbf{x}_J) W(|\mathbf{x}_I - \mathbf{x}_J|, h). \end{aligned} \quad (5.6)$$

In the above equation, the subscript  $I$  and  $J$  denote particle number,  $m_J$  and  $\rho_J$  the mass and the density of particle  $J$ ,  $N$  the number of neighbours of particle  $I$  (number of particles that interact with particle  $I$ , i.e. the support of the kernel),  $\frac{m_J}{\rho_J}$  is the volume associated to the point or particle  $J$  and  $W_{IJ} = W(|\mathbf{x}_I - \mathbf{x}_J|, h)$ .

In SPH literature, the term particles is misleading as in fact these particles have to be thought of as interpolation points rather than mass elements.

Equation (5.6) constitutes the basis of SPH method. The value of a variable at a particle, denoted by superscript  $I$ , is calculated by summing the contributions from a set of neighbouring particles (Figure 5.1), denoted by superscript  $J$  and for which the kernel function is not zero:

$$\langle f(\mathbf{x}_I) \rangle = \sum_J \frac{m_J}{\rho_J} f(\mathbf{x}_J) W(|\mathbf{x}_I - \mathbf{x}_J|, h). \quad (5.7)$$

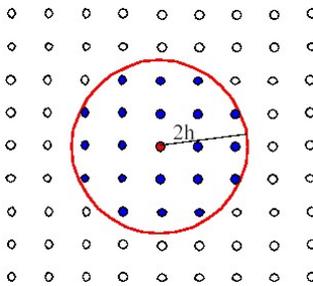


Figure 5.1: Set of neighbouring particles.

### 5.3 Conservation Equations

The conservation equations in Lagrangian framework are given by:

$$\frac{D\rho}{Dt} = -\rho \nabla \cdot \mathbf{v}, \quad (5.8)$$

$$\frac{D\mathbf{v}}{Dt} = \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} \quad \text{or} \quad \frac{D\mathbf{v}}{Dt} = \nabla \cdot \left( \frac{\boldsymbol{\sigma}}{\rho} \right) + \frac{\boldsymbol{\sigma}}{\rho^2} \cdot \nabla \rho, \quad (5.9)$$

$$\frac{DE}{Dt} = \frac{1}{\rho} \boldsymbol{\sigma} : \nabla \mathbf{v} \quad \text{or} \quad \frac{DE}{Dt} = \frac{1}{\rho^2} \boldsymbol{\sigma} : \nabla(\rho \mathbf{v}) - \frac{\boldsymbol{\sigma} \cdot \mathbf{v}}{\rho^2} \cdot \nabla \rho \quad (5.10)$$

Where  $\frac{D}{Dt}$  is the material time derivative and  $\mathbf{v} = \dot{\mathbf{x}}$ .

Equations (5.9) and (5.10) are the forms proposed by Monaghan [47]. Kernel interpolation allows the derivation of the basic SPH form of these conservation equations as:

$$\left\langle \frac{D\rho(\mathbf{x})}{Dt} \right\rangle = \int_{\Omega} \rho(\mathbf{x}') \nabla \cdot \mathbf{v}(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h) d\Omega, \quad (5.11)$$

$$\begin{aligned} \left\langle \frac{D\mathbf{v}(\mathbf{x})}{Dt} \right\rangle &= \int_{\Omega} \nabla \cdot \left( \frac{\boldsymbol{\sigma}(\mathbf{x}')}{\rho(\mathbf{x}')} \right) W(|\mathbf{x}' - \mathbf{x}|, h) d\Omega + \\ &+ \int_{\Omega} \frac{\boldsymbol{\sigma}(\mathbf{x}')}{[\rho(\mathbf{x}')]^2} \cdot \nabla \rho(\mathbf{x}') W(|\mathbf{x}' - \mathbf{x}|, h) d\Omega, \end{aligned} \quad (5.12)$$

$$\begin{aligned} \left\langle \frac{DE(\mathbf{x})}{Dt} \right\rangle &= \int_{\Omega} \frac{\boldsymbol{\sigma}(\mathbf{x}')}{[\rho(\mathbf{x}')]^2} : \nabla(\rho(\mathbf{x}') \mathbf{v}(\mathbf{x}')) W(|\mathbf{x}' - \mathbf{x}|, h) d\Omega - \\ &- \int_{\Omega} \frac{\boldsymbol{\sigma}(\mathbf{x}') \cdot \mathbf{v}(\mathbf{x}')}{[\rho(\mathbf{x}')]^2} \cdot \nabla \rho(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h) d\Omega. \end{aligned} \quad (5.13)$$

All of the above equations contain integrals of the form:

$$\int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) f(\mathbf{x}') \frac{\partial g(\mathbf{x}')}{\partial \mathbf{x}'} d\mathbf{x}'. \quad (5.14)$$

Using a Taylor series expansion at point  $\mathbf{x}' = \mathbf{x}$ , it follows:

$$\begin{aligned} &\int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) f(\mathbf{x}') \frac{\partial g(\mathbf{x}')}{\partial \mathbf{x}'} d\mathbf{x}' = \\ &= \int_{\Omega} \left\{ f(\mathbf{x}) \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} + (\mathbf{x} - \mathbf{x}') \frac{d}{d\mathbf{x}} \left( f(\mathbf{x}) \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \right) + \dots \right\} W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}', \end{aligned} \quad (5.15)$$

As  $W$  is an even function, the terms containing odd powers of  $\mathbf{x}' - \mathbf{x}$  vanish. Neglecting second and higher order terms, which is consistent with the overall order of the method, gives:

$$\begin{aligned} & \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) f(\mathbf{x}') \frac{\partial g(\mathbf{x}')}{\partial \mathbf{x}'} d\mathbf{x}' = \\ & = f(\mathbf{x}) \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}' = f(\mathbf{x}) \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}}. \end{aligned} \quad (5.16)$$

Substituting  $\left\langle \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \right\rangle$  for  $\frac{\partial g(\mathbf{x})}{\partial \mathbf{x}}$  gives:

$$f(\mathbf{x}) \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} = f(\mathbf{x}) \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \frac{\partial g(\mathbf{x}')}{\partial \mathbf{x}'} d\mathbf{x}'. \quad (5.17)$$

Using the last relation in equations (5.11), (5.12) and (5.13) yields

$$\left\langle \frac{D\rho(\mathbf{x})}{Dt} \right\rangle = -\rho(\mathbf{x}) \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \nabla \cdot \mathbf{v}(\mathbf{x}') d\mathbf{x}', \quad (5.18)$$

$$\begin{aligned} \left\langle \frac{D\mathbf{v}}{Dt} \right\rangle &= \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \nabla \cdot \left( \frac{\boldsymbol{\sigma}(\mathbf{x}')}{\rho(\mathbf{x}')} \right) d\mathbf{x}' + \\ &+ \frac{\boldsymbol{\sigma}(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \nabla \rho(\mathbf{x}') d\mathbf{x}', \end{aligned} \quad (5.19)$$

$$\begin{aligned} \left\langle \frac{DE(\mathbf{x})}{Dt} \right\rangle &= \frac{\boldsymbol{\sigma}(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \nabla (\rho(\mathbf{x}') \mathbf{v}(\mathbf{x}')) d\mathbf{x}' - \\ &- \frac{\boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} W(|\mathbf{x} - \mathbf{x}'|, h) \nabla (\rho(\mathbf{x}')) d\mathbf{x}'. \end{aligned} \quad (5.20)$$

Note that all equations include kernel approximations of spatial derivatives:

$$\langle \nabla f(\mathbf{x}) \rangle = \int_{\Omega} \nabla f(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}'. \quad (5.21)$$

Integrating by parts gives:

$$\begin{aligned} \langle \nabla f(\mathbf{x}) \rangle &= \int_{\Omega} \nabla (W(|\mathbf{x} - \mathbf{x}'|, h) f(\mathbf{x}')) d\mathbf{x}' - \\ &- \int_{\Omega} f(\mathbf{x}') \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}'. \end{aligned} \quad (5.22)$$

Using Green's theorem, the first term of the right hand side can be rewritten as:

$$\int_{\Omega} \nabla \cdot (f(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h)) d\mathbf{x}' = \int_{\Omega} f(\mathbf{x}') W(|\mathbf{x} - \mathbf{x}'|, h) \mathbf{n} dS = 0. \quad (5.23)$$

The surface integral is zero if the domain of integration is larger than the compact support of  $W$  or if the field variable assumes zero value on the boundary of the body (free surface). If none of these conditions are satisfied, modifications should be made to account for boundary conditions.

One should observe that in Equations (5.18), (5.19) and (5.20) the spatial derivatives of the field variables are substituted by the derivatives of the kernel function giving:

$$\left\langle \frac{D\rho(\mathbf{x})}{Dt} \right\rangle = \rho(\mathbf{x}) \int_{\Omega} \mathbf{v}(\mathbf{x}') \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}', \quad (5.24)$$

$$\begin{aligned} \left\langle \frac{D\mathbf{v}(\mathbf{x})}{Dt} \right\rangle &= - \int_{\Omega} \frac{\sigma(\mathbf{x}')}{\rho(\mathbf{x}')} \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}' - \\ &\quad - \frac{\sigma(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} \rho(\mathbf{x}') \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}' \end{aligned} \quad (5.25)$$

$$\begin{aligned} \left\langle \frac{DE(\mathbf{x})}{Dt} \right\rangle &= - \frac{\sigma(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} \rho(\mathbf{x}') \mathbf{v}(\mathbf{x}') \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}' + \\ &\quad + \frac{\sigma(\mathbf{x}) \mathbf{v}(\mathbf{x})}{[\rho(\mathbf{x})]^2} \int_{\Omega} \rho(\mathbf{x}') \nabla W(|\mathbf{x} - \mathbf{x}'|, h) d\mathbf{x}', \end{aligned} \quad (5.26)$$

The final step is to convert the continuous volume integrals to sums over discrete interpolation points.

Finally, after a few arrangements in order to improve the consistency between all equations, the most common form of the SPH discretised conservation equations are obtained:

$$\left\langle \frac{D\rho_I}{Dt} \right\rangle = \rho_I \sum_{J=1}^N \frac{m_J}{\rho_J} (\mathbf{v}_J - \mathbf{v}_I) \nabla W(|\mathbf{x}_I - \mathbf{x}_J|, h), \quad (5.27)$$

$$\left\langle \frac{D\mathbf{v}_I}{Dt} \right\rangle = - \sum_{J=1}^N m_J \left( \frac{\sigma_J}{\rho_J^2} + \frac{\sigma_I}{\rho_I^2} \right) \nabla W(|\mathbf{x}_I - \mathbf{x}_J|, h) \quad (5.28)$$

$$\left\langle \frac{DE_I}{Dt} \right\rangle = - \frac{\sigma_I}{\rho_I} \sum_{J=1}^N m_J (\mathbf{v}_J - \mathbf{v}_I) \nabla W(|\mathbf{x}_I - \mathbf{x}_J|, h) \quad (5.29)$$

## 5.4 Kernel Function

To perform the spatial discretisation one has to define the kernel function. Numerous possibilities exist, and a large number of kernel function types are discussed in literature, ranging from polynomial to Gaussian. The most common is the B-spline kernel that was proposed by Monaghan [59]:

$$W(v, h) = \frac{C}{h^D} \begin{cases} (1 - \frac{3}{2}v^2 + \frac{3}{4}v^3), & v < 1 \\ \frac{1}{4}(2 - v)^3, & 1 \leq v \leq 2 \\ 0, & \text{otherwise.} \end{cases} \quad (5.30)$$

Where:  $v = (|\mathbf{x} - \mathbf{x}'|)/h$ ,  $D$  is the number of dimensions of the problem (i.e. 1, 2 or 3),  $C$  is the scaling factor which depends on the number of dimensions and ensures that the consistency conditions 2 and 3 are satisfied:

$$\begin{cases} \frac{2}{3}, & D = 1, \\ \frac{10}{7\pi}, & D = 2, \\ \frac{1}{\pi}, & D = 3. \end{cases} \quad (5.31)$$

The derivatives of the kernel function have the following property

$$\frac{\partial W(|\mathbf{x} - \mathbf{x}'|, h)}{\partial x'_k} = - \frac{\partial W(|\mathbf{x} - \mathbf{x}'|, h)}{\partial x_k}.$$

It is important not to forget the dimensionality of the kernel function  $W$ . For instance, in one dimension,  $W$  has dimension of  $\text{length}^{-1}$  and the dimension of  $\frac{\partial W}{\partial x}$  is thus  $\text{length}^{-2}$ . The mass,  $m$ , should be interpreted as mass per unit area, with the cross-sectional area numerically equal to one.

## 5.5 Variable Smoothing Length

If large deformations occur, particles can move apart. In the case of conventional (Eulerian) SPH, if the smoothing length remains constant, the particle spacing can become so large that particles will no longer interact. On the other hand, in compression, the number of neighbour particles within the support can become large, which can significantly slow down the calculation. In order to avoid these problems, Gingold and Monaghan [28] suggested that it would be preferable to allow  $h$  for any particle to be related to its density according to

$$h = G \left( \frac{m}{\rho} \right)^{1/n} \quad (5.32)$$

where  $n$  is the number of dimensions and  $G$  is a constant approximately  $G \approx 1.3$ . Benz [16] proposed the use of a variable smoothing length with the intent of maintaining a healthy neighbourhood as the continuum deforms. His equation for evolution of  $h$  is:

$$h = h_0 \left( \frac{\rho_0}{\rho} \right)^{1/n} \quad (5.33)$$

where  $h_0$  and  $\rho_0$  are initial smoothing length and density and  $n$  is the number of dimensions of the problem. Another frequently used equation for evolution of  $h$  based on conservation of mass is:

$$\frac{dh}{dt} = \frac{1}{n} h \nabla \cdot \mathbf{v}. \quad (5.34)$$

According to Monaghan [65] and Price and Monaghan,  $h$  should be determined from the summation equations so that it is consistent with the density obtained from the summation, i.e.  $\rho_I = \sum_J m_J W(|\mathbf{x}_I - \mathbf{x}_J|, h_J)$ , where  $\rho_I$  is either estimated from the SPH summation.

It is important to observe that the methods for variation of  $h$  described above are empirical in nature. The evolution of  $h$  if treated rigorously should be coupled and consistent with the discretised form of the conservation equations. A rigorously derived Eulerian SPH with a variable smoothing length requires further research and is still outstanding.

## 5.6 Neighbour Search

In the SPH method the interpolation points move with the continuum and as a consequence neighbours of a particle are not fixed. This implies that the SPH kernel approximation of any field variable for a particle  $I$  requires as a first step the search of the neighbouring particles  $j$  that are within the kernel support of particle  $I$ .

Therefore, the neighbour search is an important and CPU time consuming step in an SPH computation. Based on the distance between the interpolation points the neighbour search routine must list the particles that are inside the neighbourhood of each particle at each time step. A direct search between every particle is particularly inefficient, requiring a time proportional to  $N^2$ , where  $N$  is the total number of particles.

A bucket sort algorithm is more efficient. In this method, an underlying grid of cells of size  $2h_{MAX}$  is generated and the particles are sorted according to the box, within a background grid, in which they are located (Figure 5.2). The total extent of the grid is defined to contain all particles and is updated as the problem evolves. Then for each particle, the neighbours are searched among the particles contained in the same box and its neighbouring boxes. This allows the computational time to be cut down to a time proportional to  $N \log N$ , Monaghan and Lattanzio [60].

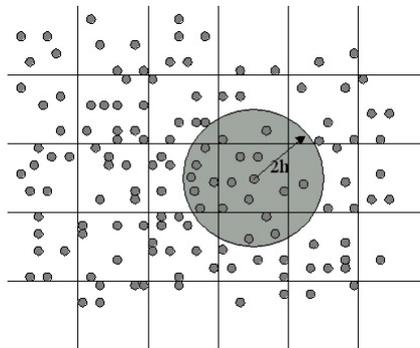


Figure 5.2: Bucket sort and neighbour search.

Each particle  $I$  carries the information about the box that currently contains the particle. Then, to determine the neighbour list of the particle a search is performed over all particles contained in the same box and its neighbouring boxes. This results in a search over three boxes in  $1D$ , nine boxes in  $2D$  and 27 boxes in  $3D$ .

Good algorithm design can minimise the computational cost of the search. For example the coordinates of each particle can also be stored in an integer format which reduces the

time for data access time. The integer coordinates  $\text{int } x_{int}^i$  are computed from the particle position vector  $\mathbf{x}_I$  as:

$$x_{int}^i = \text{floor} \left( \frac{N_{SUB}}{2h_{MAX} x_I^i} \right). \quad (5.35)$$

Where  $N_{SUB}$  is the number of box subdivisions, typically  $N_{SUB} = 1000$ .

## 5.7 SPH Shortcomings

As mentioned in the introduction, the basic SPH method has been shown to have several problems:

- Consistency,
- Tensile instability,
- Zero-energy modes.

### 5.7.1 Consistency

The SPH method even in its continuous form is inconsistent within  $2h$  of the domain boundaries due to the kernel support incompleteness. In its discrete form the method loses its  $0^{th}$  order consistency not only in the vicinity of boundaries but also over the rest of the domain if particles have an irregular distribution. Meglicki [56] showed that node disorder results in a systematic error. Therefore a proper SPH grid should be as regular as possible and not contain large discrepancies in order to perform most accurate simulation.

First order consistency of the method can be achieved in two ways. Firstly, by correcting the kernel function, and secondly, by correcting the discrete form of the convolution integral of the SPH interpolation. Johnson [40] uses this correction procedure and proposed the Normalised Smoothing Function. Vignjevic [97] also implemented a kernel normalisation and correction to lead to a Corrected Normalised Smooth Particle Hydrodynamics (CNSPH) method which is first order consistent. The full derivation of this correction is given below. In SPH methods based on a corrected kernel, it is no-longer possible to ignore boundary conditions. In basic SPH, free surface boundary conditions are not imposed and are simply ignored as variables tends to zero at boundaries because of the deficiency of neighbour particles.

## 5.8 Derivation of Normalised Corrected Gradient SPH formula

Starting from the conventional SPH method of Gingold and Monaghan, which is not even zero order consistent, a number of researchers worked on development of a first order consistent form of SPH. The result of this effort at Cranfield was named Normalised Corrected SPH (NCSPH) and the same term is used in this text for any other similar version of the method.

The approximation of fields using a NCSPH interpolation has been published by Randles and Libersky [78], Vignjevic [97], Bonet [17]. Bonet used properties of the integrals of motion (linear and angular momentum) to derive Normalisation and Gradient Correction for kernel interpolation. This approach lacks generality and does not provide the insight into the origin and the nature of the problem. A full derivation of the correction

proposed by Vignjevic [97], is given below. The derivation is based on the homogeneity and isotropy of space. These are properties of space, which have as a consequence conservation of linear and angular momentum, see Landau [45]. The mixed correction insures that homogeneity and isotropy of space are preserved in the process of spatial discretisation.

An interpolation technique should not affect homogeneity of space. One way of demonstrating this is to prove that the interpolation of the solution space itself is invariant with respect to translational transformation.

Let  $\mathbf{x}_I, I = 1, \dots, n$  be a finite set of  $n$  points in  $D \in R^3$ . Kernel interpolation/approximation of, for instance, a vector field  $\mathbf{F}(\mathbf{x})$  defined on  $D$  consists in finding, for given  $\mathbf{F}(\mathbf{x}_1), \dots, \mathbf{F}(\mathbf{x}_n)$ , we write the general expression for the SPH interpolation of a vector field on  $D$  in the  $(x_1, x_2, x_3)$  coordinate system:

$$\langle \mathbf{F}(\mathbf{x}) \rangle = \sum_J \frac{m_J}{\rho_J} \mathbf{F}(\mathbf{x}_J) W(|\mathbf{x} - \mathbf{x}_J|, h). \quad (5.36)$$

If the field to be interpolated is the domain  $D$  then  $\mathbf{F} = \mathbf{x}$ ,  $\mathbf{x} \in D$  and Equation (5.36) becomes:

$$\langle \mathbf{x} \rangle = \sum_J \frac{m_J}{\rho_J} \mathbf{x}_J W(|\mathbf{x} - \mathbf{x}_J|, h) \quad (5.37)$$

A different coordinate system  $(x'_1, x'_2, x'_3)$ , translated by a vector  $\mathbf{c}$  relative to  $(x_1, x_2, x_3)$  where the approximation  $\langle \mathbf{x} \rangle$  can be expressed as:

$$\langle \mathbf{x}' \rangle = \sum_J \frac{m_J}{\rho_J} \mathbf{x}'_J W(|\mathbf{x}' - \mathbf{x}'_J|, h) = \sum_J \frac{m_J}{\rho_J} \mathbf{x}_J W(|\mathbf{x} - \mathbf{x}_J|, h). \quad (5.38)$$

Note that the translation results in

$$\mathbf{x} = \mathbf{x}' + \mathbf{c} \quad (5.39)$$

and that

$$\mathbf{x}' - \mathbf{x}'_J = \mathbf{x} - \mathbf{x}_J \quad \Rightarrow \quad |\mathbf{x}' - \mathbf{x}'_J| = |\mathbf{x} - \mathbf{x}_J| \quad \Rightarrow \quad W(|\mathbf{x}' - \mathbf{x}'_J|, h) = W(|\mathbf{x} - \mathbf{x}_J|, h),$$

where  $\mathbf{x}'_J$  are the particle position vectors in the new coordinate system.

If the interpolation is independent of the translation of coordinate axes constant  $\mathbf{c}$  has to be approximated exactly, i.e.

$$\langle \mathbf{x} \rangle = \langle \mathbf{x}' + \mathbf{c} \rangle = \langle \mathbf{x}' \rangle + \langle \mathbf{c} \rangle = \langle \mathbf{x}' \rangle + \mathbf{c}. \quad (5.40)$$

By substituting Equations (5.37) and (5.38), the approximations for both  $\mathbf{x}_I$  and  $\mathbf{x}'_I$ , into Equation (5.40) one obtains:

$$\sum_J \frac{m_J}{\rho_J} \mathbf{x}_J W(|\mathbf{x}_I - \mathbf{x}_J|, h) = \sum_J \frac{m_J}{\rho_J} \mathbf{x}'_J W(|\mathbf{x}_I - \mathbf{x}_J|, h) + \mathbf{c} \sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h) \quad (5.41)$$

By comparison of Equation (5.39) and Equation (5.41) it is clear that the discretised space will only be homogeneous, i.e. result in the equivalent approximations, if the following condition is satisfied:

$$\sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h) = 1. \quad (5.42)$$

The requirement defined by Equation (5.42) can be interpreted as the ability of the interpolation scheme to approximate a constant field (zero order consistency). However if this condition is not satisfied an exact approximation of a constant field can be achieved by reorganising equation (5.41).

$$\begin{aligned} \mathbf{c} &= \frac{\sum_J \frac{m_J}{\rho_J} \mathbf{x}'_J W(|\mathbf{x}'_I - \mathbf{x}'_J|, h) - \sum_J \frac{m_J}{\rho_J} \mathbf{x}_J W(|\mathbf{x}_I - \mathbf{x}_J|, h)}{\sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h)} = \\ &= \sum_J \frac{m_J}{\rho_J} \mathbf{x}'_J \frac{W(|\mathbf{x}'_I - \mathbf{x}'_J|, h)}{\sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h)} - \sum_J \frac{m_J}{\rho_J} \mathbf{x}_J \frac{W(|\mathbf{x}_I - \mathbf{x}_J|, h)}{\sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h)} \end{aligned} \quad (5.43)$$

This leads to set of new shape functions, also known as Shepard functions

$$\tilde{W} = \frac{W(|\mathbf{x}_I - \mathbf{x}_J|, h)}{\sum_J \frac{m_J}{\rho_J} W(|\mathbf{x}_I - \mathbf{x}_J|, h)}$$

which have the partition of unity property [7].

Similarly, an interpolation technique should not affect isotropy of space (domain of interpolation  $D$ ). One way of demonstrating this is to prove that the interpolated space is independent of infinitesimal rotational transformations. The same holds for the SPH approximation. If the relative rotation of the two coordinate systems is defined by  $\mathbf{C}$  the rotation transformation tensor than the equivalent transformation of a position vector can be expressed as:

$$\mathbf{x}' = \mathbf{C} \cdot \mathbf{x} \quad (5.44)$$

where  $\mathbf{C}$  is an orthonormal tensor. For infinitesimal rotations this transformation can also be stated as:

$$\mathbf{x}' = \mathbf{x} - \Delta\varphi \times \mathbf{x} \quad (5.45)$$

where  $\Delta\varphi$  is an infinitesimal rotation vector.

If an SPH approximation is to maintain the isotropy of space then the approximation has to satisfy the following condition:

$$\langle \mathbf{x}' \rangle \equiv \langle \mathbf{C} \cdot \mathbf{x} \rangle. \quad (5.46)$$

For the case of infinitesimal rotations, the approximation of the product  $\langle \mathbf{C} \cdot \mathbf{x} \rangle$  can be replaced by the product of applications  $\langle \mathbf{C} \rangle \cdot \langle \mathbf{x} \rangle$  with the difference proportional to the second and higher order infinitesimals. This allows for the equation (5.46) to restated as

$$\langle \mathbf{x}' \rangle = \langle \mathbf{C} \rangle \cdot \langle \mathbf{x} \rangle. \quad (5.47)$$

In order to preserve the properties of approximation  $\langle \mathbf{x} \rangle$  the rotational matrix has to be approximately exactly, i.e.  $\langle \mathbf{C} \rangle = \mathbf{C} \Rightarrow \langle \mathbf{x}' \rangle = \mathbf{C} \cdot \langle \mathbf{x} \rangle$ . To consider this condition one can start by rewriting

$$\begin{aligned} \mathbf{x}' &= \mathbf{x} - \Delta\varphi \times \mathbf{x} = \mathbf{x} - \nabla(\Delta\varphi \times \mathbf{x}) \cdot \mathbf{x} = \\ &= \mathbf{x} - \varphi^x \mathbf{x} = (\mathbf{I} - \varphi^x) \cdot \mathbf{x}, \end{aligned} \quad (5.48)$$

where  $\boldsymbol{\varphi}^x$  is a skew-symmetric dyadic:

$$\boldsymbol{\varphi}^x = \begin{bmatrix} 0 & -\Delta\varphi_z \mathbf{e}_1 \mathbf{e}_2 & \Delta\varphi_y \mathbf{e}_1 \mathbf{e}_3 \\ \Delta\varphi_z \mathbf{e}_2 \mathbf{e}_1 & 0 & -\Delta\varphi_x \mathbf{e}_2 \mathbf{e}_3 \\ -\Delta\varphi_y \mathbf{e}_3 \mathbf{e}_1 & \Delta\varphi_x \mathbf{e}_3 \mathbf{e}_2 & 0 \end{bmatrix}. \quad (5.49)$$

For small rotations, the rotation transformation tensor is given by:

$$\mathbf{C} = \mathbf{I} - \boldsymbol{\varphi}^x. \quad (5.50)$$

The approximation of the rotated coordinates is:

$$\langle \mathbf{x}' \rangle \equiv \langle \mathbf{C}\mathbf{x} \rangle = \langle \mathbf{C} \rangle \langle \mathbf{x} \rangle = \langle \mathbf{I} - \boldsymbol{\varphi}^x \rangle \langle \mathbf{x} \rangle. \quad (5.51)$$

This means that the requirement on the interpolation is:

$$\mathbf{I} - \boldsymbol{\varphi}^x = \langle \mathbf{I} - \boldsymbol{\varphi}^x \rangle \quad (5.52)$$

or

$$\boldsymbol{\varphi}^x = \langle \boldsymbol{\varphi}^x \rangle. \quad (5.53)$$

Expanding this expression leads to:

$$\begin{aligned} \langle \boldsymbol{\varphi}^x \rangle &= \sum_J \frac{m_J}{\rho_J} \Delta\boldsymbol{\varphi} \times \mathbf{x}_J \nabla W(\mathbf{x}_I - \mathbf{x}_J, h) = \sum_J \frac{m_J}{\rho_J} (\boldsymbol{\varphi}^x \mathbf{x}_J) \nabla W(\mathbf{x}_I - \mathbf{x}_J, h) = \\ &= \boldsymbol{\varphi}^x \sum_J \frac{m_J}{\rho_J} \mathbf{x}_J \otimes \nabla W(\mathbf{x}_I - \mathbf{x}_J, h). \end{aligned} \quad (5.54)$$

Therefore to preserve space isotropy, i.e.  $\boldsymbol{\varphi} = \langle \boldsymbol{\varphi} \rangle$  the following condition has to be satisfied.

$$\sum_{J=1}^{nbr} \frac{m_J}{\rho_J} \mathbf{x}_J \otimes \nabla W(\mathbf{x}_I - \mathbf{x}_J, h) = \mathbf{I}. \quad (5.55)$$

	Space Homogeneity	Space Anisotropy
Condition which has to be satisfied	$\sum_{J=1}^{nbr} \frac{m_J}{\rho_J} W(\mathbf{x}_I - \mathbf{x}_J, h) = 1$	$\sum_{J=1}^{nbr} \frac{m_J}{\rho_J} \mathbf{x}_J \otimes \nabla W(\mathbf{x}_I - \mathbf{x}_J, h) = \mathbf{I}$
Normalised - Corrected form	$\tilde{W}_{IJ} = \frac{W(\mathbf{x}_I - \mathbf{x}_J, h)}{\sum_{J=1}^{nbr} \frac{m_J}{\rho_J} W(\mathbf{x}_I - \mathbf{x}_J, h)}$	$\tilde{\nabla} \tilde{W}_{IJ} = \frac{\nabla W(\mathbf{x}_I - \mathbf{x}_J, h)}{\left( \sum_{J=1}^{nbr} \frac{m_J}{\rho_J} \otimes \nabla W_{IJ} \right)^{-1}}$

Table 5.1: Corrected forms of the kernel function and its gradient.

The form of the normalised kernel function and the approximation of the first order derivatives which provides first order consistency is given in Table 5.1 below.

Using the NCSPH approximations the conservation equations assume the following form:

$$\left\langle \frac{D\rho_I}{Dt} \right\rangle = \rho_I \sum_{J=1}^{nbr} \frac{m_J}{\rho_J} (\mathbf{v}_J - \mathbf{v}_I) \cdot \tilde{\nabla} \tilde{\mathbf{W}}_{IJ}, \quad (5.56)$$

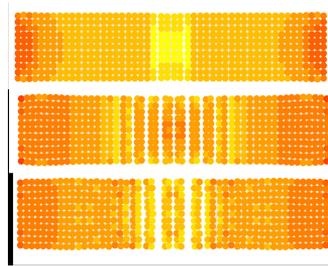
$$\left\langle \frac{D\mathbf{v}_I}{Dt} \right\rangle = - \sum_{J=1}^{nbr} m_J \left( \frac{\sigma_I}{\rho_I^2} + \frac{\sigma_J}{\rho_J^2} \right) \cdot \tilde{\nabla} \tilde{\mathbf{W}}_{IJ}, \quad (5.57)$$

$$\left\langle \frac{De}{Dt} \right\rangle = - \sum_{J=1}^{nbr} m_J (\mathbf{v}_J - \mathbf{v}_I) \cdot \tilde{\nabla} \tilde{\mathbf{W}}_{IJ}. \quad (5.58)$$

## 5.9 Tensile Instability

A Von Neumann stability analysis of the SPH method was conducted Swegle et al. [87] and Balsara [7] separately. This has revealed that the SPH method suffers from a tensile instability. This instability manifests itself as a clustering of the particles, which resembles fracture and fragmentation, but is in fact a numerical artefact, see Figure 5.3

Swegle concluded that the instability doesn't result from the numerical time integration algorithm, but rather from an effective stress resulting from a non-physical negative modulus being produced by the interaction between the constitutive relation and the kernel interpolation. In other words the kernel interpolation used in spatial discretisation changes the nature of original partial differential equations. These changes in the effective stress amplify, rather than reduce, perturbations in the strain.



**Figure 5.3:** Typical collocated Eulerian SPH behaviour under tension. Although the linear elastic model was used for this simulation (i.e. no fracture is included in the constitutive model), unphysical fracture of the 2 - D specimen occurs as a consequence of numerical instability in areas of high tensile stresses.

From Swegle's stability analysis it emerged that the criterion for stability was that:

$$W'' \sigma > 0 \quad (5.59)$$

where  $W''$  is the second derivative of  $W$  with respect to its argument and  $\sigma$  is the stress, negative in compression and positive in tension. A stability analysis leading to the stability condition (5.60) is given at the end of this section.

This criterion states that instability can also occur in compression, not only in tension. Indeed, if the slope of the derivative of the kernel function is positive, the method is

unstable in tension and stable in compression and if the slope is negative, it is unstable in compression and stable in tension.

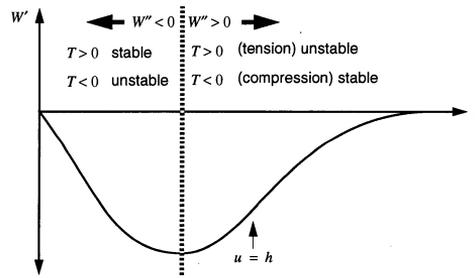


Figure 5.4: Stability regimes for the B-spline kernel function (Swegle, 1994).

The fact that this instability manifests itself most often in tension can be explained. Figure 5.4 shows the stability regime for the *B*-spline kernel function. The minimum of the derivative is situated at  $u = \frac{2}{3}h$ . In standard configurations, the smoothing length is 1.2 to 1.3 times the particle spacing. Thus, standard configurations are unstable in tension. This explains why this unstable phenomenon is generally observed in tension and hence, its misleading name "tensile instability".

In order to remedy this problem several solutions have been proposed. Guenther [29] and Wen et al. [105] proposed a solution known as Conservative Smoothing. Randles and Libersky [73] proposed adding dissipative terms, an approach related to conservative smoothing. Dyka and Ingel [23] proposed an original solution by using a non-collocated discretisation of stress and velocity points. At one set of points the stresses are evaluated, while the momentum equation is calculated at another set of points. The 'stress' points are equivalent to the Gauss quadrature points in FE, the other set of points is equivalent to the element nodes. This approach was extended to two dimensions, in combination with kernel normalisation, by Vignjevic and Campbell [97]. Other solutions were proposed, for instance see Monaghan [65] who proposes the addition of an artificial force to stabilise the computation. Recently Randles and Libersky [78] combined MLS interpolation with the stress and velocity point approach. They called this approach the Dual Particle Dynamics method.

The conservative smoothing and the artificial repulsive forces methods have limited applicability and have to be used with caution because they may affect the strength of material being modelled. At present, the most promising approach is non-collocational spatial discretisation. This problem is in the focus of attention of a number of researchers working on mesh-less methods.

### 5.9.1 Stability analysis of conventional (Eulerian) SPH

For the analysis of the SPH momentum equation in current configuration, the following 1 –  $D$  SPH momentum equation with nodal integration is considered:

$$m_I \ddot{u}_I = - \sum_{J \in S} V_J W'_I(x_I - x_J, h) \sigma_J. \quad (5.60)$$

Where:  $V_J$  is the current volume of particle  $J$ ,  $\sigma_J$  is the Cauchy stress of particle  $J$  and

$$W'_I(x_I - x_J, h) = \left. \frac{\partial W_I(x_I - x', h)}{\partial x'} \right|_{x' = x_J}.$$

In order to introduce the displacement perturbation on the right hand side of (5.60), the current volume  $V_J$  was expressed in terms of the initial particle mass and density as:

$$V_J = \left. \frac{m}{\rho} \right|_J \quad (5.61)$$

where the current density in (5.61) was defined as

$$\rho = J^{-1} \rho_0. \quad (5.62)$$

The substitution of equation (5.62) and equation (5.61) into equation (5.60) and using the fact that in  $1D$   $J = F$ , yields:

$$m_I \ddot{u}_I = - \sum_{J \in S} \frac{m_J}{\rho_0} W'_I(x_I - x_J, h) F_J \sigma_J. \quad (5.63)$$

In equation (5.63), the deformation gradient is expressed with respect to the current configuration, hence:

$$F_J = \left. \frac{\partial x}{\partial X} \right|_J = \left. \frac{\partial u}{\partial X} \right|_J + 1 = \left. \frac{\partial u}{\partial x} \frac{\partial x}{\partial X} \right|_J + 1 = \frac{1}{1 - \left. \frac{\partial u}{\partial x} \right|_J}. \quad (5.64)$$

Equation (5.63) is linearised using perturbations  $\bar{u} = u + \tilde{u}$ ,  $\bar{x} = x + \tilde{x}$ ,  $\bar{F} = F + \tilde{F}$  and  $\bar{\sigma} = \sigma + \tilde{\sigma}$  as follows:

$$m_I \ddot{\bar{u}}_I = - \sum_{J \in S} \frac{m_J}{\rho_0} W'_I(\bar{x}_I - \bar{x}_J, h) \bar{F}_J \bar{\sigma}_J, \quad (5.65)$$

$$m_I \ddot{\bar{u}}_I = - \sum_{J \in S} \frac{m_J}{\rho_0} W'_I(\bar{x}_I - \bar{x}_J, h) (\sigma + \tilde{\sigma})(F + \tilde{F}), \quad (5.66)$$

which yields

$$m_I \ddot{\bar{u}}_I = - \sum_{J \in S} \frac{m_J}{\rho_0} W'_I(\bar{x}_I - \bar{x}_J, h) (\bar{x}_I - \bar{x}_J, h) (\sigma_J F_J + \sigma_J \tilde{F}_J + \tilde{\sigma}_J F_J) \quad (5.67)$$

in (5.67) the product  $\tilde{\sigma} \tilde{F}$  was neglected.

When perturbations  $\bar{x} = x + \tilde{x}$  are introduced, the smoothing function values change as:

$$W(\bar{x}_I - \bar{x}_J, h) = W((x_I + \tilde{x}_I) - (x_J + \tilde{x}_J), h) \quad (5.68)$$

with  $\tilde{x}_I = u_I$  and  $\tilde{x}_J = u_J$  equation (5.69) can be rewritten as

$$W((x_I + \tilde{x}_I) - (x_J + \tilde{x}_J), h) = W((x_I - x_J) + (u_I - u_J), h). \quad (5.69)$$

Using Taylor's series expansion yields:

$$W(\bar{x}_I - \bar{x}_J, h) = W(x_I - x_J, h) + \Delta x W'(x_I - x_J, h) \quad (5.70)$$

where  $\Delta x = u_I - u_J$ . Similarly, the derivative of the kernel function in equation (5.70) can be approximated as:

$$W'(\bar{x}_I - \bar{x}_J, h) = W'(x_I - x_J, h) + \Delta x W''(x_I - x_J, h). \quad (5.71)$$

Hence

$$\begin{aligned} W'(\bar{x}_I - \bar{x}_J, h) - W'(x_I - x_J, h) &= (\tilde{u}_I - \tilde{u}_J) W''(x_I - x_J, h) = \\ &= \tilde{W}'(x_I - x_J, h). \end{aligned} \quad (5.72)$$

Subtracting equation (5.60) from (5.67) yields:

$$\begin{aligned} m_I \ddot{u}_I &= - \sum_{J \in S} \frac{m_J}{\rho_{0J}} W'_I(\bar{x}_I - \bar{x}_J, h) (\sigma_J F_J + \sigma_J \tilde{F}_J + \tilde{\sigma}_J F_J) + \\ &= + \sum_{J \in S} \frac{m_J}{\rho_{0J}} W_I(x_I - x_J, h) \sigma_J F_J. \end{aligned} \quad (5.73)$$

Equation (5.73) after rearranging becomes:

$$\begin{aligned} m_I \ddot{u}_I &= - \sum_{J \in S} \frac{m_J}{\rho_{0J}} \left( -W'_I(\bar{x}_I - \bar{x}_J, h) + W_I(x_I - x_J, h) \right) \sigma_J F_J - \\ &= - \sum_{J \in S} \frac{m_J}{\rho_{0J}} (\bar{x}_I - \bar{x}_J, h) (\sigma_J \tilde{F}_J + \tilde{\sigma}_J F_J). \end{aligned} \quad (5.74)$$

And after substituting equation (5.72) into (5.74) one gets:

$$m_I \ddot{u}_I = - \sum_{J \in S} \frac{m_J}{\rho_{0J}} \left[ \left( \tilde{W}'_I(\bar{x}_I - \bar{x}_J, h) \right) \sigma_J F_J + W'_I(\bar{x}_I - \bar{x}_J, h) (\sigma_J \tilde{F}_J + \tilde{\sigma}_J F_J) \right]. \quad (5.75)$$

To perform the Von Neumann stability analysis a Fourier form of perturbation was substituted into the linearised momentum equation

$$\tilde{u} = u_0 e^{i(\omega t + kX)} \quad (5.76)$$

where  $k$  is wave number and  $\omega$  is frequency.

The resulting dispersion relation for the current configuration is expressed as follows, the reader is referred to Randles et al. [78][79] and Rabczuk et al. [73] for a detailed explanation:

$$\begin{aligned} \omega^2 &= \frac{\bar{C}^{\sigma\tau}}{\rho} \left[ \sum_{j \in S} W'(j\Delta x) \sin(\kappa\Delta x) \right]^2 - \\ &= -\frac{\sigma}{\rho} \left\{ \sum_{j \in S} W''(j\Delta x) [1 - \cos(\kappa j\Delta x)] - \left[ \sum_{j \in S} W'(j\Delta x) \sin(\kappa j\Delta x) \right]^2 \right\} \end{aligned} \quad (5.77)$$

Upon inspection of equation (5.77) it is immediately clear that the three terms contained herein yield three different conditions for stability:

- 1) When the material is unstable, in other words when  $\bar{C}^{\sigma\tau}$  vanishes, which corresponds to the material instability of the continuum ( $\bar{\omega}^2 = (\bar{C}\kappa^2)/(\rho_0)$ ,  $\bar{C} \leq 0$ , equation (5.11), Reveles [81]). This implies (5.77) could have two possible solutions:  $\omega = \pm i\sqrt{x}$ , the negative solution would yield Condition 2 outlined below.
- 2) At the cut-off wave number  $\kappa = \frac{\pi}{\Delta x}$ ,  $j = -1$ , this is the onset of stability for the particle equations in current configuration for an equally spaced particle arrangement. In this case, the first term in the right hand side of equation (5.77) vanishes. Again, two possible solutions exist:  $\omega = \pm i\sqrt{x}$ , the negative solution would yield Condition 2.
- 3) when  $\sigma > 0$  and  $\bar{C}^{\sigma\tau} \neq 0$ , which is the tensile instability identified by Swegle. The second term inside the brackets on the right hand side is always positive, hence, if only

$$\frac{\bar{C}^{\sigma\tau}}{\rho} \left[ \sum_{j \in S} W'(j\Delta x) \sin(\kappa\Delta x) \right]^2 \quad \text{and} \quad \left( \sum_{j \in S} W'(j\Delta x) \sin(\kappa\Delta x) \right)^2$$

existed, the particle equation would be unconditionally stable (i.e. the only possible solution for  $\omega$  is a positive real). However, if  $\sum_{j \in S} W'(j\Delta x) [1 - \cos(\kappa j\Delta x)]$  is sufficiently positive and  $\sigma > 0$ , the product of what is in brackets in equation (5.77) and  $\sigma$  would yield a negative value, hence  $\omega = \pm i\sqrt{x}$  and again, the negative solution would yield Condition 2. This condition is given by Swegle et al. [87][88] as  $\sigma W'' > 0$  which defines the onset of tensile instability of the SPH equations with nodal integration.

Note that stability condition 1) is desirable as it represents the stability of continuum equations. Conditions 2) and 3) are the result of the type of discretisation carried out in SPH. From this analysis it is clear why some special smoothing functions can reduce or eliminate the tensile instability altogether: as long as the smoothing function is carefully selected, the second derivative might yield a negative value which can restore stability in the particle equation. For the cubic spline, (widely employed for SPH approximations) the value of the second derivative of the smoothing function at a distance  $u = 2/3$  (Figure 5.4), from particle  $I$  is positive. Therefore, the onset of tensile instability is defined by  $\sigma W'' > 0$  Swegle [87].

The stability analysis of the Eulerian SPH equations, presented above, has revealed that the stability of the system is governed by three terms: a material stability term, which is desirable since this term is also present in the continuum equations, and two more terms which are the result of the type of discretisation carried out, namely the spurious singular mode term and the tensile instability term.

An effective illustration of the distortion of material instability, by the spatial discretisation, for the case of a hyperelastic material is given in Figure 5.5 taken from [73]. The domains of material stability for the Lagrangian kernel, an Eulerian kernel and that of the governing partial differential equation, i.e. the momentum equation are clearly identified. The stable domains are defined in the space of the two principal stretches,  $\lambda_1$  and  $\lambda_2$ .

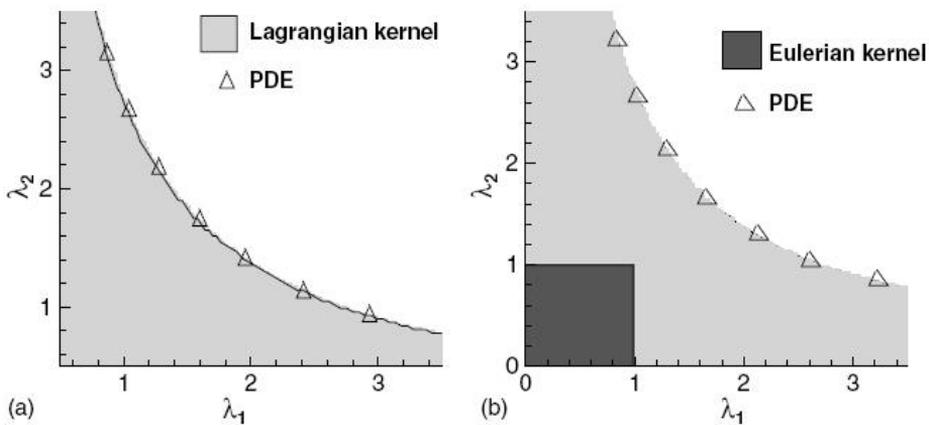


Figure 5.5: (a) Stable domain for MLS particle method with stress point integration and Lagrangian kernel compared to the stable domain for the PDE; (b) Stable domains of MLS particle methods for stress point integration with Eulerian and Lagrangian kernel for hyperelastic material; dashed and solid lines bound the stable domains for Lagrangian and Eulerian kernels, respectively [73].

## 5.10 Zero-Energy Modes

Zero-energy modes are a problem that is not unique to particle methods. These spurious modes, which correspond to modes of deformation characterised by a pattern of nodal displacement that produces zero strain energy, can also be found in the finite difference and finite element methods.

Swegle [87] was first to show that SPH suffers from zero energy modes. These modes arise from the nodal under integration. The fundamental cause is that all field variables and their derivatives are calculated at the same locations (particle positions), which makes the SPH method collocational. For instance, for a 1D oscillatory velocity field, illustrated in Figure 5.6, the kernel approximation would give negligible velocity gradients and consequently stresses at the particles. These modes of deformation are not resisted and can be easily excited by rapid impulsive loading. Another explanation can be found in the origin of the kernel approximation. As the kernel approximation, which is the basis of

SPH, is an interpolation of a set of discrete data, a constant field, can be approximated with a sinusoidal curve/surface if the order of the interpolation is high enough.

If one would approximate the derivative of the velocity field shown in Figure 5.5 with a central difference formula:

$$\left. \frac{dv}{dx} \right|_{x=x_i} = \frac{f(v_{i+1}) - f(v_{i-1})}{x_{i+1} - x_{i-1}} = 0 \quad (5.78)$$

at all sampling points. Hence this mode can not be detected, and can grow un-resisted in other words this mode could grow to a level where it dominates the solution.

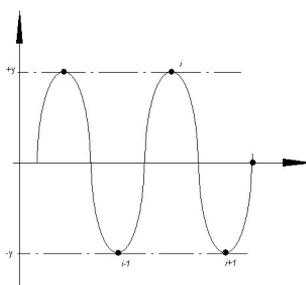


Figure 5.6: Velocity field that corresponds to a zero energy mode of deformation.

Zero energy or spurious modes are characterised by a pattern of nodal displacement that is not rigid body motion but still produces zero strain energy.

One of the key ideas to reduce spurious oscillations is to compute derivatives away from the particles where kernel functions have zero derivatives. Randles [78] proposed a stress point method. Two sets of points are created for the domain discretisation, one carries velocity, and another carries stress. The velocity gradient and stress are computed on stress points, while stress divergence is sampled at the velocity points using stress point neighbours. According to Sweigle et al. [87][88], these spurious modes can be eliminated by replacing the strain measure by a non-local approximation based on gradient approach. Beissel [11] proposed another way to stabilise nodal integration, the least square stabilisation method.

## 5.11 Non-Local properties of SPH

SPH is by nature a nonlocal method, capable of overcoming difficulties related to material softening without any additional regularisation measures. In this chapter, a local damage model resulting in material strain-softening was used in a stable Total-Lagrange SPH code, Vignjevic et al. [101]. The investigation was done by considering a simple uniaxial wave propagation problem in a symmetrically loaded homogeneous bar, in presence of damage induced strain-softening, which is defined by Bažant and Belytschko [8]. They derived an exact solution for given initial and boundary conditions for stress wave propagation

and demonstrated that for the FE spatial discretisation combined with a strain-softening material, deformation localised in a single element. With strain localised in the element undergoing strain-softening, stress wave propagation through the element stopped and the rest of the bar unloaded elastically. Consequently, the numerical results were dependent on the element size, i.e. showed pronounced mesh sensitivity.

This type of instability does not occur when SPH is used to analyse the same problem, i.e. stress wave propagation continues in the presence of strain-softening and the waves continue to propagate within the localisation zone. The SPH smoothing length represents a damage related length scale independent of the particle spacing (spatial discretisation density). This leads to the observation that the SPH method has inherent non-local properties.

### 5.11.1 Theoretical Background of Strain-Softening

According to continuum damage mechanics CDM theory, the properties of an isotropic material, including damage, have a homogeneous distribution within a representative volume element (RVE). Damage is defined by a scalar damage variable  $\omega$ , which has a value between zero and one.  $\omega = 0$  corresponds to no damage and corresponds to complete material failure.

One possible physical interpretation of damage is as a reduction in effective load carrying area within the RVE, as originally proposed by Kachanov (1958). In this case,  $\omega$  is expressed as the ratio of damaged surface area,  $\delta S_D$ , to the original undamaged surface area,  $\delta S$ :

$$\omega = \frac{\delta S_D}{\delta S}. \quad (5.79)$$

This interpretation of damage leads to a constitutive equation expressed in terms of effective stress  $\tilde{\sigma}$ , see Rabotnov [75]. The relationship between the true stress and effective stress can then be derived from a definition of effective load carrying area  $\delta \tilde{S} = \delta S - \delta S_D \tilde{\sigma} = \delta S - \delta S_D \omega \sigma$  and the force equilibrium,  $\tilde{\sigma} \delta \tilde{S} = \sigma \delta S$ :

$$\tilde{\sigma} = \sigma \frac{1}{1 - \omega}. \quad (5.80)$$

Making use of the effective stress, combined with equivalent strain principle, Hooke's law can be expressed in two equivalent forms, i.e.  $\sigma = \tilde{E} \varepsilon_e$  or  $\tilde{\sigma} = E \varepsilon_e$ , where  $\tilde{E}$  is the effective Young's modulus and  $\varepsilon_e$  is the elastic strain. Note that the true stress  $\sigma$  results in the same elastic strain for a damaged material as  $\tilde{\sigma}$  for the virgin material. This provides a relationship between  $\omega$  and  $\tilde{E}$ :

$$\omega = 1 - \frac{\tilde{E}}{E}, \quad \tilde{E} = E(1 - \omega). \quad (5.81)$$

The development of localised deformation is caused by a physical process occurring on a sub-continuum scale. The process is defined by the initiation, growth and interaction of cracks and voids, which finally lead to complete material fracture. In this investigation, as proposed by Rudnicki [76], "localization is defined as instability in the macroscopic constitutive description of inelastic deformation of the material". The instability allows the

constitutive equations of an originally homogeneous material to reach a bifurcation point and become unstable. Consequently, the deformation localises and becomes non-uniform, while outside this instability zone, the material continues to be stable, Rudicki [76].

When the FE method, combined with a conventional local constitutive equation, is applied to model the response of a softening material the results are nonphysical and mesh dependent. This is a consequence of the ill-posed description of the governing differential equations in the material softening zone. In static problems the partial differential equations (PDEs) change from elliptic to hyperbolic in the softening zone, while in dynamic problems they change from hyperbolic to elliptic.

A material is considered to be stable and stay in equilibrium, when the double contraction of stress-rate  $\dot{\sigma}_{ij}$  and strain-rate  $\dot{\epsilon}_{ij}$  is positive. This criterion is also called general bifurcation criterion, according to Neilsen [68], and is true as long as the stress-strain relationship of the material has a positive slope.

$$\dot{\sigma}_{ij}\dot{\epsilon}_{ij} > 0. \quad (5.82)$$

The constitutive equations are written in the rate form, with a piecewise linear relationship between stress and strain rates through the material tangent stiffness tensor  $D_{ijkl}$  as:

$$\dot{\sigma}_{ij} = D_{ijkl}\dot{\epsilon}_{kl} \quad (5.83)$$

Therefore, the inequality (5.82) reads:

$$D_{ijkl}\dot{\epsilon}_{ij}\dot{\epsilon}_{kl} > 0. \quad (5.84)$$

The material becomes unstable when the material reaches its limiting point, which occurs when the condition (5.84) is violated. This criterion defines the bifurcation point and is mathematically expressed as:

$$D_{ijkl}\dot{\epsilon}_{ij}\dot{\epsilon}_{kl} = 0, \quad \dot{\epsilon}_{ij}D_{ijkl}\dot{\epsilon}_{kl} = 0. \quad (5.85)$$

The bifurcation criterion is reached when the tangent stiffness tensor becomes singular (not positive-definite) anymore:

$$\det(D_{ijkl}) = 0. \quad (5.86)$$

The test problem chosen to illustrate nonlocal properties of SPH is based on the 1D stress state dynamic softening problem for which Bažant and Belytschko [8] derived an analytical solution. To make the problem more suited to analysis codes a 1D strain was assumed, then a new analytical solution developed following procedure presented in aforementioned work.

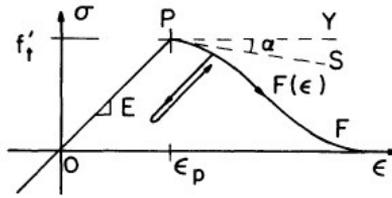


Figure 5.7: Stress-strain diagram of softening material Bažant and Belytschko [8].

Figure 5.7 shows the stress-strain curve of a general strain-softening material, as considered by Bažant and Belytschko [8]. For the linear elastic behaviour between the points  $O$  and  $P$ , the material stiffness is defined by Young's modulus  $E$ . The maximum strength  $f'_t$  is reached for the strain  $\epsilon_p$ . The curve in the strain softening zone (between points  $P$  and  $F$ ) is defined by the function  $F(\epsilon)$ , and the slope of this part of the curve,  $F'(\epsilon)$ , is negative. Function  $F(\epsilon)$  reaches a zero stress for a finite strain  $\epsilon$  or an asymptotic strain  $\epsilon \rightarrow \infty$ . In the original paper of Bažant and Belytschko [8], unloading  $\dot{\epsilon} < 0$  and reloading  $\dot{\epsilon} \geq 0$  is considered to be elastic and occur with the undamaged Young's modulus  $E$ .

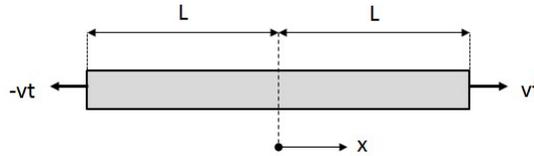


Figure 5.8: Geometry and loading of softening bar, Bažant and Belytschko [8].

The geometry and the loading conditions of the problem are shown in Figure 5.8. The bar length is  $2L$ , material density per unit length is  $\rho$  and the coordinate system is chosen so that the longitudinal coordinate  $x$  is measured from the bar centre. The bar is loaded at both ends with a constant velocity  $v$ , applied in opposite directions.

Two tensile step waves are generated in the bar, one travelling from the right boundary in the negative  $x$ -direction and the other travelling from the left boundary in the positive  $x$ -direction. These two step waves of constant strain meet in the centre of the bar ( $x = 0$ ) at time  $t = \frac{L}{c_e}$ . At this point the strain doubles instantaneously, and the midsection zone of the bar enters the strain-softening regime.

For elastic material response the wave equation is hyperbolic:

$$c_e^2 \frac{\partial^2 u}{\partial x^2} = \frac{\partial^2 u}{\partial t^2} \quad (5.87)$$

where  $c_e$  is the elastic wave speed, which for the 1D state of strain, is:

$$c_e = \sqrt{\frac{E(1-\nu)}{\rho(1-2\nu)(1+\nu)}}. \quad (5.88)$$

The longitudinal displacement function for the linear elastic response is derived from appropriate initial and boundary conditions:

$$u(x,t) = -v \left\langle t - \frac{x+L}{c_e} \right\rangle + v \left\langle t + \frac{x+L}{c_e} \right\rangle \quad (5.89)$$

where the expressions in the brackets  $\langle \cdot \rangle$  need to be positive-definite. The corresponding strain is obtained as:

$$\varepsilon_x = \frac{\partial u}{\partial x} = \frac{v}{c_e} \left[ H \left( t - \frac{x+L}{c_e} \right) + H \left( t + \frac{x-L}{c_e} \right) \right] \quad (5.90)$$

where  $H(\cdot)$  is the Heaviside step function. The stress state induced by this deformation is:

$$\sigma_x = \frac{E(1-\nu)}{(1-2\nu)(1+\nu)} \varepsilon_x \quad (5.91)$$

This assumption of linear elasticity (5.89) only holds for  $t < L/c_e$ , before the centre of the bar enters the strain-softening regime at response time  $t = L/c_e$ , when the two stress step waves meet. At the centre of the bar, the slope of the stress-strain curve (the strain in the bar satisfies the condition:  $\varepsilon_p/2 < \varepsilon \leq \varepsilon_p$ ) becomes negative, i.e.  $F'(\varepsilon) < 0$ , and the wave speed  $c$  becomes imaginary. Consequently, the equation of motion in the softening domain becomes an elliptic PDE:

$$c_e^2 \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial t^2} = 0 \quad \text{with} \quad c^2 = \frac{F'(\varepsilon)}{\rho}. \quad (5.92)$$

Theoretically, the softening is limited to an area of zero width at  $x = 0$ . So, a discontinuity with a displacement jump develops at this point, giving a difference in magnitude of strain  $\varepsilon = 4v \langle t - L/c_e \rangle$ . Strain increases towards infinity and stress drops to zero within the softening zone. Release waves are generated from this point and propagate into the bar.

The infinite strain in the softening domain can be expressed by the Dirac Delta function  $\delta(x)$  as:

$$\varepsilon_x = 4vt - \frac{L}{c_e} \delta(x). \quad (5.93)$$

The solution for the strain field outside the softening zone,  $t > L/c_e$  and  $x < 0$ , (consequently in  $x > 0$  part of the bar, due to symmetry of the problem), is then:

$$\varepsilon_x = \frac{v}{c_e} \left[ H \left( t - \frac{x+L}{c_e} \right) - H \left( t - \frac{L-x}{c_e} \right) + 4c_e t - Lv \delta(x) \right]. \quad (5.94)$$

Following Bažant and Belytschko [8] this analytical solution was used to derive a comparison between an elastic ( $\varepsilon < \varepsilon_p/2$ ) and a strain-softening ( $\varepsilon_p/2 < \varepsilon < \varepsilon_p$ ) wave propagation problem.

Figure 5.9 to Figure 5.10 show the solutions for longitudinal displacement, strain and stress along the bar at time  $t = 3L/2c_e$ , for both the elastic and strain-softening responses of a local continuum. The elastic solution represents continuous wave propagation after

superposition of the waves. The results for the strain-softening solution show clearly the consequences of strain-softening; a displacement discontinuity develops after the superposition in the localisation zone at  $x = 0$  and this zone localises in an area of zero width (see Figure 5.9). This discontinuity cannot propagate outside this zone, as the type of PDE in this zone has changed to elliptic and interaction with areas  $x \neq 0$ , which are governed by hyperbolic PDEs, is not possible. Consequently, strain grows to infinity, as illustrated in Figure 5.10, and simultaneously stress in the localisation zone drops to zero, see Figure 5.11. Outside the localisation zone, the bar unloads as release waves travel to the bar ends. One can observe that the softening zone effectively acts as a free boundary.

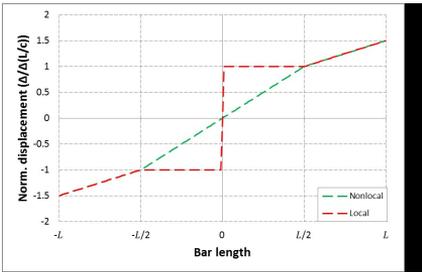


Figure 5.9: Elastic local and nonlocal solutions for normalised longitudinal displacement at  $t = 3L/2c$ .

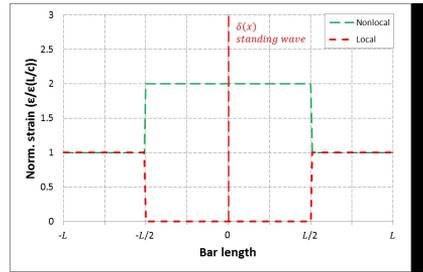


Figure 5.10: Elastic local and nonlocal solutions for normalised longitudinal strain at  $t = 3L/2c$ .

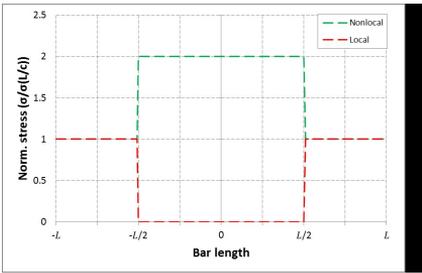


Figure 5.11: Elastic local and nonlocal solutions for normalised longitudinal stress at  $t = 3L/2c$ .

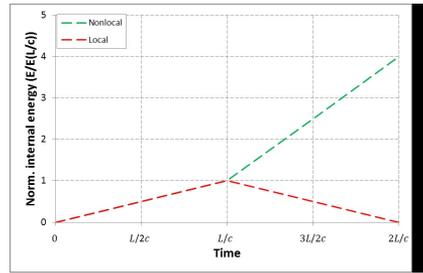


Figure 5.12: Internal energy history for the local and nonlocal solutions.

### 5.11.2 Main Aspects of the Smoothed Particle Hydrodynamics (SPH) Method as Nonlocal Regularisation Method

In the local theory of continuum stress only depends on the deformation history at a single point  $x$ . A nonlocal theory considers additionally the influence of the deformation of surrounding points,  $\xi$ , in a representative volume element (RVE). This is done by substituting the local variable  $\eta(x)$  by a weighted average  $\bar{\eta}(x)$  of the variable in the point's spatial neighbourhood.

Nonlocal integral theory aims to describe spatial interactions with weighted spatial averages. A transformation rule is defined by an integral over a RVE domain, denoted as  $V$  in the integral below:

$$\bar{\eta}(x) = \int_V \alpha(\mathbf{x}, \xi) \eta(\xi) d\xi \quad (5.95)$$

where  $\alpha(\mathbf{x}, \xi)$  is a **weighting function** for a local state variable in the spatial domain  $V$ .

The size of the RVE is quantified by a characteristic length  $\ell$ , also called the internal length. It is understood as a material property which depends on the size of material heterogeneities on the micro-scale. An example of a common weighting function  $\alpha(\mathbf{x}, \xi)$  is Gaussian function:

$$\alpha(r) = \alpha(\|\mathbf{x} - \xi\|) = \frac{1}{(2\pi)^{3/2} \ell^3} \exp\left(-\frac{r^2}{2\ell^2}\right). \quad (5.96)$$

With increase in distance from the point  $\mathbf{x}$ , the influence of the surrounding material reduces and reaches zero at the boundaries of the RVE. This averaging process is often called smoothing.

As already stated, the SPH approximations of field variables are values smoothed over the kernel function domains or, in its discrete form, over a number of neighbours for a given particle  $I$ . This kernel smoothing/interpolation gives the SPH method non-local properties. More specifically, the density, stress and velocity fields in SPH are smoothed (discretised) using the kernel interpolation. Furthermore, in SPH constitutive equations are integrated for all particles locally, i.e. all particles carry information about density, velocity, stress and internal state variables, which makes this method collocational. In FE, based on the isoparametric element formulation, constitutive equations are integrated at the Gauss points and the discrete values for the velocity field are determined for nodal points, which makes this method **non-collocational**.

An outline of a stable Total Lagrangian form of SPH, used in this work, is given below, whilst the full information about this form of SPH can be found in Vignjevic et al. [101].

In the Total Lagrangian formulation the balance equations are written in the initial configuration and expressed in terms of material coordinates. The Total Lagrangian SPH form of the discretised balance equations are given in Table 5.2, Vignjevic et al. [101]. These equations are discretised using the Total Lagrangian kernel function and its derivatives are evaluated in the initial configuration, in terms of the initial coordinates  $x_i^0$  and  $x_j^0$  of the  $i$  and  $j$  particles.

Balance equations in continual and the SPH discrete forms		
mass	$\rho = J^{-1} \rho^0$	$\rho = J^{-1} \rho^0 = [\det(\mathbf{F})]^{-1} \rho^0$
momentum	$\rho^0 \ddot{\mathbf{u}} = \nabla^0 \cdot \mathbf{P} + \mathbf{b}$	$\langle \ddot{\mathbf{u}}_i \rangle = - \sum_{j=1}^{np} m_j \left( \frac{\mathbf{P}_i}{\rho_i^{02}} + \frac{\mathbf{P}_j}{\rho_j^{02}} \right) \nabla_{x_i^0} W( \mathbf{x}_i^0 - \mathbf{x}_j^0 , h^0) + \mathbf{b}$
energy	$\rho^0 \dot{e} = \mathbf{P} : \dot{\mathbf{F}}$	$\langle \dot{e}_i \rangle = \frac{\mathbf{P}_i}{\rho_i} : \sum_{j=1}^{np} \frac{m_j}{\rho_j^0} (\mathbf{v}_i - \mathbf{v}_j) \otimes \nabla_{x_i^0} W( \mathbf{x}_i^0 - \mathbf{x}_j^0 , h^0)$

Table 5.2: Balance equations in continuum and SPH discrete forms Vignjevic at al. [101].

Where:  $\mathbf{F}$  is deformation gradient,  $J = \det \mathbf{F}$  is the Jacobian of deformation gradient,  $\rho$  is material density,  $\mathbf{u}$  is displacement,  $\mathbf{P}$  is first Piola-Kirchhoff stress,  $\mathbf{b}$  is a body force per unit mass,  $e$  is internal energy. The superscript  $^0$  in the equation above indicates initial configuration and dot above a variable denotes time derivative.

This SPH discretisation results in the following equation for the deformation gradient, Vignjevic at al. [101]:

$$\langle \mathbf{F}_i \rangle = - \sum_{j=1}^{np} \frac{m_j}{\rho_j^0} (\mathbf{v}_i - \mathbf{v}_j) \otimes \nabla_{x_i^0} W(|\mathbf{x}_i^0 - \mathbf{x}_j^0|, h^0) \quad (5.97)$$

A normalised corrected version of SPH based on cubic  $B$ -spline kernel function was used (for more details, see Vignjevic at al. [97]). These semi-discretised equations are integrated in time using a central difference integration scheme (explicit time integration). The update of particle positions was performed using a smoothed velocity (XSPH), Randles and Libersky [77].

### 5.11.3 Numerical Experiments for the Evaluation of the SPH Method

The objective of the numerical experiments was to investigate the behaviour of the SPH method, when used with a local continuum damage mechanics (CDM) material model with strain-softening. Then to compare the results with equivalent analyses performed with the FE method. The tests were conducted with an in-house Total-Lagrangian SPH code (MCM) and with the FE code DYNA3D, Lin [50]. The 1D strain, wave propagation problem, described above was used as the benchmark example.

An isotropic elastic material model with damage was used in this study, with a stress-strain relationship is illustrated in Figure 5.13. The onset of strain-softening occurs when strain reaches the damage initiation strain  $\varepsilon_i$ , which corresponds to the maximum strength. After the onset of strain-softening, material strength reduces gradually until it reaches zero at a strain equal to the critical failure strain  $\varepsilon_f$ .

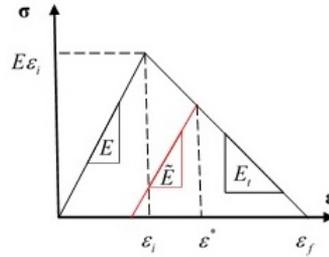


Figure 5.13: Material model with strain-softening implemented into the MCM SPH code and FE code DYNA3D.

The evolution of the damage variable was defined to give linear stress-strain behaviour following the onset of damage growth. In the model,  $E$  is the Young's modulus of a virgin material, which defines material linear elastic behaviour. Material stiffness in the softening regime is defined by the tangent stiffness (softening part of the stress strain curve),  $E_t$ . Material loading/unloading response in the softening regime was defined using the secant stiffness, denoted as  $\tilde{E} = E$ . The secant stiffness is equivalent to virgin material Young's modulus. The parameters  $\varepsilon_i$  and  $\varepsilon_f$  define the initiation and critical failure strains, respectively, and  $\varepsilon^*$  denotes the current strain.

The stiffness of the softening material was defined by the slope of the strain-softening part of the stress-strain curve:

$$E_t = \frac{-E \varepsilon_i}{\varepsilon_f - \varepsilon_i}. \quad (5.98)$$

Using equation (5.80) damage variable  $\omega$  can then be expressed as:

$$\omega = \frac{\varepsilon_f(\varepsilon^* - \varepsilon_i)}{\varepsilon^*(\varepsilon_f - \varepsilon_i)}. \quad (5.99)$$

For  $\varepsilon^* = \varepsilon_i$ , and for  $\varepsilon^* = \varepsilon_f$ , i.e. at the point of material failure  $\omega_f = 1$ .

A number of numerical tests were performed to illustrate the inherent nonlocal properties of the SPH method, where the smoothing length, in addition to its interpolation meaning, represents material characteristic length.

The bar, used in the numerical experiments, has an overall length of  $2\ell = 200 [mm]$  and a square cross section. In the SPH models symmetry planes were used to properly enforce the boundary conditions on the long edges. The origin of the coordinate system was located in the centre of the bar with the  $x$  axis aligned with the bar. All degrees of freedom except for the longitudinal direction are restricted in order to ensure uniaxial strain conditions. The material input data used in the simulations is given in Table 5.3.

Value	Label	Samples Magnitude	Unit
Density	$\rho$	$1.55 \cdot 10^{-9}$	Tonnes/mm <sup>3</sup>
Young's Modulus	$E$	70800	MPa
Poisson's ratio	$\nu$	0.125	
Damage initiation strain	$\varepsilon$	0.022	
Failure strain	$\varepsilon_f$	0.060	

Table 5.3: Input data for isotropic CDM model with linear strain-softening for FE (DYNA3D) and SPH (MCM) codes.

The bar was loaded in tension by applying constant velocity in opposite directions to its ends. In order to induce the softening regime in material, the applied velocity has to be between.

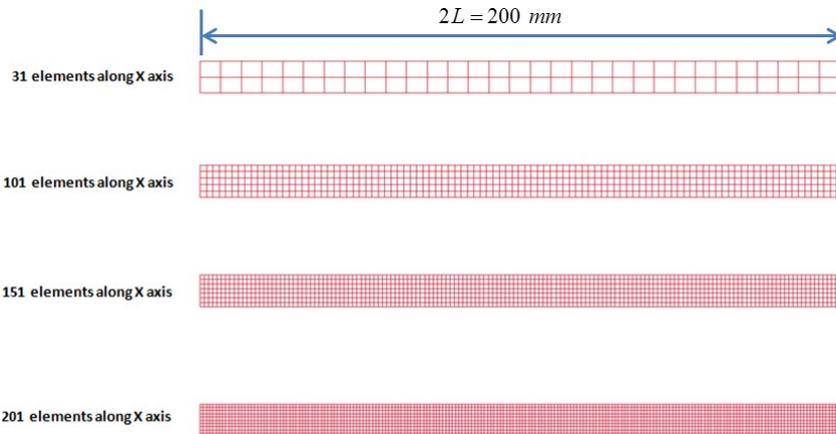


Figure 5.14: Spatial discretisations used in the FE (DYNA3D) simulation of the strain-softening bar.

To provide a reference data set for comparison with SPH, and to illustrate the mesh dependency of the FE model, the bar problem was simulated with the nonlinear transient FE code DYNA3D, using the local constitutive model described above. Four different spatial discretisation densities (mesh densities) were considered: the bar was discretised with 31, 101, 151 and 201 elements along  $x$  axis, as shown in Figure 5.14

Similarly, in the SPH simulations, the bar was discretised with three different particle densities, determined by inter-particle spacings:  $\Delta p = 1.98, 1.32$  and  $0.995 \text{ [mm]}$ , as shown in Figure 5.15.

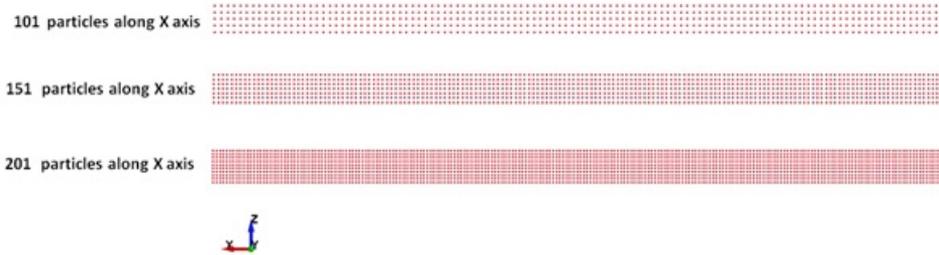


Figure 5.15: Particle discretisation in SPH (MCM) of strain-softening bar.

The smoothing length, which determines the range over which particle velocities and stresses are smoothed, is defined as:

$$h = \lambda \cdot \Delta p \tag{5.100}$$

where  $\lambda$  is a factor, which relates the interparticle spacing to the smoothing length. For the *B*-Spline kernel, the smoothing domain has radius  $2h$ , which is user defined parameter in SPH simulations.

The influence of the smoothing length on the results is illustrated with three numerical experiments summarised in Table 5.4.

- The first experiment investigates the influence of variable  $h$ , where  $h$  was varied by changing the interparticle distance  $\Delta p$  while keeping parameter  $\lambda$  constant at  $\lambda = 1.3$ . This is a typical value used in SPH analyses. In this experiment, the number of neighbours if a given particle, i.e. particles that lie within a spherical domain of radius  $2h$ , was the same for all models.
- The second experiment investigates the influence of  $h$  on the size of softening zone in the case of constant discretisation density (interparticle distance). The interparticle distance was fixed as  $\Delta p = 200[mm]/201$ , and  $\lambda$  varied. Parameter  $\lambda$  was given values of  $\lambda = 1.25, 2.25$  and  $3.25$ , corresponding to 5, 9 and 13 neighbour particles in the loading direction, respectively.
- The third experiment investigated a fixed smoothing length of  $h = 2.5[mm]$  for different discretisation densities. In these tests, both the smoothing length parameter,  $\lambda$  and the interparticle distance  $\Delta p$ , were varied.

Interparticle distance $\Delta p[mm]$	Support domain factor $\lambda(-)$	Physical smoothing length ( $h = \lambda \Delta p$ )	Particles through thickness (y&z Directions)		
Experiment 1: Influence of interparticle distance, $\Delta p$ =variable, $\lambda = 1.3 = \text{constant}$					
200/101	1.3	260/101	5	101	2525
200/151		260/151	9	151	12231
200/201		260/201	11		
Experiment 2: Influence of averaging over several neighbouring particles, $\Delta p$ =variable, $\lambda = 1.3 = \text{variable}$					
200/101	1.25	250/201	11	201	24321
	2.25	150/67			
	2.25	650/201			
Experiment 3: Influence of averaging over several neighbouring particles, $\Delta p$ =variable, $\lambda = 1.3 = \text{variable}$ , $h = 25[mm]$					
200/101	1.2625	2.5	5	101	2525
200/151	1.8875		9	151	12231
200/201	2.5125		11	201	24321

Table 5.4: Summary of the SPH discretisation parameters used in the three numerical experiments.

### 5.11.4 Numerical Results of the Strain-Softening in FE

The conventional strain-softening solutions, obtained with FE, are compared with the analytical solution for longitudinal displacement, strain and stress. The stress-strain curves, obtained at the elements located at  $x = 0$  for the FE models with different mesh densities, are shown in Figure 5.16.

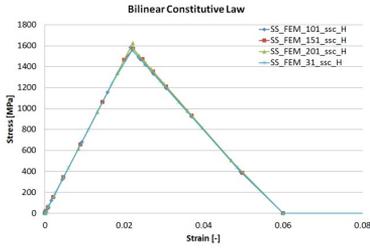


Figure 5.16: Longitudinal stress vs. longitudinal strain curves for the central element for different FE mesh densities.

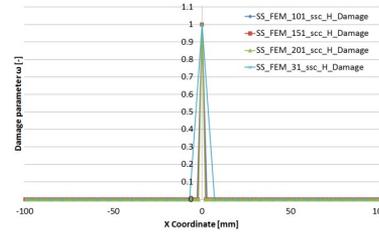


Figure 5.17: Damage distribution for different FE mesh densities at response time  $t = 3/2 \cdot L/c_e$ .

The damage was limited to this element only and propagates towards the bar ends by the deformation of this element which undergoes softening. The size of these elements, i.e. softening zones at response time  $t = 3/2 \cdot L/c_e$  is shown in Figure 5.17 and Figure 5.22. The size of the softening zone in which damage accumulates, was influenced by the initial element size (mesh sensitive).

Figure 5.18, Figure 5.19 and Figure 5.20 respectively show the analytical solution and the FE numerical results for longitudinal displacement, strain and stress at response time  $t = 3/2 \cdot L/c_e$ . The results show a strong dependence on the mesh density in the strain-softening area  $-L/2 \leq x \leq L/2$ , as a consequence of the local strain-softening. It can be observed that numerical results are converging to the analytical solution with the increase in mesh density. The areas outside of  $-L/2 \leq x \leq L/2$  are still governed by the elastic solution.

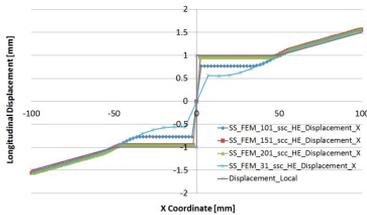


Figure 5.18: Analytical solution and FE results for longitudinal displacement at  $t = 3/2 \cdot L/c_e$ .

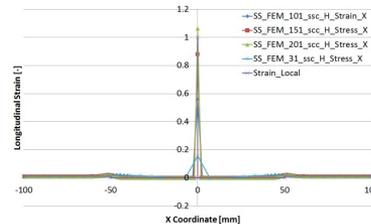


Figure 5.19: Analytical solution and FE results for longitudinal strain at  $t = 3/2 \cdot L/c_e$ .

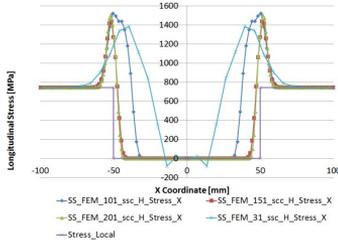


Figure 5.20: Analytical solution and FE results for longitudinal displacement at  $t = 3/2 \cdot L/c_e$ .

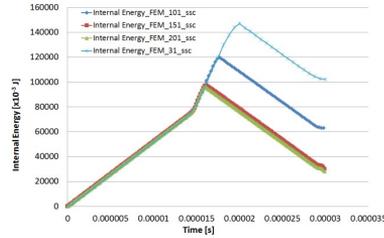


Figure 5.21: FE results for internal energy.

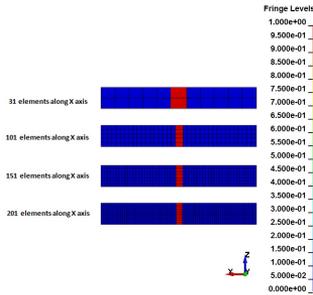


Figure 5.22: Strain localisation at response time  $t = 3/2 \cdot L/c_e$  in a single element due to material strain-softening (fringe level: strain [-]).

Figure 5.22 illustrates the size of the strain/damage localisation zone at response time  $t = 3/2 \cdot L/c_e$ . It can be clearly seen that strain grows in a single element. Consequently, the localisation zone reduces in size with the increase in mesh density and the strain magnitude in the central element increases with the reduction in element size.

### 5.11.5 Strain-Softening in SPH Numerical Results of the Strain-Softening in SPH

**Experiment 1 - Influence of interparticle distance,  $\Delta p = \text{variable}$ ,  $\lambda = 1.3 = \text{const}$ .**

The smoothing length,  $h = 1.3 \cdot \Delta p$ , was varied in these three test cases by changing the inter particle distance, while maintaining a constant number of neighbouring particles. Figure 5.23 shows the stress-strain curves, obtained for the central particle of the bar ( $x = 0$ ). It is clear that for all particle densities, strain-softening behaviour initiates at the centre of the bar and propagates outwards.

Figure 5.24 shows the distribution of damage at response time  $t = 3/2 \cdot L/c_e$ . It can be seen that the width of the damaged area depends on the chosen interparticle distance, as the smoothing length is a function of  $\Delta p$  as  $\lambda$  is constant. The damage affected zone was largest for the largest  $\Delta p$ .

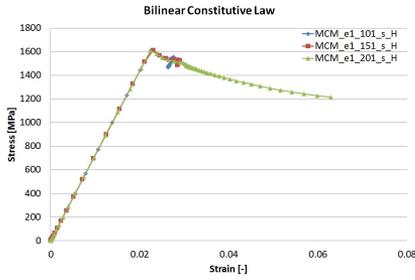


Figure 5.23: Longitudinal stress vs. longitudinal strain curves at the central particle obtained with different particle densities, SPH-experiment 1.

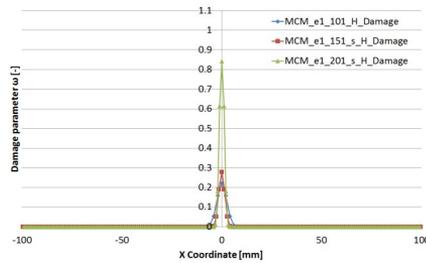
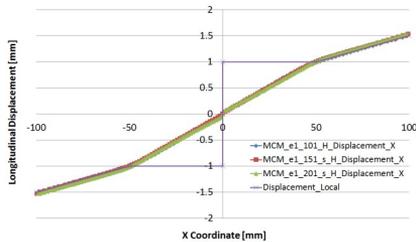
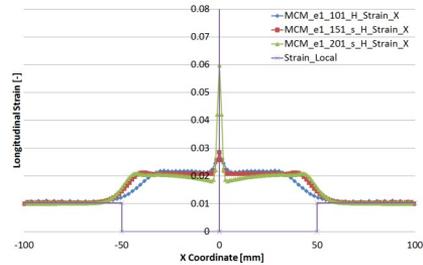


Figure 5.24: : Damage distribution for different particle densities at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 1.

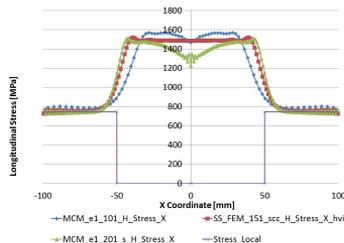
Figure 5.25a, Figure 5.25b and Figure 5.25c respectively show the profiles of longitudinal displacement, strain and stress, plotted along the bar length for the three inter-particle distances, along with the analytical solution. These results indicate that the stress waves continue to propagate through the softening zone and are close to the nonlocal solution. Damage stays limited to the softening zone.



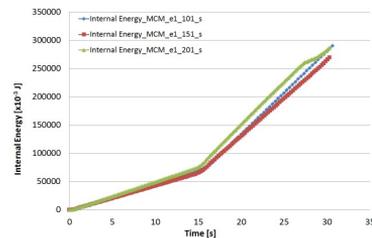
(a) Analytical solution and numerical results for longitudinal displacement at  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 1.



(b) Analytical solution and numerical results for longitudinal displacement at  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 1.



(c) Analytical solution and numerical results for longitudinal displacement at  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 1.



(d) Internal energy for  $0 \leq t \leq 2 \cdot L/c$ ; SPH-experiment 1.

Figure 5.25: My composed figure.

Figure 5.26 shows a graphical representation of the damage zone at response time  $t = 3/2 \cdot L/c_e$ . Damage is averaged over 5 particles along the length of the bar. This number is constant for all particle spacings ( $\Delta p$ ) considered, however this may be difficult to observe in Figure 5.26 because of the significant difference in maximum damage magnitude (see Figure 5.24). The width of the damage zone depends on the chosen interparticle distance.

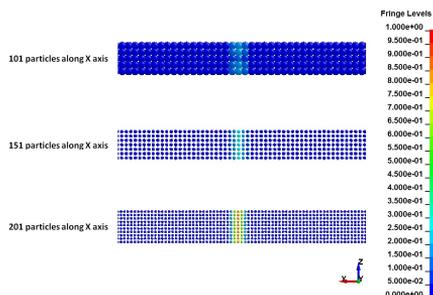


Figure 5.26: Localisation of damage within a limited area  $4h$  in size  $h = 1.3 \cdot \Delta p$  around the bar symmetry plane at response time  $t = 3/2 \cdot L/c_e$  (fringe level: damage [-]. SPH experiment 1.

Development of damage in the centre of the bar, presented in Figure 5.26, shows that the increase of the smoothing length results in increase of the size of the damage zone.

**Experiment 2 - Influence of averaging over several neighbouring particles,  $\Delta p = \text{constant}$ ,  $\lambda = \text{variable}$**

This experiment demonstrates the effects of the smoothing length size, varied by changing the parameter  $\lambda$  and keeping constant the inter-particle distance. The bar with inter-particle distance  $\Delta p = 200[\text{mm}]/201$  was used in these experiments with  $\lambda = 1.25, 2.25$  and  $3.25$ , which correspond to 5, 9 and 13 neighbouring particles respectively.

Again, the particles in the centre of the bar undergo strain-softening, as illustrated in the longitudinal stress - longitudinal strain curves in Figure 5.27. The damage distribution at response time  $t = 3/2 \cdot L/c_e$ , shown in Figure 5.28, indicates that the size of the damaged (softening zone) was dependent on the size of the smoothing domain. Furthermore, maximum peak value of damage was obtained with the smallest smoothing length.

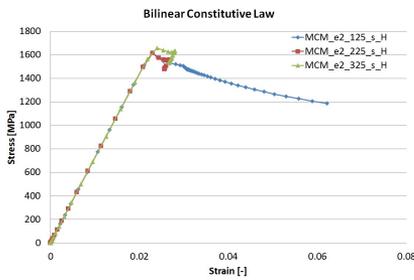


Figure 5.27: Longitudinal stress vs. longitudinal strain curves for the central particle obtained with different values of  $\lambda$ ; SPH-experiment 2.

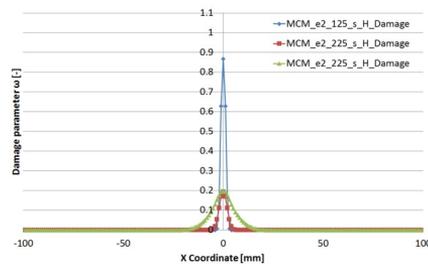


Figure 5.28: : Damage distribution obtained with different values of smoothing lengths at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 2.

Figure 5.29, Figure 5.30, Figure 5.31 and Figure 5.32 respectively show the profiles of longitudinal displacement, strain and stress plotted along the bar length for the three particle densities, along with the analytical solution. These results indicate that the stress waves continue to propagate through the softening zone and are close to the nonlocal solution. Damage stays limited to the softening zone.

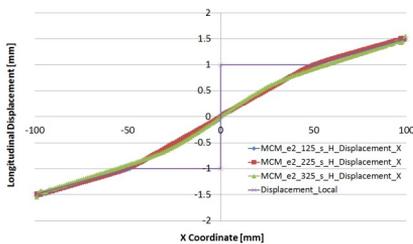


Figure 5.29: Analytical solution and the numerical results for longitudinal displacement at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 2.

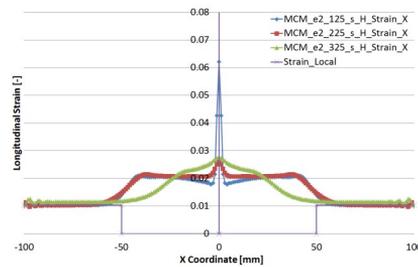


Figure 5.30: : Analytical solution and the numerical results for longitudinal strain at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 2.

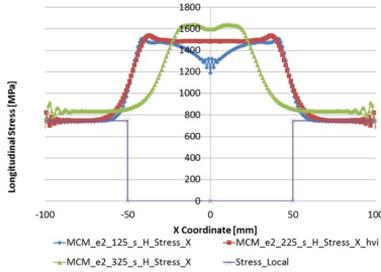


Figure 5.31: Analytical solution and the numerical results for longitudinal displacement at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 2.

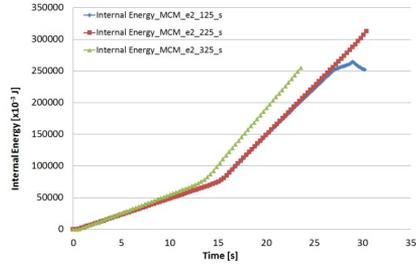


Figure 5.32: : Internal energy for  $0 \leq t \leq 2 \cdot L/c$ ; SPH experiment 2.

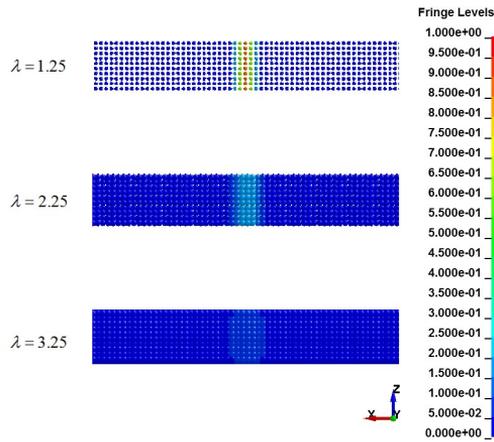


Figure 5.33: Localisation of damage within a limited area  $4h$  in size ( $h = \lambda \cdot 200[mm]/201$ ) around the bar symmetry plane at response time  $t = 3/2 \cdot L/c_e$  (fringe level: damage [-]. SPH experiment 2.

Figure 5.33 shows the size of the damage zone, at response time  $t = 3/2 \cdot L/c_e$ , obtained in the SPH experiment 2. The three models have the same interparticle distance  $\Delta p = 200[mm]/201$ ; however, the smoothing distance  $h$  was varied and consequently the number of neighbouring particles increased with increase of  $\lambda$ .

**Experiment 3 - Influence of constant smoothing length,  $\Delta p = \text{variable}$ ,  $\lambda = \text{variable}$ ,  $h = 2.5[\text{mm}]$**

Experiment 3 demonstrates the behaviour of SPH for different interparticle distances, subjected to a fixed smoothing length size,  $h = 2.5[\text{mm}]$ . The particle density, i.e. the number of neighbouring particles, was changed, by varying the interparticle distance used in the model. Figure 5.34 shows the stress-strain curves for the three particle densities considered. It is clear that all bars underwent linear strain-softening behaviour.

Damage distribution at response time  $t = 3/2 \cdot L/c_e$  is shown in Figure 5.35 for the central particle of the bar. The results indicate that the size of the damaged zone did not depend on the interparticle distance  $\Delta p$ , i.e. particle density. However, the damage peak value was dependant on particle density (the lowest value obtained for the highest particle density).

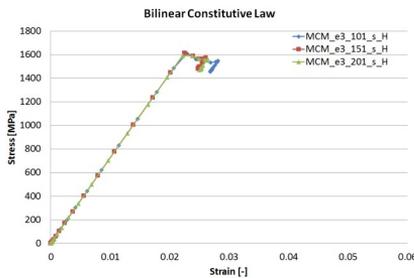


Figure 5.34: Longitudinal stress vs. longitudinal strain curves for the central particle obtained with different values of  $\Delta p$ ; SPH-experiment 3.

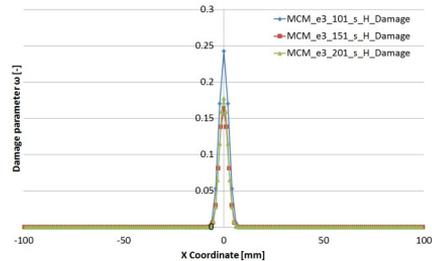


Figure 5.35: : Damage distribution obtained with different particle densities at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 3.

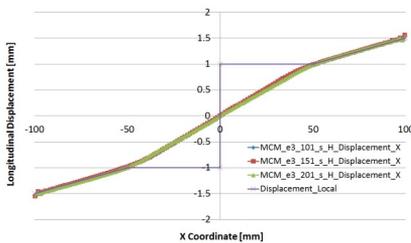


Figure 5.36: Analytical solution and the numerical results for longitudinal displacement at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 3.

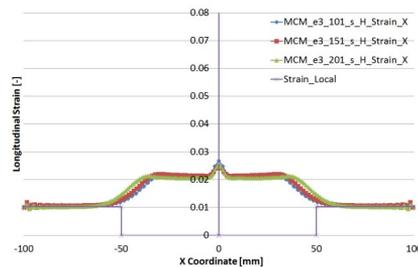


Figure 5.37: : Analytical solution and the numerical results for longitudinal strain at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 3.

The distribution of longitudinal displacement, longitudinal strain and longitudinal stress shown in Figure 5.36, Figure 5.37 and Figure 5.38, respectively, corroborate the above statement and are independent of the particle density. The effects of damage are smoothed out over an increasing number of particles with increasing interparticle distance.

These results indicate that the stress waves continue to propagate through the softening zone and are close to the nonlocal solution.

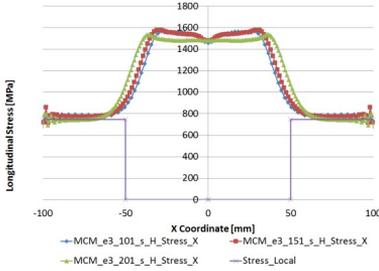


Figure 5.38: Analytical solution and the numerical results for longitudinal stress at response time  $t = 3/2 \cdot L/c_e$ ; SPH-experiment 3.

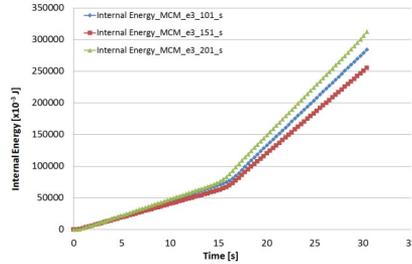


Figure 5.39: Internal energy for  $0 \leq t \leq 2 \cdot L/c$ ; SPH experiment 3.

Finally, the physical size of the damage area, shown in Figure 5.40, is constant with a finite size of  $4h = 10[mm]$  and is independent of the interparticle distance.

Together the results from experiments 2 and 3 show that the width of the damage zone is dependent on the smoothing length  $h$ , not the interparticle distance  $\Delta p$ . This suggests that when modelling a problem including material damage the smoothing length should not be smaller than the characteristic length of the damage and should be set to the characteristic length if the particle resolution permits.

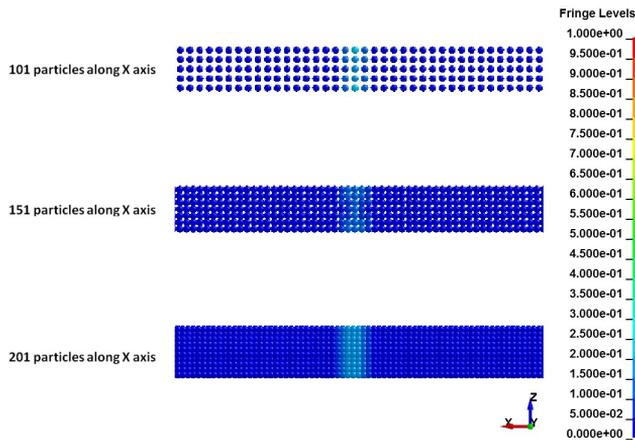


Figure 5.40: Localisation of damage within a limited area  $4h$  in size ( $h = 2.5[mm]$ ) around the bar symmetry plane at response time  $t = 3/2 \cdot L/c_e$  (fringe level: damage [-]). The damage distribution is independent of the interparticle distance  $\Delta p$ ; SPH experiment 3.

## 5.12 Summary

This chapter provides an overview of the development of the SPH method. Especial attention is given to the main shortcomings of the original form of the method namely consistency, tensile instability and zero energy modes. These are important to understand as the SPH method is increasingly used in engineering analysis, and without an understanding of the method inappropriate conclusions may be drawn from numerical results.

The topics covered include:

- A discussion of the kernel interpolation that forms the basis of the SPH spatial interpolation method. In kernel interpolation the variable at a particle is calculated from summing the contribution from all neighbouring particles. The use of a differentiable kernel function allows the spatial gradient of the variable to be approximated.
- An overview of the conventional SPH forms of the conservation equations of Lagrangian continuum mechanics. It should be noted that in these forms the boundary is diffuse and not clearly defined, which is important to understand when modelling solids or liquids.
- A summary of the kernel function, varying smoothing length and neighbour search algorithm that are commonly used in SPH implementations.
- An example of the derivation of a correction necessary to assure first order consistency is given. The conventional SPH method is not first order consistent and is not even zero order consistent except in the special case when particles are evenly distributed.
- The introduction of corrected SPH required boundary conditions to rigorously treated. In the conventional SPH method the deficiency in neighbour particles at the boundary of the domain leads to an error in the interpolation that allows free surfaces to be approximately treated.
- A summary of a stability analysis of SPH is presented and used to explain the so called tensile instability problem. This problem is relevant for solid mechanics simulations where the instability can lead to fracture occurring and preventing accurate analyses of problems involving material fracture. A few proposed solutions to this problem are described. Similar consideration is given with respect to the zero energy modes typical for the collocational SPH method.
- In the SPH simulations of the softening materials, damage distribution is controlled by the smoothing domain used in the kernel interpolation. Consequently, the smoothing domain size represents a material characteristic length. The user can control the damage localisation process by varying the smoothing length  $h$ . Stress waves propagated through the softening zone.
- The numerical results in Section 8 demonstrate that, in the problem considered, SPH performed as nonlocal method and did not suffer from the same instabilities as FE. The sensitivity of results to the spatial discretisation can be removed in SPH by adjusting the smoothing length appropriately, as the smoothing length represents a characteristic length that controls damage/softening localisation. Consequently, physically representative values for a material should be used when modelling damage.

The key aspects of the SPH method covered and discussed any user of the method should be aware of.



## Bibliography

- [1] Abu Al-Rub, R.K. and Voyiadjis, G.Z., 2004. "Analytical and Experimental Determination of the Material intrinsic Length Scale of Strain Gradient Theory from Micro-and Nano-Indentation Experiments," *International Journal of Plasticity*, 20(6), 1139-1182.
- [2] Aifantis, E.C., 1984. "On the Microstructural Origin of Certain Inelastic Models," *Journal of Engineering Materials and Technology* , 106, 326-334
- [3] Aifantis, E.C., 1995. "From Micro-Plasticity to Macro-Plasticity: The Scale- Invariance Approach," *Journal of Engineering Materials and Technology*, 117 (4), 352-355
- [4] Atluri S., Zhu T. (2000) A new meshless local Petrov Galerkin (MLPG) approach in computational mechanics, *Computational Mechanics*, Vol 22, pp 117-127
- [5] Atluri S., H. G. Kim, J. Y. Cho, (2000), A critical assessment of the truly meshless local Petrov Galerkin (MLPG) and local boundary integral equation (LBIE), *Computational Mechanics*, Vol 24, pp 348-372
- [6] Attaway S. W., Heinstejn M.W., and Swegle J.W. (1994) Coupling of smooth particle hydrodynamics with the finite element method. *Nuclear Engineering and Design*, 150:199—205.

- [7] Bammann, D.J. and Solanki, K., 2010. "On Kinematic, Thermodynamic and Coupling of a Damage Theory for Polycrystalline Material," *International Journal of Plasticity*, 26(6), 775793
- [8] Bažant, Z. P. and Belytschko, T. B., 1985. "Wave Propagation In A Strain- Softening Bar: Exact Solution," *Journal of Engineering Mechanics*, 111 (3), 381-389
- [9] Bažant, Z.P. and Pijaudier-Cabot, G., 1988. "Nonlocal Continuum Damage, Localization, Instability and Convergence," *Journal of Applied Mechanics*, 55, 287-293
- [10] Balsara D.S., Von Neumann stability analysis of smoothed particle hydrodynamics – Suggestions for optimal algorithms, *J. Comput. Phys.*, 121 (1995), 357–372.
- [11] Beissel, S. and Belytschko T. (1996). Nodal integration of the element-free Galerkin method. *Computational Methods Applied Mechanical Engineering*, Vol. 139 , pp. 49–71.
- [12] Belytschko T., Krongauz Y., Organ D., Fleming M., Krysl P. (1996) Meshless Methods: An overview and recent developments *Computer methods in applied mechanics and engineering*, 139:3–47.
- [13] Belytschko T., Y.Y. Lu, and L. Gu. (1994) Element-free Galerkin methods. *International Journal for Numerical methods in Engineering*, 37:229–256, 1994.
- [14] Belytschko T., Xiao S. (2002), 'Stability Analysis of Particle Methods with Corrected Derivatives', <http://www.tam.northwestern.edu/xiaop/stable.html>
- [15] Belytschko T, Xiao SP. (2004), A bridging domain method for coupling continua with molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*; 193: pp. 1645-1669.
- [16] Benz, W. (1990). Smooth particle hydrodynamics: a review. In J.R. Buchler, editor, *The Numerical Modelling of Nonlinear Stellar Pulsations*, pp. 269-288. Kluwer Academic Publishers. Cited in: Campbell J. (1998). *Lagrangian hydrocode modelling of hypervelocity impact on spacecraft*. PhD thesis, Cranfield University
- [17] Bonet J. and Kulasegaram S. (1999). Correction and stabilization of smooth particle hydrodynamics methods with application in metal forming simulation. *International Journal of Numerical Methods Engineering*, Vol. 47, pp. 1189–1214.
- [18] Campbell J., Vignjevic R., Libersky L., (2000), A Contact Algorithm for Smoothed Particle Hydrodynamics, *Computer Methods in Applied Mechanics and Engineering*, Vol. 184/1, pp 49-65, March 2000.
- [19] Campbell P. M., (1989), Some new algorithms for boundary value problems in smooth particle hydrodynamics. Technical Report DNA-TR-88-286, Mission Research Corporation.
- [20] Cha S. H. and Whitworth A.P., (2003), Implementations and tests of Godunov-type particle hydrodynamics, *Mon. Not. R. Astron. Soc.*, 340 pp. 73-90

- [21] De Vuyst T., Vignjevic R. and Campbell J. (2005), Modelling of Fluid-Structure Impact Problems using a Coupled SPH-FE solver, *Journal of Impact Engineering*, Volume 31, Issue 8, pp. 1054-1064
- [22] Dilts G. A., (1997), Moving –least squares-particle hydrodynamics I, consistency and stability. *International Journal for Numerical Methods in Engineering*, 44, pp. 1115-55
- [23] Dillon, O.W. and Kratochvill, J., 1970. "A Strain Gradient Theory of Plasticity," *International Journal of Solids and Structures*, 6, 1533-1566
- [24] Dyka C. T., Ingel R.P., (1995), An approach for tension instability in smoothed particle hydrodynamics (SPH). *Computers and Structures*, 57(4), pp. 573-580
- [25] Enakoutsa, K., Leblond, J.B. and Perrin, G., 2007. "Numerical Implementation and Assessment of a Phenomenological Nonlocal Model of Ductile Rupture," *Computer Methods in Applied Mechanics and Engineering*, 196, 1946-1957
- [26] Enakoutsa, K., Solanki, K., Bammann, D. and Ahad, F., 2012. "Using Damage Delocalization to Model Localization Phenomena in Bammann-Chiesa-Johnson Metals," *Journal of Engineering Materials and Technology*, 134(4)
- [27] Fleck, N.A., Muller, G.M., Ashby, M.F., and Hutchinson, J.W., 1994. "Strain Gradient Plasticity: Theory and Experiment," *Acta Metallurgic and Materialia*, 42(2), 475-487
- [28] Gingold, R.A. and Monaghan, J.J. (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly Notices Royal Astronomical Society*, Vol. 181, pp. 375-389.
- [29] Guenther C., Hicks D.L. and Swegle J.W. (1994). Conservative smoothing versus artificial viscosity. Technical Report SAND94-1853.
- [30] Harten A., P. D. Lax P. D. and van Leer B., (1983). On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Review*, V 25, no 1, pp 35 -61.
- [31] Hiermaier S., Peter J., Sauer M., Thoma K. (2001): Coupled FE-Particle Codes Applied to Material Characterization and Crash Simulation. European Conference on Computational Mechanics (ECCM), Krakau, Polen, 26.-29.06..
- [32] Hiermaier S. and Sauer M., (2003), Adaptive FE-Meshfree Modelling for impacts of liquid filled vessels on thin walled structures, *Proceedings of IMECE'03, ASME International Mechanical Engineering Congress and Exposition, Washington DC*, paper IMECE2003-44189
- [33] Huerta A, Fernandez-Mendez S., (2000), Enrichment and coupling of the finite element and meshless method, *International Journal for Numerical Methods in Engineering*; 48: pp. 1615-1636.

- [34] Huerta A, Fernandez-Mendez S, Liu WK., (2004), A comparison of two formulations to blend finite elements and meshfree methods. *Computer Methods in Applied Mechanics and Engineering*; 193 (12–14): pp. 1105-1117
- [35] Hughes T. J. R. (1987). *The finite element method*, Prentice Hall.
- [36] Inutsuka S.-I., Reformulation of smoothed particle hydrodynamics with Riemann solver, *J. Comput. Phys.*, 179 (2002), pp. 238–267
- [37] Johnson, G.R., Petersen, E.H. and Stryk, R.A. (1993). Incorporation of an SPH option in the EPIC code for a wide range of high velocity impact computations. *International Journal of Impact Engineering*, Vol. 14, pp. 385-394.
- [38] Johnson, G.R. (1994). Linking of Lagrangian particle methods to standard finite element methods for high velocity impact computations *Nuclear Engineering and Design*, Vol. 150, pp. 265-274.
- [39] Johnson G. R. and S.R. Beissel. (1996a). Normalised smoothing functions for SPH impact computations. *International Journal for Numerical Methods in Engineering*, 39:2725–2741.
- [40] Johnson, G.R., Stryk, R.A. and Beissel, S.R. (1996b). SPH for high velocity impact computations. *Computer Methods in Applied Mechanics and Engineering*, Vol. 139, pp. 347-373.
- [41] Kachanov, L.M., 1958. “Time of the rupture process under creep conditions”, *Izv Akad Nauk SSR Otd Tech Nauk*, 8(8),26-31
- [42] Kadowaki H, Liu WK., (2005), A multiscale approach for the micropolar continuum model. *Computer Modeling in Engineering and Sciences*; 7(3): pp. 269-282.
- [43] Krongauz Y., Belytschko T. (1997). Enforcement of essential boundary conditions in meshes approximations using finite elements. Submitted to *International Journal for Numerical Methods in Engineering*.
- [44] Krongauz Y., Belytschko T. (1997). Consistent pseudo derivatives in meshless methods *Computer methods in applied mechanics and engineering*, 146:371–386.
- [45] Landau L. D. Lifshitz E.M. (1960). *Mechanics, Course of Theoretical Physics, V 1*, Pergamon Press
- [46] Lattanzo J. C., J.J. Monaghan, H. Pongracic, P. Schwarz. (1996). Controlling penetration. *SIAM, Journal for Scientific and Statistic Computation*, V 7, No 2, pp 591—598.
- [47] Libersky, L.D. and Petschek, A.G., (1990), Smooth particle hydrodynamics with strength of materials. *Advances in the Free Lagrange Method, Lecture Notes in Physics*, Vol. 395, pp. 248-257.
- [48] Libersky L.D., Petschek A.G., Carney T.C., Hipp J.R. and Allahdadi F.A., (1993), High Strain Lagrangian Hydrodynamics: A Three-Dimensional SPH Code for Dynamic Material Response, *Journal of Computational Physics*, Vol. 109, Issue 1, November, pp. 67-75.

- [49] Li S, Liu WK., (2004), Meshfree Particle Methods. Springer: Berlin
- [50] Lin, J.I., 2004. "DYNA3D: A nonlinear, explicit, three-dimensional finite element code for solid and structural mechanics User manual" Lawrence Livermore National Laboratory
- [51] Liu M. B., Liu G. R. and Lam K. Y., (2003), Constructing smoothing functions in smoothed particle hydrodynamics with applications, *J. Comp. Appl. Math*, 155, 263-284
- [52] Liu W. K., Jun S. and Zhang Y. F., (1995), Reproducing kernel particle methods, *Int. J. Num. Meth. Engrng.*, 20, pp. 1081-1106
- [53] Liu W. L., Jun Li S, Adee J. and Belytschko T., (1995), Reproducing kernel particle methods for structural dynamics, *International Journal for Numerical Methods in Engineering*, 38: pp. 1655-1679
- [55] Lucy, L.B., (1977), A numerical approach to the testing of fusion process. *Astronomical journal*, Vol. 88, pp. 1013-1024
- [56] Meglicki Z., Analysis and Application of Smoothed Particle Magneto-Hydrodynamics, PhD thesis, Australian National University
- [57] Mitchell A. R. and Griffiths D. F., (1980), The finite difference method in partial differential equations, John Wiley
- [58] Monaghan J.J., (1982), Why particle methods work, *SIAM Journal on Scientific and Statistical Computing*, 3(4): pp. 422-433
- [59] Monaghan J.J. and Gingold R.A., (1983), Shock simulation by the particle method SPH, *Journal of Computational Physics*, Vol. 52, pp. 374-389
- [60] Monaghan J.J. and Lattanzio J.C., (1985), A refined particle method for astrophysical problems. *Astronomy and Astrophysics*, Vol. 149, Issue 1, pp. 135-143
- [61] Monaghan J.J., Pongracic H., (1985), Artificial viscosity for particle methods. *Applied Numerical Mathematics*, 1: pp. 187-194
- [62] Monaghan J.J., (1989), On the problem of penetration in particle methods. *Journal of Computational Physics*, 82:, pp. 1-15
- [63] Monaghan J.J., (1992), Smoothed Particle Hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30: pp. 543-574
- [64] Monaghan, J.J., (2000), SPH without a Tensile Instability. *Journal of Computational Physics*, Vol. 159, pp. 290-311
- [65] Monaghan J., (2002), SPH compressible turbulence *Mon. Not. R. Astron. Soc.* 335, pp. 843-52

- [66] Moussa B., (2000), Meshless Particle methods: Recent developments for non-linear conservation laws in bounded domain, in *Godunov Methods: Theory and Applications*, E. F. Toro (Editor), Kluwer Academic/Plenum Publishers
- [67] Moussa B., J. P. Vila, (2000), Convergence of the SPH method for scalar nonlinear conservation laws, *SIAM Journal of Numerical Analysis*, Vol 37 number 3, pp 863-887.
- [68] Neilsen, M. K. and Schreyer, H. L., 1993. "Bifurcations in Elastic-Plastic Materials", *International Journal of Solids and Structures*, 30 (4), 521-544
- [69] Parshikov A. N., Medin S. A., Loukashenko I. I. and Milekhin V. A., (2000), Improvements in SPH method by means of inter-particle contact algorithm and analysis of perforation tests at moderate projectile velocities, *Int. J. Impact Engng.*, 24 pp 779-796
- [70] Petschek A. G. and Libersky, L. D., (1993), Cylindrical smoothed particle hydrodynamics. *Journal of Computational Physics*, Vol. 109, pp. 76-83.
- [71] Pijaudier-Cabot, G. and Bažant, Z.P., 1987. "Nonlocal Damage Theory", *Journal of Engineering Mechanics*, ASCE , 113, 1512-1533
- [72] Price J. and Monaghan J., (2004), Smoothed particle magneto-hydrodynamics: II. Variational principles and variable smoothing length terms, *Mon. Not. R. Astron. Soc.* 348, pp. 139-52
- [73] Rabczuk T, Belytschko T, Xiao SP., (2004), Stable particle methods based on Lagrangian kernels. *Computer Methods in Applied Mechanics and Engineering*; 193: pp. 1035-1063
- [74] Ramaswamy, S. and Aravas, N., 1998. "Finite Element Implementation of Gradient Plasticity Models part I: Gradient-Dependent Yield Functions," *Computer Methods in Applied Mechanics and Engineering*, 163, 11-32
- [75] Rabotnov, Y. N., 1968. "Creep Rupture", 12th International Congress of Applied Mechanics, Stanford, Springer-Verlag, Berlin, 342
- [76] Rudnicki, J. W. and Rice, J. R., 1975. "Conditions for the Localization of Deformation in Pressure-Sensitive Dilatant Materials", *Journal of Mechanics and Physics of Solids*, 23, 371-394
- [77] Randles P. W. and Libersky L. D., (1996), Smoothed particle hydrodynamics: Some recent improvements and applications, *Computer methods in applied mechanics and engineering*, 139, pp. 375-408.
- [78] Randles, P.W., Libersky, L.D. and Petschek, A.G., (1999), On neighbors, derivatives, and viscosity in particle codes, in: *Proceedings of ECCM Conference*, Munich, Germany
- [79] Randles, P.W., Libersky, L.D., (2000), Normalised SPH with stress points, *Int. Journal Numerical Methods in Engineering*, Vol. 48, pp. 1445-1462

- [80] Resnyansky, A. D., (2002), DYNA-modelling of the high-velocity impact problems with a split-element algorithm, *International Journal of Impact Engineering*, Vol. 27, Issue 7, pp. 709-727.
- [81] Reveles J., (2006), Development of a Total Lagrangian SPH Code for the Simulation of Solids Under Dynamic Loading, PhD Thesis, Cranfield University
- [82] Sauer M., (2000), Adaptive Kopplung des netzfreien SPH-Verfahrens mit finiten Elementen zur Berechnung von Impaktvorgaengen. Dissertation, Universitaet der Bundeswehr Muenchen, Institut fuer Mechanik und Statik
- [83] Sauer M., Hiermaier S., Scheffer U. (2001): Modeling Penetration Events using FE/MLSPH Adaptive Coupling. Intern. Symp. on the Interaction of the Effects of Munitions with Structures (10th ISIEMS), San Diego, California, USA, 07.-11.06.
- [84] Sauer, M.; Hiermaier, S.; Thoma, K., (2002): Modelling the Continuum/Discrete Transition Using Adaptive Meshfree Methods, Proceedings of the Fifth World Congress on Computational Mechanics (WCCM V), July 7-12, 2002, Vienna, Austria,
- [85] Solanki, K. and Bammann, D.J., 2010. "A Thermodynamic Framework for a Gradient Theory of Continuum Damage," *Acta Mechanica* 213, 27-38
- [86] Stryk R. A., Johnson G.R. and Beissel S.R., (1996), SPH for high velocity impact computations. *Computer methods in applied mechanics and engineering*, 139, pp. 347-373
- [87] Swegle, J.W., Attaway, S.W., Heinstein, M.W., Mello, F.J. and Hicks, D.L., (1994), An analysis of smooth particle hydrodynamics, Sandia Report SAND93-2513.
- [88] Swegle J, (2000) Conservation of momentum and tensile instability in particle methods, Sandia Report 2000-1223
- [89] Takeda H. T., S. M. Miyama, M. Sekiya. (1994). Numerical simulation of viscous flow by smooth particle hydrodynamics, *Progress of Theoretical Physics*, 92 pp 939-960.
- [90] Tvergaard, V. and Needleman, A., 1995. "Effects of Nonlocal Damage in Porous Plastic Solids," *International Journal of Solids Structures*, 32, 1063-1077
- [91] Tvergaard, V. and Needleman, A., 1997. "Nonlocal Effects on Localization in a Void-sheet", *International Journal of Solids Structures*, 34, 2221-2238
- [92] Van der Vegt L.J. J. W., Van der Ven H. and Boelens O. J., (1995), Discontinuous Galerkin methods for partial differential equations, in *Godunov Methods: Theory and Applications*.
- [93] Van Leer B., (1979), Towards the ultimate conservative difference scheme, *J. Comput. Phys.*, 32 pp. 101-136
- [94] Van Leer B., (2006), Upwind and high-resolution methods for compressible flow: From donor cell to residual-distribution schemes, *Commun. Comput. Phys.*, 1, pp. 192-206

- [95] Vila J. P., On particle weighted methods and smooth particle hydrodynamics, *Math. Models Methods Appl. Sci.*, 9 (1999) 2, pp. 161-209
- [96] Vignjevic R., De Vuyst T., Campbell J., (2000), Modelling of Spall in an Anisotropic Aluminium Alloy, *Journal of Space Debris*, Volume 2, Number 4, pp. 225-232
- [97] Vignjevic R., Campbell J., Libersky L., (2000), A Treatment of Zero Energy Modes in the Smoothed Particle Hydrodynamics Method, *Computer Methods in Applied Mechanics and Engineering*, Vol. 184/1, pp. 67-85
- [98] Vignjevic R., De Vuyst T., Campbell J., (2002a), The use of a homogeneous repulsive force for contact treatment in SPH, *Fifth World Congress on Computational Mechanics*, Vienna, Austria, July 7-12
- [99] Vignjevic R., Hughes K. and Taylor E.A., (2002b), Finite element modelling of failure of a multi-material target due to high velocity space debris impacts. *Space Debris*, Vol. 2, pp. 41-50.
- [100] Vignjevic R., De Vuyst T., Gourma M., (2001), Interpolation Techniques in Meshless Methods, *Computer Modelling in Engineering and Science Journal*, V 2, No. 3, pp 319-337
- [101] Vignjevic R. Reveles J., Campbell J., (2006a), SPH in a Total Lagrangian Formalism, *Computer Methods in Applied Mechanics and Engineering*, Vol. 14, No. 3, pp. 181-198,
- [102] Vignjevic R.; De Vuyst T.; and Campbell J., (2006b), A Frictionless Contact Algorithm for Meshless Methods, *Computer Methods in Applied Mechanics and Engineering*, Vol. 13, No. 1, pp. 35-48,
- [103] Vignjevic, R., Campbell, J., Jaric, J. and Powell, S., 2009; "Derivation of SPH Equations in a Moving Referential Coordinate System, *Computer Methods in Applied Mechanics and Engineering* , 198, 2403–2411
- [104] Wagner GJ, Liu WK. (2001), Hierarchical enrichment for bridging scales and meshfree boundary conditions. *International Journal for Numerical Methods in Engineering*; 50: pp. 507-524.
- [105] Wen Y., Hicks D.L. and Swegle, J.W. (1994), Stabilising SPH with conservative smoothing. *Technical Report SAND94-1932*
- [106] Xiao SP, Belytschko T. (2005) Material stability analysis of particle methods. *Advances in Computational Mathematics*; 23: pp. 171-190
- [107] Zbib, H.M. and Aifantis, E.C., 1992. "On the Gradient-Dependent Theory of Plasticity and Shear Banding", *Acta Mechanica*, 92, 209-225
- [108] MCM, (2002), A 1D/2D/3D Meshless Continuum Mechanic Code for solid and fluid mechanics, User Manual, Scholl of Engineering, Cranfield University. Version 2.0
- [109] Campbell J., (1998), Lagrangian hydrocode modelling of hypervelocity impact on spacecraft, PhD thesis, College of Aeronautics, Cranfield University

- 
- [110] Gourma M., (2003), Towards Better Understanding of the Smooth Particle Hydrodynamic Method, PhD Thesis, School of Engineering, Cranfield University
  - [111] De Vuyst T., (2003), Hydrocode Modelling of Water Impact, PhD Thesis, School of Engineering, Cranfield University
  - [112] Reveles J., (2006), Development of a Total Lagrangian SPH Code for the Simulation of Solids Under Dynamic Loading, PhD Thesis, School of Engineering, Cranfield University



# Numerical method - CMD

<b>6</b>	<b>Introduction to Computational Mechanics of Discontinua</b> .....	<b>215</b>
6.1	Introduction	
6.2	Molecular Dynamics	
6.3	Particle Dynamics	
6.4	Lattices	
6.5	Discrete Element Methods	
	<b>Bibliography</b> .....	<b>273</b>
	<b>Index</b> .....	<b>283</b>



# 6. Introduction to Computational Mechanics of Discontinua

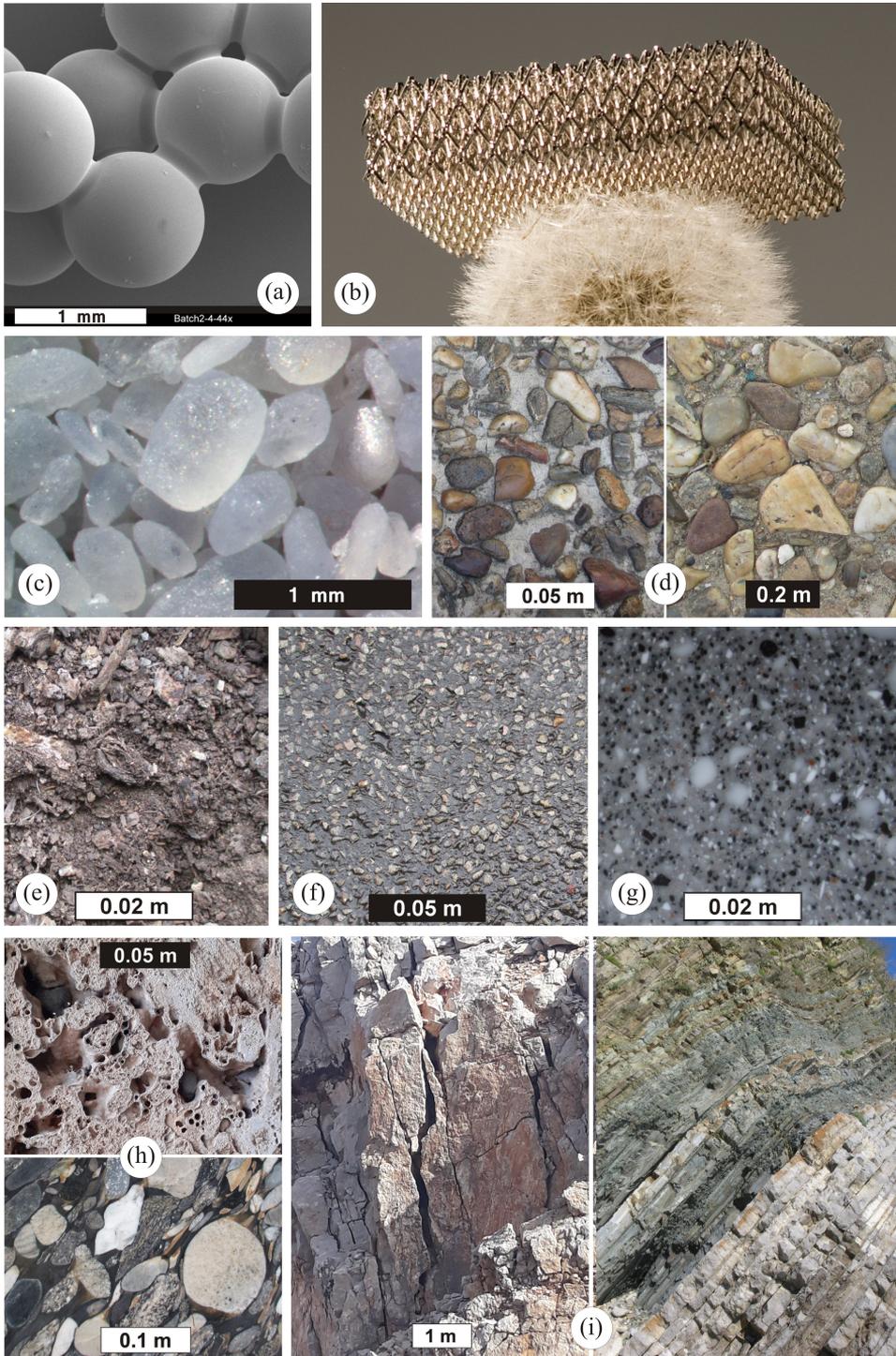
## 6.1 Introduction

The rapid development of computational mechanics of discontinua (CMD) emerged from the need to model objectively the deformation, damage and fracture of quasi-brittle materials with random, heterogeneous (or discontinuous) micro-/meso-structure and inferior (or non-existent) tensile strength. For decades, prior to that, many researchers have tried to model these material systems using classical methods of continuum mechanics but with limited success. One of the main reasons for this "elusiveness" is that behavior of the subject materials is essentially defined by their heterogeneous/discrete character, which affects the **localization of deformation** and damage evolution through the processes of nucleation, propagation and coalescence of cracks on various sub-macroscopic spatial scales. These phenomena are inextricably linked to the discontinuity of displacement, which clearly violates the continuum hypothesis and the fundamental assumptions of differential calculus. These difficulties have led to development of CMD whose basics are briefly summarized in this introduction. Over the past few decades, CMD models have fought for their place among the tools in the structural analysis and design. Finally, nowadays, they have become complementary to continuum mechanics models and experimental methods thanks to their ability to improve our understanding of damage and fracture and the ways they affect the effective material properties.

After introductory considerations, a summary of the basic concepts of traditional methods of CMD is presented. It should be noted that this classification of methods is somewhat subjective since their boundaries are blurred, domains overlap and distinctions are iffy. Be it as it may, the tentative classification is as follows: molecular dynamics (MD) and its coarser-scale offshoot dubbed herein particle dynamics (PD), the lattice methods and the discrete element methods (DEM). Although these (particle-based) methods are now widely used to model different classes of materials and various physical phenomena and industrial processes, the most natural applications seem to be the simulations of deformation and flow, damage, and fracture of systems that have the same topology as the representative model structure. Therefore, the modern, advanced applications advocate a modeling approach where it is insisted upon, as far as possible, the direct correspondence between the experimentally determined material and the structure explicitly represented by the numerical model.

All materials have a discontinuous (and heterogeneous) structure on some spatial scale (if not macroscopic, perhaps mesoscopic, microscopic, by definition atomic) as illustrated by Figure 6.1. When this scale is “very small” from the standpoint of engineering applications, the materials are considered continuous (and homogeneous) (Figure 6.2). This discontinuity and heterogeneity lead to complex mechanical behavior difficult to reproduce with models based on the classical theory of continuum mechanics since the material substance: (i) does not fill entirely the space it occupies, and (ii) the physical and mechanical properties may vary significantly within that space and across various directions. Among these complex phenomena notable is the evolution of damage with nucleation, propagation, branching, mutual interaction and coalescence of (micro-/meso-/macro-scale) cracks and other pre-existing flaws and features of material texture that can lead to appearance of flow, diffuse or localized deformation and damage, fracture, and fragmentation. All CMD methods described herein have in common that they cope with these complexities by establishing a computational domain (approximating the material structure) by a collection of discrete building units that are, or may be, interconnected. These models differ from the computational models of continuum mechanics in the definition of the displacement field only in the finite number of nodal points and, accordingly, in the formulation of the problem using algebraic, instead of partial differential, equations.

During the 1960s, researchers and engineers working in various fields of mechanics of materials and materials science noticed that solutions obtained using traditional continuum mechanics often exhibited singularities or yielded results inconsistent with experimental observations. In the decade that followed, the awesome development of computer capabilities and the accompanying advances of numerical methods enabled the emergence of novel particle-based computational methods that used various distinct structural-building elements (atoms, springs, trusses, beams, particles or various shapes) to model materials. CMD is nowadays firmly established as an integral part of not only the cutting-edge research in various fields (e.g., nanotechnology, stem cell research, biomedical engineering, space propulsion) but also industrial processes covering a wide range of different application fields (e.g., mining, machining, pharmaceuticals, civil construction, industrial and systems engineering).



**Figure 6.1:** Examples of discontinuous and heterogeneous material systems with representative spatial scales. (a) epoxy-cemented glass bead material (courtesy of Idar Larsen (SINTEF) and Prof. Rune M. Holt (NTNU)), (b) ultralight metallic microlattice (on Dandelion, courtesy of Dr. Tobias A. Schaedler), (c) Gypsum sand (photographed by Prof. Mark A. Wilson), (d) concrete, (e) soil, (f) asphalt, (g) particle composite (Kerrock<sup>®</sup> 9057 – aluminum hydroxide particles with acrylic binder), (h) rock (Travertine /top/ and Nero Marinace Granite /bottom/), (i) rock masiffs (courtesy of Prof. Radojica Lapcevic). (Note: black/white rectangles indicate length scales.)

Modeling using computer simulations is more flexible than analytical modeling and has the advantage over experimental research in that the data are available at every stage of the virtual experiment. This flexibility extends to the ability to configure loads, initial and boundary conditions, and to tailor the custom-made models in accordance with the topological, geometric, and structural disorder of material.

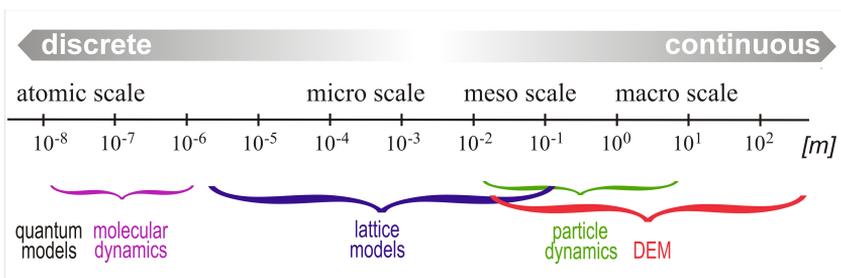
These, so called, virtual experiments are in main aspects very similar to the laboratory experiments. First, a sample is prepared from the subject material rendering a simulation object in virtual space ("numerical" or "virtual" material). The sample created in this way is then connected to the necessary "virtual" measuring instruments so that the parameters of the state can be recorded over a period of time. Since most measurements are subject to statistical variations, the more time is available to average the results, the more accurate the measurements become.

However, the virtual measurement resolution is inversely proportional to the size of the averaging period and it is necessary to find a compromise between these conflicting requirements taking into account the nature of the physical phenomenon being simulated. Consequently, the most common errors made during virtual experiments are very similar to those that can contaminate the results of actual laboratory experiments: the sample is not adequately prepared; the measurement is too short; due to conceptual oversights, we do not measure what we intend to measure...

All CMD methods offer some common advantages in damage and fracture analysis compared to corresponding conventional computational methods based on continuum mechanics (typically, FEM). Damage and its evolution are presented explicitly through broken bonds or contacts; it is not necessary to use any empirical relations to define damage or determine its effect on material behavior.

Variety of structural flaws nucleate, extend and merge into different types of macroscopic damage without the need to use numerical "ingenuities" such as convenient mesh orientation, mesh reformulation or constant adoptive meshing. There is no need to develop constitutive laws or damage models in order to represent complex nonlinear responses of materials as they emerge naturally through the collective behavior of discrete units whose interaction is guided by relatively simple rules.

A summary of the CMD models that will be considered herein is shown in Figure 6.2 in conjunction with the natural spatial scale on which they are most commonly used.



**Figure 6.2:** Spatial scales and corresponding traditional CMD models as tentatively classified herein for typical brittle materials with random microstructure (e.g., concrete).

It cannot be overemphasized that this classification is somewhat arbitrary, the boundaries between the models are hard to draw, and the associated length scales are subject to definition for each material separately. For example, the spatial scale corresponding to the meso-structure of concrete (of the order of centimeters or even decimeters as illustrated in Figure 6.1d) or rock massifs (of the order of decimeters or meters, Figure 6.1i) exceeds that of silicon carbide (SiC) by several orders of magnitude although they can all be classified the quasi-brittle systems.

With reference to Figure 6.2, on the nano-scale (e.g., the crystal lattice), it is possible to use the atomic models based on quantum mechanics. These rigorous methods<sup>1</sup> are based on the Schrödinger wave equation and semiempirical effective potentials which approximate quantum effects [1]. In contrast, classical MD relies on Newton's equations of motion and empirical potentials. Therefore, the traditional MD sacrifices the quantum mechanics rigor for the benefit of a much larger spatio-temporal modeling range. Observation scales and corresponding numerical models of mechanics of discontinua (conditionally divided into the three broad, intertwined categories: lattices, PD and DEM, as discussed above) correspond to concrete (Figure 6.1d) (inspired by [2]).

The MD refers to models where the basic building object is a point mass which may represent an atom, a molecule, a nanocluster, as well as a planet in a galaxy. In this short introduction, we always consider atoms (thus, the length scale in Figure 6.2) but, in general, it could be any of the above. From now on, the term MD, unless specified otherwise, is used for the traditional (classic) MD where each atom is treated as a point mass  $m_i$  (and, generally, a fixed charge  $q_i$ ).

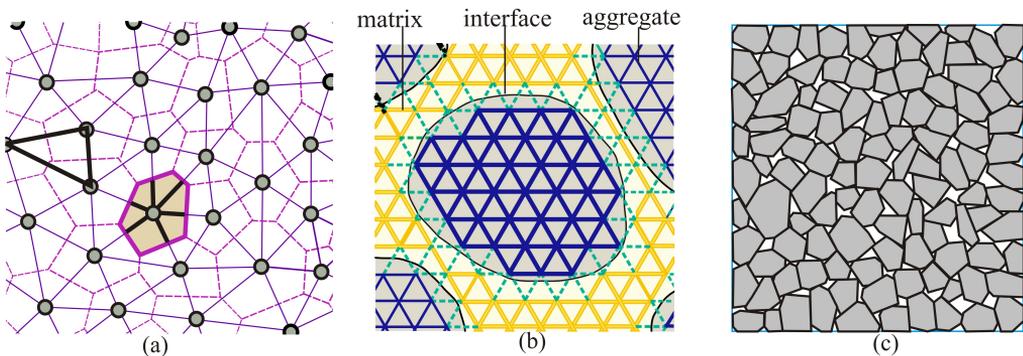
The term particle dynamics (PD), as used herein, designates "a coarse scale cousin to molecular dynamics" [3] sometimes also called the quasi-MD [4] to emphasize this kinship. It is developed to simulate phenomena on coarser spatial scales—the dynamic response of a material, either solid or fluid—based on a generalization of the MD modeling approach. The role of an atom is taken over by aggregates of atoms or molecules, represented by a material point called a "**continuum particle**" or a "quasi-particle". Depending on a particular application, this entity can represent, for example, a nanocluster, a ceramic grain, a concrete aggregate, a composite particle, a clastic rock granule and can, therefore, cover a wide range of spatial scales (up to the above-mentioned cosmological scales). Particle models use tried and tested MD techniques to directly confront various challenges of extremely complex physics. A critical step in the PD modeling is the transition from an adopted atomic potential to an interparticle potential (bottom-up approach) or a definition of an interparticle potential (a set of constitutive rules) on macro-scale (top-down), which is a common theme in all CMD numerical approaches. Traditional references for particle modeling are [4], [5], while [6] can be consulted for a review of recent developments.

Lattices are arguably the simplest CMD models (specifically, the spring-networks among them), comprised of one-dimensional discrete structural units such as springs, trusses or beams (Figure 6.3a,b). These elementary building units are assigned both geometric and structural properties and fracture characteristics that allow them to mimic elastic and inelastic deformation and fracture of the abstracted material. Lattice models could be

---

<sup>1</sup>So-called, "ab initio (first principle) MD", also known as "Born-Oppenheimer" or "Carr-Parrinello" MD.

considered meso-scale offshoots of both MD (micro-scale) but also the engineering truss and frame structures well-known from structural mechanics (macro-scale). Lattices were the original systems for modeling discontinuous media - various ideas of application in engineering mechanics date, at least, to Hrennikoff's pioneering work [7]. The modeling of network structures on a much coarser spatial scale than the atomic one eliminates the obvious need to work with a huge number of degrees of freedom, which could result in both computer congestion and data overload (so-called "data glut"), which would be inevitable if the atomic methods were used for modeling even the smallest structures on the macro-scale. This approach also reduces to a relatively modest level the number of nodes necessary to model the heterogeneity of the material structure. Comprehensive reviews of lattice models were published by Ostoja-Starzewski [3,8].



**Figure 6.3:** (a) Irregular triangular Delaney lattice dual to Voronoi grain tessellation. (b) Mesostructure of a three-phase composite projected on a regular triangular lattice. (c) Assembly of polygonal particles.

Unlike lattice models in which the basic structural elements are one-dimensional, in DEM (discrete element method) models the basic building blocks are typically of the same dimensionality as the considered problem. For example, planar DEMs include models of discontinuous systems comprised of 2D basic constituent elements such as circles, ellipses, or polygons (Figure 6.3c). These discrete elements are provided with geometric, structural, and contact properties that allow their “assemblies” (conglomerates, agglomerates) to approximate the complex phenomenological response of the subject material. In the most concise terms, DEMs enable the simulation of the motion and interactions of a huge number of discrete objects. It is important to note, that DEM unit blocks are actual geometric objects characterized by their dimension and shape, unlike the MD atoms, lattice nodal points, and particles (of PD) that are essentially material points. The macroscopic behavior of DEM models emerges as a consequent feature of the system derived from a small set of meso-properties of individual elements and their interactions. Contacts among discrete elements are endowed with the proscribed cohesive strength (including zero cohesive strength for non-cohesive, loose, material systems) and the ability to dissipate energy that allows representation of both elastic and inelastic phenomena and the nucleation of cracks and cooperative phenomena among them. It is important to emphasize that the properties of the contacts between discrete elements should be, in principle, identifiable based on the

material properties and the limited number of available experiments. Comprehensive DEM reviews were published in [9-11].

Regardless of the type of the CMD model, the evolution of a system of particles (the discrete structural elements in the most general sense) of known masses  $m_i$ , moments of inertia  $I_i$ , and positions  $r_i$  ( $i = 1, 2, \dots, N$ ) is obtained by solving a system of equations of motion for each particle. Within classical mechanics, the equations that define the translational and rotational motion are those that correspond to Newton's second law

$$\begin{aligned} \frac{d\mathbf{p}_i}{dt} &= \frac{d}{dt}(m_i\dot{\mathbf{r}}_i) = \mathbf{F}_i, \\ \frac{d\mathbf{L}_i}{dt} &= \frac{d}{dt}(I_i\boldsymbol{\omega}_i) = \mathbf{M}_i \end{aligned} \quad (6.1)$$

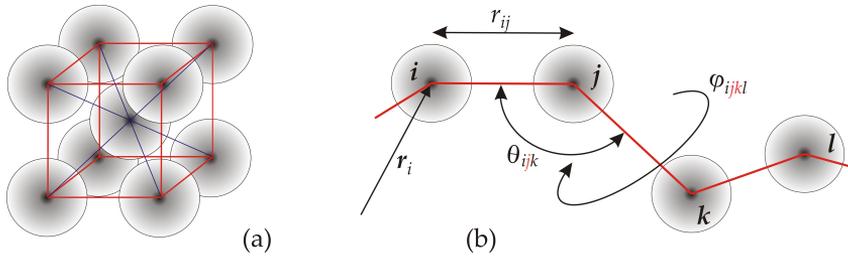
where  $\mathbf{p}_i$  and  $\mathbf{L}_i$  mark, respectively, the linear and angular momenta of  $i$ -th particle and  $\mathbf{F}_i$  and  $\mathbf{M}_i$  corresponding forces and moments. Obviously, the motion of material points (all CMD systems except DEM) is completely defined by Equation (6.1)<sub>1</sub>. (Hereinafter, symbols in boldface designate vectors and tensors.)

Lastly, the following brief introduction to CMD stems out of the two-part essay published in coauthorship with Antonio Rinaldi in "Handbook of Damage Mechanics: Nano to Macro Scale for Materials and Structures" [1]. Despite renewed efforts to make it as representative and comprehensive as possible, its content suffers unavoidably from the author's bias due to research background and interests.

## 6.2 Molecular Dynamics

This short introduction aims to outline the basics of the traditional molecular dynamics (MD) method based on the classical mechanics. This computational simulation technique allows prediction of the temporal evolution of a system of material points (Figure 6.4) interacting via empirical interatomic potentials or molecular mechanics force fields by numerically solving Newton's equations of motion. Simulation methods based on quantum mechanics are beyond the scope of this overview, as well as many other advanced topics.

The first research article [12] in which Alder and Wainwright used MD to simulate perfectly elastic collisions of hard spheres was published in 1957. In 1960, Gibson and co-authors used a Born-Mayer potential to simulate a radiation damage of solid copper. Rahman (1964) simulated liquid argon by using 864 atoms interacting with a Lennard-Jones potential [13]. The first computer simulation of a simplified protein folding was produced in 1975 [14]. These pioneering articles were published more than half a century ago. Therefore, it is not surprising that many outstanding monographs devoted to computer simulations in condensed matter physics are available to the interested researcher (e.g., [5,15-17]).



**Figure 6.4:** Schematic representation of two different solid structures: (a) the body-centered cubic crystal lattice, and (b) a molecular chain. The latter indicates 2-atoms (chemical bond), 3-atoms (angle bending), and 4-atoms (torsion; dihedral) interactions typical of the intramolecular bonding interactions.

The main feature of the MD method is the ability to analyze dynamics of (non)equilibrium processes with spatial resolution on the atomic scale. Thus, MD simulations play a role of computational microscope and have no computational alternative for many problems including atomic-scale phenomena that cannot be observed directly.

MD can be considered a numerical simulation offshoot of statistical mechanics. It has found research application in a wide range of problems prevalent in various scientific fields; such as, for example:

- Theoretical and statistical physics: fluid theory; properties of a statistical ensemble; structures and properties of small clusters; phase transitions,...
- Materials science and mechanics of materials: point, linear and plane defects in crystals and their interactions; stable and metastable structures of complex alloys and related phase diagrams; amorphous materials; radiation damage to materials; microscopic damage and fracture mechanisms; surface reconstructions; melting; growth of thin films; friction,...
- Biology, biochemistry and biophysics: molecular structure; chemical reactions; protein structure, functional mechanisms and folding process; drug design; vibrational relaxation and energy transfer; membrane structure; dynamics of biomolecules,..."Everything that living things do can be understood in terms of the jiggings and wiggings of atoms" [18].

### 6.2.1 Basic Idea of MD

The basic idea is simple. First, to setup the atomic system one must:

- i) define a set of initial conditions (initial positions  $\mathbf{r}_i$  and velocities  $\mathbf{v}_i$  of all atoms in the system), then
- ii) adopt the interatomic potential to define interatomic forces (internal forces), and finally
- iii) introduce (externally applied) load acting on the system.

After that, the evolution of a system of atoms  $(m_i, \mathbf{r}_i, \mathbf{v}_i)$  ( $i = 1, 2, \dots, N$ ) is determined by solving a system of equations of motion (6.1)<sub>1</sub> for each atom.

The resulting force acting on each atom at a given moment

$$f_{ij} = |\mathbf{f}_{ij}| = -\frac{d\Phi}{dr_{ij}}; \quad \mathbf{F}_i = -\sum_j f_{ij} \frac{\mathbf{r}_{ij}}{r_{ij}} \quad (6.2)$$

can be obtained from the interatomic potential  $\Phi$  which is, in general, a function of the position of all atoms of the system ( $r_{ij} = |\mathbf{r}_{ij}| = |\mathbf{r}_i - \mathbf{r}_j|$  being the intensity of the distance vector between the atoms  $i$  and  $j$ , that is, the interatomic distance, Figure (6.4b)). Thus, when the initial conditions and the interaction potential are defined, Equations (6.1)<sub>1</sub> can be solved numerically. Namely, the positions and velocities of all atoms of the system as a function of time are obtained as a result of solving a system of algebraic equations that approximates the system of differential Equations (6.1)<sub>1</sub>. Thus, the motion of each individual atom (and each ensemble of atoms) is completely deterministic. In most cases, analysts are not interested in the trajectories of individual atoms but in the macroscopic properties of materials that result from the motion of a multitude of atoms. The information resulting from computer simulations can be averaged at certain time intervals for all (or selected) atoms of the system to obtain thermodynamic parameters (Chapter 6.2.4).

### 6.2.2 Empirical Interatomic Potentials

Empirical potentials used in materials science and mechanics of materials are called interatomic potentials. The role of interatomic forces (6.2) is crucial since the MD simulation is realistic only insofar as the interatomic forces are similar to those operating between real atoms in the corresponding atomic configuration [16]. As already noted, the classical definition of interatomic interaction, based on empirical potentials, represents the rigorous quantum mechanical nature of materials in a limited way through impromptu approximations.

The interatomic potentials depend on the states of the electrons, thus, the electrons are the origin of the interatomic forces. Nonetheless, the electrons are not directly present in the traditional MD model – their influence is introduced indirectly through analytical functions that define potential energy solely on the basis of the atomic (nuclei) positions (6.3). The creation of the analytical function of potential energy and the choice of input parameters is often based on the fitting of the available experimental data that are of greatest interest for the specific problem being studied (e.g., modulus of elasticity, cohesive energy, phase transition temperature, vibration frequencies).

When forming an MD model, the interatomic potential is adopted either on the basis of knowledge of the atomic nature of the simulated material or a priori. The construction of interatomic potential is as much an art as it is a science, but from the point of view of users, the choice is, nowadays, quite simplified thanks to the available literature. This choice is essential not only because the adequacy and accuracy of the potential dictate the quality of the simulation results but also due to the fact that its complexity determines the efficiency of the code in terms of simulation duration. Although some compact potentials may seem inadequate, many fundamental, generic aspects of a physical phenomenon can be observed thanks to the advantages provided by their simplicity.

The empirical interatomic potential

$$\begin{aligned} \Phi &= \Phi(r_1, r_2, \dots, r_N) = \\ &= \sum_i \Phi_1(r_i) + \sum_i \sum_{j>i} \Phi_2(r_i, r_j) + \sum_i \sum_{j>i} \sum_{k>j>i} \Phi_3(r_i, r_j, r_k) + \dots \end{aligned} \quad (6.3)$$

represents the potential hypersurface of the non-bonded interactions [5]. In expression

(6.3), the terms  $\Phi_1, \Phi_2, \Phi_3, \dots$  are, respectively, contributions due to external fields (e.g., gravitational or the container wall), pair, triple and higher order interatomic interactions. In traditional MD, rigor is often sacrificed for the sake of efficiency, and interatomic interactions among all atoms have been replaced by less computer-intensive approximations in which each individual atom interacts only with a certain number of **nearest neighbors** (so-called, first-nearest, second-nearest, etc.). Further, by neglecting the three-body interactions (involving interatomic angles) and higher, the potential energy of the system can be approximated by the sum of isolated empirical biatomic potentials (**pairwise additivity** assumption)

$$\Phi = \frac{1}{2} \sum_i \sum_j \phi(r_{ij}). \quad (6.4)$$

It is obvious that the concept of pairwise additivity (6.4) represents a huge simplification with far-reaching consequences. For some crystal lattices, the pairwise interaction is not able to take into account a good portion of the cohesive interaction [19]. Furthermore, the interaction in ionic crystals may be a consequence of the polarization effects, attributed to the action of the electric fields of the surrounding ions, which cannot be described by simple pair potentials. However, the main interactions in the ionic and Van der Waals crystal lattices are believed to be essentially pairwise [19]. Many more complex forms of potential can be used as needed at the cost of increasing the duration of the simulation. The consequence of choosing the central-force potential (6.4) is that the total energy of the system is conserved. In the language of statistical physics, classical MD generates a microcanonical ensemble ( $N, E, V$ ).

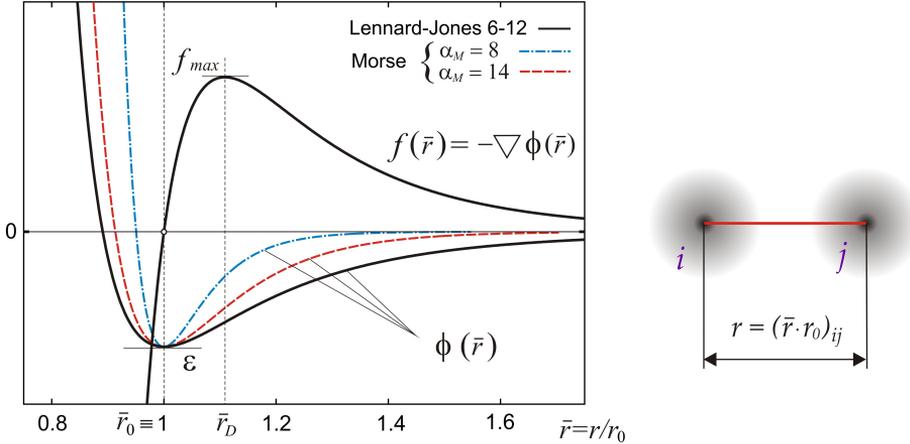
The **pair potentials** are the simplest potentials since the force of interaction of two atoms is completely determined by their mutual distance. There is an extensive literature on the ways in which these potentials are experimentally determined or theoretically modeled (e.g., [20]). Strictly speaking, they realistically describe only noble gases. The simplest potential of this type is the discontinuous potential of a "rigid sphere" which implies that the value of the interatomic force is equal to either: (i) zero, if the interatomic distance is greater than the prescribed value; or (ii) infinity, if the interatomic distance is equal to or less than the prescribed value.

A more realistic interatomic interactions are obtained under the assumption that the interaction force gradually varies from strongly repulsive (at small interatomic distances) to attractive (at medium distances) until it finally converge asymptotically to zero (with further increase in distance) (Figure 6.5). The best known potential of this type, which has been widely used in the past when the focus of research was on the study of qualitative trends (essential physics) rather than narrowly specific issues, is the **Lennard-Jones 6-12 potential**

$$\phi_{ij} = -\varepsilon_{LJ} \left[ 2 \left( \frac{1}{\bar{r}_{ij}} \right)^6 - \left( \frac{1}{\bar{r}_{ij}} \right)^{12} \right] \quad (6.5)$$

originally developed for noble gases from van der Waals cohesion [21]. In expression (6.5),  $\varepsilon_{LJ}$  represents the depth of the potential well, and  $\bar{r}_{ij} = (r/r_0)_{ij}$  ratio of current and equilibrium distance between atoms  $i$  and  $j$  (Figure 6.5). These model parameters are chosen with the aim of optimally reproducing the most desirable physical and mechanical

properties of the material. The amount of published work with Leonard-Jones systems since the creation of MD is unsurpassed by any other potential, and only for that reason, not to mention others, its importance cannot be overemphasized. However, it can be ill-advised to apply this potential for the quantitative study of more complex materials.



**Figure 6.5:** Schematic representation of pair potentials with unit depth of the **potential well**: Lennard-Jones 6-12 with corresponding force (solid black line), Morse for two parameter values  $\alpha_M$  which define the width of the potential well. These curves illustrate the atomic hypothesis that „all things are made of atoms—little particles that move around in perpetual motion, attracting each other when they are a little distance apart, but repelling upon being squeezed into one another” [18].

The term  $\bar{r}_{ij}^{-12}$  in (6.5), which dominates the interatomic repulsion, is selected for application convenience: as far as the physics is concerned, the exponential term is a more appropriate choice. The simplest potential that takes this into account is the Morse potential:

$$\phi(\bar{r}_{ij}) = \varepsilon_M \left\{ \exp[-2\alpha_M(\bar{r}_{ij} - 1)] - 2\exp[-\alpha_M(\bar{r}_{ij} - 1)] \right\}$$

The adoptable parameter  $\alpha_M$  defines the slope of the repulsive wall (that is, the width of the potential well; Figure 6.5). The parameters  $\varepsilon_M$ ,  $r_0$  and  $\alpha_M$  are, as in the previous case, usually determined by fitting the material properties that are most relevant to the problem being analyzed. The Morse potential was originally developed for covalent bonds that are strongly spatially oriented, so the interatomic distance is not sufficient for a realistic description of the interaction [22]. Although the inverse-power form of the interatomic potential (6.5) is most often used in heuristic studies, the results of quantum mechanical calculations, as already mentioned, favor the exponential form, which is later used in some particle models (6.26).

If electrostatic charges are present, it is necessary to add the appropriate Coulomb potentials

$$\phi(r_{ij}) = \frac{q_i q_j}{4\pi \bar{\varepsilon}_p r_{ij}}$$

where  $q_i$  and  $q_j$  are charges, and  $\bar{\varepsilon}_p$  the permittivity.

The simple potentials also provide a direct way to estimate the modulus of elasticity and the theoretical strength using expressions

$$E = \frac{1}{r_0} \left( \frac{d^2\phi}{dr^2} \right) \Big|_{r=r_0}, \quad \sigma_m = \frac{1}{r_0^2} \left( \frac{d\phi}{dr} \right) \Big|_{r=r_D} \quad (6.6)$$

which follows from its physical nature [23]. In Equation (6.6),  $r_D$  stands for the so-called separation distance defined by the maximum value of the interatomic force  $df(r)/dr = 0$ , that is, the inflection point of the interparticle potential  $d^2\phi(r)/dr^2 = 0$  (Figure 6.5).

The basic purpose of any interatomic potential is to correctly reproduce the most prominent characteristics of atomic bonds. Therefore, it should always be borne in mind that the pair of potentials were “originally developed to describe atomic interaction in systems in which these forms of potentials are physically justified” [24] and resist the temptation to use them injudiciously for their simplicity.

The pair potentials cannot accurately describe interatomic interactions in more complex systems [16] such as, for example, strongly covalent systems (e.g., SiC), most ceramics characterized by fully populated orbitals, metals characterized by delocalized “sea of electrons”, or semiconductors. However, in MD simulations from the 1950s to the 1980s, a couple of these simplified potentials were used almost exclusively. Significant progress was made during the 1980s with the development of **many-body potentials** for metals based on the concept of atomic density (e.g., [25]). The main observation that needed to be modeled was that interatomic bonds become weaker when nested in a “dense” local environment. Accordingly, the force acting on an atom depends not only on the distance separating its nucleus from nuclei of its neighbors, but also on the local atomic density. In other words, the forces between ions are characteristically dependent on many bodies (“many-body in character”), instead of simply being pairwise additive. The focus is usually on the attractive part of the potential [16]. Accordingly, the so-called “glue model” potentials have been developed (e.g., [26] and references cited therein). Among these potentials, the best known is the so-called **embedded atom method (EAM)**

$$\Phi(r_{ij}, \rho_i) = \frac{1}{2} \sum_i \sum_{j \neq i} \phi(r_{ij}) + \Psi = \sum_i \left[ \frac{1}{2} \sum_{j \neq i} \phi(r_{ij}) + \psi(\bar{\rho}_i) \right], \quad (6.7)$$

$$\bar{\rho}_i = \sum_{j \neq i} \rho(r_{ij}).$$

developed to approximate the interaction between ions in metals. The various forms of (6.7) differ from each other only in the forms of functions:  $\phi$  (pairwise term depending entirely on the interatomic distance),  $\psi$  (density-dependent contribution - the embedding energy necessary to insert the  $i$ -th atom into the background of the electron density  $\bar{\rho}_i$ ), and  $\rho_{ij}$  (atomic density function) [26].

Differences between pair potentials and many-body potentials have been discussed in detail in the literature (e.g., [27], [28]). EAM potential has been used extensively to model ductile metals.

Finally, empirical potentials in organic chemistry and molecular biology are often called force fields. It should be noted that for complex macro-molecular chains (e.g.,

synthetic polymers, biopolymers), in addition to non-bonding forces (van der Waals and electrostatic), it is necessary to consider the **intramolecular bonding interactions** illustrated in Figure 6.4b. The elementary models of this kind include contributions due to:

- (i) bond stretching: 2-atoms vibrations about the equilibrium bond length (6.8)<sub>1</sub>
- (ii) angle bending: 3-atoms vibrations about the equilibrium angle they define (6.8)<sub>2</sub>

$$\frac{1}{2} \sum_{\text{bonds}} k_{ij}^{\alpha} (r_{ij} - r_{0ij})^2, \quad \frac{1}{2} \sum_{\text{bend angles}} k_{ijk}^{\beta} (\theta_{ijk} - \theta_{0ijk})^2, \quad (6.8)$$

- (iii) torsion (dihedrals, proper and improper, 4-atoms torsional vibrations), and
- (iv) various cross-terms [29].

These so-called bonded interactions are not further discussed in this introduction. The details are available in [29], [30].

Further considerations of empirical potentials go beyond the objectives of this introduction and can be found in literature (e.g., [24], [30]; and many others). In the last thirty years, empirical potentials have been developed in a targeted way - for specific material systems with a range of applicability in mind. The ultimate test of any empirical potential is its success in simulating properties of interest. However, it seems appropriate to conclude this brief review by adding that in the constant competition between the more sophisticated and the spatially larger MD models, under the constraints imposed by computational capabilities, the latter are still considered more advisable in terms of meaningful results. In other words, it is generally accepted that it is better to increase the size of the MD system and simplify the potential, than to do the opposite. This trend has resulted in the development of parallel processing [11], [31], [32] without which MD simulations in the contemporary research cannot be imagined.

### Shortening the range of potential

The empirical potentials presented in Chapter 6.2.2 have an unlimited range. In MD simulations, it is a common custom to establish the cut-off distance ( $r_{cut}$ ) and to neglect interatomic interactions for distances that exceed it because the corresponding forces are insignificant (Figure 6.5). This neglect of interatomic action in the range of potential asymptotic approach to zero leads to program simplification and huge computational savings due to a drastic reduction in the number of interacting atomic pairs.

However, a simple shortening of the potential would lead to a new problem: whenever the mutual distance between pairs of atoms "crossed" over the cutting distance, there would be a small, abrupt change in the energy of the system. A large number of such events could, on the one hand, have an impact on the law of conservation of energy, and on the other, affect the physics of subtle micro-processes that depend on the details of the local energy state. Therefore, limiting the range of potentials is most often done by a smooth transition in the attractive range, for example, by using a cubic spline. As an example, the approach of Holian et al. [27] is based on the Lennard-Jones potential (6.5) interrupted at  $r_{spl} \approx 1.109r_0$  and replaced by the cubic spline in  $r^2$

$$\phi_{spl}(r) = -\mathcal{A}(r_{cut}^2 - r^2)^2 + \mathcal{B}(r_{cut}^2 - r^2)^3$$

which reaches zero at  $r_{cut}$ . The spline parameters are chosen to ensure continuity of coordinates, inclination and curvature ( $C0$ ,  $C1$ , and  $C2$ ) at the point of intersection ( $r = r_{spl}$ ):

$\phi_{LJ} = \phi_{spl}$ ,  $\phi'_{LJ} = \phi'_{spl}$  and  $\phi''_{LJ} = \phi''_{spl}$ . Note that, since the cubic spline is defined in  $r^2$  instead of  $r$ , rooting and division operations with  $r$  are avoided in the MD simulation, while differences in the appearance of the function are often practically imperceptible [27].

Physical quantities are, of course, somewhat influenced by this shortening of the range of potentials. Overall, the eventual assessment of these effects is a problem specific to each application (e.g., [16], [17]).

### 6.2.3 Integration Algorithms

The core of the MD program is the integration algorithm necessary for the integration of the equations of motion (6.1)<sub>1</sub>. These algorithms are based on the finite difference method in which time is discretized within a finite network and the time step,  $\delta t$ , represents the distance between successive points on the network. Knowing the position and some of its time derivatives in the current time ( $t$ ), allows calculation of the same quantities in the next time ( $t + \delta t$ ). In addition to being as accurate as possible, the integration algorithm should be both fast and of modest memory requirements, to allow the application of the longest possible time step, and to be simple for implementation.

Several different methods are available for this purpose but only one will be discussed herein. The **Verlet algorithm** is still the most widely used, although more powerful techniques for integrating finite difference equations are available. The derivation of this algorithm is based on the development of the atomic position function in the Taylor series at an arbitrary time  $t$

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\delta t + \mathbf{a}_i(t)\delta t^2 + \frac{d^3\mathbf{r}_i(t)}{dt^3}\delta t^3 + O(\delta t^4)$$

that is

$$\mathbf{r}_i(t - \delta t) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\delta t + \mathbf{a}_i(t)\delta t^2 - \frac{d^3\mathbf{r}_i(t)}{dt^3}\delta t^3 + O(\delta t^4).$$

Adding the previous two expressions yields

$$\mathbf{r}_i(t + \delta t) + \mathbf{r}_i(t - \delta t) = 2\mathbf{r}_i(t) + 2\mathbf{a}_i(t)\delta t^2 + O(\delta t^4).$$

The time-reversible equation for calculating the next position has the form

$$\mathbf{r}_i(t + \delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \delta t) + \delta t^2 \sum_{j \neq i} \frac{\mathbf{f}_{ij}(\mathbf{r}_i, \mathbf{r}_j)}{m_i} + O(\delta t^4). \quad (6.9)$$

This method of integration is very compact and easy to implement. Since no dissipative forces act between the atoms, the dynamic system is conservative. Therefore, the force by which the atom  $j$  acts on the atom  $i$  is  $f_{ij} = -\nabla_i \phi(r_{ij})$ , and its calculation is by far the most demanding part of MD simulations. The Verlet method seems to be the least time-consuming and thus the most suitable for modest computing resources, which explains its popularity. The main disadvantages of the Verlet algorithm are clumsy handling of velocities

$$\mathbf{v}_i(t) = \frac{\mathbf{r}_i(t + \delta t) - \mathbf{r}_i(t - \delta t)}{2\delta t} + O(\delta t^2)$$

which requires knowledge of the next position  $r_i(t + \delta t)$  and is susceptible to rounding error due to relatively large residue  $O(\delta t^2)$ . However, it is possible to obtain more accurate estimates of the velocity (and thus the kinetic energy of the system) using various modifications of the Verlet algorithm.

The Störmer algorithm is a such modification of the Verlet algorithm, where the most pronounced computational advantage stems from the fact that at no time is the difference between two large numbers used to calculate a small number [5]. The computational scheme has the form

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \delta t \mathbf{v}_i \left( t + \frac{\delta t}{2} \right). \quad (6.10)$$

In addition to current positions and accelerations, recorded quantities include mid-step speeds

$$\mathbf{v}_i \left( t + \frac{\delta t}{2} \right) = \mathbf{v}_i \left( t - \frac{\delta t}{2} \right) + \delta t \mathbf{a}_i(t). \quad (6.11)$$

Since the Störmer algorithm is only a modification of the Verlet algorithm, it produces identical trajectories. The problem, evident in expressions (6.10), arises because the velocities are not calculated at the same time points as their positions, which complicates the calculation of the total energy of the system.

The above-mentioned algorithms, and similar ones available in literature, are completely adequate for most MD simulations. However, it is sometimes necessary to use higher-order integration schemes which use higher-order position vector derivatives in the Taylor approximation. These algorithms not only achieve higher calculation accuracy for the same time step but also allow use of a longer time step without losing accuracy (at least in a short run). Unfortunately, the use of these higher-order algorithms (an example is the popular predictor-corrector method) is coupled with many implementation difficulties that go beyond the scope of this introduction and have been discussed in detail in the literature (e.g., [5], [17]).

#### 6.2.4 Calculation of Macro-parameters of State

Statistical physics provides the connection between microscopic behavior of the system and the macroscopic world described by thermodynamics. In order to calculate a certain physical parameter of the state of a macroscopic system (such as stress, strength, temperature, damage) it is necessary to define it as a function of the raw MD output data (that is, the atomic positions, velocity and forces ( $\mathbf{r}$ ,  $\mathbf{v}$ ,  $\mathbf{a}$ )). Strictly speaking, this can be achieved only when the thermodynamic system is: (i) large enough to be statistically homogeneous, and (ii) either in equilibrium or close enough to equilibrium (measured by the Deborah number [33]). If these preconditions are not met, the meaning of the continuum concepts becomes disputable.

### Stress, strain and effective stiffness

The elastic strain energy of the system in the (proximity of) equilibrium can be developed in the Taylor series [34]

$$\begin{aligned} U &= U_0 + \frac{1}{1!} \frac{\partial U}{\partial \varepsilon_{\alpha\beta}} \varepsilon_{\alpha\beta} + \frac{1}{2!} \frac{\partial^2 U}{\partial \varepsilon_{\alpha\beta} \partial \varepsilon_{\gamma\delta}} \varepsilon_{\alpha\beta} \varepsilon_{\gamma\delta} + \dots = \\ &= U_0 + \sigma_{\alpha\beta} \varepsilon_{\alpha\beta} + \frac{1}{2} C_{\alpha\beta\gamma\delta} \varepsilon_{\alpha\beta} \varepsilon_{\gamma\delta} + \dots \end{aligned} \quad (6.12)$$

In the (proximity of) equilibrium state, the resultant forces acting on any atom of the system (6.2)<sub>2</sub> are (close to) zero, which implies that each atom lies in its potential well. Such system must be, by definition, stable in the event of an infinitesimal disturbance, such as the one imposed by the homogeneous strain tensor  $\varepsilon_{\alpha\beta}$  acting as an external load.

The linear (second) term in the Taylor series (6.12) represents the stress tensor. It is important to note that this is a general thermodynamic relation independent of the applicability of Hooke's law [35]. With regards to the potential energy of a system, if the interatomic actions can be successfully approximated with the EAM (6.4), the stress tensor components are

$$\sigma_{\alpha\beta} = \frac{1}{2\bar{V}N} \sum_{\substack{i,j \\ j \neq i}} \frac{d\phi}{dr_{ij}} \frac{(r_{ij})_{\alpha} (r_{ij})_{\beta}}{r_{ij}} + \frac{d\Psi}{d\bar{V}} \delta_{\alpha\beta} = -\frac{1}{2\bar{V}N} \sum_{\substack{i,j \\ j \neq i}} f_{ij} \frac{(r_{ij})_{\alpha} (r_{ij})_{\beta}}{r_{ij}} + \frac{d\Psi}{d\bar{V}} \delta_{\alpha\beta} \quad (6.13)$$

where  $\bar{V}$  is the average volume per atom, while  $(r_{ij})_{\alpha}$  and  $(r_{ij})_{\beta}$  are corresponding ( $\alpha$  and  $\beta$ ) projections of the distance vectors  $\mathbf{r}_{ij}$  [35], [36]. Since the stress definition (6.13) is inherently related to the static equilibrium state, it is, strictly speaking, applicable only to static (or quasi-static) deformation where the resultant force acting on each atom (6.2)<sub>2</sub> are equal (or "close enough") to zero. On the other hand, dynamic deformation implies wave propagation and, in order for expression (6.13) to be applicable, it must be tacitly assumed that the nonequilibrium process can be represented by a successive series of equilibrium processes. This concept is routinely used, out of necessity, in thermodynamics of nonequilibrium processes.

The third term in the Taylor series (6.12) defines the elastic stiffness tensor

$$\begin{aligned} C_{\alpha\beta\gamma\delta} &= \frac{1}{2\bar{V}N} \sum_{\substack{i,j \\ j \neq i}} \left( \frac{d^2\phi}{dr_{ij}^2} - \frac{1}{r_{ij}} \frac{d\phi}{dr_{ij}} \right) \frac{(r_{ij})_{\alpha} (r_{ij})_{\beta} (r_{ij})_{\gamma} (r_{ij})_{\delta}}{r_{ij}^2} - \\ &- \frac{1}{2} \frac{d\Psi}{d\bar{V}} \left\{ (\delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\beta\gamma} \delta_{\alpha\delta}) (2 - \delta_{\alpha\beta}) - \frac{1}{2} \delta_{\alpha\beta} \delta_{\gamma\delta} (4 - \delta_{\alpha\gamma} - \delta_{\beta\gamma} - \delta_{\alpha\delta} - \delta_{\beta\delta}) \right\} + \\ &+ \bar{V} \frac{d^2\Psi}{d\bar{V}^2} \delta_{\alpha\beta} \delta_{\gamma\delta} \end{aligned} \quad (6.14)$$

In the case of pair potentials (6.4),  $\Psi = 0$  and only the first terms of Equations (6.13) and (6.14) remain. When the EAM (6.7) or related methods are used to model interatomic interactions, it is necessary to use the complete Equations (6.13) and (6.14) that take into account the density (the average volume per atom) dependence of the potential [24].

The strain tensor components are calculated based on the deviation of the current network configuration from the reference configuration defined at the initial moment. Since atomic positions are known at all times (therefore, in both the initial and the current configuration), calculating the strain components is a straightforward task. For example, the components of the left Cauchy-Green strain tensor, corresponding to  $i$ -th atom of the plane system, are commonly (e.g., [37]) defined as follows

$$b_{\alpha\beta} = \frac{1}{3} \sum_{j=1}^6 (\bar{r}_{ij})_{\alpha} (\bar{r}_{ij})_{\beta}, \quad (\bar{r}_{ij})_{\alpha} = (r_{ij})_{\alpha}/r_0. \quad (6.15)$$

It should be noted that, unlike the stress definition (6.13), the virial equation (6.15) is instantaneously valid in time and space, that is, it does not require time averaging.

### Temperature

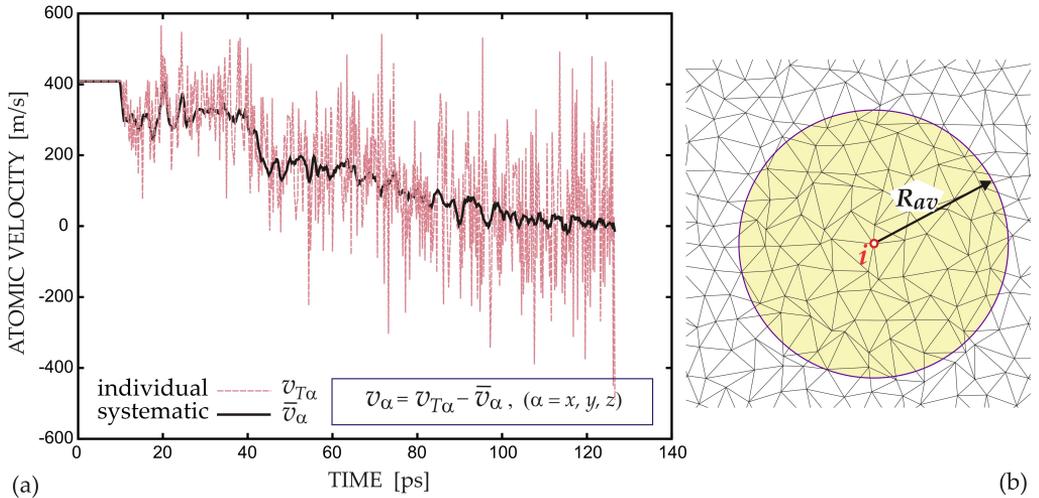
The temperature evaluation during dynamic processes raises some fundamental questions related to the thermodynamics of nonequilibrium processes in relation to basic thermostatic concepts such as entropy and absolute temperature of nonequilibrated systems (e.g., [34], [38], [39]). That discussion is bypassed herein. It is deemed sufficient just to point out the Callen's [40] claim that the nonequilibrium definitions of entropy and temperature are based on uncertain premises. Be it as it may, a consensus has been reached, over time, on the use of a standard definition of temperature, known from the kinetic theory of gases. This definition is based on the equipartition theorem that provides a relationship between the average kinetic energy and the instantaneous kinetic temperature of the system: atomic velocities establish a thermometer. According to this theorem, each degree of freedom contributes  $k_B T/2$  to the internal energy of the system from where it follows

$$T \equiv \frac{2}{3k_B} \langle E_k \rangle = \frac{2}{3k_B} \left\langle \frac{mv^2}{2} \right\rangle. \quad (6.16)$$

where  $k_B$  is the Boltzmann constant. The definition (6.16) is firmly established in statistical mechanics as it derives from the distribution of the highest probability of a canonical  $(N, V, T)$  ensemble [34], [41]. Thus, expression (6.16) tacitly implies that the system is thermally equilibrated, and the atomic velocities distributed in accordance with the **Maxwell-Boltzmann distribution** (6.17) [42]. As argued by Holian and coworkers [39], the instantaneous kinetic temperature is the only meaningful definition in nonequilibrium situations.

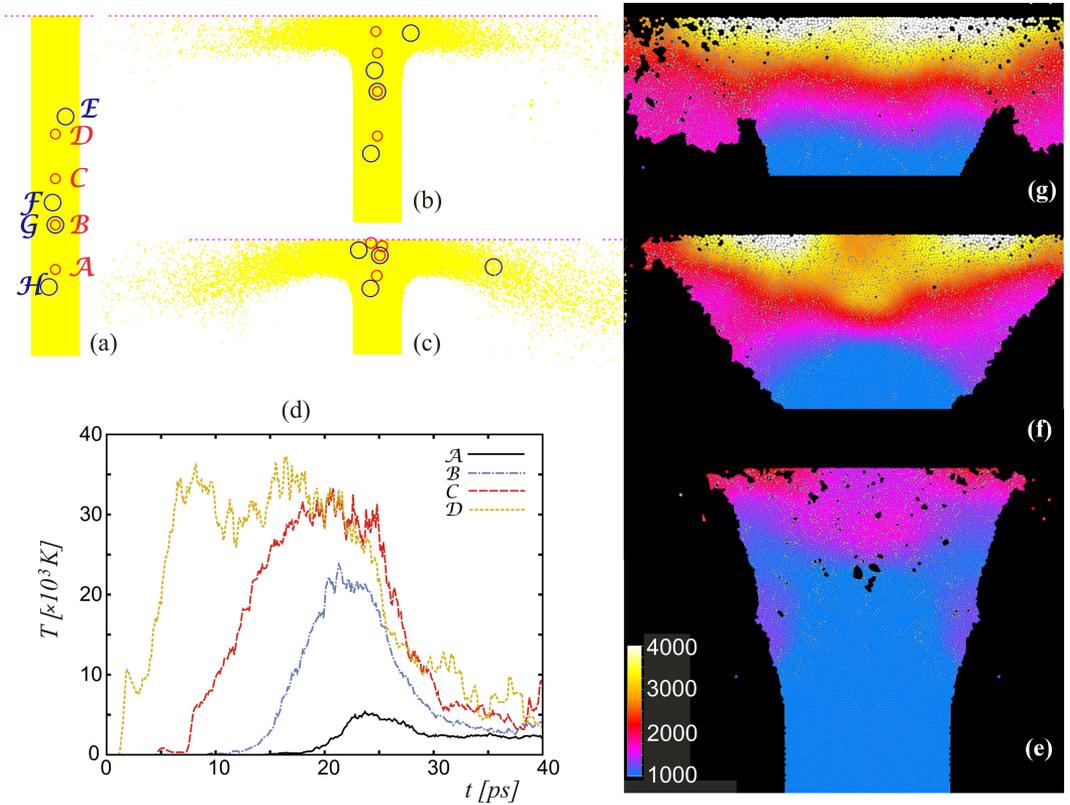
Importantly, the intensity of velocity vector  $\mathbf{v}(v_x, v_y, v_z)$  appearing in Equation (6.16) corresponds to the vibratory motion and, therefore, does not contribute to the resultant momentum. Nonetheless, it possesses a finite kinetic energy that is identified with the thermal energy and is related to the absolute temperature in MD simulations. With reference to Figure 6.6a, the vibrational velocity components could be obtained by subtracting the velocity of correlative (systematic) motion (full black line) from the total (individual) atomic velocity (dashed red line). The total velocity of each atom is obtained directly by solving Newton's equations of motion (6.1)<sub>1</sub>. The associated velocity of correlative motion can be calculated by spatially averaging the total velocities of all atoms belonging to a particular averaging region centered at the atom in question (Figure 6.6b). This averaging

volume (surface in 2D) plays a role of a virtual sensor (thermometer in this case). This approach is reminiscent of the division of the motion of molecules, in the kinetic theory of gases, into a "random" and a "systematic" part [44].



**Figure 6.6:** (a) A typical time history of atomic speed recorded in the course of 0.408 [km/s] ballistic Taylor test [45]. The full black line represents the systematic velocity of (correlative) motion while the dashed red line represents the corresponding total velocity of the individual atom. The difference between the total velocity and the associated systematic velocity results in the velocity of vibratory motion which defines the thermal energy of random vibrations related to the instantaneous kinetic temperature (6.16) (Adopted from [43]). (b) Averaging area (in general, volume) superimposed on an irregular lattice for evaluation of the macroscopic state parameters in MD simulations (**virtual sensor**).

The instantaneous kinetic temperature (6.16) is averaged in both time and space. In 2D simulations, the averaging area is usually taken to be the same circular region (shaded in Figure 6.6b) that was previously used to calculate the correlative motion velocity. Therefore, each node in the network also represents the center of the averaging area of radius  $R_{av}$ , characterized by the corresponding velocity of correlative motion and the temperature obtained by averaging within the specified area. The size of the averaging region is the result of a compromise between contradictory requirements for the largest possible size of statistical sample and the finest possible calculation (measurement) resolution. An example of the arrangement of the averaging regions is shown in Figure 6.7 for the case of a slender flat-head projectile hitting a rigid wall (the Taylor ballistic test [45]). The circular averaging areas (6.6b), which play the role of measuring gages, follow the movement of the atoms on which they are centered during the sample deformation. Accordingly, if during deformation the atoms, on which the averaging surfaces are centered, approach the edge of the deformed object, incomplete averaging may occur, which has different consequences for different macroscopic parameters (depending on their definitions; Equations (6.13)-(6.15)) which should be carefully examined in each specific case.



**Figure 6.7:** Temperature evolution for the simulation of the Taylor ballistic test (the (nano)projectile collision with the rigid wall). (a-c) Sequence of deformed nanoparticle configurations upon  $v_{imp} = 7$  [km/s] rigid-wall collision with the marked positions of the eight averaging areas ("measurement gages") used to evaluate the macroparameters of state: (a-c) 0.2[ps], 6[ps], 12[ps], respectively (Adopted from [46]). (d) An example of time histories of temperature recorded at four measurement locations  $A - D$  equidistantly distributed along the longitudinal axis of symmetry of the nanoparticle at the impact velocity  $v_{imp} = 4$  [km/s] (Adopted from [43]). (e-g) Examples of temperature field evolutions during the  $v_{imp} = 0.7$  [km/s] simulation; the images correspond 10[ps], 20[ps], and 50[ps], respectively. The scale of values on the label refers to the absolute temperature in degree Kelvin.

### Thermal initialization of the system

Assignments of the initial and boundary conditions necessary for an MD simulation imply definition of the initial positions and initial velocities of all atoms of the system. The vibrational part of the initial velocities are generally defined by selecting the velocity intensities for each atom from the Maxwell-Boltzmann distribution for the desired initial sample temperature ( $T_0$ ) while the velocity directions are assigned randomly.

The Maxwell-Boltzmann distribution of the vibrational velocities provides the probability density for atoms with the velocity intensity  $v$  and has the form

$$pdf(v) = 4\pi \left( \frac{m}{2\pi k_B T_0} \right)^{3/2} v^2 \exp\left( -\frac{mv^2}{2k_B T_0} \right). \quad (6.17)$$

Figure 6.8 illustrates the Maxwell-Boltzmann velocity intensity distribution for tungsten (W) at temperatures of 300 [K] and 1000 [K]. As can be seen, the distribution is not symmetric: the average value of the velocity intensity  $(6.18)_1$  is always greater than the most probable value  $(6.18)_2$  (corresponding to the maximum of the distribution curve)

$$\langle v \rangle = \sqrt{\frac{8k_B T_0}{\pi m}}, \quad v_m = \sqrt{\frac{2k_B T_0}{m}} = \frac{\sqrt{\pi}}{2} \langle v \rangle. \quad (6.18)$$

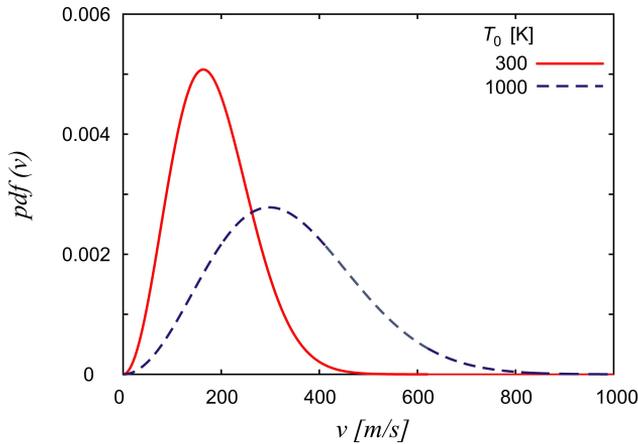


Figure 6.8: The Maxwell-Boltzmann speed distributions for tungsten for two different temperatures.

### 6.2.5 MD Simulation Cell and List of Neighbors

The response of small systems can be dominated by surface effects that may obscure the physical response of the bulk material. A simulation cell with **periodic boundary conditions** (Figure 6.9) is introduced to eliminate these contaminating surface effects whenever it is of interest to study behavior of the bulk material. The volume (surface area in 2D) of a periodic cell is representative of bulk material in the sense that it is considered to be composed of periodic cells surrounded on all sides by their exact replicas [5] as illustrated in Figure 6.9. This periodicity implies that the atom, in the bottom-right cell corner of Figure 6.9, leaving the MD simulation cell instantaneously reappears at the bottom-left cell corner.

This elimination of unwanted surface effects using these periodic cells is achieved at the cost of introducing non-physical periodicity into the atomic system. Adverse consequences include: (i) unrealistically rigid response, (ii) unnatural wavelengths in the solution fields, (iii) suppression of localization that might otherwise occur [47], and (iv) violation of the conservation of angular momentum [15].

In order to simulate the most general loading conditions, it is necessary to be able to change the shape and size of the periodic cell and maintain constant values of certain state parameters (typically, temperature or pressure; the *NTP* ensemble). The restriction that the shape of the periodic cell must remain unchanged has a negative impact on the applicability

of the periodic cell. Therefore, several alternative MD methods have been proposed as potential solutions for above mentioned problems in a series of papers [48]-[51]. In these studies, the shape of a periodic cell is treated as a phenomenological variable subject to change over time.

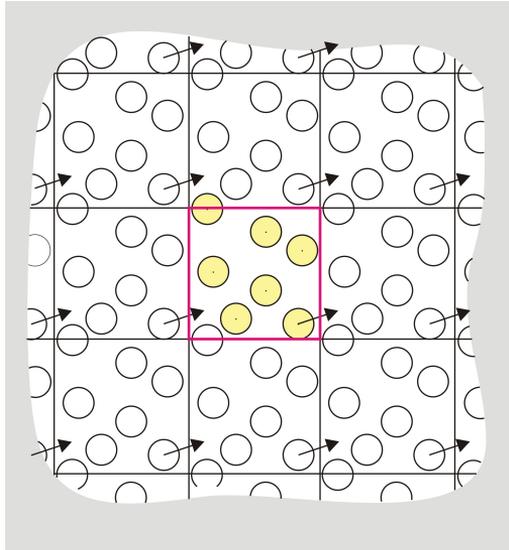


Figure 6.9: Periodic boundary conditions and the simulation cell.

The time required to calculate the interactions between the  $N$  atoms of a 2D system using algorithm (6.9) is proportional to  $N^2$  [5]. In order to improve the speed of program execution, it is very useful to maintain a **list of neighbors** of each atom that Verlet originated in his classic work [52]. For example, the list of neighbors of the  $i$ -th atom is formed by including all atoms at a distance less than the prescribed cut-off distance. Between periodic updates of the neighbor list (typically, several calculation cycles), the program does not check all other atoms in the system but only those that appear in the list for a given atom.

### 6.2.6 Temperature and Pressure Control

In the absence of dissipative forces, the equation of motion of classical mechanics results in the conservation of the total energy of the system which corresponds to a microcanonical  $(N, V, E)$  ensemble. The calculated temperature (6.16) and pressure in the standard MD formulation are not constrained and can vary significantly during the simulation (Figure 6.7). Most often, it is of utmost importance to evaluate these temperature and pressure changes. On the other hand, sometimes it is of interest in MD simulations to constrain temperature or pressure or both. The importance of temperature and pressure control has made the topics of virtual thermostats and barostats very popular among researchers, resulting in an abundance of different methods (e.g., [53]-[55]).

### Nose-Hoover thermostat

Atomic velocities establish a thermometer, as argued in Chapter 6.2.4. All algorithms performing the role of thermostat use some modifications of Newton's second law of motion to provide a constant average temperature of the particle ensemble by adding and removing its energy [55]. The temperature control is illustrated here by the Nose-Hoover algorithm as formulated by Holian and co-authors [39].

The Nose-Hoover thermostatted MD has a theoretical basis in the classical thermodynamic concept of "connecting" a thermodynamic system (in this case, the MD cell) with a heat reservoir that ensures a constant temperature during the simulation. Consequently, the given atomic ensemble becomes, by definition, canonical  $(N, V, T)$ . The Nose-Hoover model uses the classical temperature concept (discussed in Chapter 6.2.4) based on the instantaneous kinetic energy of the system. The technique of providing thermostatic conditions is of the integral-feedback type. First, an additional term is inserted into Newton's second law of motion

$$\ddot{\mathbf{r}}_i = \frac{\mathbf{F}_i}{m_i} - \vartheta_H \xi_H \dot{\mathbf{r}}_i \quad (6.19)$$

to control the „jiggings and wiggings“ of atoms. Second, the dynamic variable,  $\xi_H$ , must satisfy the additional equation of motion

$$\dot{\xi}_H = \vartheta_H \left( \frac{T}{T_0} - 1 \right) \quad (6.20)$$

which provides the necessary feedback. In the differential Equation (6.19),  $\vartheta_H$  is the coupling speed of the atom with the thermal reservoir. The standard form of Newton's equation of motion is recovered from expression (6.19) for  $\vartheta_H = 0$ . The variation of the heat distribution variable  $\xi_H$  ensures that the long-term average kinetic energy (consequently, the average temperature as well) remains constant, while allowing fluctuations in its current value (Figure 6.10).

Numerical integration of Nose-Hoover equations of motion using the Störmer central difference algorithm ((6.10) and (6.11)) [56] reduces to the following expressions

$$\begin{aligned} \mathbf{r}(t) &= \mathbf{r}(t - \delta t) + \delta t \dot{\mathbf{r}}(t - \delta t/2) + O(\delta t^3), \\ \xi_H(t) &= \xi_H(t - \delta t) + \vartheta_H \left[ \frac{T(t - \delta t/2)}{T_0} - 1 \right] \delta t + O(\delta t^3), \\ \dot{\mathbf{r}} \left( t + \frac{\delta t}{2} \right) &= \frac{1}{1 + \vartheta_H \xi_H(t) \delta t/2} \left\{ \dot{\mathbf{r}} \left( t - \frac{\delta t}{2} \right) \left[ 1 - \frac{1}{2} \delta t \vartheta_H \xi_H(t) \right] + \frac{\mathbf{F}(t)}{m} \delta t \right\} + O(\delta t^3). \end{aligned} \quad (6.21)$$

A detailed consideration of this method is available in [39]. A more advanced version of the thermostat is available in [57].

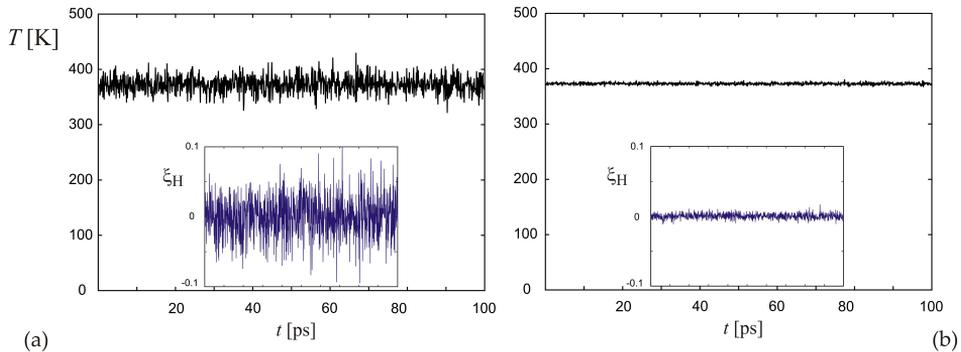


Figure 6.10: Main panels: Examples of current temperature fluctuations in time for a constant mean value of 373[K] in systems with the number of atoms: (a) 1300, and (b) 34000. Inserts: Corresponding fluctuations of the dynamic variable  $\xi_H$  which ensure the constant temperature.

### 6.2.7 Advantages and Disadvantages of Traditional MD

The main advantages of conventional MD methods are:

- (i) The only model input (excluding boundary and initial conditions) is the interatomic potential. Other than that, there are no classical constitutive models that postulate the relationships between the main thermomechanical parameters of the state.
- (ii) No assumptions are made regarding the nature of the physical mechanisms or simulated processes that are the subject of the study. Accordingly, the possibility opens up for a well-designed virtual experiment to provide a detailed and unbiased insight into the main mechanisms of the studied physical phenomena.
- (iii) The above-mentioned insight into the mechanisms of the studied physical phenomena can be achieved with extremely high spatio-temporal resolution.

The main disadvantages of classical MD methods are:

- (i) **Classical description of interatomic interactions**, based on empirical potentials, greatly simplifies the rigorous quantum mechanical nature of the material.
- (ii) **Classical description of interatomic forces** in which Newton's second law of motion replaces the Schrödinger equation of quantum mechanics. One of the indicators of the validity of this approximation is the de Broglie wavelength, which for thermal motions has the form

$$\Lambda_{th} = \frac{h}{\sqrt{2\pi mk_B T}}$$

where  $h$  is the Planck constant,  $m$  the atomic mass,  $k_B$  is the Boltzmann constant and  $T$  is the absolute temperature [16]. Quantum effects are considered to be negligible if the de Broglie wavelength is less than the interatomic distance,  $\Lambda_{th} < r_0$ , or if the frequency of vibrational motion,  $\nu$ , is such that  $h\nu < k_B T$ . In other cases, the atoms of most chemical elements (except the lightest ones like  $H$ ,  $He$ , noble gases) can be considered the material points at sufficiently high temperatures and classical mechanics can be used to describe their motion.

- (iii) **Spatio-temporal constraints** which include the size of the MD model and the

duration of the simulated phenomenon, limit the range of problems that can be solved by MD methods.

Although MD has advanced tremendously and gained in popularity on the wave of the extremely rapid development of computer technology, the current state the matter still makes unthinkable the macroscopic modeling of even the shortest physical phenomena. For example, MD modeling of a  $1\text{ [mm]}^3$  copper sample is not even closely achievable, because it consists of approximately  $10^{21}$  copper atoms, which exceeds by far the capabilities of even the most powerful computer systems at present. The size of the MD simulation cell is limited typically up to  $10^8$  atoms nowadays, which ordinarily corresponds to a few tens of nanometers. According to the available data, it appears that the current record of  $4.1 \cdot 10^{12}$  atoms (equivalent to a  $4.5 \times 4.5 \times 4.5\text{ [\mu m]}$  cubic simulation cell) has been achieved on the computer platform “SuperMUC” (Leibniz Supercomputer Center of the Bavarian Academy of Sciences) using 131072 processors and over 500 TFLOPS.

The simulation duration is defined by the size of the time step and the number of cycles. A typical time step in MD simulations is of the order of femtosecond. As an example, for 2D Leonard Jones systems, the initial time step estimate is based on the expression

$$\delta t = \left( \frac{1}{60} \div \frac{1}{30} \right) \frac{2\pi}{\omega_E}, \quad \omega_E = \sqrt{3} \omega_0 \quad (6.22)$$

where  $\omega_E$  denotes the Einstein frequency associated with the fundamental harmonic frequency  $\omega_0 = C_0/r_0$  defined in terms of the speed of sound and the equilibrium interatomic distance [39].

The small time step required for MD simulations of atomic systems severely limits the total duration of the event simulated. By using modern computers and massively parallel processing it is possible to calculate approximately  $10^8$  time cycles (that is, to simulate physical phenomena that occur within 100[ns]). This is a serious limitation for many problems involving thermally activated processes (Table 6.1, Table 6.2), which is why several methods have been developed for their acceleration for the purpose of studying surface diffusion, film deposition, and the evolution of point defects. A record MD simulation of the order of  $10\text{ [\mu s]}$  describes the rapid bending of protein chains [58].

Increasing the physical duration of simulated processes is a constant, active research area that includes the improvement of numerical algorithms, parallel processing, and the continuous development of hardware capabilities.

**Table 6.1:** Examples of representative temporal and spatial scales necessary to observe some typical deformation mechanisms under step-pressure loading of single-phase metals (adopted from [59]).

Mechanism	Representative Length Scale	Representative Time Scale
Phase transformation	10 nm	10 ps
Dislocation nucleation	50 nm	50 ps
Twin formation	1 nm	1 ns
Interaction of dislocations	100 nm	100 ns
Spallation; tensile damage	1 $\mu\text{m}$	100 ns
Adiabatic shear	100 $\mu\text{m}$	10 $\mu\text{s}$

**Table 6.2:** Examples of representative temporal and spatial scales necessary to observe some typical deformation mechanisms under step-pressure loading of amorphous brittle materials (adopted from [59]).

Mechanism	Representative Length Scale	Representative Time Scale
Crack nucleation	100 $\mu\text{m}$	10 ns
Crack coalescence	1 mm	100 ns
Comminution	1 mm	1 $\mu\text{s}$
Fragment flow	1 mm	1 $\mu\text{s}$
Interfragment friction	10 mm	10 $\mu\text{s}$
Interfragment rotation	10 mm	10 $\mu\text{s}$

- (iv) **Output oversaturation** (“data glut”). The above-mentioned detailed insight into the mechanisms of the studied phenomena on the atomic scale with extremely fine time resolution can lead to the oversaturation with the raw MD output data ( $\mathbf{r}, \mathbf{v}, \mathbf{a}$ ).

## 6.3 Particle Dynamics

Particle dynamics (PD) is one of many computational methods developed to bridge the gap between the microscopic and macroscopic spatial scales (Figure 6.2). As presented herein, it is an engineering offshoot of MD on an arbitrarily selected spatial scale (Figure 6.11). Since PD has MD techniques at the root, it is sometimes called quasi-MD. Thus, the basic distinguishing features of PD in relation to MD lie in the coarser spatial scale and, in that regard, the phenomenological constitutive model that defines the interparticle interaction. It will be shown in this chapter that this constitutive model may or may not have a functional form of the empirical interatomic potentials. Other than that, the computational simulation techniques used in PD modeling are largely the same as those well known from the traditional MD literature [5], [15], [17]. Greenspan [60] contributed the most to the early development of particle methods as presented in this introduction.

### 6.3.1 Basic Idea of PD

Basically, the PD system consists of material points (referred to as particles in this chapter) of known masses  $m_i$ , and positions  $\mathbf{r}_i$  ( $i = 1, \dots, N$ ). Depending on the material being modeled (fluid, amorphous or crystalline solids), these material points can be arranged randomly or regularly according to the topology of an underlying network. As discussed in Chapter 6.2, the known initial configuration defines the reference state. The calculation methodology requires an approximate solution of the system of differential equations with given initial conditions. At an arbitrary time ( $t > 0$ ), the position and momentum of each particle are completely determined by Newton’s laws of motion (6.1)<sub>1</sub>. Therefore, the movement of each particle is deterministic.

As already mentioned, the well-established MD techniques have been adapted to simulate such coarser-scale material systems with the role of atoms being taken over by a different kind of material points—the large chunks of material—often called a continuum particle or quasi-particle. For simplicity, these material points often interact with each

other according to a simple central-force rule that completely determines the strain energy density function based on their mutual position. A system of Newton differential equations of motion for a system of particles with defined momenta  $\mathbf{p}_i = m_i \dot{\mathbf{r}}_i$  is then approximated by an appropriate system of finite difference equations and then solved using one of the many available integration algorithms as outlined in Chapter 6.2.3.

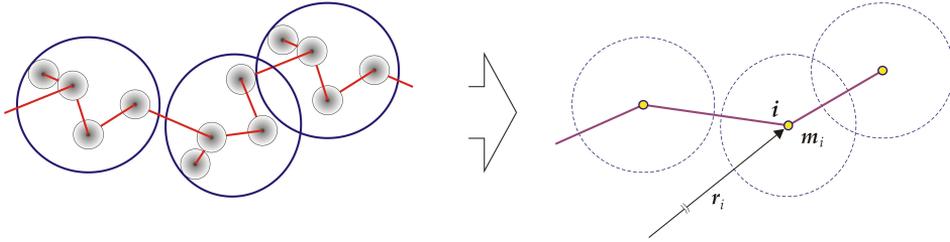


Figure 6.11: Schematic illustration of mapping between a molecular (or atomic) structure and the PD model (represented by a coarse-grained MD model).

It is interesting to note that the time-reversible equation for calculating the next particle position of the system (6.9) can be modified into the following form

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \eta [\mathbf{r}_i(t) - \mathbf{r}_i(t - \delta t)] + \delta t^2 \mathbf{a}_i(t) \quad (6.23)$$

where  $\eta$  marks the dumping coefficient. The reversible scheme (6.23) is used (often with  $\eta = 0.95$ ) in PD simulations of quasi-static problems for the purpose of dissipating kinetic energy in order to obtain an equilibrium configuration in a time-efficient manner. Assuming that the interparticle forces are conservative, the intensities of the central force with which the particle  $j$  acts on the neighboring particle  $i$  and the resulting force with which all the first neighbors act on the particle  $i$  can be calculated using expression (6.2). The standard form of time-reversible Equation (6.9) is recovered from (6.23) for  $\eta = 1$  (no dissipation).

### 6.3.2 Interparticle Potentials

The interparticle potential (which, as already mentioned, in the PD models plays the role of constitutive law) must be adopted as an initial modeling step. Along with the spatio-temporal scale of simulations, this potential represents the basic difference between MD and PD. One of the most commonly used pair potentials for interparticle interactions

$$\begin{aligned} \phi(r_{ij}) &= -\left(\frac{1}{p-1}\right) \frac{P}{r_{ij}^{p-1}} + \left(\frac{1}{q-1}\right) \frac{Q}{r_{ij}^{q-1}}, \quad q > p > 1, \\ f_{ij} &= -\frac{d\phi(r_{ij})}{dr_{ij}} = -\frac{P}{r_{ij}^p} + \frac{Q}{r_{ij}^q} \end{aligned} \quad (6.24)$$

represents a generalization of the well-known Lennard-Jones 6-12 potential (6.7). The limit case of the exponent  $p = 1$  was analyzed in detail by Wang and Ostojca-Starzewski [61] with an alternative form of potential that is necessary due to the singularity of expression

(6.24) for this value of the exponent. With regards to relation (6.24),  $P$ ,  $Q$ ,  $p$  and  $q$  are positive constants, and  $q > p$  in order to obtain a repulsive force necessarily much stronger than the attractive one. The Greenspan trailblazing method [60], based on interparticle interaction (6.24), exerted a profound influence on development of PD.

Note that the condition of zero-stress state in the equilibrium configuration (6.24)<sub>2</sub> results in the expression for the equilibrium interparticle distance

$$r_{ij0} = \left( \frac{Q}{P} \right)^{\frac{1}{q-p}} \quad (6.25)$$

defined in terms of the four parameters of the interparticle potential (6.24)<sub>1</sub>. In principle, the equilibrium interparticle distance can be shrunk to the level of a few angstroms in which case the PD model is reduced to the MD model.

An ubiquitous step, common to all CMD methods, is to ensure that the assignment of model parameters results in appropriate physical and mechanical properties of the material. With regards to that crucial step, Wang and Ostoja-Starzewski [61] introduced a novel principle of equivalence between the meso-scale particle model and the micro-scale MD model, while using the Greenspan modeling approach based on the interparticle potential (6.24). This was achieved by establishment of four equality conditions (related to mass, elastic strain energy, modulus of elasticity and tensile strength) for the purpose of determination of four unknown model parameters in (6.24). In this way, the equations for parameters  $P$ ,  $Q$ ,  $p$  and  $q$  are derived and an appropriate parametric study conducted in order to determine the model parameters under given conditions.

For example, Figure 6.12 shows the dependence of the modulus of elasticity and tensile strength on the parameters  $(p, q)$  of the interparticle potential (6.24)<sub>1</sub>, arrived at by simulations at constant values of the sample volume and the equilibrium interparticle distance [61].

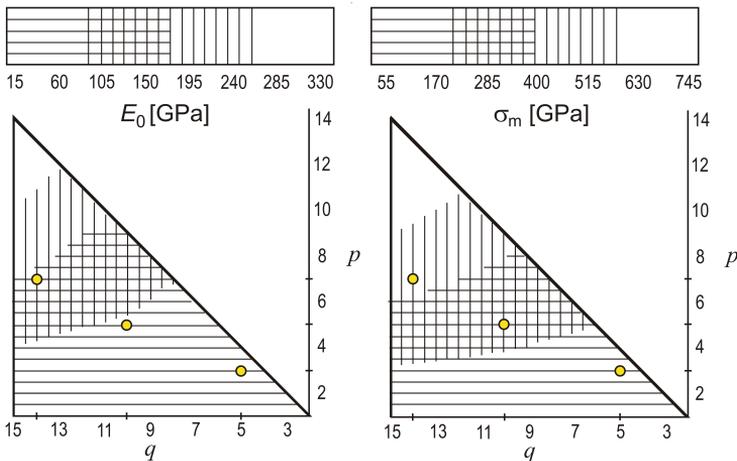


Figure 6.12: Modulus of elasticity and tensile strength dependence upon a pair of parameters  $(p, q)$  at fixed values of the interparticle distance ( $0.2$  [cm]) and sample volume ( $8.0 \times 11.4 \times 3.1$  [ $cm^3$ ]). (Reproduced from reference data [61].)

Given the range of physical characteristics of copper and its alloys ( $E_0 = 120 - 150$  [GPa] and  $\sigma_m = 250 - 1000$  [MPa]) it is obvious that, among the three options offered in Table 6.3, the most favorable is  $(p, q) = (7, 14)$ .

It is important to emphasize that this selection is not unique: the similar values of  $E_0$  and  $\sigma_m$  could be obtained with slightly different combinations of parameters  $(p, q)$  from which, on the other hand, different values of fracture toughness of materials on the macro-scale can be derived. Accordingly, the analyst has an additional "degree of freedom" to adjust the physical and mechanical properties of the target material to a certain extent.

Another potential that has found wide application in PD [62], for simulations of the behavior of brittle materials with random mesostructure, is a combination of Born-Mayer (6.26)<sub>1</sub> and Hookean (6.26)<sub>2</sub> potentials.

$$\begin{aligned}
 \phi^r(\bar{r}_{ij}) &= \frac{k_{ij}r_{0ij}^2}{(B-2)} \left( \frac{1}{B} e^{B(1-\bar{r}_{ij})} - \bar{r}_{ij}^{-1} \right), \quad \bar{r}_{ij} < 1; \\
 \phi^a(\bar{r}_{ij}) &= \frac{1}{2} k_{ij}r_{0ij}^2 (\bar{r}_{ij} - 1)^2, \quad \bar{r}_{ij} \geq 1; \\
 f^r(\bar{r}_{ij}) &= \frac{k_{ij}r_{0ij}}{(B-2)} \left( e^{B(1-\bar{r}_{ij})} - \bar{r}_{ij}^{-2} \right), \quad \bar{r}_{ij} < 1; \\
 f^a(\bar{r}_{ij}) &= k_{ij}r_{0ij}(\bar{r}_{ij} - 1), \quad \bar{r}_{ij} \geq 1.
 \end{aligned} \tag{6.26}$$

In expression for this hybrid potential (6.26), superscripts  $r$  and  $a$  designate, respectively, a repulsive and attractive branch of interaction,  $k_{ij}$  bond stiffness (related to modulus of elasticity by (6.27), while the adjusting parameter  $B$  defines the slope (steepness) of the repulsive wall (Figure 6.13, on page 243). The parameter  $B$  is, in principle, identifiable from shock experiments (e.g., it is conceptually related to the particle-velocity multiplier in the linear form of the ballistic equation of state) [62].

**Table 6.3:** Parameters  $P$  and  $Q$  of potential (6.24)<sub>1</sub>, modulus of elasticity and tensile strength determined for different pairs of  $(p, q)$  parameters shown in Figure 6.12. The reference [61] offers a more precise estimate of these quantities as well as a change in the modulus of elasticity with increasing initial interparticle distance in a given range which is conveyed here in the last line.

$(p, q)$	(3,5)	(5,10)	(7,14)
P	$2.5 \cdot 10^7$	$1.8 \cdot 10^6$	$1.1 \cdot 10^5$
Q	$9.9 \cdot 10^5$	$5.7 \cdot 10^2$	1.4
$E_0$ [GPa]	15.	70.	150.
$\sigma_m$ [MPa]	90.	270.	440.
$\Delta E_0$ [%] $r_0: (0.1 \rightarrow 0.5) cm$	-15.	-16.	-16.

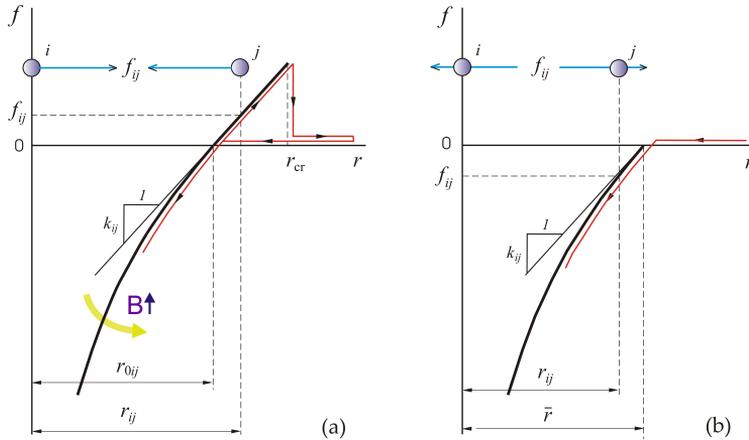


Figure 6.13: Nonlinear hybrid potential (6.26). Schematic representation of the interaction between particles that (a) are or were first-nearest neighbors, and (b) were not first-nearest neighbors initially. The yellow arrow indicates the effect of increasing the parameter  $B$ , which defines the slope of the repulsive wall [62].

The latter can be identified in dynamic simulations, for example, by matching the ballistic equation of state. Thus, the average link stiffness of the interparticle potential is determined uniquely by the value of modulus of elasticity of the pristine material

$$k_{ij} = \frac{8}{5\sqrt{3}} E_0^{(3D)}. \quad (6.27)$$

The expression (6.27) is equivalent to (6.41) of the corresponding triangular lattice model with central interactions. This equivalence also implies the fixed value of Poisson's ratio,  $\nu^{(3D)} = 1/4$  (6.39)<sub>1</sub>. The hybrid potential (6.26) was introduced to reproduce some underlying features of the deformation process typical for the considered materials, such as: brittle behavior in tension, increase of shock wave velocity and decrease of compressibility with increasing pressure.

It should be noted that linear interparticle interactions, characteristic of elastic-brittle behavior of materials (and the traditional spring-network models of Chapter 6.4.2), can be considered a modification of the above hybrid potential, with the attractive part of the potential (6.26)<sub>2</sub> being used in the repulsion domain as well [63].

Finally, Watson and Steinhauser [64] used a conceptual solution of interparticle interaction very similar to the one of Mastilović and Krajčinović [62] to model the phenomena of hypervelocity impact. The main difference is that their interparticle repulsion instead of the Born-Meyer (6.26)<sub>1</sub> used the ubiquitous Lennard-Jones 6-12 potential (6.7), while the Hooke potential (6.26)<sub>2</sub> was used in unaltered form in the attractive branch. It is interesting to note that for the purposes of their 3D simulations, Watson and Steinhauser [64] adjusted the remaining two parameters of the model—corresponding to the depth of the potential well in (6.7) and the bond stiffness in (6.26)<sub>2</sub>—by a fitting procedure based on a series of hypervelocity impact experiments (a sphere colliding with a thin plate). This nicely illustrates the connection between physical (laboratory) and virtual experiments that is becoming an integral part of contemporary computational modeling.

As already noted, the PD simulation is completely deterministic. However, the particle configurations that mimic the modeled material are not necessarily associated with an ideal lattice. As an example, in order to describe the deformation of the brittle material with a random microstructure, it is necessary to introduce the **quenched disorder** into the computer model. This initial disorder can be topological (unequal coordination number), geometric (unequal bond length) or structural (unequal bond stiffness and/or strength). It can be introduced through the probability distributions of strength, stiffness, and missing bonds (which mimics porosity or pre-existing damage). This (so called, quenched) disorder increases with damage evolution (the induced disorder) in the course of deformation process. Therefore, the nature of damage evolution is inherently stochastic due to the initial stochasticity of the PD model (although each individual physical realization of the given statistics is deterministic).

## 6.4 Lattices

Lattice models are a class of CMD models based on the concept of computation domain discretization with an assembly of one-dimensional elements (springs, trusses, beams) endowed by elementary constitutive rules and rupture criteria. They are closely related to PD, as outlined in the preceding chapter, since every system of particles can be associated with a lattice, especially in the case of solids. (As an example, the mentioned modification of the hybrid potential (6.26), with the attractive part of the potential (6.26)<sub>2</sub> being used in the repulsion domain as well, constitutes the simplest lattice model - the spring network.) The first application of the lattice method is attributed to the Russo-Canadian engineer Alexander Hrennikoff [7], who devised it to solve the plane stress problem of a thin elastic plate loaded with in-plane forces. The method then fell into oblivion and remained dormant until the 1980s when its remarkable capabilities for introducing material disorder and heterogeneities into the computational model in a simple and natural way were noticed. Ever since, simplicity and inherent ability to capture localized failure mechanisms led to rapid development of lattice models. Not surprisingly, these models are of special importance for studies of mechanical fracture of quasi-brittle materials (e.g., concrete), which were necessarily of phenomenological character. Nonetheless, applications for metals, ceramics, polymers, composites, granular materials are available in literature ([65]-[67]).

The chronological development and adaptations of lattice models to different types of materials and loads are encapsulated herein in a most concise form. To begin with, it should be noted that the continuum can be discretized by lattice models in various ways (plane or spatial lattices, of regular or irregular (random) geometry, with overlapping or non-overlapping elements). Lattice models can also differ in the number of degrees of freedom per node (truss vs. beam), which has proven to be a source of important distinction when it comes to their ability to realistically reproduce physical phenomena. Over time, a consensus was reached that lattice models of irregular geometry with beam interactions were most suitable for fracture simulations, especially in materials characterized with a distinctly heterogeneous structure.

### 6.4.1 Basic idea of Lattice Models

When formulating the lattice model, it is crucial to establish the relationship between the lattice parameters and the mechanical properties of the solid material being simulated. The gist is that under the uniformly applied load, the lattice should reproduce the linear elastic behavior of the corresponding equivalent continuum and its uniform deformation. Different approaches have been proposed in this regard (e.g., [67]-[71]).

The methodological approach employed herein is based on the **deformation energy** equivalence and follows closely the original works of Ostoja-Starzewski and co-workers [3], [8], [72], [73]. The basic idea is to ensure the equivalence of the deformation energy contained in the deformed unit cell of the lattice (e.g., the hexagonal unit cell in Figure 6.14a) with that in the associated continuum structure (of the same volume  $V$ )

$$U_{cell} = U_{continuum}. \quad (6.28)$$

The deformation energy is defined in continuum mechanics by expression

$$U_{continuum} = \frac{1}{2} \int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon} dV = \frac{1}{2} \int_V \sigma_{\alpha\beta} \varepsilon_{\alpha\beta} dV. \quad (6.29)$$

If we restrict ourselves to a uniform strain field  $\boldsymbol{\varepsilon}$ , Equation (6.29) becomes

$$U_{continuum} = \frac{V}{2} \boldsymbol{\varepsilon} : \mathbf{C} : \boldsymbol{\varepsilon} = \frac{V}{2} C_{\alpha\beta\gamma\delta} \varepsilon_{\alpha\beta} \varepsilon_{\gamma\delta} \quad (6.30)$$

In order to establish relationships between the macroscopic material properties (e.g., the effective stiffness components  $C_{\alpha\beta\gamma\delta}$ ) and the lattice parameters, it is necessary to define a lattice unit cell based on specific periodic arrangements of the associated nodal points and their mutual interactions. For brevity, only the regular triangular lattice with the hexagonal unit cell will be used in the following deliberations (Figures 6.14-6.16).

### 6.4.2 Lattices with Central Interactions ( $\alpha$ Models)

The lattice with central-force interactions (also known as the spring-network or  $\alpha$  model) is the basic model in the sense that each bond represents a truss or a spring that transmits only an axial force:  $\mathbf{f} = f_n \mathbf{n}$  directed along the bond direction defined by the unit vector  $\mathbf{n}$  (Figure 6.14). The deformation energy contained in a unit lattice cell with central interactions is the sum of deformation energies of the constituent lattice elements (bonds)

$$U_{cell} = \sum_b E^{(b)} = \frac{1}{2} \sum_b^{N_b} (\mathbf{f} \cdot \mathbf{u})^{(b)}. \quad (6.31)$$

In Equation (6.31),  $\mathbf{u} = \mathbf{u}_{ij} = \mathbf{u}_i - \mathbf{u}_j$  designates the resulting change of length of the lattice element that connects the lattice nodes (material points)  $i$  and  $j$ ,  $b$  is bond index ( $b$ -th truss/spring), and  $N_b$  is their total number. If we restrict ourselves to linearly elastic interactions, Equation (6.31) can be written as

$$U_{cell} = \frac{1}{2} \sum_b^{N_b} (k^n \mathbf{u} \cdot \mathbf{u})^{(b)} \quad (6.32)$$

where  $k^n$  designated the bond stiffness. (Compare with (6.8)<sub>1</sub> and (6.26)<sub>2</sub>.)

At this point, it is necessary to define an associated unit lattice cell. The derivation of the equations of the connection between the bond (spring) constants and the effective stiffness components is based on equivalence (6.28). A key step in this process is to establish a connection between  $\mathbf{u}$  and  $\boldsymbol{\varepsilon}$ , which depends on the specific geometry of the lattice unit cell and the specific model of the interaction between lattice nodes.

### Triangular lattice with central interactions

The only truss lattice to be considered in detail herein is an equilateral triangular lattice with the central-force interactions between the first-nearest neighbors. The lattice, illustrated in Figure 6.14, is based on a hexagonal unit cell and six lattice elements (springs or trusses) of length  $\ell$ , equal to a half the equilibrium distance,  $r_0$ , between lattice nodes which defines the equilibrium lattice geometry (the reference state). The area (volume of unit thickness) of a hexagonal unit cell is  $V = 2\sqrt{3}\ell^2$ . Each bond  $b$ , belonging to a given unit cell, is characterized by a spring constant  $\alpha^{(b)}$  and unit vector  $\mathbf{n}^{(b)}$  defining the bond direction, with corresponding angles  $\theta^{(b)} = (b-1)\pi/3$  ( $b = 1, \dots, 6$ ) with respect to the horizontal.

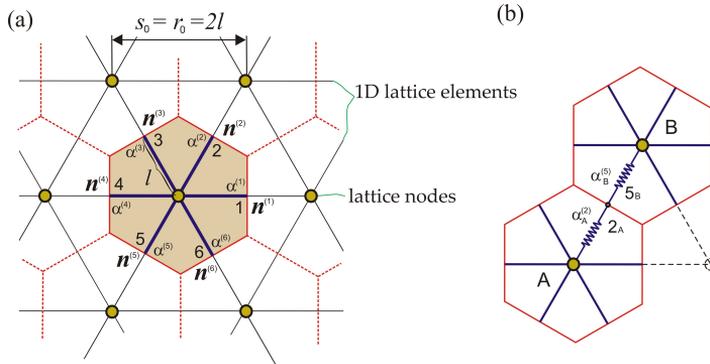


Figure 6.14: (a) Regular triangular lattice with a hexagonal unit cell and central interactions between the first neighbors ( $\alpha$  model); (b) a serial connection associated with a link between two adjacent unit cells.

The deformation energy stored in the unit hexagonal cell consisting of six evenly stretched connections is

$$U_{cell} = \frac{1}{2} \sum_{b=1}^6 (\alpha \mathbf{u} \cdot \mathbf{u})^{(b)} = \frac{\ell^2}{2} \sum_{b=1}^6 \alpha^{(b)} n_{\alpha}^{(b)} n_{\beta}^{(b)} n_{\gamma}^{(b)} n_{\delta}^{(b)} \varepsilon_{\alpha\beta} \varepsilon_{\gamma\delta}. \quad (6.33)$$

Based on equivalence (6.28), the components of the elasticity tensor can be identified in the following form

$$C_{\alpha\beta\gamma\delta} = \frac{1}{2\sqrt{3}} \sum_{b=1}^6 \alpha^{(b)} n_{\alpha}^{(b)} n_{\beta}^{(b)} n_{\gamma}^{(b)} n_{\delta}^{(b)} \quad (6.34)$$

which, in the case of equal spring constants  $\alpha(b) = \alpha$  ( $b = 1, \dots, 6$ ), reduces to

$$\begin{aligned} C_{1111} = C_{2222} &= \frac{9}{8\sqrt{3}}\alpha = \frac{E}{1-\nu^2}, \\ C_{1122} = C_{2211} &= \frac{3}{8\sqrt{3}}\alpha = \frac{E\nu}{1-\nu^2}, \\ C_{1212} &= \frac{3}{8\sqrt{3}}\alpha = \frac{E}{2(1+\nu)}. \end{aligned} \quad (6.35)$$

It should be noted that expressions (6.35) satisfy the isotropy condition

$$C_{1212} = (C_{1111} - C_{1122})/2. \quad (6.36)$$

In addition, since the value of the apparent plane-strain Poisson's ratio<sup>2</sup> is fixed, the spring constant defines only the plane modulus of elasticity of the unit lattice cell

$$\nu = \frac{C_{1122}}{C_{1111}} = \frac{1}{3}, \quad E = \frac{\alpha}{\sqrt{3}}. \quad (6.37)$$

Importantly, for all types of lattices, the upper limit of the plane Poisson's ratio is defined by the value (6.37)<sub>1</sub>. This value could be modified in various ways. For example, it is possible to define the so-called "triple honeycomb network" (sometimes the term  $\alpha - \beta - \gamma$  model is also used [8]) in which the base of the unit cell is a regular hexagon but, in the notation used in Figure 6.14,  $\alpha^{(1)} = \alpha^{(4)} \neq \alpha^{(2)} = \alpha^{(5)} \neq \alpha^{(3)} = \alpha^{(6)}$ . This selection of spring stiffnesses results in the following expression

$$\nu = 1 - \frac{2}{\left[1 + \frac{2}{9}(\alpha^{(1)} + \alpha^{(2)} + \alpha^{(3)}) \left(\frac{1}{\alpha^{(1)}} + \frac{1}{\alpha^{(2)}} + \frac{1}{\alpha^{(3)}}\right)\right]}$$

for the plane Poisson's ratio, which can reproduce values in the range 1/3 to 1. When  $\alpha^{(1)} = \alpha^{(4)} = \alpha^{(2)} = \alpha^{(5)} = \alpha^{(3)} = \alpha^{(6)} = \alpha$ , the value (6.37)<sub>1</sub> is recovered.

The plane coefficients  $E$  and  $\nu$ , defined by (6.37) in terms of the spring stiffness  $\alpha$ , are not material properties but simply their 2D counterparts, mere parameters, obtained by combining the elastic properties under the plane-strain (or, generally, the plane-stress) conditions. Since, the 2D triangular lattice (Figure 6.14) is equivalent to three-dimensional continuum under the plane strain conditions [74], the corresponding relationships between the real (3D) and the **apparent (2D) material properties** are

$$\nu = \frac{\nu^{(3D)}}{1 - \nu^{(3D)}}, \quad E = \frac{E^{(3D)}}{1 - [\nu^{(3D)}]^2}. \quad (6.38)$$

Consequently, Poisson's ratio and the modulus of elasticity corresponding to Equations (6.37) are

$$\nu^{(3D)} = \frac{1}{4}, \quad E^{(3D)} = \frac{5\sqrt{3}}{16}\alpha. \quad (6.39)$$

<sup>2</sup>The apparent plane-strain Poisson's ratio and the apparent plane-strain modulus of elasticity are henceforth, for brevity, referred to as the plane Poisson's ratio and the plane modulus of elasticity.

Note that the fixed value of Poisson's ratio,  $1/4$  (6.39)<sub>1</sub>, is a reasonable approximation for many engineering materials.

Finally, in lattice modeling it is often useful to designate the spring stiffnesses directly to the bond between two lattice nodes (instead of using the half values associated with the hexagonal unit cell). With regards to Figure 6.14b, since the two springs between the adjacent unit cells  $A$  and  $B$  are connected in series, their equivalent spring stiffness is

$$\frac{1}{\alpha_{eq}} = \frac{1}{\alpha_A^{(2)}} + \frac{1}{\alpha_B^{(5)}} \Rightarrow \alpha_{eq} = \frac{\alpha_A^{(2)} \alpha_B^{(5)}}{\alpha_A^{(2)} + \alpha_B^{(5)}}. \quad (6.40)$$

For the case of equal spring stiffnesses,  $\alpha^{(b)} = \alpha$  ( $b = 1, \dots, 6$ ), for all unit cells, the preceding expression ( $\alpha_{eq} = \alpha/2$ ) in combination with (6.39)<sub>2</sub> yields

$$\alpha_{eq} = \frac{8}{5\sqrt{3}} E^{(3D)}. \quad (6.41)$$

Thus, Equation (6.41) provides a direct link between the (meso-scale) spring stiffness  $\alpha_{eq}$  of the equilateral triangular lattice with the central interactions and the (macro-scale) material property  $E^{(3D)}$ . (Compare with expression (6.27).)

### Chronology of development of lattices with central interactions

Lattices with central interactions, as the simplest lattices, gained popularity very early in the failure modeling of heterogeneous materials with disordered microstructure. Although they began to be used earlier, they achieved the greatest momentum through the pioneering works of Bažant and his associates [67], [68]. Schlangen and van Mier [75], [76] noticed very early the possibilities of the  $\alpha$  model to simulate qualitatively the process of damage and fracture of concrete represented by a network of aggregates (as lattice nodes) bonded by the cement (as lattice elements) of inferior tensile strength. However, it should be borne in mind that these simple models are inherently unable to realistically capture more complex crack propagation patterns and reproduce fracture shapes resulting from a combination of basic failure modes [69], [74], [77], [78]. For example, if  $\alpha$  models are calibrated to reproduce cracks due to tension and fracture in the first mode, they will do that reasonably well; however, that same model will significantly exaggerate the compressive strength and will not realistically reproduce the stress-strain curve in the post-critical (softening) regime. Also,  $\alpha$  models are not able to accurately predict fracture envelopes at complex stress states due to the oversimplified unit-cell stiffness. Despite all of the above, thanks to their simplicity and computational efficiency, these models are—regardless of the abundance of more sophisticated methods—still popular among researchers for simulations in which these shortcomings are not fully expressed [79], [80].

Bažant and co-authors [67], [68], [81] used a central-force lattice with irregular geometry to model the brittle heterogeneous material. In contrast to lattice models in which the distance between nodes is an arbitrary input parameter, they selected the lattice node initial locations by mapping the actual meso-structure of concrete. In other words, the positions of the lattice nodes coincide with the centers of aggregates; therefore, the lattice topology reflects the actual concrete texture. As it will be discussed later, this modeling

approach innovatively introduces the post-critical behavior as a part of the micro-scale constitutive model. Ever since, various approaches were devised to deal with the softening of lattice elements [97]. This model has been immensely influential in development of numerical techniques for these classes of materials (e.g., [70], [82]-[85]).

It cannot be overemphasized that Bažant and co-authors [67], [68] used the **post-critical softening regime** (i.e., the progressive degradation of stiffness) in the constitutive stress-strain relations for the bonding matrix and the interface despite the fact that it is not considered an inherent material property. The softening, thus introduced into the model, is then defined indirectly by the fracture energy of the interparticle bond. More than a decade after this model gained prominence, the assignment of the softening properties to the micro/meso scale has not been not fully defined despite great efforts in that direction (e.g., [86]-[88]). The standard test to determine softening parameters is still elusive due to the unavoidable experimental material-structural interaction. Consequently, the introduction of softening in constitutive relations on the meso-scale was criticized by van Mier [87], on the grounds that there is a danger that the basic mechanisms may be missed due to the fact that the desired results are achieved by introducing additional model parameters. At the same time, many authors advocated the use of softening in constitutive modeling as indispensable (e.g., [89]) since, in their opinion, the meso-scale heterogeneity alone cannot account fully for the experimentally-observed dissipative response at the macro-scale. This difference of opinion is important to point out since the introduction of softening on micro- and meso-scales (based on macroscopic observations) is a common practice as well as an active research topic nowadays due to the computational benefits it provides.

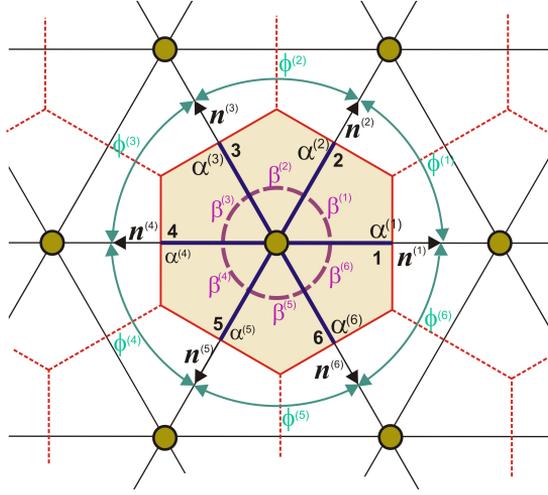
### 6.4.3 Lattices with Central and Angular Interactions ( $\alpha - \beta$ Models)

The lattice with a central-angular interaction (also known as the  $\alpha - \beta$  model) was created by upgrading the  $\alpha$  model by adding an angular spring between adjacent connections that meet at the same node [90]. Consequently, the energy is necessary to overcome the bending angle resistance reminiscent of (6.8)<sub>2</sub>.

The triangular  $\alpha - \beta$  model was considered in detail by Kale and Ostoja-Starzewski [91] in the context of studying the material damage.

#### Triangular lattice with central and angular interactions

A triangular lattice with central and angular interactions was obtained by introducing angular springs in the manner illustrated in Figure 6.15. The stiffness of these angular springs is defined by the spring constant  $\beta^{(b)}$ . According to the conditions of symmetry of the unit cell with respect to the corresponding lattice node, the elastic properties of a hexagonal unit cell are completely determined by six independent spring constants  $\{\alpha^{(b)}, \beta^{(b)}, b = 1, 2, 3\}$ :  $\alpha^{(b)} = \alpha^{(b+3)}$  and  $\beta^{(b)} = \beta^{(b+3)}$  for  $b = 1, 2, 3$  (isotropic Kirkwood model).



**Figure 6.15:** A regular triangular lattice with a hexagonal unit cell and a central and angular interaction between the first neighbors ( $\alpha - \beta$  model).

The angle of the angular spring with respect to the horizontal axis, in the reference configuration, illustrated in Figure 6.15, is  $\theta^{(b)} = (b-1)\pi/3$  ( $b = 1, \dots, 6$ ). The infinitesimal change of that angle is

$$\Delta\theta^{(b)} = \Delta\theta_{\gamma}^{(b)} = e_{\gamma\alpha\beta} \varepsilon_{\beta\delta} n_{\alpha} n_{\delta}, \quad (\alpha, \beta, \delta = 1, 2 \text{ and } \gamma = 3) \quad (6.42)$$

with remark that  $e_{\gamma\alpha\beta}$  is the permutation tensor (with  $\gamma = 3$  for 2D). With regards to Figure 6.15, the corresponding infinitesimal change of the angle between two adjacent bonds ( $b$  and  $b+1$ ), associated with the angular spring constant  $\beta^{(b)}$ , is  $\Delta\phi^{(b)} = \Delta\theta^{(b+1)} - \Delta\theta^{(b)}$ . Therefore, the deformation energy stored in the angular spring  $\beta^{(b)}$  is

$$E_{\beta}^{(b)} = \frac{1}{2} \beta^{(b)} |\Delta\phi^{(b)}|^2 = \frac{1}{2} \beta^{(b)} \left\{ e_{\gamma\alpha\beta} \varepsilon_{\beta\delta} \left[ n_{\alpha}^{(b+1)} n_{\delta}^{(b+1)} - n_{\alpha}^{(b)} n_{\delta}^{(b)} \right] \right\}. \quad (6.43)$$

The components of the effective stiffness tensor of the triangular  $\alpha - \beta$  model can be derived by summation of the deformation energies of the central interactions (6.33) and all six angular springs ((6.43)).

Ostojca-Starzewski [3], [8] arrived at the following non-zero components of the effective stiffness tensor

$$\begin{aligned} C_{1111} = C_{2222} &= \frac{1}{2\sqrt{3}} \frac{9}{4} \left( \alpha + \frac{\beta}{\ell^2} \right) = \frac{E}{1-\nu^2} \\ C_{1122} = C_{2211} &= \frac{1}{2\sqrt{3}} \left( \frac{3}{4} \alpha - \frac{9}{4} \frac{\beta}{\ell^2} \right) = \frac{E\nu}{1-\nu^2} \\ C_{1212} &= \frac{1}{2\sqrt{3}} \left( \frac{3}{4} \alpha + \frac{9}{4} \frac{\beta}{\ell^2} \right) = \frac{E}{2(1-\nu)} \end{aligned} \quad (6.44)$$

With respect to Equations (6.44), it can be shown that the isotropy condition (6.36) is satisfied, so that only two independent elastic constants remain. Based on the same

equations, the plane Poisson's ratio and the plane modulus of elasticity can be expressed as functions of the axial ( $\alpha$ ) and angular ( $\beta$ ) spring constants

$$\nu = \frac{C_{1122}}{C_{1111}} = \frac{1}{3} \left[ \frac{1 - 3\beta/(\alpha\ell^2)}{1 + \beta/(\alpha\ell^2)} \right], \quad E = \frac{\alpha}{\sqrt{3}} \left[ \frac{1 + 3\beta/(\alpha\ell^2)}{1 + \beta/(\alpha\ell^2)} \right] \quad (6.45)$$

Importantly, the two 2D parameters (6.45) are also dependent upon the model resolution defined by  $\ell$ .

Substitution of  $\beta = 0$  in Equations (6.45) recovers the plane moduli of the  $\alpha$  model defined by Equations (6.37). Complete range of definition of the plane-strain Poisson's ratio

$$-1 < \nu < 1/3$$

can be obtained from expression (6.45)<sub>1</sub> for two boundary cases:  $\beta/\alpha \rightarrow \infty$  and  $\beta/\alpha \rightarrow 0$  ( $\alpha$  model).

The plane moduli of compression and shear are defined as functions of the axial ( $\alpha$ ) and angular ( $\beta$ ) springs constants by the following expressions

$$K = \frac{\sqrt{3}}{4} \alpha, \quad \mu = \frac{\sqrt{3}}{8} \alpha \left( 1 + 3 \frac{\beta}{\alpha\ell^2} \right)$$

which demonstrates that angular springs have no effect on volume change.

#### 6.4.4 Lattices with Beam Interactions

The lattice with beam interactions (beam lattice) is the result of upgrading the  $\alpha$  model (truss lattice, spring network) by replacing structural elements capable of transmitting only axial force ( $F$ ) with beam elements that can also transmit shear forces and moments ( $Q$ ,  $M$ ; Figure 6.16). The beam lattice represents a **micro-polar continuum** with independent nodal displacement and nodal microrotations fields. As a result there are six (three) degrees of freedom per lattice node in 3D (2D, Figure 6.16c) models. The beam lattice presentation that follows is based mainly on articles of Ostoja-Starzewski [8] and Karihaloo and co-authors [89].

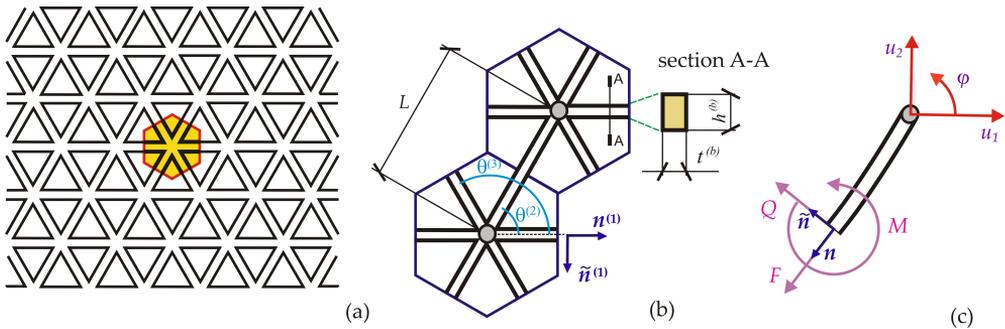


Figure 6.16: (a) Triangular beam lattice with (b) two adjacent hexagonal unit cells; (c) Degrees of freedom per 2D beam lattice node: two translational and one rotational.

### Triangular Bernoulli-Euler beam lattice

**Bernoulli-Euler beams** that transmit axial and shear forces and bending moments are commonly used in beam lattice models to simulate crack propagation and fracture (Figure 6.16). In 2D micropolar continuum ( $u_3 = 0$ ,  $\varphi_1 = \varphi_2 = 0$ ), kinematics of such beam lattice is described with three functions: a nodal displacement ( $u_1, u_2$ ), and a nodal rotation ( $\varphi_3 = \varphi$ ). The additional kinematic function  $\varphi = \varphi_3$  is completely independent of the displacement field (i.e., it is independent and different from the antisymmetric rotation  $(u_{j,i} - u_{i,j})/2$  of the classical continuum theory). Adoption of the linearity assumption of the three kinematic functions leads to the expressions for the local asymmetric strain ( $\gamma$ ) and the torsional strain (curvature,  $\kappa$ )

$$\gamma_{\alpha\beta} = u_{\beta,\alpha} + e_{\beta\alpha 3}\varphi, \quad \kappa_\delta = \varphi_{,\delta} \quad (6.46)$$

that fully describe the micropolar deformation. With regards to (6.46), recall that  $e_{\beta\alpha 3}$  stands for the permutation tensor, and the repeated Greek indices imply summation. Thus, the nodal rotation  $\varphi$  does not contribute to the normal strains (i.e., the elongation of the generic material fiber), which implies that there is no difference between the normal strains in the micropolar and the classical continuum theories ( $\gamma_{11} = \varepsilon_{11}$ ,  $\gamma_{22} = \varepsilon_{22}$ ). The “micro-polar strain” defined by  $(6.46)_1$  has the same form as the one used in Cosserat model [92].

The average normal (axial) strain in the half-beam of the unit cell is

$$\gamma^{(b)} = n_\alpha^{(b)} n_\beta^{(b)} \gamma_{\alpha\beta}, \quad (6.47)$$

which is to say that  $\gamma^{(b)}L^{(b)}$  is the average change of beam length.

The difference between the angle of rotation of the beam chord and the rotation of its end nodes is

$$\tilde{\gamma}^{(b)} = n_\alpha^{(b)} \tilde{n}_\beta^{(b)} \gamma_{\alpha\beta} = n_\alpha^{(b)} \tilde{n}_\beta^{(b)} u_{\alpha,\beta} - \varphi. \quad (6.48)$$

Where  $\mathbf{n}^{(b)}$  and  $\tilde{\mathbf{n}}^{(b)}$  designate, respectively, the unit vectors in the longitudinal and lateral directions (Figure 6.16c).

Therefore, the difference between the angles of rotation of the beam ends (corresponding to the lattice nodes) is

$$\kappa^{(b)} \equiv n_\delta^{(b)} \kappa_\delta. \quad (6.49)$$

The Bernoulli-Euler beam theory implies that in each beam the relations between forces and displacements, and moments and angles of rotation, are of the following form

$$F^{(b)} = E^{(b)} A^{(b)} \gamma^{(b)}, \quad Q^{(b)} = \frac{12E^{(b)} I^{(b)}}{(L^{(b)})^2} \tilde{\gamma}^{(b)}, \quad M^{(b)} = E^{(b)} I^{(b)} \kappa^{(b)}. \quad (6.50)$$

In Equations (6.50) the familiar relations for the area,  $A^{(b)} = t^{(b)} \cdot h^{(b)}$ , and axial moment of inertia,  $I^{(b)} = t^{(b)} [h^{(b)}]^3 / 12$ , of the beam cross-section are used (Figure 6.30b). For the triangular beam lattice with a hexagonal unit cell and the spacing of the mesh  $L^{(b)}$  (Figure 6.30), the deformation energy of the unit cell is

$$U_{cell} = \frac{1}{2} \sum_{b=1}^6 \left[ F^{(b)} \gamma^{(b)} + Q^{(b)} \tilde{\gamma}^{(b)} + M^{(b)} \kappa^{(b)} \right] \frac{L^{(b)}}{2}. \quad (6.51)$$

The deformation energy of the equivalent micropolar continuum is

$$U_{\text{continuum}} = \frac{1}{2} (\gamma_{\alpha\beta} C_{\alpha\beta\gamma\delta} \gamma_{\gamma\delta} + \kappa_{\alpha} D_{\alpha\beta} \kappa_{\beta}) V \quad (6.52)$$

where  $V = (\sqrt{3}/2)L^2t$  denotes the volume of the continuum corresponding to the unit cell (Figure 6.16b), while  $t$  is the thickness (which, in the general case, should be distinguished from the thickness of the half-beam  $t^{(b)}$  perpendicular to the lattice plane, shown in Figure 6.16).

The condition of equivalence (6.28) of deformation energies (6.51) and (6.52), leads to the following expressions

$$C_{\alpha\beta\gamma\delta} = \sum_{b=1}^6 n_{\alpha}^{(b)} n_{\gamma}^{(b)} \left[ n_{\beta}^{(b)} n_{\delta}^{(b)} R^{(b)} + \tilde{n}_{\beta}^{(b)} \tilde{n}_{\delta}^{(b)} \tilde{R}^{(b)} \right], \quad D_{\alpha\beta} = \sum_{b=1}^6 n_{\alpha}^{(b)} n_{\beta}^{(b)} S^{(b)} \quad (6.53)$$

where

$$R^{(b)} = \frac{E^{(b)}A^{(b)}}{\sqrt{3}Lt}, \quad \tilde{R}^{(b)} = \frac{12E^{(b)}I^{(b)}}{\sqrt{3}L^3t}, \quad S^{(b)} = \frac{E^{(b)}I^{(b)}}{\sqrt{3}Lt}. \quad (6.54)$$

It should be noted that, given (6.54)<sub>1</sub> and (6.54)<sub>2</sub> and the above-mentioned definitions of the cross-sectional properties, it is straightforward to show that

$$\frac{\tilde{R}^{(b)}}{R^{(b)}} = \left( \frac{h^{(b)}}{L^{(b)}} \right)^2 = \left( \bar{h}^{(b)} \right)^2 \quad (6.55)$$

where the height of the beam cross section normalized with its length (Figure 6.16),  $\bar{h}^{(b)} = h^{(b)}/L^{(b)}$ , represents a geometric parameter inverse of the slenderness of the beam element.

Referring to the properties of the unit cell illustrated in Figure 6.16b

$$\begin{aligned} \theta^{(b)} &= (b-1)\pi/3, \quad \mathbf{n}^{(b)} = \left( n_1^{(b)}, n_2^{(b)} \right) = (\cos \theta^{(b)}, \sin \theta^{(b)}), \\ \tilde{\mathbf{n}}^{(b)} &= \left( \tilde{n}_1^{(b)}, \tilde{n}_2^{(b)} \right) = (-\sin \theta^{(b)}, \cos \theta^{(b)}), \quad b = 1, 2, 3, 4, 5, 6 \end{aligned}$$

and assuming equality of beams ( $R^{(b)} = R, \dots$ ), the non-zero stiffness components are obtained from (6.53) in the following form

$$\begin{aligned} C_{1111} = C_{2222} &= \frac{3}{4}R \left( 3 + \frac{\tilde{R}}{R} \right), \quad C_{1122} = C_{2211} = \frac{3}{4}R \left( 1 - \frac{\tilde{R}}{R} \right) \\ C_{1221} = C_{2112} &= \frac{3}{4}R \left( 1 - \frac{\tilde{R}}{R} \right), \quad C_{1212} = \frac{3}{4}R \left( 1 + 3\frac{\tilde{R}}{R} \right) \\ C_{2121} &= \frac{3}{4}R \left( 1 + 3\frac{\tilde{R}}{R} \right), \quad D_1 = D_{22} = 2S. \end{aligned} \quad (6.56)$$

Corresponding Lamé coefficients are

$$\mu = \frac{3}{4}R \left( 1 + \frac{\tilde{R}}{R} \right) = \frac{\sqrt{3}}{4} \frac{t^{(b)}}{t} \bar{h} (1 + \bar{h}^2) E^{(b)}, \quad \lambda = \frac{3}{4}R \left( 1 - \frac{\tilde{R}}{R} \right) = \frac{\sqrt{3}}{4} \frac{t^{(b)}}{t} \bar{h} (1 - \bar{h}^2) E^{(b)} \quad (6.57)$$

and the plane Poisson's coefficient and the modulus of elasticity

$$\nu = \frac{1 - \tilde{R}/R}{3 + \tilde{R}/R} = \frac{1 - \bar{h}^2}{3 + \bar{h}^2}, \quad E = 6R \left( \frac{1 + \tilde{R}/R}{3 + \tilde{R}/R} \right) = 2\sqrt{3} \frac{t^{(b)}}{t} \left( \frac{1 + \bar{h}^2}{3 + \bar{h}^2} \right) E^{(b)} \quad (6.58)$$

based on well-known elasticity relations (e.g., [106]).

Given the expression (6.58)<sub>1</sub>, it is evident that for slender beams ( $h^{(b)} \ll L^{(b)} \Leftrightarrow \bar{h}^{(b)} \ll 1$ ), shear bending loses importance and the value of the plane Poisson's ratio approaches the upper limit of 1/3. Given the assumption of beam slenderness inherent in the Bernoulli-Euler formulation, the lower limit of the plane Poisson's ratio is  $\approx 0.2$  [89].

Finally, it is easy to show that in the limit case of negligible shear bending stiffness ( $\bar{h} \rightarrow 0$ ), the plane-strain parameters (6.58) are reduced to their  $\alpha$  model counterparts (6.37).

The disadvantages of the Bernoulli-Euler beam lattices are: (i) the results are sensitive to the size of the beam elements and the direction of application of the load, (ii) the material response is excessively brittle (especially if ideally-brittle behavior is used for individual beams), (iii) the beam elements in the pressure zones could overlap, and (iv) exceptional computational effort is required at the structural level. All these shortcomings can be reduced in various ways. For example, the first one can be remedied by using an irregular geometry [94]. The second, by 3D modeling, using very small unit cells [95], as well as using a nonlocal approach in calculating the deformations of beam elements [94].

### Triangular Timoshenko beam lattice

When the lattice model contains short beams it is more appropriate to use lattice elements based on **Timoshenko beam theory** that takes into account shear deformation and rotational bending effects. This formulation is presented herein in a much abbreviated form in comparison with Bernoulli-Euler beam presented in the preceding chapter. Unlike Bernoulli-Euler formulation, during the actual beam deformation, the cross sections of the beam remain neither perpendicular to the neutral line nor flat/straight (i.e., warping takes place). Timoshenko kept the assumption of a flat section, but introduced an (additional) shear-induced angular deformation, so that the cross section is no longer perpendicular to the neutral line. Thus, Timoshenko beam theory implies

$$Q^{(b)} = \frac{12E^{(b)}I^{(b)}}{(1 + \zeta_T)(L^{(b)})^3} L^{(b)} \tilde{\gamma}^{(b)}, \quad \zeta_T = \frac{12E^{(b)}I^{(b)}}{G^{(b)}A^{(b)}(L^{(b)})^2} = \frac{E^{(b)}}{G^{(b)}} \bar{h}^{(b)} \quad (6.59)$$

while the relations  $F^{(b)} - \gamma^{(b)}$  (6.50)<sub>1</sub> and  $M^{(b)} - \kappa^{(b)}$  (6.50)<sub>3</sub> remain unchanged. The dimensionless parameter (6.59)<sub>2</sub> is the key ingredient of Timoshenko beam theory. When the shear stiffness is dominant,  $\zeta_T \ll 1$ , the shear displacement is relatively small and Bernoulli-Euler beam theory is applicable. In contrast, at low shear stiffness, the shear displacement is no longer negligible and it is necessary to use Timoshenko beam theory.

From the equivalence of strain energies (6.28), identical expressions for the components of the elasticity tensor (6.56) as for the Bernoulli-Euler beam follow, with the difference that

$$\frac{\tilde{R}}{R} = \frac{12I}{L^2A} \frac{1}{1 + \zeta_T} = \left( \frac{h}{L} \right)^2 \frac{1}{1 + \zeta_T} = \frac{\bar{h}^2}{1 + \zeta_T} \quad (6.60)$$

Note that identical geometry of beams is assumed in derivation of Equation (6.60) and the subscript ( $b$ ) is, consequently, omitted from geometric parameters for brevity. By comparing expressions (6.55) and (6.60), it follows that the stiffness components and elastic constants of Timoshenko beam theory are obtained from Bernoulli-Euler beam theory when  $\bar{h}^2$  is replaced by  $\bar{h}^2/(1 + \zeta_T)$ . Therefore, the plane Poisson's ratio and the modulus of elasticity expressions

$$\nu = \frac{1 - \bar{h}^2/(1 + \zeta_T)}{3 + \bar{h}^2/(1 + \zeta_T)}, \quad E = 2\sqrt{3} \frac{t^{(b)}}{t} \bar{h} \left( \frac{1 + \bar{h}^2/(1 + \zeta_T)}{3 + \bar{h}^2/(1 + \zeta_T)} \right) E^{(b)} \quad (6.61)$$

should be used in lieu of (6.58).

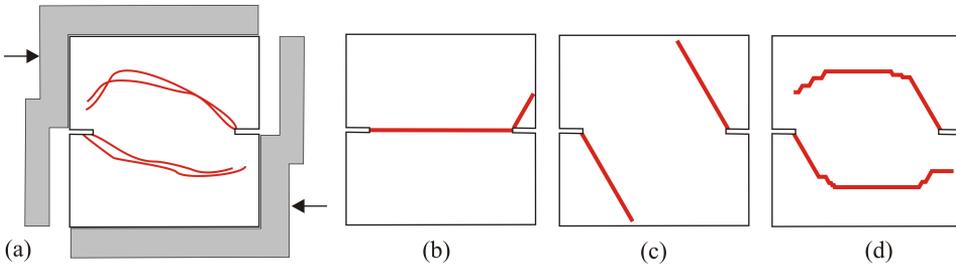
In general, Timoshenko beams are by definition more appropriate for use than Bernoulli-Euler beams when lattice elements are not very slender but rather stocky. As an example, in order to directly include the interface layers in the model (Figure 6.3b), the network resolution (i.e., the beam span) must be limited by the interface thickness. Consequently, there are practical difficulties and extremely large computational efforts. This problem has led to the development of alternative lattice models based on generalized beams [96].

#### 6.4.5 Various Aspects of Lattice Modeling

The selection of lattice elements is crucial for simulation of complex cracking. Schlangen and Garboczi [93], [94] performed a paramount, and later very influential, comparative analysis of simulation techniques using lattice modeling of heterogeneous materials with random micro structure. Experimental crack propagation patterns obtained using a concrete double-edge-notched specimen were compared with those obtained by computer simulations using lattice models with different types of interactions and spatial lattice orientations. The selected results are illustrated in Figure 6.17.

It is important to note that the geometric disorder was not used in these lattice models to emphasize the ability of a particular type of lattice element to capture the fracture pattern. With respect to Figure 6.17, it is obvious that—in the absence of geometric disorder—the beam elements are superior in the reproduction of experimentally observed complex crack patterns compared to the two truss lattices. Nevertheless, it is noticeable that the shape of the cracks even in that case (Figure 6.17d) reveals the bias inevitable in geometrically regular lattices.

Thus, another important selection for correct prediction of complex crack patterns is related to the lattice geometry, especially its regularity. The cracking directions of regular lattices are strongly predetermined (as illustrated in Figure 6.17) but it is easy to achieve the uniform deformation. On the other hand, the irregular lattices are characterized by less biased cracking patterns but, in general, do not behave homogeneously under uniform loading. Schlangen and Garboczi [93] demonstrated importance of the lattice geometric disorder for realistic simulation of crack propagation by comparing crack patterns obtained by computer simulations using four different lattices (based on a square grid, two differently oriented triangular grids, and irregular triangular grids). They also proposed an approach (based on iterative adjustment of the beam properties) to obtain an irregular lattice with the elastically uniform deformation.



**Figure 6.17:** Schematic illustration of a prediction of crack path in shear test: (a) the geometries of the experimental setup of the Nuru-Mohammed shear-load test specimen with a double-edge-notched specimen and corresponding crack propagation patterns in a concrete slab (red). Crack propagation patterns were obtained by computational simulations using a **geometrically regular** triangular lattice with: (b) central interaction (Chapter 6.4.2), (c) central-angular interaction (Chapter 6.4.3), and (d) beam interaction (Chapter 6.4.4). (Reproduced based on [93]).

As far as computer implementation procedures are concerned, lattice models simulate the process of damage and fracture by performing an analysis for a given load with the removal from the network of those bonds that meet the prescribed fracture criteria. In the case of beam lattices, the forces and moments are calculated using the appropriate beam theory. A global stiffness matrix is formed for the whole lattice, the corresponding inverse matrix is calculated which is then multiplied by the load vector to obtain the displacement vector. The heterogeneity of the material structure can be taken into account in different ways by: (i) assigning to the beams different tensile strengths, (ii) assuming a random distribution of cross-sectional dimensions and/or beam lengths, or (iii) mapping to beams different material properties (aggregates, cement matrix, interface layers,...; Figure 6.3b). These various types of **disorder** (geometrical, topological, chemical,...) could be introduced by using statistical distributions.

When lattice models are used for fracture analysis, the breaking rule of the basic one-dimensional structural element (that is, the rupture criterion on the micro- or meso-scale) must be defined in advance. Again, depending on the type of material that is the subject of modeling, several bond-removal criteria can be applied based on strength theory, energy dissipation, fracture mechanics. The simplest examples include those used for the lattices with central interactions

$$f^{(b)} = f_{cr}, \quad \varepsilon^{(b)} = \varepsilon_{cr}, \quad E^{(b)} = E_{cr}, \quad (6.62)$$

where the critical parameter—as indicated by subscript (*cr*)—refers to the bond axial force, elongation, or elastic strain energy, respectively.

Naturally, more complex **bond rupture criteria** are commonly used for beam lattices, characterized by more complex stress states. This complexity necessitates the failure definition in terms of the failure envelopes that take into account contributions of all relevant deformation types. For example, van Mier and co-workers [65], [66] established

the following criteria for removal of beam elements

$$\sigma_{eff}^{(b)} = \frac{F^{(b)}}{A^{(b)}} \pm \zeta \frac{(|M_i^{(b)}|, |M_j^{(b)}|)_{max}}{W^{(b)}} = \sigma_{cr}, \quad (6.63)$$

$$\frac{F^{(b)}/A^{(b)}}{\sigma_{cr}} + \frac{|Q^{(b)}|/A^{(b)}}{\sigma_{cr}} > 1$$

where  $W^{(b)}$  and  $\zeta$  designate the section modulus of the beam, and the fitting parameter; while,  $\sigma_{cr}$  and  $\tau_{cr}$  are the tensile strength and the shear strength, respectively.

With beam lattices, it is possible to reproduce very complex patterns of damage. They are able to simulate the nucleation and propagation of microcracks, crack branching, crack curvature, bridging and coalescence of cracks etc., which results in a complete picture of macroscopic damage and fracture. The model also makes it possible to “capture” the effect of sample size. The advantages of this approach are simplicity and direct insight into the fracture process at the level of the microstructure.

The recent development of lattice models and their “peculiarities” are reviewed recently by Nikolić and co-authors [97].

## 6.5 Discrete Element Methods

### 6.5.1 Basic idea of DEM

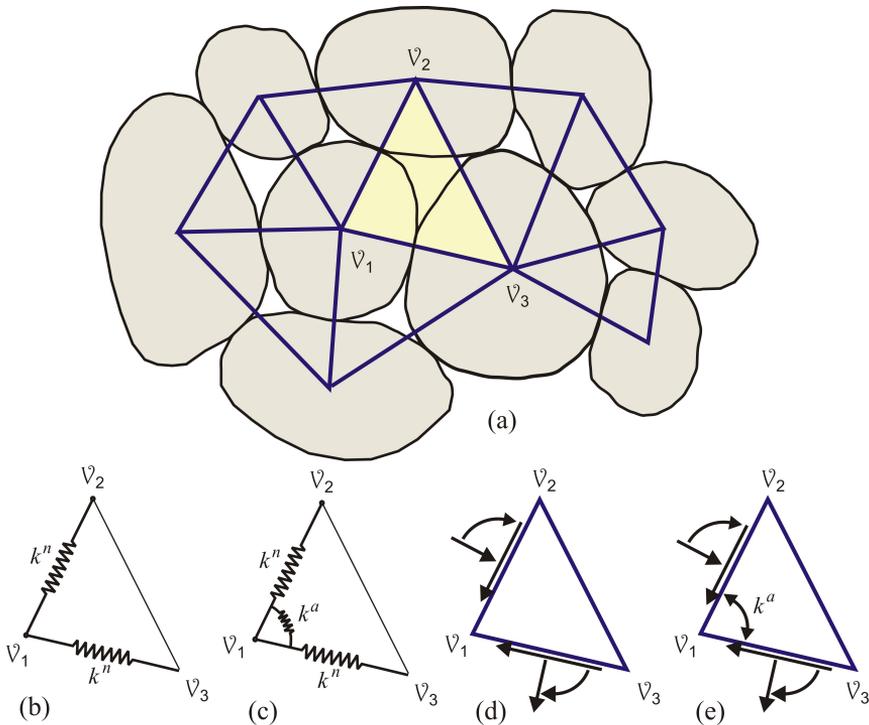
DEM is a Lagrangian technique of computational simulations in which the computational model is made of discrete (rigid or deformable) elements of the same (or even higher) Euclidean dimensionality as the analyzed problem. These discrete elements of different shapes interact through contact algorithms (smooth or rough contacts) [98]. The material is, therefore, modeled by a set of **Voronoi cells** (mimicking grains, granules, particles, aggregates) representative from the point of view of heterogeneity of material texture, whose meso-scale dynamic interactions determine its macroscopic behavior. Obviously, the contact algorithms are at the physical core of a group of computational techniques custom-made to solve problems characterized by extremely large discontinuities in the internal structure or geometry of materials or both [99]-[101].

DEM was introduced by Cundall [99] to analyze the intermittent progressive fracture of rocky slopes, to be later applied to the analysis of granular assemblies by Cundall and Strack [100]. Although DEM is now widely used for modeling different classes of materials (such as geomaterials, biomaterials, composites), the most natural applications are for simulations of deformation or flow of material systems that have the same topology as the representative group of discrete elements (e.g, Figure 6.1a,c-g). Cundall and Hart [102] summarized succinctly the methodological approach by defining DEM as a method that allows finite translations and rotations of discrete bodies<sup>3</sup> (rigid, solid, breakable), including complete separation of their mutual contacts, as well as automatic recognition of newly established contacts during simulation. The mechanical behavior of the whole

<sup>3</sup>Throughout this chapter, the term “particle” (grain, as well) is used, when convenient, for the DEM building block – the discrete element; it should not be confused with particles used occasionally in relation with other CMD models in this introduction, which are material points.

conglomerate is described on the basis of the motion of these individual elements and specification of the constitutive rules or contact forces among them. Like other methods of CMD, DEM provides a detailed temporal evolution of the system by solving Newton's equations of motion of individual discrete elements (6.1), including the complex damage mechanisms that naturally arise from such simulations. This chapter outlines only the traditional DEM; the review of more advanced DEM models is beyond the scope of this short introduction. The same goes for advanced topics and specifics of DEM technique such as packing and grain shape, flow laws, capillary effects, high deformation loadings, which are available, for example, in the review paper by Donze and co-authors [10] and references cited therein.

It is common to classify DEM models based on the load transfer mechanisms illustrated in Figure 6.18.



**Figure 6.18:** (a) A group of particles or grains (with a cluster of three highlighted) with prominent interacting lines forming an associated network. Based on the load transfer mechanism used in the model, DEM are divided into models with: (b) Central interactions (these models represent the generalization of the  $\alpha$  model of Chapter 6.4.2); (c) Central and angular interactions (generalization of the  $\alpha - \beta$  model of Chapter 6.4.3); (d) Central, shear and bending interactions (generalization of the beam lattices described in Chapter 6.4.4; this is a typical DEM, which can be called a “local inhomogeneous micropolar continuum” [3]); and (e) Central, shear, bending and angular interactions. (Adopted from [3].)

The DEM problem-solving methodology is based on the MD formalism, which includes explicit finite difference schemes in which the computational cycle involves

applying Newton's second law of motion (6.1) to each discrete element using a specified load-displacement rule to all contacts to determine new element positions. During the global conglomerate (sample) evolution, new contacts may appear (between particles that have not previously touched) and existing contacts may disappear. Therefore, the global stiffness matrix of the whole group of discrete elements must be constantly updated, from cycle to cycle. For non-cohesive (loose) materials and particle systems, there is another reason why it is necessary to update the global stiffness matrix: namely, the interactions among discrete elements, both in the normal and shear directions, are not necessarily linear, which means that the stiffnesses  $k^n$  and  $k^t$ , which define these contacts (e.g., Figure 6.19), must be recalculated continuously [103]-[105].

Unlike computational methods of continuum mechanics, such as FEM, in DEM the role of primary variables is played by forces and displacements. Accordingly, it is necessary to develop and apply methods for describing the continuum parameters (stresses and strains) based on these forces and displacements (i.e., mesoscopic parameters of the state of the individual elements that make up the assembly). This process is called **homogenization**. The starting point in this procedure is to define a **representative volume element**, which serves as an averaging volume for calculating mean values of macroscopic parameters (e.g., [106], [107]). Thus, homogenization and constitutive modeling techniques make it possible to take into account the micro-/meso-structure within the application of continuum methods, but only indirectly in terms of the "mean field". The DEM model parameters are typically adjusted using experimentally observed behavior and a large number of such parameters are necessary to reproduce complex phenomena. The homogenization is, very often, a demanding job because the parameters that control geometric properties and constitutive behavior do not always have a clear physical meaning and can also show complex interdependencies [10]. Despite the open questions highlighted throughout this chapter, DEM modeling (especially the mechanical behavior of geomaterials) is on the rise not only in research but also in geotechnical engineering (e.g., [108]). The main cause of this DEM popularity is the ability of natural reproduction of localization (a phenomenon ubiquitous in quasi-brittle materials with random texture) that is difficult to capture objectively by computational methods of continuum mechanics based on network discretization (e.g., FEM).

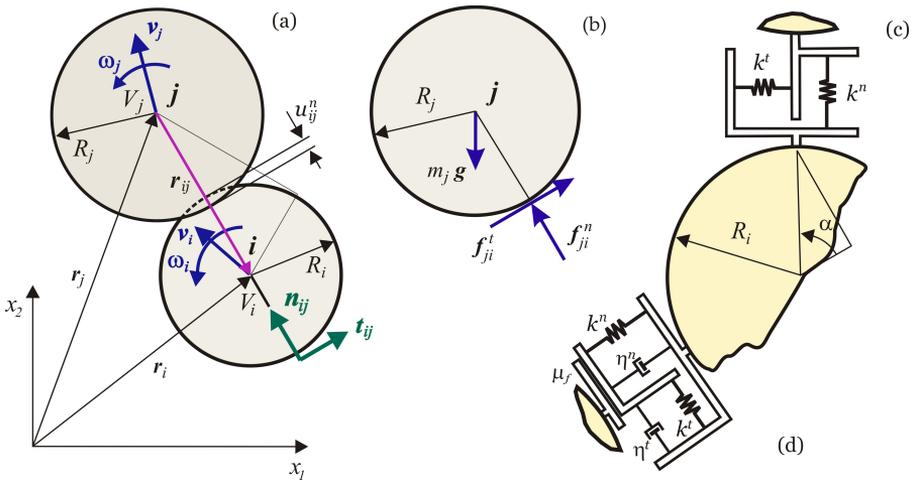
### 6.5.2 Contact Algorithms

It is obvious from the DEM basics outlined above that contact algorithms are the essential DEM feature. Most often, realistic and detailed modeling of particle contacts is not only too complex but also unnecessary. Therefore, in the following considerations, the force of the interaction of the particles  $i$  and  $j$  is related to their overlap illustrated in Figure 6.19a. The Hertz theory of contact mechanics [109] defines the basic law of elastic contact of two spheres of radius  $R_i$  and  $R_j$  by a nonlinear relation

$$(f_{ij}^n)_{ela} = \frac{4}{3} \left( \frac{1 - \nu_i^2}{E_i} + \frac{1 - \nu_j^2}{E_j} \right)^{-1} \sqrt{R_{ij}} (u_{ij}^n)^{3/2} \quad (6.64)$$

between the normal contact force,  $f_{ij}^n$ , and the maximum overlap,  $u_{ij}$ . In Equation (6.64),  $(E, \nu)$  are pairs of elastic constants of two materials in contact and  $R_{ij}^{-1} = R_i^{-1} + R_j^{-1}$  the

equivalent radius. The force on particle  $j$  exerted by particle  $i$  is obtained by applying Newton's third law,  $\mathbf{f}_{ji}^n = -\mathbf{f}_{ij}^n$ . Equation (6.64) is derived under the following assumptions: (i) the spherical elements are ideally smooth, (ii) the materials are elastic and isotropic, (iii) the shear component of the elastic contact force has no effect on the normal force, and (iv) the overlap is small relative to the size of spherical elements. This law is essential for the simulation of certain phenomena in granular materials, such as the propagation of elastic waves. However, it should be remembered that the **Hertzian contact** model is adequate only for the elastic contact (i.e., when the forces do not exceed the yield strength anywhere in the contact zone). For more complex cases, contact models based on viscoelasticity and elastoplasticity have been developed ([110] and references cited therein).



**Figure 6.19:** Schematic representation of (dry) contact of two circular elements according to the classical theory. (a) Definition of meso-parameters; (b) contact forces; (c) basic interaction model - bonding contact without damping; and (d) a more complex contact model involving damping and friction.

At small deformations, compact geomaterials (e.g., sand in Figure 6.1c) are characterized by a linear elastic response. The force with which two spherical elements of such material act on each other can be decomposed into the elastic normal force,  $\mathbf{f}_{ela}^n$ , and the incremental shear force,  $\mathbf{f}^t$ , which are (in the classical interpretation) related to the relative normal and the incremental tangential displacements, respectively, through the coefficients of normal and secant shear stiffness ( $k^n$  and  $k^t$ )

$$(\mathbf{f}_{ij}^n)_{ela} = -k^n \mathbf{u}_{ij}^n, \quad \mathbf{f}_{ij}^t = \{\mathbf{f}_{ij}^t\}_{updated} - k^t \Delta \mathbf{u}_{ij}^t.$$

The elastic response is completely defined by this pair of constants ( $k^n, k^t$ ). The incremental shear force should reset to zero,  $\{\mathbf{f}_{ij}^t\}_{updated} = \mathbf{0}$ , whenever the elements can slide relative to each other, which happens when the **Mohr-Coulomb type limit** is reached

$$|f_{ij}^t| = f_{coh}^t + \mu_f f_{ij}^n. \quad (6.65)$$

The limit value (6.65) is defined by the local values of the contact friction coefficient,  $\mu_f$ , and the cohesion,  $f_{coh}^t$  (which is by definition equal to zero for non-cohesive materials).

It cannot be overemphasized that DEM is essentially defined by contact interactions. In addition to the above simple interactions of the discrete elements in contact, more complex contact relationships should be mentioned that take into account subtle details of bonds such as: rotational stiffness (corresponding to rolling stiffness in 2D [66]), capillary cohesion [111], solid cohesion [112], plasticity [113], plasticity with temperature and relaxation [114]. Finally, the general criteria for DEM contact failure are discussed at length by Ibrahimbegovic and Delaplace [115], Tavarez and Plesha [116], Sheng and co-authors [117].

### 6.5.3 DEM Modeling of Particulate Systems

Particulate systems, as used throughout this DEM overview, include not only non-cohesive (loose) materials but also systems of distinct objects in general (e.g., various industrial transport processes involving flow of "particles" representing a wide variety of objects). The emphasis herein is on the former. Their overall behavior can be described as a contact problem of a large number of bodies, making them ideal for DEM. Therefore, DEM has been used extensively to study the deformation, transport processes, and flow of these **non-cohesive systems** as seen from a number of references such as [66], [118]-[120]. The corresponding numerical techniques are based on the trailblazing work of Cundall and Strack [100]. Since loose materials are large conglomerates of particles, this model, in the absence of adhesion, was based on the "primordial" properties of these discrete elements: their shapes, sizes, and interactions. Discrete elements have two forms of movement, translational and rotational, or three (six) degrees of freedom per element in the case of 2D (3D) problems. Inherently discrete, DEM models represent the non-cohesive system as a group of interacting distinct objects, so computational implementation techniques are based on alternating transitions from the application of Newton's second law of motion (6.1) and the contact force-displacement laws (e.g., Figure 6.19) at every single contact.

Thus, the three main aspects of the particulate system dynamics are:

- (i) discrete element shape and size distribution (physical parameters),
- (ii) contact behavior of discrete elements (mechanical parameters; e.g., coefficient of contact friction, contact stiffnesses, contact tensile strength), and
- (iii) numerical techniques for solving systems of equations of motion (6.1).

The existence or non-existence of the ability of a material to carry tensile loading represents the essential difference between cohesive (adhesive, solid) and non-cohesive (loose, particulate, granular, fluid) materials. The slip aspect of the DEM model takes into account the limited shear resistance—defined by Coulomb's law of friction (6.65) - that the contact provides before sliding.

As particulate systems evolve, collision, sliding, and rolling contacts give rise to forces and moments (Figure 6.19) that the DEM tends to calculate in order to determine new particle positions. The discrete elements - the basic building blocks of the DEM model, can be randomly generalized geometric objects whose size distribution (log-normal is a frequent choice) reflects the inherent heterogeneity of the system. The circular/spherical elements are the simplest option. They are fully described by only one parameter—radius—that defines both their geometry and the one and only type of contact that can be easily observed (Figure 6.19). Accordingly, circles and spheres are often adopted for their simplicity. (For

the same reason - to keep the formalism as simple as possible, the basic equations are presented only for circular geometry in this introduction.) However, this computational convenience comes at a price. First of all, the circular/spherical shape of discrete elements significantly underestimates the rolling resistance among them. Second, they cannot reproduce more complex configurational rearrangements of particles (such as the particle interlock), which typically occur when discrete elements are of more complex shape. These two computational aspects result in an inherent underestimation of macro-strength. An additional artifact introduced by the circular/spherical particle geometry is the numerical porosity, which should not be confused with actual porosity.

Moreover, the discrete elements are often modeled as rigid, but a certain overlap between them (as indicated in Figure 6.19a) is allowed to model the occurrence of relative displacement and localized contact deformation (soft or **smooth contact**). The contact dynamics methods based on “non-smooth” formulations, which exclude the possibility of the particle overlap, are not addressed in this introduction; the interested reader is referred to the review paper of Donze and co-authors [10] and the references cited therein. The rigidity assumption is reasonable when the movements along the interfacial surface represent the largest part of the deformation in the assembly of discrete elements, which is typical for loose materials (such as dry sand Figure 6.1c) and industrial transport processes and flow phenomena. Whatever model is adopted or developed for a certain problem, it will naturally be based on a greater or lesser simplification of the actual physical processes on the meso-scale which is not only inevitable but also desirable, given that many details of the meso-scale contact do not have to be significant for the macro-scale response of the system as a whole. Research challenges include not only realistic quantitative simulations of large particulate systems, with the ability to predict responses and their experimental validation, but also the transition from the meso-scale contact properties to the macroscopic properties of materials. This meso-macro transition should make it possible to understand the collective behavior of the large conglomerate of discrete elements as a function of their contact properties [110].

A typical example of a non-cohesive system are dry granular materials (e.g., dry sands) mentioned above. They are characterized by the dominance of non-cohesive interparticle actions of short range: elastic or inelastic contact forces and contact friction between touching grains. The simplest **rheological model** of such contact interactions is presented in Figure 6.19c with one spring in the normal and tangential directions at the point of contact. According to the more complex approach of Xiang et al. [121], any contact between particles can be rheologically represented by a Kelvin spring-damper element in the normal direction and a spring-damper-slider element in the tangent direction (Figure 6.19d). The contact of the particles in the normal direction is ideally elastic in the case of mutual pressure while the tensile strength of non-cohesive materials is by definition equal to zero. Thus, the generalized contact behavior of particles of a simple central-angular type of interaction (Figures 6.16 and 6.18d represent two isomorphic interaction models) takes into account: normal interactions, shear interactions, and slip. In the general case, the forces acting on the particle  $j$  include: the gravitational force ( $m_j \mathbf{g}$ ), and normal ( $\mathbf{f}_{ji}^n$ ) and shear ( $\mathbf{f}_{ji}^t$ ) components of the elastic contact force between elements  $i$  and  $j$  (Figure 6.19b). Accordingly, taking into account the basic law of dynamics (6.1) and 2D geometry,

the translational and rotational motion of the particle is defined as follows

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \mathbf{g} + \sum_{j=1}^{N_i} (\mathbf{f}_{ij}^n + \mathbf{f}_{ij}^t), \quad I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_{j=1}^{N_i} (\mathbf{T}_{ij}^t + \mathbf{T}_{ij}^r) \quad (6.66)$$

where  $I_i$  designates the moment of inertia of particle  $i$ , while the translational ( $\mathbf{v}$ ) and rotational (angular) ( $\boldsymbol{\omega}$ ) particle velocities are indicated on Figure 6.19a.  $\mathbf{T}_{ij}^t$  is the torque of the tangential component of the contact force, while  $\mathbf{T}_{ij}^r$  is the torque of the rolling friction force.

In the case of multiple interactions, the forces and torques on each contact points are evaluated and added to calculate the resultant action on the discrete element. When calculating the contact forces, the contact between the particles is modeled with a pair of linear rheological models of the spring-damper-slider type [100] in both tangential and normal directions (Figure 6.19). Contact force vector  $\mathbf{f}_{ij}$ , which represents the action of a discrete element  $j$  on its neighbor  $i$ , can be decomposed into tangential (shear) and normal directions

$$\mathbf{f}_{ij} = \mathbf{f}_{ij}^n + \mathbf{f}_{ij}^t \quad \begin{cases} \mathbf{f}_{ij}^n &= - \left[ k^n \mathbf{u}_{ij}^n + \eta^n (\mathbf{v}_{ij} \cdot \mathbf{n}_{ij}) \mathbf{n}_{ij} \right], \\ \mathbf{f}_{ij}^t &= \min \{ k^t \mathbf{u}_{ij}^t - \eta^t \mathbf{v}_{ij}^t, \mu_f |\mathbf{f}_{ij}^n| \mathbf{t}_{ij} \} \end{cases} \quad (6.67)$$

wherein both components of the force include dissipative terms,  $\eta$ . The detailed discussion of the this model is presented by Xiang and co-authors [121]. Obviously, Equation (6.67) with  $\mu_f = 0$  and  $\eta^n = \eta^t = 0$  (no energy dissipation by friction and **damping**) corresponds to the simpler case of Figure 6.19c.

Alonso-Marroquin and Herrmann [122] give an almost identical representation of the DEM methodology, with the discrete elements being in the form of convex polygons. For tetrahedra, one can consult, for example, Munjiza's book [11], and for clusters [123]. Naturally, all grain shapes more complex than circular and spherical ones require much more complex algorithms, for both contact detection and definition of appropriate interactions [122], [124], [125], which will not be subject of this elementary review. However, it should be noted that complex grain shapes are more demanding in terms of computer memory and processing time, which may reduce number of grains that can be used in modeling.

The model parameters (e.g., those figuring in Equation (6.67) and Figure 6.19d) in DEM infancy were typically determined by an ad hoc procedure of calibrating the results of numerical simulations of standard laboratory tests with the corresponding experimental results (e.g., [126]). However, the parameters determined in this way depend on the size of the typical discrete element. Tavares and Plesha [116] systematically approached the determination of meso-parameters of the model by deriving expressions for elastic stiffness coefficients

$$k^n = \frac{Et}{\sqrt{3}(1-\nu)}, \quad k^t = \frac{1-3\nu}{1+\nu} k^n \quad (6.68)$$

as a function of unit cell thickness  $t$ , and the plane elastic coefficients  $E = E^{(2D)}$  and  $\nu = \nu^{(2D)}$ . It should be noted that Equations (6.68) are derived for a densely-packed, ideal triangular 2D lattice using displacement equivalence conditions. The expressions (6.68) also apply to the corresponding irregular lattice obtained by the DEM cluster consolidation process [116]. Also, it should be noted that the condition of non-negativity of

the tangential stiffness coefficient (6.68)<sub>2</sub> defines the upper limit for the value of Poisson's ratio  $\nu^{(2D)} = 1/3$ , which implies  $\nu_{\max} = 1/3$  for the plane stress and  $\nu_{\max} = 1/4$  the plane strain. Similar expressions for determining the meso-parameters of the DEM model are available in the literature for different spatial arrangements of particles. For example, Masuya et al. [127] and Potyondy and Cundall [128] derived  $k^n = Et$ , which corresponds to a simple cubic lattice; while Wang and Mora [129] obtained

$$k^n = \frac{E^{(3D)}}{\sqrt{2}(1 - 2\nu^{(3D)})}, \quad k^t = \frac{1 - 3\nu^{(3D)}}{1 + \nu^{(3D)}} k^n \quad (6.69)$$

for a face-centered cubic lattice.

Equations (6.66) are usually solved by different finite-difference schemes. The Verlet and Störmer algorithms, summarized in Chapter 6.2.3, are often used for this purpose, and the force-displacement law (6.67) and Newton's second law of motion (6.1) are used in each computational cycle. In order to obtain the correct motion of each discrete element by integration, the time step must be chosen carefully as already discussed in Chapter 6.2. By analogy between the contact (with stiffness  $k^n$ ) and the oscillating material point  $m$ , it can be shown that the time step must be chosen as a small part (usually the tenth or twentieth) of the half-period of oscillation [10]. Numerical stability can, for example, be improved by using local, contact dissipative damping as  $(\eta^n, \eta^t)$  in Equations (6.67).

Finally, the macroscopic constitutive laws of continuum mechanics connect the stress tensor and the specific deformation tensor, while the meso-scale contact constitutive relations connect the contact force with the relative displacement at the points of contact of the particles. Kruyt and Rothenburg [130] derived micromechanical expressions for stress tensors and relative deformations as a function of microscopic contact parameters. The same authors also developed statistical theories of elastic modules for plane groups of particles [131].

### 6.5.4 DEM Modeling of Solid Materials

The defining property of solid (cohesive) materials is the ability to transfer tensile force between connected particles. Thus, the Cundall's original concept [99], [132], developed for blocky rock systems, has been extended to take into account the interface tensile strength [81], [133], [134]. This DEM adaptation for solid materials was achieved usually by adding a bond at the point of contact of two discrete elements. This bond mimics the presence of a cement matrix attached to the contacting particles, which is able to impart cohesion [135]. This approach has been used to model a wide variety of classes of heterogeneous cohesive materials such as sedimentary rocks, concrete, ceramics, grouted soils, solid rocket propellants, explosives, biomaterials. All these materials, in principle, can be represented by the simple model shown in Chapter 6.5.3 (Figure 6.19) with an important proviso that in the case of cohesive (adhesive) contact, the spring in the normal direction provides resistance to both compressive and tensile loads. If, during the deformation process, the bonded contact between two discrete elements is broken (according to some prescribed bond-rupture criterion), the contact becomes purely compressive and frictional (as illustrated in Figure 6.19), if it survives at all (two grains could be separated instead of pressing against each other). Regarding the simple central and angular interaction (presented in Figure 6.18c and generalized in Chapter 6.4.3) as a combination of a viscoelastic

contact strength model (represented by the Kelvin element) and a slip model, the only difference in non-cohesive and cohesive applications is the ability transmission of the tensile force in the normal direction.

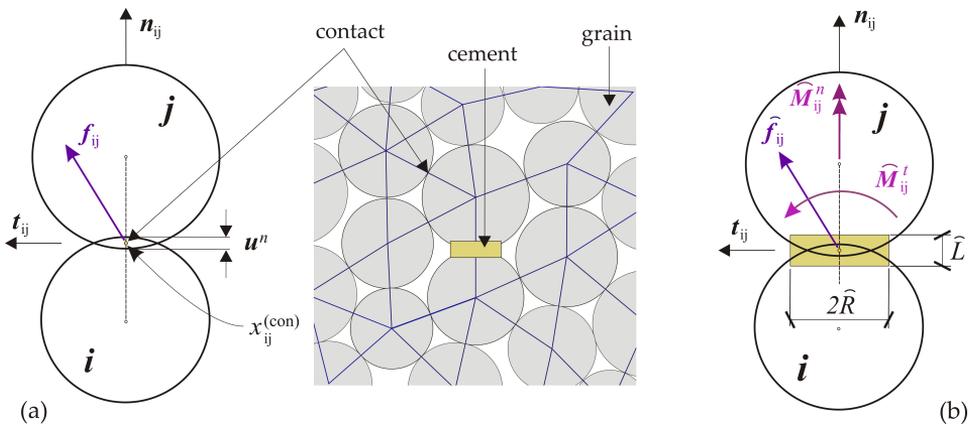
As noted above, DEM models allow the particles of the cohesive material to be interconnected but also separated, if the bond-rupture criterion is met. In the case of the action of an external load such that either the tensile strength or the limit deformation or the fracture energy are exceeded, the bonds between the particles are separated and a crack is formed on the scale of the model (meso, macro). Therefore, damage modes and their interactions naturally arise from the process of gradual particle separation. The DEM approach to discretization of the computational domain is the most pronounced advantage over the continuum-based methods of computational mechanics, since common problems (such as dynamic composite response, crack singularity, crack formulation criterion itself) can be avoided due to naturally discontinuous and random representation of material meso-structure [136].

### Application of DEM for rocks

Rock mechanics is the discipline from which DEM originated. The basic idea is to reproduce the quasi-brittle behavior of rocks by simulating the nucleation, growth, branching and merging of local cracks. Although rocks may not look like granular materials at first glance, the main features of many types of rocks and, especially, rock massifs (Figures 6.2h,i) are the pre-existing damage and the high degree of heterogeneity and discontinuity of their structure (at various spatial scales). That is the reason why rock massifs can be considered as conglomerates of discrete blocks interconnected by different models of cohesive forces (“blocky rock systems” [99]). Therefore, the mechanical behavior of the whole jointed-rock assembly evolves from the collective contribution of these discrete blocks during loading. Accordingly, the separation of two discrete units mimics the elementary meso-damage event, which represents the basic building-block of complex damage-evolution phenomena. Detailed reviews by Jing [137] and Jing and Stephansson [98] include techniques, advances, problems, and then predictions of future directions of development in computational rock modeling.

In general, the discrete elements may represent separate rock blocks of (up to) tone and meter levels that, in 2D-DEM rock simulations, can be modeled with randomly generated circles, ellipses, or convex polygons interconnected by introduction of a specific bond into the contact area. The shapes and methods of packing of discrete elements have far-reaching effects on the distribution and intensity of interaction forces. The bond strengths may be allowed to vary from contact to contact, which may represent another source of heterogeneity in the simulated material. The very influential explicit DEM method in rock mechanics is Cundall’s method of “distinct” elements [132], [138] with quadrilateral / prismatic blocks developed in the computer programs UDEC and 3DEC (Itasca<sup>TM</sup> Consulting Group, [www.itascacg.com](http://www.itascacg.com)). The focus of this brief introduction to DEM implementation for rocks will be on a simpler DEM – the **bonded-particle model** (also, often called the parallel-bond model) [128]. This model is based on circular / spherical discrete elements as illustrated in Figure 6.20. This approach to modeling has been generalized by Potyondy [139] and developed over the years through the commercial packages PFC<sup>2D</sup> and PFC<sup>3D</sup> (Itasca<sup>TM</sup> Consulting Group).

Potyondy and Cundall [128] devised the bonded-particle model to simulate rock massifs and other heterogeneous, brittle systems of intermittent sub-structure. In the original variant, the model consisted of a densely packed group of rigid circular/spherical grains of different sizes interconnected at contact points by additional parallel bonds that represent the cohesive action of the cement (Figure 6.20). The rigid discrete elements interact only through soft contacts (i.e., small overlaps are allowed) that possess limited normal and shear stiffness. The model is fully dynamic. It is able to describe complex phenomena of rock damage evolution such as nucleation, growth, branching and merging of microcracks resulting in the damage-induced anisotropy, hysteresis, dilatation, postcritical softening, strength-increasing with lateral confinement. At the beginning of 2000s when this modeling approach commenced, the DEM modeling of discontinuous media was still in its infancy compared to the mechanics of the continuum. Therefore, the authors paid great attention to the systematic development of an appropriate modeling methodology that included not only careful virtual experimentation and qualitative comparison of results with physically observed mechanisms on micro- and macro-scale but also quantitative comparison with experimentally measured properties.



**Figure 6.20:** Bonded-particle model for simulation of heterogeneous materials composed of cemented grains. Schematics of (a) the behavior of grains in contact (the non-cohesive bond), and (b) cement behavior (the parallel bond). If the cement (which provides the parallel bond) is not present, then only the grain behavior remains and the slip model with rolling (outlined in Chapter 6.5.3) is recovered. In general, the model is defined by the density of the particles, their shape and the size distribution, their packing, and the meso-properties of the grains and the cement. Potyondy and Cundall [128] used circular / spherical elements corresponding to the PFC<sup>2D</sup> / PFC<sup>3D</sup> programs (Itasca Consulting Group). Note the striking similarity with the epoxy-cemented glass beads of Figure 6.1a.

It cannot be overemphasized that, unlike some other computational methods in CMD, the term "particle" in this context means a discrete element that occupies a finite part of space (the term "grain" will be also used interchangeably). The radii of the circular discrete elements (particles, grains) are drawn from a uniform distribution bounded by  $R_{\min}$  and  $R_{\max}$  and dense packaging was obtained by following the appropriate material generation procedure. The rigid particles can independently translate and rotate and interact via soft

contacts (Figure 6.20). They can act on each other exclusively through contacts which are, having in mind their simple shape, always uniquely defined. The grain overlaps are assumed to be small compared to their size to ensure that the contacts occur “at the point”. The set of meso-properties consists of the parameters of stiffness and strength of both elements and bonds. The bonds are of finite stiffness and limited strength. The force-displacement dependence in each contact relates the relative displacements of the discrete elements with the forces and moments acting on each of them.

Dynamic response of the system is calculated using a finite difference algorithm comprehensively presented by Cundall and Hart [102]. The DEM simulation technique is based on the assumption that the time step is small enough so that, in one calculation cycle, the perturbations cannot extend beyond the first neighbors of each grain. Potyondy and Cundall [128] discussed in detail the advantages of an explicit numerical scheme. Since DEM is a fully dynamic formulation, attenuation can be introduced as needed to dissipate kinetic energy. This damping mimics microscopic dissipative processes in real materials, such as the internal friction or wave scattering.

Figure 6.20 illustrates the way in which the bonded-particle model simulates the mechanical behavior of a group of circular grains connected by parallel bonds. The total force and moment acting in each contact consist of the contact force  $\mathbf{f}_{ij}$ , which is the result of the overlap of the particles (Figure 6.20a) and represents the grain behavior (Equation (6.67) with or without damping), and the force and moment,  $\hat{\mathbf{f}}_{ij}$  and  $\hat{\mathbf{M}}_{ij}$  which are transmitted by the parallel bond and represent the cement behavior (Figure 6.20b). These quantities contribute to the resultant force and moment acting on both circular elements (by virtue of Newton’s third law of motion) involved in the contact and represent the input data for computational integration (Newton’s second law for a dynamic system (6.1)) using an explicit finite difference scheme to obtain grain trajectories.

The constitutive rule of the contacting grains is described by the same non-cohesive interaction with friction (Chapter 6.5.3 for the case without contact damping) defined with the normal and shear stiffness,  $k^n$  and  $k^t$ , and the friction coefficient,  $\mu_f$ . This contact is established as soon as the two grains overlap. The contact stiffnesses of the bond thus established (in the directions normal to, and in, the contact plane, designated respectively by superscripts  $n$  and  $t$ ) are determined by the serial connection (6.40). The overlap, although physically impermissible, mimics, in a sense, the local deformation of the grains (especially when the contact surfaces are not smooth but rough). The contact force vector of each bond can be decomposed into a normal and a shear component as already shown by (6.67)<sub>1</sub>. The contact behavior of the circular particle is already discussed in Chapter 6.5.3: if  $u_{ij}^n \leq 0$  there is a gap (note the sign convention), and the normal and shear forces are equal to zero by definition; if  $u_{ij}^n > 0$ , there is an overlap, and sliding is defined using the Mohr-Coulomb type limit (6.65). In doing so, in contrast to the normal force (which is at any time proportional to the size of the overlap with secant stiffness,  $k^n$ , as the coefficient of proportionality), the shear force is calculated in an incremental manner: after establishing contact,  $f^t$  is initialized to zero; from each subsequent increase in the relative displacement of the particles in the direction of the tangent,  $u^t$ , there is an increase in shear force,  $\Delta f^t = -k^t \Delta u^t$ , where  $k^t$  tangential stiffness (as opposed to secant from which it is distinguished by the subscript). Contact displacements are calculated in each calculation cycle on the basis of the contact velocity which depends on the translational and angular

velocities of the grains; see the original reference [128] for details.

The behavior of the parallel bond mimicking cement behavior is defined by five parameters: normal and shear stiffness coefficient ( $\hat{k}^n, \hat{k}^t$ ), the tensile and shear strength ( $\hat{\sigma}_m, \hat{\tau}_m$ ), and multiplier  $\hat{\lambda}$  which defines the diameter of the parallel bond depending on the diameter of the circular elements in contact. The interaction of the grains is represented by the total force and moment,  $\hat{\mathbf{f}}_{ij}$  and  $\hat{\mathbf{M}}_{ij}$ , transmitted by the parallel bond (Figure 6.20b). The force and moment can be projected in the directions of normal and tangent in the following way

$$\hat{\mathbf{f}}_{ij} = \hat{f}_{ij}^n \mathbf{n}_{ij} + \hat{f}_{ij}^t \mathbf{t}_{ij}, \quad \hat{\mathbf{M}}_{ij} = \hat{M}_{ij}^n \mathbf{n}_{ij} + \hat{M}_{ij}^t \mathbf{t}_{ij} \quad (6.70)$$

where  $\hat{f}_{ij}^n, \hat{f}_{ij}^t$  denote the normal and shear forces, and  $\hat{M}_{ij}^n, \hat{M}_{ij}^t$  the twisting and bending (rolling) moments (naturally, there are two of the later in 3D), respectively. (Note that  $\hat{M}_{ij}^n \equiv 0$  for 2D models and the bending moment acts in the out-of-plane direction.) At the initialization of the parallel bond,  $\hat{f}_{ij}$  and  $\hat{M}_{ij}$  are set to zero; each subsequent relative increment of translation and angle of rotation ( $\Delta u_{ij}^n, \Delta u_{ij}^t, \Delta \theta_{ij} = (\omega_j - \omega_i) \Delta t$ ) lead to the corresponding increase in the components of force and moment (Figure 6.20)

$$\Delta \hat{f}^n = \hat{k}^n A \Delta u^n, \quad \Delta \hat{f}^t = -\hat{k}^t A \Delta u^t, \quad \Delta \hat{M}^n = -\hat{k}^t J \Delta \theta^n, \quad \Delta \hat{M}^t = -\hat{k}^n I \Delta \theta^t \quad (6.71)$$

which are added to the current values. The geometrical properties of the parallel-bond cross section in (6.71) — area ( $A$ ), axial ( $I$ ) and polar ( $J$ ) moment of inertia — are defined by well-known expressions in terms of the parallel-bond radius,  $\hat{R}$ , for the 2D ( $PF C^{2D}$ ) and 3D ( $PF C^{3D}$ ) models.

The maximum normal and shear stresses acting on the circumferences of the parallel bond are calculated using the elementary beam theory

$$\hat{\sigma}^{\max} = -\frac{\hat{f}^n}{A} + \frac{|\hat{M}^t|}{I} \hat{R}, \quad \hat{\tau}^{\max} = -\frac{|\hat{f}^t|}{A} + \frac{|\hat{M}^n|}{J} \hat{R} \quad (6.72)$$

If the value of the maximum normal stress (6.72)<sub>1</sub> exceeds the tensile strength ( $\hat{\sigma}^{\max} \geq \hat{\sigma}_m$ ) or the maximum shear stress (6.72)<sub>2</sub> exceeds shear strength ( $\hat{\tau}^{\max} \geq \hat{\tau}_m$ ), the parallel bond is broken and removed from the model, which corresponds to the nucleation of the tension/shear mesocracks. The cracking of the cement reduces the contact of the corresponding pair of grains (e.g., the glass beads in Figure 6.1a) to the usual non-cohesive interaction with friction.

Potyondy and Cundall [128] demonstrated the ability of this model to reproduce a number of rock behavior characteristics such as fracture, damage-induced anisotropy, dilatation, softening, and confinement-driven strengthening. The evolution of damage is explicitly presented as a process of progressive accumulation of broken ties; “no empirical relations are needed to define damage or to quantify its effect on material behavior” [128]. The obtained damage patterns agree well with the experimental observations and reveal some subtle details of the influence of lateral confinement.

As for the model’s ability to reproduce the macroproperties of granite; the modulus of elasticity, Poisson’s ratio and uniaxial compressive strength were reproduced with satisfactory accuracy (especially in the case of 3D models). On the other hand, the tensile strength and friction angle obtained by the simulations show large discrepancies with the

experimental results. In particular, there is a marked disparity in strength in the presence of lateral confinement.

Potyondy and Cundall [128] also considered in detail the effect of particle size (given by the minimum diameter of  $2R_{min}$  as the defining length scale) on the properties of the material macro-properties. They demonstrated that the discrete-element size is not only an independent parameter of the model that controls the DEM resolution but also an essential part of the material characterization coupled with the macro-properties of tensile strength and fracture toughness. Elastic constants are rather independent of the discrete-element size due to the scaling of the stiffness of the parallel bond as a function of particle size. The unconfined compressive strength seems to show a similar trend of model objectivity. On the other hand, the splitting tensile strength in the Brazilian test ("Brazilian strength") shows a clear dependence on particle size as well as fracture toughness in the first mode (opening). This observation, that the discrete-element size, which controls the model resolution, cannot be chosen arbitrarily because it is coupled with the fracture toughness of the material (expressed by the critical stress intensity factor;  $K_{Ic}$ ), is not surprising because the very definition of the stress intensity factor implies an internal length scale. Moon and co-authors [140] developed a general procedure for calculating the fracture toughness of random packing of grains of unequal sizes.

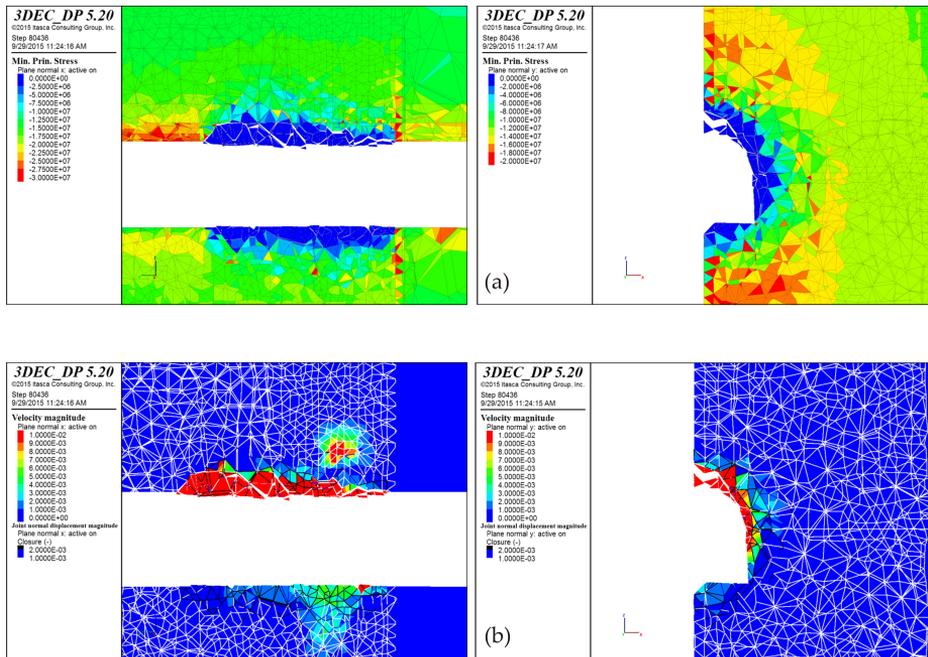


Figure 6.21: 3DEC simulation of collapse of jointed rock masses around an underground excavation due to seismic loading. Field (a) of the maximum principal stress and (b) velocity magnitudes in two planes with given color interpreters. (Courtesy of Dr. Branko Damjanac.)

In line with the observed problems with oversimplified particle shapes, the later model improvements include the use of breakable, deformable, polygonal discrete elements which, in the long run, led to the development of UDEC and 3DEC (Itasca<sup>TM</sup> Consulting Group) software packages [108]. These programs nowadays represent an indispensable tools for the calculation of intermittent (fractured and jointed) rock masses in the vicinity of underground facilities (Figure 6.21).

Potyondy [141] improved this model by including time-dependent behavior by introducing into the parallel bond the law of damage rate that mimics stress corrosion. Mas Ivars and co-authors [142] performed an important synthesis and rounding up of the whole approach by extending the model of parallel bonds to simulate natural cracks in rock masses by including joints on scales larger than the grain scale. When 3DEC is used to study the stability of cracked rock masses, the considered block is usually divided into several discrete segments defined by the intersection of natural cracks or joints that are piecewise straight (Figure 6.21). These segments (which emulate jointed rock blocks; see Figure 6.1h) are then assigned constitutive properties. The external boundary conditions of discrete segments are set while the internal boundary conditions are calculated from contact interactions which allows explicit modeling of existing cracks in fractured rock [108].

Finally, a detailed overview of the bonded-particle method as a research tool in rock mechanics and engineering applications with current trends and future directions of development is presented by Potyondy [139]. In that reference, an exhaustive list of papers is given in which the method is applied in a wide range of phenomena in rock mechanics and beyond.

### Equivalence of beam lattice and DEM representations

A set of nodal points (Figure 6.22a), randomly located in general, can be connected with beams and represented by the beam lattice (Figure 6.22b). Alternatively, these nodal points could be considered centers of circular/spherical particles (Figure 6.22c) connected in the manner illustrated by Figure 6.18d; this particle network is the typical DEM. In general, the stiffness between two particles engaged in contact is represented by the three types of springs (a normal, a shear and a rotational defined, respectively, by the spring constants  $k^n, k^t, k^a$ ), which is depicted by the symbol shown in Figure 6.22d.

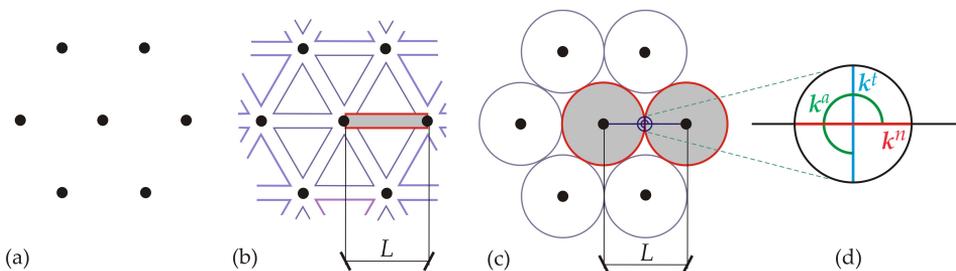


Figure 6.22: (a) A set of nodal points in space and the corresponding: (b) beam lattice and (c) DEM (network of circular particles). (d) The symbol for the particle contact reflecting three types of springs: a normal, a shear, and a rotational.

It is demonstrated by Chang and co-authors [66] that, for a 2D case (the inter-particle twisting stiffness is by definition equal to zero), the behavior of the particle network is equivalent to that of the beam lattice if the following relationships

$$k^n = \frac{E^{(b)}A^{(b)}}{L^{(b)}}, \quad k^t = \frac{12E^{(b)}I^{(b)}}{(L^{(b)})^3}, \quad k^a = \frac{E^{(b)}I^{(b)}}{L^{(b)}}, \quad (6.73)$$

hold true between the spring constants and the beam properties.





## Bibliography

- [1] Voyiadjis G. (Ed.), 2015, “Handbook of Damage Mechanics: Nano to Macro Scale for Materials and Structures,” Springer Science+Business Media, New York.
- [2] D’Addetta, G.A., Kun, F., Ramm, E., Herrmann, H.J., 2001, “From solids to granulates - Discrete element simulations of fracture and fragmentation processes in geomaterials,” P.A.Vermeer et al. (Eds.): LNP 568, pp. 231–258.
- [3] Ostoja-Starzewski, M., 2007, “Microstructural randomness and scaling in mechanics of materials.” Taylor & Francis Group, LLC, Boca Raton, FL, USA.
- [4] Greenspan, D., 1997, “Particle Modeling,” Birkhäuser Publishing, Boston.
- [5] Allen, M.P., Tildesley, D.J., 1987, “Computer Simulation of Liquids,” Oxford University Press Inc., New York.
- [6] Wang, G., Cheng, A.H.-D., Ostoja-Starzewski, M., Al-Ostaz, A., Radziszewski, P., 2010, “Hybrid Lattice Particle Modelling Approach for Polymeric Materials Subject to High Strain Rate Loads,” *Polymers*, 2, 3-30.
- [7] Hrennikoff, A., 1941, “Solution of problems of elasticity by the framework method,” *J. Appl. Mech.*, 8, A169–A175.
- [8] Ostoja-Starzewski, M., 2002, “Lattice models in micromechanics,” *Appl. Mech. Rev.*, 55(1), 35-60.

- [9] Bicanic N., 2004, "Discrete Element Methods;" in Encyclopedia of Computational Mechanics: Fundamentals (eds. Stein et al.), 311-337, DOI: 10.1002/0470091355.ecm006
- [10] Donze, F.V., Richefeu, V., Magnier, S-A., 2008, "Advances in Discrete Element Method Applied to Soil, Rock and Concrete Mechanics," *Electr. J. Geotech. Engng.*, Bouquet 08, 1-44.
- [11] Munjiza, A., 2004, "The Combined Finite-Discrete Element Method," Wiley, New York.
- [12] Alder, B.J., Wainwright, T.E., 1957, "Phase transition for a hardsphere system," *J. Chem. Phys.*, 27, 1208-1209.
- [13] Rahman, A., 1964, "Correlations in the Motion of Atoms in Liquid Argon," *Phys. Rev.*, 136, A405.
- [14] Levitt, M., Warshel, A., 1975, "Computer simulation of protein folding," *Nature*, 253, 694.
- [15] Haile, J.M., 1992, "Molecular Dynamics Simulations," John Wiley & Sons.
- [16] Ercolessi, F., 1997, "A molecular dynamics primer," Spring College in Computational Physics, ICTP, Trieste, Italy.
- [17] Rapaport, D.C., 2004, "The Art of Molecular Dynamics Simulations," Cambridge University Press.
- [18] Feynman, R.P., 1963, "The Feynman Lectures on Physics," Addison-Wesley.
- [19] Born, M., Huang, K., 1956, "Dynamical Theory of Crystal Lattices." Clarendon Press, Oxford.
- [20] Stone, A.J., 1996, "The Theory of Intermolecular Forces," Clarendon Press, Oxford.
- [21] Lennard-Jones, J.E., Devonshire, A.F., 1939, "Critical and co-operative phenomena," *Proc. Royal Soc. A*, 169(938), 317-333.
- [22] Marenic, E., Soric, J., Ibrahimbegovic, A., 2012, "Adaptive modelling in atomistic-to-continuum multiscale models," *J. Serbian Soc. Comput. Mech.*, 6(1), 169-198.
- [23] Ashby, M.F., Jones, D.R.H., 1996, "Engineering Materials 1: An Introduction to Their Properties and Applications," 2nd Edition, Butterworth & Heinemann, Oxford.
- [24] Vitek, V., 1996, "Pair potentials in atomistic computer simulations." In: *Interatomic potentials for atomistic simulations*, MRS Bulletin 21: 20-23.
- [25] Daw, M.S., Baskes, M.I., 1984, "Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals," *Phys. Rev. B*, 29, 6443-6462.

- [26] Mishin, Y., Farkas, D., Mehl, M.J., Papaconstantopoulos, D.A., 1999, "Interatomic potentials for monoatomic metals from experimental data and ab initio calculations," *Phys. Rev. B*, 59, 3393-3407.
- [27] Holian, B.L., Voter, A.F., Wagner, N.J., Ravelo, R.J., Chen, S.P., Hoover, W.G., Hoover, C.G., Dontje, T.D., 1991, "Effects on pairwise versus many-body forces on high-stress plastic deformation," *Phys. Rev. A*, 43(6), 2655-2661.
- [28] Gullet, P.M., Wagner, G., Slepoy, A., 2004, "Numerical Tools for Atomistic Simulations," SAND2003-8782, Sandia National Laboratories, Albuquerque, NM, USA.
- [29] Allen, M.P., 2004, "Introduction to Molecular Dynamics Simulation". In: *Computational Soft Matter: From Synthetic Polymers to Proteins*, John von Neumann Institute for Computing, Julich, NIC Series, Vol. 23., ISBN 3-00-012641-4, pp. 1-28.
- [30] Yip, S. (Ed.), 2005, "Handbook of Materials Modelling," Springer, Berlin.
- [31] Plimpton, S., 1995, "Fast parallel algorithms for short-range molecular-dynamics," *J. Comp. Phys.*, 117(1), 1-19.
- [32] Janežič D., Borštnik U., Praprotnik M., 2009, "Parallel Approaches in Molecular Dynamics Simulations". In: Trobec R., Vajteršic M., Zinterhof P. (eds) *Parallel Computing*. Springer, London. <https://doi.org/10.1007/978-1-84882-409-6-10>
- [33] Krajcinovic, D., 1996, "Damage Mechanics," North-Holland, Amsterdam.
- [34] Hoover, W.G., 1985, "Canonical dynamics: Equilibrium phase space distributions," *Phys. Rev. A*, 31, 1695-1697.
- [35] Landau, L.D, Lifshitz, E.M., 1970, "Theory of Elasticity." Pergamon Press, Oxford.
- [36] Egami, T., 2011, "Atomic level stress," *Progress Mater. Sci.*, 56, 637-653.
- [37] Buehler, M.J., Abraham, F.F., Gao, H., 2003, "Hyperelasticity governs dynamic fracture at a critical length scale," *Nature* 426: 141-146.
- [38] Jou, D., Casas-Vazquez, J., 1992, "Possible experiment to check the reality of a nonequilibrium temperature," *Phys. Rev. A*, 45, 8371-8373.
- [39] Holian, B.L., Voter, A.F., Ravelo, R., 1995. "Thermostatted molecular dynamics: how to avoid the Tada demon hidden in Nose-Hoover dynamics," *Phys. Rev. E* 52, 2338-2347.
- [40] Callen, H.B., 1961, "Thermodynamics," John Wiley, New York.
- [41] Wannier, G.H., 1987, "Statistical Physics," Dover, New York.
- [42] Chandler, D., 1987, "Introduction to Modern Statistical Mechanics," Oxford University Press, New York.
- [43] Mastilovic, S., 2016, "Molecular-dynamics simulations of the nanoscale Taylor test under extreme loading conditions," *Math. Mech. Solids*, 21(3), 326-338.

- [44] Fung, Y.C., 1977, "A First Course In Continuum Mechanics," Prentice Hall, Englewood Cliffs.
- [45] Taylor, G.E., 1948, "The use of flat-ended projectiles for determining dynamic yield stress I. Theoretical considerations", Proc. Royal Soc. London A 194 (1038): 289-299.
- [46] Mastilovic, S., 2015, "Impact fragmentation of nanoscale projectiles at ultrahigh striking velocities", *Meccanica* 50: 2353-2367.
- [47] Mesarovic, S.D., Padbidri, J., 2005, "Minimal Kinematic Boundary Conditions for Simulations of Disordered Microstructures," *Philosophical Magazine*, 85, 65-78.
- [48] Parrinello, M., Rahman, A., 1981, "Polymorphic transitions in single crystals: a new molecular dynamics method," *J. Appl. Phys.*, 52, 7182-7190.
- [49] Nose, S., Klein, M.L., 1983, "Constant pressure molecular dynamics for molecular systems," *Molec. Phys.*, 50, 1055-1076.
- [50] Li, M., Johnson, W.L., 1992, "Fluctuations and thermodynamic response functions in a Lennard-Jones solid," *Phys. Rev. B*, 46 (9), 5237-5241.
- [51] Li, M., Johnson, W.L., 1993, "Ergodicity and convergence of fluctuations in Parrinello-Rahman molecular dynamics," *Mat. Res. Soc. Symp. Proc.*, 291, 285-290.
- [52] Verlet, L., 1967, "Computer Experiments on Classical Fluids. I. Thermodynamical Properties on Lennard-Jones Molecules," *Phys. Rev.*, 159(1), 98-103.
- [53] Andersen, H.C., 1980, "Molecular dynamics at constant pressure and temperature," *J. Chem. Phys.*, 72, 2384-2393.
- [54] Berendsen, H.J.C. et al, 1984, "Molecular dynamics with coupling to an external bath," *J. Chem. Phys.*, 81(8), 3684-3690.
- [55] Nose, S., 1991, "Constant Temperature Molecular Dynamics Methods," *Progress Theor. Phys. Supplement*, 103, 1-46.
- [56] Holian, B.L., De Groot, A.J., Hoover W.G., Hoover C.G., 1990, "Time-reversible equilibrium and nonequilibrium isothermal-isobaric simulations with centered-difference Störmer algorithms," *Phys. Rev. A*, 41, 4552-4553.
- [57] Nose S., 2001, "An improved symplectic integrator for Nose-Poincare thermostat," *J. Phys. Soc. Jpn.* 70, 75-77.
- [58] Freddolino, P.L., Liu, F., Gruebele, M., Schulten, K., 2008, "Ten-microsecond molecular dynamics simulation of a fast folding WW domain," *Biophys J.* 94(10): L75-L77.
- [59] Bourne, N.K., 2011, "Materials' Physics in Extremes: Akrology," *Metallurgical Mater Trans A*; 42A: 2975-2984.
- [60] Greenspan, D., 1997, "Particle Modeling," Birkhäuser.

- [61] Wang, G., Ostoja-Starzewski, M., 2005, "Particle modeling of dynamic fragmentation-I: theoretical considerations," *Comput. Mater. Sci.*, 33, 429–442.
- [62] Mastilovic, S., Krajcinovic, K., 1999, "High-velocity expansion of a cavity within a brittle material," *J. Mech. Phys. Solids*, 36, 466-509.
- [63] Krajcinovic, D., Vujosevic, M., 1998, "Strain Localization – Short to Long Correlation Length Transition," *Int. J. Solids Struct.*, 35(31-32), 4147-4166.
- [64] Watson, E., Steinhauser, M.O., 2017, "Discrete Particle Method for Simulating Hypervelocity Impact Phenomena," *Materials*, 10, 379; doi:10.3390/ma10040379.
- [65] Van Mier, J.G.M., van Vliet, M.R.A., Wang, T.K., 2002, "Fracture mechanisms in particle composites: statistical aspects in lattice type analysis," *Mech. Mater.*, 34, 705–724.
- [66] Chang, C.S., Wang, T.K., Sluys, L.J., Van Mier, J.G.M., van Vliet, M.R.A., Wang, T.K., 2002, "Fracture modeling using a micro-structural mechanics approach—I. Theory and formulation," *Engng. Fract. Mech.* 69, 1941–1958.
- [67] Bažant, Z.P., Tabbara, M.R., Kazemi, M.T., Pyaudier-Cabot, G., 1990, "Random Particle Model for Fracture of Aggregate and Fiber Composites," *J. Eng. Mech.*, 116(8), 1686–1705.
- [68] Jirásek, M., Bažant, Z.P., 1995, "Microscopic fracture characteristics of random particle system," *Int. J. Fract.* 69, 201-228.
- [69] Cusatis, G., Bažant, Z.P., Cedolin, L., 2003, "Confinement-shear lattice model for concrete damage in tension and compression, I. Theory," *J. Eng. Mech.*, 129(12), 1439–1448.
- [70] Cusatis, G., Bažant, Z.P., Cedolin, L., 2006, "Confinement-shear lattice CSL model for fracture propagation in concrete," *Comp. Meth. Appl. Mech. Eng.*, 195, 7154–7171.
- [71] Kozicki, J., Tejchman, J., 2008, "Modelling of fracture process in concrete using a novel lattice model," *Granular Matter.*, 10, 377–388.
- [72] Ostoja-Starzewski, M., Wang, G., 2006, "Particle modeling of random crack patterns in epoxy plates," *Probabilistic Engng. Mech.*, 21, 267–275.
- [73] Wang, G., Al-Ostaz, A., Cheng, A.H.-D., Mantena, P.R., 2009, "Hybrid lattice particle modeling: Theoretical considerations for a 2D elastic spring network for dynamic fracture simulations," *Comput. Mater. Sci.* 44, 1126–1134.
- [74] Monette, L., Anderson, M.P., 1994, "Elastic and fracture properties of the two-dimensional triangular and square lattices," *Modelling Simul. Mater. Sci. Engng.*, 2, 53-66.
- [75] Schlangen, E., van Mier, J.G.M., 1992, "Simple lattice model for numerical simulation of fracture of concrete materials and structures," *Mater. Struct.*, 25, 534–542.

- [76] Schlangen, E., van Mier, J.G.M., 1992, "Experimental and numerical analysis of micro-mechanisms of fracture of cement-based composites," *Cem. Concr. Compos.*, 14, 105–118.
- [77] Curtin, W.A., Scher, H., 1990, "Brittle fracture in disordered materials: A spring network model," *J. Mater. Res.*, 5(3), 535-553.
- [78] Jagota, A., Bennison, S.J., 1994, "Spring-Network and Finite Element Models for Elasticity and Fracture." Proceedings of a workshop on Breakdown and non-linearity in soft condensed matter. K.K. Bardhan, B.K. Chakrabarti, A. Hansen (eds). Springer-Verlag Lecture Notes in Physics (Berlin, Heidelberg, New York) ISBN 3-540-58652-0.
- [79] Rinaldi, A., 2009, "A Rational Model for 2D Disordered Lattices Under Uniaxial Loading," *Int. J. Damage Mech.*, 18, 233-257.
- [80] Spagnoli, A., 2009, "A micromechanical lattice model to describe the fracture behavior of engineered cementitious composites," *Comput. Mater. Sci.*, 46, 7–14.
- [81] Zubelewicz, A., Bažant, Z.P., 1987, "Interface element modeling of fracture in aggregate composites," *J. Eng. Mech.*, 113(11), 1619–1630.
- [82] Riera, J.D., Miguel, L.F.F., Iturrioz, I., 2014, "Assessment of Brazilian tensile test by means of the truss-like Discrete Element Method (DEM) with imperfect mesh," *Eng. Struct.*, 81:10–21.
- [83] Rodrigues, R.S., Birck, G., Iturrioz, I., 2016, "Damage index proposals applied to quasi-brittle materials simulated using the lattice discrete element method," *Int. J. Damage Mech.*, 25 (7), 1017–1039
- [84] Đorđević, N., Vignjević, R., De Vuyst, T., Gemko, S., Campbell, J., Hughes, K., 2017, "Localization and Damage Induced Softening using Finite Element and Smooth Particle Hydrodynamic Methods," *J. Serbian Soc. Computat. Mech.*, 11 (2), 120-129.
- [85] Iturrioz, I., 2018, "Numerical DEM simulation of the evolution of damage and AE preceding failure of structural components," *Engng. Fract. Mech.*, <https://doi.org/10.1016/j.engfracmech.2018.02.023>
- [86] Kitsutaka, Y., Uchida, Y., Mihashi, H., Kaneko, Y., Nakamura, S., Kurihara, N., 2001, "Draft on the JCI standard test method for determining tension softening properties of concrete". In: de Borst et al. (eds). *Fracture mechanics of concrete structures*, Swets & Zeitlinger, Lisse, 371–376.
- [87] Van Mier, J.G.M., 2004, "Lattice modelling of size effect in concrete strength by Ince R, Arslan A, Karihaloo BL [Engineering Fracture Mechanics 2003;70:2307–20]," *Eng. Fract. Mech.*, 71, 1625–1628.
- [88] Van Mier, J.G.M., 2007, "Multi-scale interaction potentials (F - r) for describing fracture of brittle disordered materials like cement and concrete," *Int. J. Fract.*, 143, 41–78.

- [89] Karihaloo, B.L., Shao, P.F., Xiao, Q.Z., 2003, "Lattice modelling of the failure of particle composites," *Eng. Fract. Mech.*, 70, 2385–2406.
- [90] Wang, G., Al-Ostaz, A., Cheng, A.H.-D., Mantena, P.R., 2009a, "Hybrid lattice particle modeling: Theoretical considerations for a 2D elastic spring network for dynamic fracture simulations," *Comput. Mater. Sci.*, 44, 1126–1134.
- [91] Kale, S., Ostoja-Starzewski, M., 2015, „Lattice and Particle Modelling of Damage Phenomena;” in *Handbook of Damage Mechanics: Nano to Macro Scale for Materials and Structures*, Ed. Voyiadjis G., Springer Science+Business Media, New York (DOI 10.1007/978-1-4614-5589-9-23, ISBN 978-1-4614-5588-2), 203-238.
- [92] Cosserat, E., Cosserat, F., 1909, "Théorie des corps déformables. Herman, Paris, 1909.
- [93] Schlangen, E., Garboczi, E.J., 1996, "New method for simulating fracture using an elastically uniform random geometry lattice," *Int. J. Eng. Science*, 34, 1131–1144.
- [94] Schlangen, E., Garboczi, E.J., 1997, "Fracture simulations of concrete using lattice models: computational aspects," *Engng. Fract. Mech.*, 57, 319–332.
- [95] Lilliu, G., van Mier, J.G.M., 2003, "3D lattice type fracture model for concrete," *Engng. Fract. Mech.*, 70, 927–941.
- [96] Liu, J.X., Deng, S.C., Zhang, J., Liang, N.G., 2007, "Lattice type of fracture model for concrete", *Theor. Appl. Fract. Mech.*, 48, 269–284.
- [97] Nikolić, M., Karavelić, E., Ibrahimbegovic, A., Mišćević P., 2018, "Lattice Element Models and Their Peculiarities," *Arch. Computat. Methods Eng.*, 25, 753–784.
- [98] Jing, L.; Stephansson, O., 2008, "Fundamentals of discrete element methods for rock engineering: Theory and applications," Elsevier, Amsterdam.
- [99] Cundall, P.A., 1971, "A computer model for simulating progressive large scale movements in blocky rock systems," In: *Proceedings of the Symposium of International Society of Rock Mechanics*, Vol. 1, Nancy: France; 1971. Paper No. II-8.
- [100] Cundall, P.A., Strack, O.D.L., 1979, "A discrete numerical model for granular assemblies," *Geotechnique*, 29(1), 47–65.
- [101] Munjiza, A.A., Knight, E.E., Rougier, E., "Computational Mechanics of Discontinua," John Wiley & Sons, UK.
- [102] Cundall, P.A., Hart, R., 1992, "Numerical modeling of discontinua," *J. Eng. Comp.*, 9, 101–113.
- [103] Van Baars, S., 1996, "Discrete element modelling of granular materials," Heron.
- [104] Li, B., Pandolfi, A., Ravichandran, G., Ortiz, M., 2015, "Material-point erosion simulation of dynamic fragmentation of metals," *Mech. Mater.*, 80, 288–297.
- [105] Kruyt, N.P., Rothenburg, L., 2001, "Statistics of the elastic behavior of granular materials," *Int. J. Solids Struct.*, 38, 4879-4899.

- [106] Nemat-Nasser, S., Hori, M., 1993, "Micromechanics: Overall Properties of Heterogeneous Materials," North-Holland, Amsterdam.
- [107] Wellman, C., Lillie, C., Wriggers, P., 2008, "Homogenization of granular material modeled by a three-dimensional discrete element method," *Computers and Geotechnics*, 35, 394–405.
- [108] Damjanac, B., Board, M., Lin, M., Kicker, D., Leem, J., 2007, "Mechanical degradation of emplacement drifts at Yucca Mountain: A modeling case study, part 2: Lithophysal rock," *Int. J. Rock Mech. Mining Sci.*, 44, 368–399.
- [109] Maugis, D., 1999, "Contact, adhesion and rupture of elastic solids," *Solid-State Sciences*, Berlin.
- [110] Luding, S., 2008, "Cohesive, frictional powders: contact models for tension," *Granular Matter*, 10, 235–246.
- [111] Yang, S.C., Hsiau, S.S., 2001, "The simulation of powders with liquid bridges in a 2D vibrated bed," *Chem. Engng. Sci.*, 56, 6837–6849.
- [112] Delenne, J.-Y., El Youssoufi, M.S., Cherblanc, F., B enet, J.-C., 2004, "Mechanical behaviour and failure of cohesive granular materials," *Int. J. Numer. Analytical Meth. Geomech.*, 28, 1577–1594.
- [113] Shiu, W., Donz e, F.V., Daudeville, L., 2008, "Compaction process in concrete during missile impact: a DEM analysis," *Computers and Concrete*, 5(4), 329-342.
- [114] Luding, S., Manetsberger, K., Muellers, J., 2005, "A discrete model for long time sintering," *J. Mech. Phys. Solids*, 53(2), 455-491.
- [115] Ibrahimbegovic, A., Delaplace, A., 2003, "Microscale and mesoscale discrete models for dynamic fracture of structures built of brittle material," *Comput. Struct.*, 81, 1255–1265.
- [116] Tavarez, F.A., Plesha, M.E., 2007, "Discrete element method for modelling solid and particulate materials," *Int. J. Numer. Meth. Engng.*, 70, 379–404.
- [117] Sheng, Y., Yang, D., Tan, Y., Ye, J., 2010, "Microstructure effects on transverse cracking in composite laminae by DEM," *Composites Science and Technology*, 70, 2093–2101.
- [118] Cleary, P.W., Sawley, M.L., 2002, "DEM modeling of industrial granular flows: 3D case studies and the effect of particle shape on hopper discharge," *Appl. Math. Model.*, 26, 89-111.
- [119] Alonso-Marroqu n, F., Muhlhouse, H.B., Herrmann, H.J., 2008, "Micromechanical investigation of soil plasticity using a discrete model of polygonal particles," *Theoret. Appl. Mech.*, 35, 11–28.
- [120] Bierwisch, C., Kraft, T., Riedel, X., Moseler, M., 2009, "Three-dimensional discrete element models for the granular statics and dynamics of powders in cavity filling," *J. Mech. Phys. Solids*, 57(1), 10-31.

- [121] Xiang, J., Munjiza, A., Latham, J.-P., Guises, R., 2009, "On the validation of DEM and FEM/DEM models in 2D and 3D," *Engng. Comput.*, 26(6), 673-687.
- [122] Alonso-Marroquín, F., Herrmann, H.J., 2005, "The incremental response of soils. An investigation using a discrete-element model," *J. Engng. Math.*, 52, 11-34.
- [123] Salot, C., Gotteland, P., Villard, P., 2009, "Influence of relative density on granular materials behavior: DEM simulations of triaxial tests," *Granular Matter*, 11(4), 221-236.
- [124] Kun, F., Herrmann, H.J., 2000, "Damage development under gradual loading of composites," *J. Mater. Sci.* 35(18), 4685-4693.
- [125] Tykhoniuk, R., Tomas, J., Luding, S., 2006, "A microstructure-based simulation environment on the basis of an interface enhanced particle Model," *Granular Matter* 8, 159-174.
- [126] Magnier, S.A., Donze, F.V., 1998, "Numerical simulations of impacts using a discrete element method," *Mechanics of Cohesive-Frictional Materials*, 3, 257-276.
- [127] Masuya, H., Kajukawa, Y., Nakata, Y., 1994, "Application of the distinct element method to the analysis of concrete members under impact," *Nuclear Engng. Design*, 150 (2-3), 367-377.
- [128] Potyondy, D.O., Cundall, P.A., 2004, "A bonded-particle model for rock," *Int. J. Rock Mech. Mining Sci.*, 41, 1329-1364.
- [129] Wang, Y.C., Mora, P., 2008., "Macroscopic elastic properties of regular lattices," *J. Mech. Phys. Solids*, 56, 3459-3474.
- [130] Kruyt, N.P., Rothenburg, L., 1996, "A micro-mechanical definition of the strain tensor for two dimensional assemblies of particles," *J. Appl. Mech.*, 63, 706-711.
- [131] Kruyt, N.P., Rothenburg, L., 1998, "Statistical theories for the elastic moduli of two-dimensional assemblies of granular materials," *Int. J. Engng. Sci.*, 36, 1127-1142.
- [132] Cundall, P.A., 1980, "UDEC—a generalized distinct element program for modelling jointed rock." Report PCAR-1-80, Peter Cundall Associates, European Research Office, US Army Corps of Engineers.
- [133] Zubelewicz, A., Mroz, Z., 1983, "Numerical simulation of rockburst processes treated as problems of dynamic instability." *Rock Mech. Engng.*, 16, 253-274.
- [134] Plesha, M.E., Aifantis, E.C., 1983, "On the modeling of rocks with microstructure." Proc. 24th U.S. Symp. Rock Mech., Texas A&M Univ., College Station, Texas, 27-39.
- [135] Topin, V., Delenne, J.-Y., Radjaï, F., Brendel, L., Mabilille, F., 2007, "Strength and failure of cemented granular matter," *Eur. Phys. J. E* 23, 413-429.
- [136] Yang, D., Sheng, Y., Ye, J., Tan, Y., 2011b, "Dynamic simulation of crack initiation and propagation in cross-ply laminates by DEM," *Composites Sci. Tech.*, 71, 1410-1418.

- 
- [137] Jing, L., 2003, "A review of techniques, advances and outstanding issues in numerical modelling for rock mechanics and rock engineering," *Int. J. Rock Mech. Mining Sci.*, 40, 283–353.
- [138] Cundall, P.A., 1988, "Formulation of a three-dimensional distinct element model—Part I: a scheme to detect and represent contacts in a system composed of many polyhedral blocks. *Int. J. Rock Mech. Min. Sci. Geomech. Abstr.*, 25(3), 107–116.
- [139] Potyondy, D.O., 2014, "The bonded-particle model as a tool for rock mechanics research and application: current trends and future directions," *Geosystem Engng.*, 17 (6), 342–369.
- [140] Moon, T., Nakagawa, M., Berger, J., 2007, "Measurement of fracture toughness using the distinct element method," *Int. J. Rock Mech. Mining Sci.*, 44(3), 449–456.
- [141] Potyondy, D.O., 2007, "Simulating stress corrosion with a bonded-particle model for rock," *Int. J. Rock Mech. Mining Sci.*, 44, 677–691.
- [142] Mas Ivars, D., Pierce, M.E., Darcel, C., Reyes-Montes, J., Potyondy, D.O., Young, R.P., Cundall, P.A., 2011, "The synthetic rock mass approach for jointed rock mass modelling," *Int. J. Rock Mech. Mining Sci.*, 48, 219–244.

# Index

- non-collocational method, 188
- ADI schemes, 120
- algebraic non-orthogonal grid, 115
- apparent (2D) material properties, 247
- backward difference, 14
- base function
  - rectangular element, 83
  - triangular element, 83
- base functions, 37
  - finite element, 37
  - global, 66
- Bernoulli-Euler beams, 252
- block-diagonal procedure, 131
- bond rupture criteria, 256
- bonded-particle model, 265
- boundary condition
  - essential, 88
- boundary conditions
  - essential, 34, 82
  - natural, 34, 82
- numerical, 113
  - physical, 113
- canon triangle, 63
- central difference scheme, 111
- Characteristic form
  - of Euler equations, 138
- characteristic line, 126
- characteristic lines, 127
- characteristic variables, 119, 137
- consistency, 12
- constitutive equation, 29
- continuum particle, 219
- contravariant base vector, 80
- control volume, 110
- control volumes, 110
- Courant number, 114
- Crank - Nicolson method, 14
- damping, 263
- deformation energy, 245
- difference operators, 112
- diffusion equation, 14

- disorder, 256
- dissipative terms, 112
- EAM, 226
- eigenvalue, 126
- eigenvalues of flux matrices, 118
- element
  - master, 75
- elementary flux, 110
- embedded atom method, 226
- entropy condition, 148
- Euler equations, 110, 115
- explicit method, 14
- finite difference method, 11
- finite element
  - approximation, 42
  - approximations, 67
  - interpolation, 62
- finite element matrices, 42, 79
- finite element matrix, 74
- finite element method
  - 2D, 72
- finite elements, 42
- finite volume method, 109
- flow
  - quasi one-dimensional, 128
- flux of flow
  - quantities, 111
- fluxes, 110
- forward difference, 14
- geometric discretization, 110
- geometrically regular, 256
- Godunov scheme, 123
- Hertzian contact, 260
- homogenization, 259
- implicit method, 14
- integration weight function, 81
- interpolation
  - within triangles, 64
- interpolation error, 67
- intramolecular bonding interactions, 227
- Jacobian, 117
- Lagrange interpolation, 39
- Lennard-Jones 6-12 potential, 224
- linear space, 36
- list of neighbors, 235
- load vector, 37
- localization of deformation, 215
- Mach number, 119
  - undisturbed flow, 115
- many-body potentials, 226
- Maxwell-Boltzmann distribution, 231
- mesh, 11
- meshless
  - Lagrangian method, 161
- micro-polar continuum, 251
- Mohr-Coulomb type limit, 260
- monotonic numerical scheme, 148
- natural boundary condition, 34
- Navier-Stokes equations, 115
- nearest neighbors, 224
- Neumann problem, 25
- node, 11
- non-cohesive systems, 261
- non-stationary flow, 121
- numerical boundary conditions, 139
- numerical flux, 127
- pair potentials, 224
- pairwise additivity, 224
- partial differential equation, 12
  - linear, 82
- periodic boundary conditions, 234
- physical boundary conditions, 138
- post-critical softening regime, 249
- postprocessor, 83
- potential
  - Lennard-Jones 6-12, 224
- potential theory, 110
- potential well, 225
- Prandtl number, 150
- preprocessor, 83
- procedure

- of two-factor **LU** implicit factorization, 115
- processor, 83
- program
  - finite element method
    - solving, 82
  - in fortran, 90
- quenched disorder, 244
- Rankine-Hugoniot equations, 131
- representative volume element, 259
- residual, 118
- rheological model, 262
- Roe scheme, 131
- Runge-Kutta method, 113
- shock wave zone, 131
- shock waves, 110
- smooth contact, 262
- Smooth Particle Hydrodynamics, 161
- solution
  - stable, 13
- spectral radius, 131
- square integrability, 62
- stability, 13
  - of the numerical scheme, 112
- stationary flow, 121
- stiffness matrix, 37
- test function class, 35
- Timoshenko beam theory, 254
- total variation, 148
- transformations
  - global, 75
  - local, 75
- transonic flow field, 110
- variation formulation, 62
- Verlet algorithm, 228
- virtual sensor, 232
- Voronoi cells, 257
- weight function, 35, 68
- weighting function, 188





