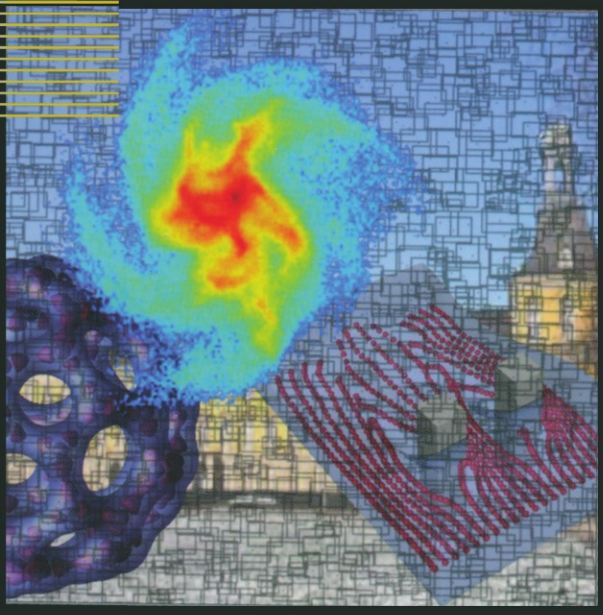
Lecture Notes in Computational Science and Engineering

43



Michael Griebel
Marc A. Schweitzer
Editors

Meshfree Methods for Partial Differential Equations II

Springer

Lecture Notes
in Computational Science
and Engineering

43

Michael Griebel
Marc A. Schweitzer
Editors

Meshfree Methods for Partial Differential Equations II

With 157 Figures and 11 Tables

Springer

Editors

Michael Griebel
Marc A. Schweitzer

Universität Bonn
Institut für Numerische Simulation
Wegelerstraße 6
53115 Bonn, Germany

e-mail: griebel@ins.uni-bonn.de
schweitzer@ins.uni-bonn.de

Library of Congress Control Number: 2002030507

Mathematics Subject Classification (2000): 65N99, 64M99, 65M12, 65Y99

ISSN 1439-7358
ISBN 3-540-23026-2 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media
springeronline.com

Printed in Germany

Cover design: Friedhelm Steinen-Broo, Estudio Calamar, Spain
Cover production: *design & production*, Heidelberg
Typeset by the authors using a Springer TEX macro package
Production: LE-TEX Jelonek, Schmidt & Vöckler GbR, Leipzig

Printed on acid-free paper 46/3142/YL - 5 4 3 2 1 0

Preface

Over the past years meshfree methods for the solution of partial differential equations have significantly matured and are used in various fields of applications. One of the reasons for this development is the fact that meshfree discretizations and particle models are often better suited to cope with geometric changes of the domain of interest than mesh-based discretization techniques such as finite differences, finite elements or finite volumes. Furthermore, the computational costs associated with mesh generation are eliminated in meshfree approaches, since they are based only on a set of independent points. From the modelling point of view, meshfree methods gained much interest in recent years since they may provide an efficient and reliable approach to the coupling of contiuum models to particle models.

In light of these developments the Sonderforschungsbereich 611 and the Gesellschaft für Mathematik und Mechanik sponsored the second international workshop on *Meshfree Methods for Partial Differential Equations*. It was hosted by the Institut für Numerische Simulation at the Rheinische Friedrich–Wilhelms Universität Bonn from September 15 to September 17, 2003. The organizers Ivo Babuška, Ted Belytschko, Michael Griebel, Wing Kam Liu, Helmut Neunzert, and Harry Yserentant invited scientist from twelve countries to Bonn with the aim to bring together European, American and Asian researchers working in this exciting area of interdisciplinary research. The objective of the workshop was not only to strengthen the mathematical understanding and analysis of meshfree discretizations but also to promote the exchange of ideas on their implementation and application.

This volume of LNCSE now comprises selected contributions of attendees of the workshop. Their content ranges from applied mathematics to physics and engineering.

Bonn,
September, 2004

Michael Griebel
Marc Alexander Schweitzer

Contents

A Particle Strategy for Solving the Fokker-Planck Equation Modelling the Fiber Orientation Distribution in Steady Recirculating Flows Involving Short Fiber Suspensions

Amine Ammar[1⋆] and Francisco Chinesta[2⋆⋆]

[1] Laboratoire de Rhéologie, Univesité Joseph Fourier, 1301 Rue de la Piscine, BP 53 Domaine Universitaire, F-38041 Grenoble cedex 9, France.
[2] Laboratoire de Mécanique des Systèmes et des Procédés, 151 Boulevard de l'Hôpital, F-75013 Paris, France.

Abstract In this work we have analyzed the application of a deterministic approximation of the diffusion term in the Fokker-Planck equation using smooth particles for computing its steady solution in a steady recirculating flow. The main idea of this approach lies in the introduction of the Fokker-Planck diffusion term into the advection one, which allows to proceed in a Lagrangian deterministic manner without a mesh support requirement.

1 Introduction

As indicated by Chaubal et al. [CHA97], "complex fluid" is the term commonly used to describe a wide class of liquid-like materials, in which the relaxation time towards the equilibrium state occurs sufficiently slowly that significant changes in the microstructural configuration, and thus in the macroscopic properties, can be induced by the flow. Viscoelastic fluids or short fiber suspensions may be considered as examples of complex fluids. The Fokker-Planck formalism is a commonly used description of kinetic theory problems, for describing the evolution of the configuration distribution function. This function represents the probability of finding the microstructural element in a particular configuration.

In the case of a short fiber suspension, the configuration distribution function (also known as orientation distribution function) gives the probability of finding the fiber in a given direction. Obviously, this function depends on the

⋆ Amine.Ammar@ujf-grenoble.fr
⋆⋆ francisco.chinesta@paris.ensam.fr

physical coordinates (space and time) as well as on the configuration coordinates, that taking into account the rigid character of the fibers, are defined on the surface of the unit sphere. Thus, we can write $\Psi(\underline{x}, t, \underline{p})$, where $\underline{x}$ defines the position of the fiber center of mass, t the time and $\underline{p}$ the unit vector defining the fiber orientation. The evolution of the distribution function is given by the Fokker-Planck equation

$$\frac{d\Psi}{dt} = -\frac{\partial}{\partial \underline{p}}(\Psi \dot{\underline{p}}) + \frac{\partial}{\partial \underline{p}}\left(D_r \frac{\partial \Psi}{\partial \underline{p}}\right) \tag{1.1}$$

where d/dt represents the material derivative, D_r is a diffusion coefficient and $\dot{\underline{p}}$ is the fiber rotation velocity. When the fibers are assumed with an ellipsoidal shape and the suspension is dilute enough, the rotation velocity can be obtained from the Jeffery's equation

$$\dot{\underline{p}} = \underline{\underline{\Omega}}\ \underline{p} + k\ \underline{\underline{D}}\ \underline{p} - k(\underline{p}^T\ \underline{\underline{D}}\ \underline{p})\ \underline{p} \tag{1.2}$$

where $\underline{\underline{\Omega}}$ and $\underline{\underline{D}}$ are the vorticity and the strain rate tensors respectively, associated with the fluid flow undisturbed by the presence of the fiber, and k is a scalar which depends on the fiber aspect ratio λ (ratio between the fiber length and the fiber diameter)

$$k = \frac{\lambda^2 - 1}{\lambda^2 + 1} \tag{1.3}$$

Although this work focuses in the flow of short fiber suspensions, the numerical procedures here developed could be applied successfully to other fluids whose microstructure is described by similar kinetic theory models. This is the case for example of some viscoelastic fluids. In the FENE model [BIR87], the probability distribution function depends on the physical coordinates and on the conformation coordinates that in this case are defined by the end-to-end molecule vector $\underline{q}$. The equation governing its evolution results:

$$\frac{d\Psi}{dt} = -\frac{\partial}{\partial \underline{q}}(\Psi \dot{\underline{q}}) + \frac{1}{2}\frac{\partial}{\partial \underline{q}}\left(\frac{\partial \Psi}{\partial \underline{q}}\right) \tag{1.4}$$

where the evolution of $\underline{q}$ can be evaluated from

$$\dot{\underline{q}} = \mathrm{Grad}_{\underline{v}}\ \underline{q} - \frac{1}{2}\underline{F}(\underline{q}) \tag{1.5}$$

with the spring force $\underline{F}$ given by

$$\underline{F} = \frac{1}{1 - \frac{\|\underline{q}\|^2}{b^2}}\ \frac{\underline{q}}{\|\underline{q}\|} \tag{1.6}$$

being b the maximum molecule length. The FENE-P model is obtained by averaging the square norm of the molecule length in the expression of the spring force [BIR80].

In these cases the Fokker-Planck equations remain linear, even if the advection fields depend in a non-linear manner on the configuration coordinates (see for instance Eqs. (1.2) and (1.5)).

Another more complex example consists in the model describing the behavior of Liquid Crystals Polymers (LCP) due to Doi. This model has been treated using a particle technique in [CHA97]. In this case the microstructural element is a rigid axis-symmetric rod of infinite aspect ratio. Under some simplifying assumptions the orientation distribution (also defined on the surface of the unit sphere) evolves according to

$$\frac{d\Psi}{dt} \;=\; -\frac{\partial}{\partial \underline{q}}(\Psi \dot{\underline{p}}) \;+\; \frac{\partial}{\partial \underline{p}}\left(D_r\left\{\frac{\partial \Psi}{\partial \underline{p}} \;+\; \Psi \frac{\partial U}{\partial \underline{p}}\right\}\right) \tag{1.7}$$

where the advection field $\dot{\underline{p}}$ is given by

$$\dot{\underline{p}} \;=\; \mathrm{Grad}_{\underline{v}}\, \underline{p} - (\underline{p}^T\, \underline{\underline{D}}\, \underline{p})\, \underline{p} \tag{1.8}$$

As the potential U depends on the second moment of the distribution function $\langle \underline{p} \otimes \underline{p} \rangle$ defined by

$$\langle \underline{p} \otimes \underline{p} \rangle \;=\; \oint \underline{p} \otimes \underline{p}\; \Psi(\underline{p})\; d\underline{p} \tag{1.9}$$

and the Fokker-Planck equation results in this case non-linear.

Many of the experimental and industrial flows show recirculating areas or recirculate themselves. For example, many rheometric devices involve this type of flows, whose steady and recirculating character introduce some additional difficulties in their numerical simulation. Actually, the Fokker-Planck equation which defines an advection problem in physical coordinates, is supposed to have a steady solution in these steady recirculating flows but neither boundary conditions nor initial conditions are known in such flows.

In a former paper [CHI03] the discretisation of the advection dominated Fokker-Planck equation, governing the fiber orientation in short fiber suspension flows, was carried out using a particle technique, where the diffusion term was modelled from random motions. It was pointed out that the number of fibers required in this stochastic simulation to describe the fiber distribution increases significantly with the diffusion coefficient D_r. Thus, it was argued that for practical applications the use of the particle method in the framework of a stochastic simulation, is restricted to very slight diffusion effects. When the diffusion becomes dominant, continuous approximations using a fixed mesh seem to be suitable, but in this case accurate stabilizations are required for dealing with small diffusion effects, and a lack of accuracy is noticed in the treatment of the advection dominated case.

Chaubal et al. [CHA97] propose the use of the SPH (Smooth Particle Hydrodynamics) to solve the dynamics of a liquid crystalline polymer (LCP) using the Doi's model. Thus, the diffusion term is treated in a determinist

manner, and high accurate results were obtained due to the meshless and Lagrangian character of the SPH technique considered by the authors.

In this work we examine the application of this kind of techniques for solving the steady Fokker-Planck equation in steady recirculating flows, but we will limit our attention to the 2D linear Fokker-Planck equations.

2 A Particle Discretisation

In this section we will consider the simplest linear form of the Fokker-Planck equation (1.1), that in the 2D case results

$$\frac{d\Psi}{dt} \;=\; -\frac{\partial}{\partial\varphi}(\Psi\dot{\varphi}) \;+\; \frac{\partial}{\partial\varphi}\left(D_r\frac{\partial\Psi}{\partial\varphi}\right) \tag{2.1}$$

where $\Psi(\underline{x},t,\varphi)$, $\dot{\varphi}=\dot{\varphi}(\underline{x},t,\varphi)$ and D_r is assumed constant. Due to the steady character of the flow kinematics we can remove the temporal variable in the expression of $\dot{\varphi}$. A general approximation of the steady probability distribution function at point $\underline{x}_0$ can be written as

$$\Psi(\underline{x}_0,\varphi) \;\approx\; \sum_{i=1}^{i=N}\alpha_i(\underline{x}_0)\;F_i(\varphi) \tag{2.2}$$

Due to the linearity and homogeneity of Eq. (2.1) we can compute its solution for each function $F_i(\varphi)$ along the pathline related to the point $\underline{x}_0$. These solutions are denoted by $\Psi_i(\underline{x}(\underline{x}_0,t_0;t),\varphi)$, where the notation $\underline{x}(\underline{x}_0,t_0;t)$ refers to the position at time t of a particle located at point $\underline{x}_0$ at time t_0. The trajectory is defined by the equation

$$\underline{x}(\underline{x}_0,t_0;t) \;=\underline{x}_0 \;+\; \int_{t_0}^{t}\underline{v}(\underline{x}(\underline{x}_0,t_0;t'))\;dt' \tag{2.3}$$

Thus, the general solution of Eq. (2.1) results

$$\Psi(\underline{x}(\underline{x}_0,t_0;t),\varphi) \;\approx\; \sum_{i=1}^{i=N}\alpha_i(\underline{x}_0)\;\Psi_i(\underline{x}(\underline{x}_0,t_0;t),\varphi) \tag{2.4}$$

The particle method lies in taking the Dirac masses in Eq. (2.2). Thus, if we consider the N directions defined by $\varphi_i=(i-1)\;h_\varphi$, $i\in[1,\cdots,N]$, with $h_\varphi=2\pi/N$, Eq. (2.2) becomes

$$\Psi(\underline{x}_0,\varphi) \;\approx\; \sum_{i=1}^{i=N}\alpha_i(\underline{x}_0)\;\delta(\varphi-\varphi_i) \tag{2.5}$$

where $\alpha_i(\underline{x}_0)$ represents the "mass" of the particle aligned in the φ_i direction.

2.1 The Advection Equation

In the case of neglecting all the diffusion effects, the solution $\Psi_i(\underline{x}(\underline{x}_0, t_0; t), \varphi)$ results

$$\Psi_i(\underline{x}(\underline{x}_0, t_0; t), \varphi) = \delta(\varphi - \varphi(\underline{x}_0, t_0, \varphi_i; t)) \tag{2.6}$$

where $\varphi(\underline{x}_0, t_0, \varphi_i; t)$ denotes the orientation at time t of a fiber located at time t_0 in $\underline{x}_0$ and whose orientation was defined by φ_i. Obviously, the spatial location of a fiber refers to the position of its center of mass, that at time t is given by Eq. (2.3).

In the pure advection case $\varphi(\underline{x}_0, t_0, \varphi_i; t)$ becomes

$$\varphi(\underline{x}_0, t_0, \varphi_i; t) \; = \varphi_i \; + \; \int_{t_0}^{t} \dot{\varphi}(\underline{x}(\underline{x}_0, t_0; t'), \varphi(\underline{x}_0, t_0, \varphi_i; t'))\; dt' \tag{2.7}$$

Thus, Eqs. (2.3) and (2.7) describe the position and orientation of each fiber along the flow trajectory. Several discretisations of these equations exist, being the simplest one the backward Newton method, that consider the time interval $[t_0, t]$ divided into M intervals $[t_m, t_{m+1}]$ of length Δt such that $M\Delta t = t - t_0$, and the fibers updating given by

$$\begin{cases} \underline{x}(\underline{x}_0, t_0; t_{m+1}) \; = \; \underline{x}(\underline{x}_0, t_0; t_m) \; + \; \underline{v}(\underline{x}(\underline{x}_0, t_0; t_m))\; \Delta t \\ \varphi(\underline{x}_0, t_0, \varphi_i; t_{m+1}) \; = \; \varphi(\underline{x}_0, t_0, \varphi_i; t_m) \; + \\ \qquad + \; \dot{\varphi}(\underline{x}(\underline{x}_0, t_0; t_m), \varphi(\underline{x}_0, t_0, \varphi_i; t_m))\; \Delta t, \qquad \forall i \in [1, \cdots, N] \end{cases} \tag{2.8}$$

where

$$\begin{cases} \underline{x}(\underline{x}_0, t_0; t_0) \; = \; \underline{x}_0 \\ \varphi(\underline{x}_0, t_0, \varphi_i; t_0) \; = \; \varphi_i, \qquad \forall i \in [1, \cdots, N] \end{cases} \tag{2.9}$$

2.2 Steady Recirculating Flows

In the case of a steady recirculating flow the particles come back to the departure position after a time T, that corresponds with the period of the considered trajectory. Thus, we can write

$$\underline{x}(\underline{x}_0, t_0; T) \; = \; \underline{x}(\underline{x}_0, t_0; t_0) \; = \; \underline{x}_0 \tag{2.10}$$

However, the final orientation will be, in general, different to the initial one, i.e.

$$\varphi(\underline{x}_0, t_0, \varphi_i; T) \neq \varphi_i, \; \forall i \tag{2.11}$$

On the other hand, the steady solution of the probability distribution at point $\underline{x}_0$, $\Psi(\underline{x}_0, \varphi)$, requires its periodicity along the closed streamline

$$\sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\ \delta(\varphi - \varphi(\underline{x}_0, t_0, \varphi_i; T)) \ = \ \sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\ \delta(\varphi - \varphi_i) \tag{2.12}$$

but this expression, (Eq. (2.12)), cannot be used in its present form because the Dirac masses are concentrated in different angles at t_0 and T.

To use this expression we need to transfer the masses concentrated in the directions $\varphi(\underline{x}_0, t_0, \varphi_i; T)$ to the initial ones φ_i. In [CHI03] this transfer is performed from each angle $\varphi(\underline{x}_0, t_0, \varphi_i; T)$ towards the neighbor directions, being the mass transferred to each neighbor direction proportional to the distance between them. Thus, we can finally write (see [CHI03] for more details)

$$\sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\ \delta(\varphi - \varphi(\underline{x}_0, t_0, \varphi_i; T)) \ = \ \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} \beta_{ij}\alpha_j(\underline{x}_0)\ \delta(\varphi - \varphi_i) \tag{2.13}$$

where β_{ij} depends in the considered probability transfer from $\varphi(\underline{x}_0, t_0, \varphi_i; T)$ to φ_j. If a linear transfer is used, then it results

$$\beta_{ij} \ = \ \begin{cases} \frac{1-|\varphi(\underline{x}_0, t_0, \varphi_i; T) - \varphi_j|}{h_\varphi} & if\ |\varphi(\underline{x}_0, t_0, \varphi_i; T) - \varphi_j| \leq h_\varphi \\ 0 & if\ |\varphi(\underline{x}_0, t_0, \varphi_i; T) - \varphi_j| > h_\varphi \end{cases}$$

The periodicity condition becomes in this case

$$\sum_{i=1}^{i=N} \sum_{j=1}^{j=N} \beta_{ij}\alpha_j(\underline{x}_0)\ \delta(\varphi - \varphi_i) \ =$$

$$= \ \sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\ \delta(\varphi - \varphi_i) \ = \ \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} \delta_{ij}\alpha_j(\underline{x}_0)\ \delta(\varphi - \varphi_i) \tag{2.14}$$

where δ_{ij} is the unit tensor. The previous equation implies

$$\sum_{j=1}^{j=N} (\beta_{ij} - \delta_{ij})\alpha_j(\underline{x}_0) \ = \ 0, \qquad \forall i \in [1, \cdots, N] \tag{2.15}$$

that with the normality condition

$$1 \ = \ \int_0^{2\pi} \Psi(\underline{x}_0, \varphi)\ d\varphi \ = \ \sum_{i=1}^{i=N} \alpha_i(\underline{x}_0) \tag{2.16}$$

allows to compute the N coefficients $\alpha_i(\underline{x}_0)$, and then, the steady solution of the probability distribution at point $\underline{x}_0$

$$\Psi(\underline{x}_0, \varphi) \ \approx \ \sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\ \delta(\varphi - \varphi_i) \tag{2.17}$$

2.3 Introducing Diffusion Effects Using a Deterministic Approximation

When diffusion effects are retained in the kinetic model the application of a particle technique becomes delicate as we will describe in this section. The first possibility for introducing of the diffusion effects lies in the use of a stochastic technique [CHI03]. Thus, a great amount of particles are considered at point $\underline{x}_0$ aligned in each direction φ_i. Then, each fiber is subjected to three actions during its movement along its closed trajectory: (i) the advection of its center of mass, (ii) the rotation induced by the term $\dot{\underline{p}}$ and (iii) the rotation associated with the diffusion that is modelled from a random motion. In spite of the simplicity of its computational implementation and the very accurate results obtained for slight diffusion coefficients, the great number of particles required when the diffusion coefficient increases makes this strategy useless for treating the problems encountered in practical applications.

To circumvent these computational drawbacks, we transform the linear Fokker-Planck advection-diffusion equation in a pure advection problem. For this purpose we define the new advection field $\dot{\underline{p}}^*$

$$\dot{\underline{p}}^* \;=\; \dot{\underline{p}} \;-\; D_r \,\frac{\frac{\partial \Psi}{\partial \underline{p}}}{\Psi} \tag{2.18}$$

that introduced in the pure advection Fokker-Planck equation

$$\frac{d\Psi}{dt} \;=\; -\frac{\partial}{\partial \underline{p}}(\Psi \dot{\underline{p}}^*) \tag{2.19}$$

leads to the standard linear advection-diffusion Fokker-Planck equation (1.1).

In the 2D case developed in the previous section, both equations result in the following purely-advection problem

$$\dot{\varphi}^* \;=\; \dot{\varphi} \;-\; D_r \,\frac{\frac{\partial \Psi}{\partial \varphi}}{\Psi} \tag{2.20}$$

$$\frac{d\Psi}{dt} \;=\; -\frac{\partial}{\partial \varphi}(\Psi \dot{\varphi}^*) \tag{2.21}$$

Now, Eq. (2.21) could be used for computing the different solutions associated with the Dirac's distributions $\delta(\varphi - \varphi_i),\ \forall i$, but two difficulties appear: (i) the derivative and the ratio of linear combinations of Dirac masses cannot be defined in a proper way, and (ii) due to the distribution function derivative required to compute the modified advection field $\dot{\varphi}^*$ in Eq. (2.20), a coupling between the different fibers takes place, and consequently, in this case, the evolution of each fiber cannot be computed independently of the other ones.

In order to circumvent the first difficulty we introduce a smoothed approximation of the Dirac mass given by

$$\zeta(x) \;=\; \frac{e^{-x^2}}{\sqrt{\pi}} \tag{2.22}$$

which verifies the normality condition

$$\int_{-\infty}^{\infty} \zeta(x)\; dx \;=\; 1 \tag{2.23}$$

To adapt its sharpness we modify this function in the following manner

$$\zeta_\epsilon(x) \;=\; \frac{\zeta\left(\frac{x}{\epsilon}\right)}{\epsilon} \tag{2.24}$$

which verifies also the normality condition. Fig. 2.1 depicts this function for three values of the ϵ parameter.

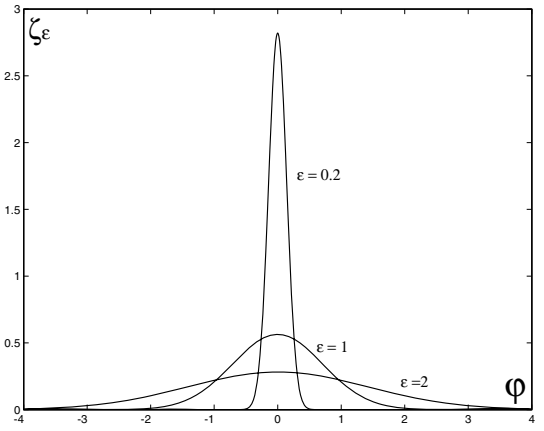


Figure 2.1. Representation of the cut-off function for different values of the ϵ parameter

In this form, the gradients and ratios can be computed in a proper way. On the other hand to avoid the second limitation two simple possibilities exist: (i) a direct procedure that compute N times, the evolution history of N particles along the closed trajectory, and (ii) an iteration algorithm that computes the evolution of each particle along the closed trajectory independently of the other ones due to the fact that the coupling term is evaluated from the solution at the previous iteration. In the following sections we describe both procedures.

A Direct Procedure.

We write the searched solution at point $\underline{x}_0$ in a matrix form

$$\Psi(\underline{x}_0,\varphi) \;\approx\; \sum_{i=1}^{i=N} \alpha_i(\underline{x}_0)\;\zeta_\epsilon(\varphi-\varphi_i) \;=\; \underline{\alpha}(\underline{x}_0)\cdot\underline{\zeta}_\epsilon(\varphi) \tag{2.25}$$

where the unknown coefficients $\alpha_i(\underline{x}_0)$, contained in the vector $\underline{\alpha}(\underline{x}_0)$, must be determined in order to satisfy the periodicity condition imposed by both the steady character of the searched solution and the steady and recirculating character of the flow.

We define the following N alpha-vectors:

$$\underline{\alpha}^k(\underline{x}_0) \;=\; \begin{pmatrix} \alpha_1^k(\underline{x}_0) \\ \vdots \\ \alpha_{k-1}^k(\underline{x}_0) \\ \alpha_k^k(\underline{x}_0) \\ \alpha_{k+1}^k(\underline{x}_0) \\ \vdots \\ \alpha_N^k(\underline{x}_0) \end{pmatrix} \;=\; \begin{pmatrix} \nu \\ \vdots \\ \nu \\ \mu \\ \nu \\ \vdots \\ \nu \end{pmatrix}, \qquad \forall k\in[1,\cdots,N] \tag{2.26}$$

where ν and μ can be chosen arbitrarily with the only restriction of verifying the normality condition, i.e. $(N-1)\nu+\mu=1$. We can notice that the uncoupled procedure, applied when diffusion effects are neglected, uses $\nu=0$ and $\mu=1$.

Introducing this notation, Eq. (2.25) can be rewritten as

$$\Psi(\underline{x}_0,\varphi) \;\approx\; \sum_{k=1}^{k=N} \beta^k\;\underline{\alpha}^k(\underline{x}_0)\cdot\underline{\zeta} \;=\; \sum_{k=1}^{k=N} \beta^k\;\Psi^k(\underline{x}_0,\varphi) \tag{2.27}$$

Now, we can compute the evolution along a closed trajectory of each solution

$$\Psi^k(\underline{x}_0,\varphi) \;=\; \underline{\alpha}^k(\underline{x}_0)\cdot\underline{\zeta}_\epsilon(\varphi) \tag{2.28}$$

which in fact requires the tracking of N particles. For this purpose, we need to integrate the evolution of the orientation for each fiber (represented by the subscript i) in each problem (noted by the superscript k). If we consider, for a sake of simplicity, the backward Newton method again, it results

$$\varphi(\underline{x}_0,t_0,\varphi_i^k;t_{m+1}) = \varphi(\underline{x}_0,t_0,\varphi_i^k;t_m) + \dot{\varphi}(\underline{x}(\underline{x}_0,t_0;t_m),\varphi(\underline{x}_0,t_0,\varphi_i^k;t_m))\;\Delta t -$$

$$-\;D_r\;\frac{\frac{\partial}{\partial\varphi}\left(\sum_{j=1}^{j=N}\alpha_j^k(\underline{x}_0)\zeta_\epsilon(\varphi-\varphi(\underline{x}_0,t_0,\varphi_j^k;t_m))\right)_{\varphi=\varphi(\underline{x}_0,t_0,\varphi_i^k;t_m)}}{\sum_{j=1}^{j=N}\alpha_j^k(\underline{x}_0)\zeta_\epsilon(\varphi(\underline{x}_0,t_0,\varphi_i^k;t_m)-\varphi(\underline{x}_0,t_0,\varphi_j^k;t_m))}\;\Delta t \tag{2.29}$$

$$\forall k\in[1,\cdots,N],\quad \forall i\in[1,\cdots,N],\quad \forall m\in[0,\cdots,M-1]$$

After a complete turn of period T we obtain

$$\varphi(\underline{x}_0, t_0, \varphi_i^k; T) \quad \forall i \in [1, \cdots, N], \quad \forall k \in [1, \cdots, N] \tag{2.30}$$

Now the weights associated to each orientation $\varphi(\underline{x}_0, t_0, \varphi_i^k; T)$, $\beta^k \alpha_i^k(\underline{x}_0)$, must be transferred to the angles φ_i to enforce the periodicity. This operation can be accurately carried out by using the algorithm proposed in [CHI03] as previously described.

This algorithm offers the steady solution of the problem from the resolution of the N problems, requiring each one of them, the integration of the orientation evolution of N fibers along the closed trajectory. In this form the algorithm results of order N^2.

A Fixed Point Iteration Algorithm. In this case we start from the solution obtained assuming $D_r = 0$. This solution can be written as

$$\Psi^0(\underline{x}(\underline{x}_0, t_0; t), \varphi) \approx \sum_{i=1}^{i=N} \alpha_i^0(\underline{x}_0)\ \zeta_\epsilon(\varphi - \varphi(\underline{x}_0, t_0, \varphi_i; t)), \quad \forall t \in [t_0, T] \tag{2.31}$$

Now, the iteration k results

$$\begin{aligned}\varphi^k(\underline{x}_0, t_0, \varphi_i; t_{m+1}) = \varphi^k(\underline{x}_0, t_0, \varphi_i; t_m) + \dot{\varphi}(\underline{x}(\underline{x}_0, t_0; t_m), \varphi^k(\underline{x}_0, t_0, \varphi_i; t_m))\Delta t \\ - D_r \frac{\frac{\partial}{\partial \varphi}\left(\Psi^{k-1}(\underline{x}(\underline{x}_0, t_0; t), \varphi)\right)_{\varphi=\varphi^k(\underline{x}_0, t_0, \varphi_i; t_m)}}{\Psi^{k-1}(\underline{x}(\underline{x}_0, t_0; t), \varphi^k(\underline{x}_0, t_0, \varphi_i; t_m))}\ \Delta t\end{aligned} \tag{2.32}$$

for all $i \in [1, \cdots, N]$, and all $m \in [0, \cdots, M-1]$, where in this case the superscript $k \geq 1$ refers to the iteration.

This algorithm results of order N but the convergence is not ensured. Some improvements can be introduced, as for example the introduction of the diffusion coefficient step by step.

3 Numerical Examples

In this section two numerical examples involving steady recirculating flows will be considered. The first one is defined by the following kinematics:

$$\underline{v} = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -y\sqrt{x^2+y^2} \\ x\sqrt{x^2+y^2} \end{pmatrix} \tag{3.1}$$

The steady solution of the probability distribution is searched at point $(0, 1)$. For this problem the exact solution can be computed due to the solution symmetry as described in [CHI03]. Thus, for a diffusion coefficient $D_r = 0.2$ and fibers with an aspect ratio related to $k = 0.6$, the fixed point algorithm proposed in the section 2.3 with $\epsilon = 0.5$ and $N = 72$ converges in around 10 iterations to the reference solution as depicted in Fig. 3.2.

Now, the direct method described in the section 2.3 is applied. Obviously, the numerical solution accuracy will depends on the μ coefficient (for $\mu \in [3, 5]$

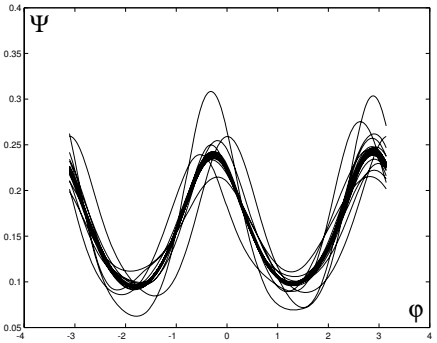


Figure 3.2. Numerical solution computed at point $(0, 1)$ using the fixed point algorithm

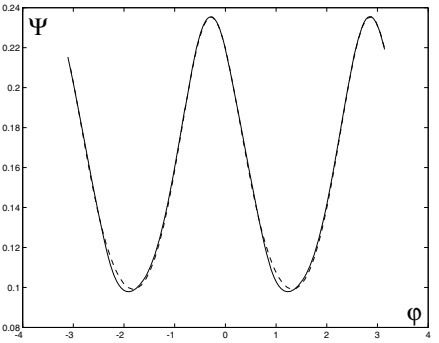


Figure 3.3. Numerical solution computed at point $(0, 1)$ using the direct procedure with $\mu = 1.2/N$

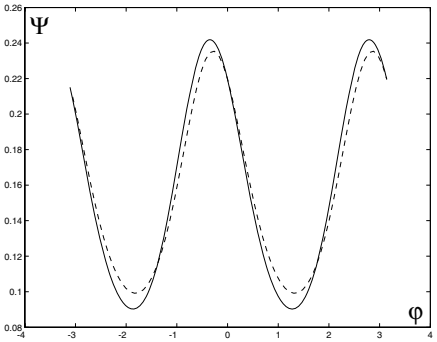


Figure 3.4. Numerical solution computed at point $(0, 1)$ using the direct procedure with $\mu = 10/N$

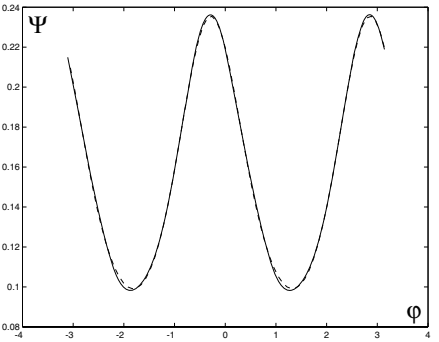


Figure 3.5. Numerical solution computed at point $(0, 1)$ using the direct procedure with $\mu = 0$

the extact and the computed solutions are in very good agreement). Figs. 3.3 and 3.4 depicts the fiber distribution related to $\mu = 1.2/N$ and $\mu = 10/N$ respectively as well as the exact solution (dashed line). The case of $\mu = 0$ (where $N - 1$ fibers with equal weights are considered) is depicted in Fig. 3.5. The solution accuracy can be improved by increasing the number of particles N involved in the simulation. A first order convergence is noticed when the number of particles N is increased. In these examples we don't take μ in the interval $[3, 5]$ because in that case the numerical and reference solutions are completely superposed.

Fig. 3.6 depicts the fiber distribution in some points on the closed streamline related to point $(0, 1)$ computed using the following model parameters: $k = 0.8$, $\epsilon = 0.5$, $D_r = 0.2$ and $N = 72$. The fiber orientation is then defined on the unit circle, where the orientation of each fiber is depicted by a small

circle. Thus, for each one of the eight points considered, the orientation of the 72 fibers involved in the simulation are depicted.

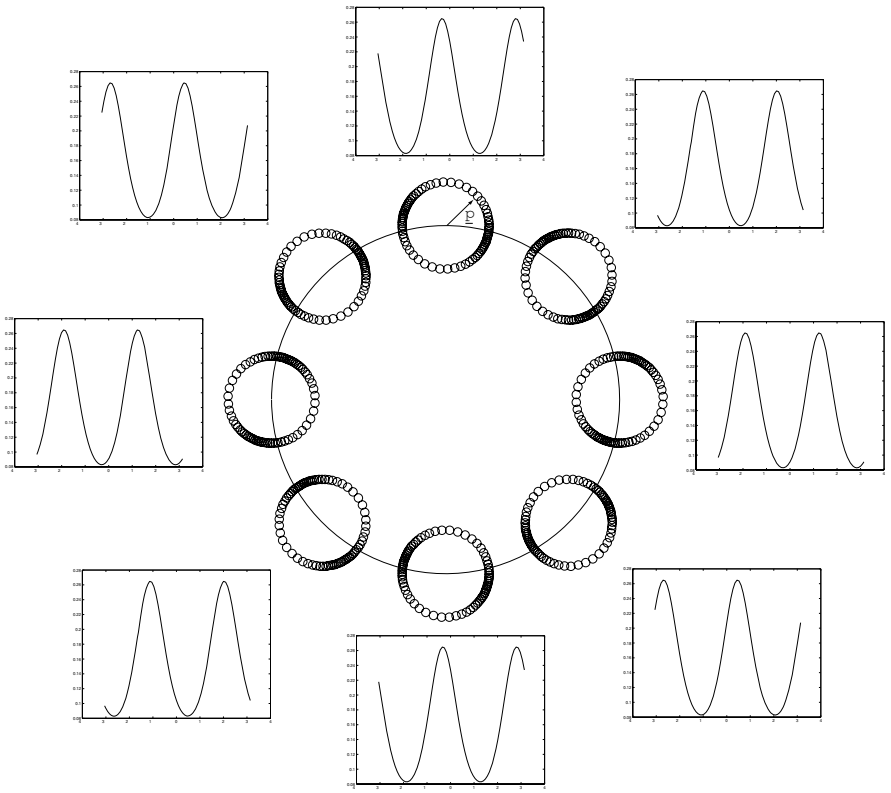


Figure 3.6. Fiber orientation distribution along the streamline related to the point $(0, 1)$

The second example concerns the flow kinematics induced by two-eccentric rotating cylinders. The velocity field was computed assuming a Newtonian fluid behavior and using the well known lubrification hypothesis (justified by the fact that the flow gap is much more lower than the cylinder radius). The inner cylinder is rotating with a velocity of 0.1 rad/s. The cylinder radius are 0.01 m. and 0.015 m., being the distance between the both cylinder axes of 0.0031 m. The model parameters are $k = 0.8$, $D_r = 0.2$, $\epsilon = 0.5$ and $N = 0.72$. Figs. 3.7 and 3.8 depict the fiber orientation distribution along a streamline located in the secondary vortex and the main recirculation respectively.

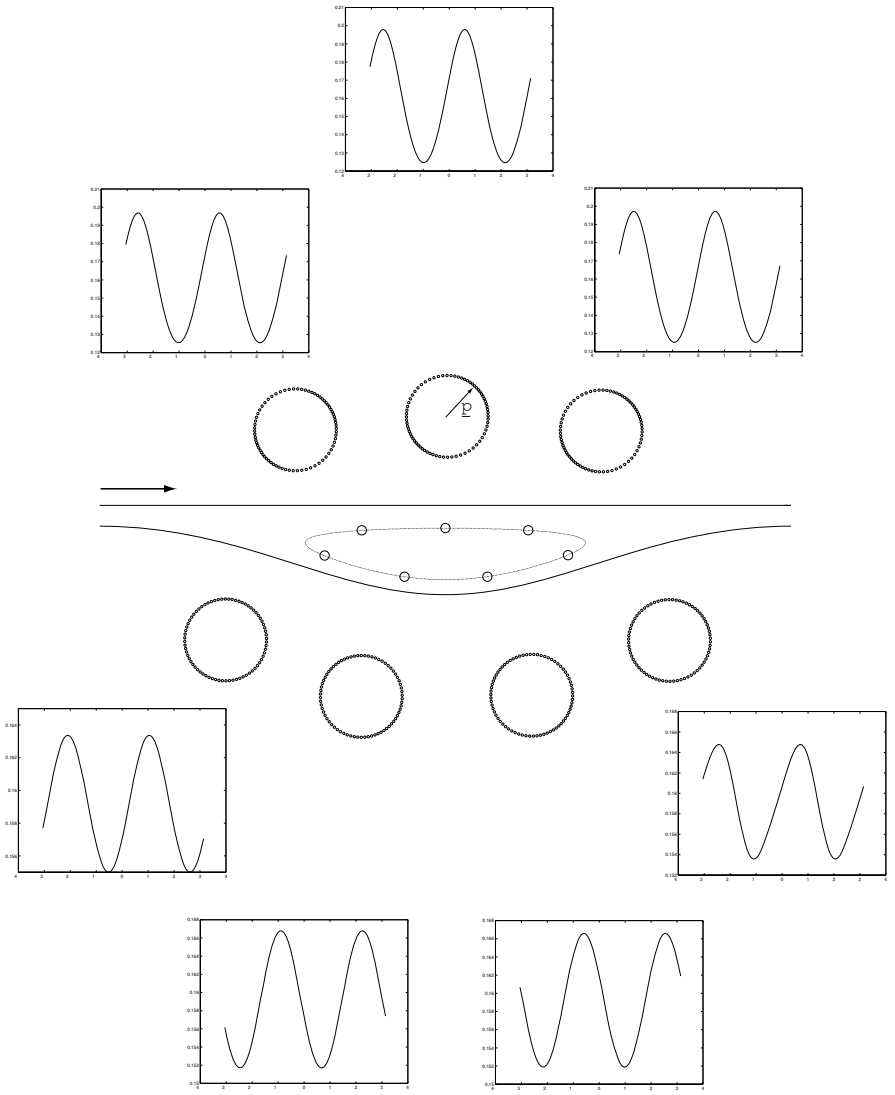


Figure 3.7. Fiber orientation distribution on a streamline located in the secondary vortex of a two-eccentric rotating cylinders device

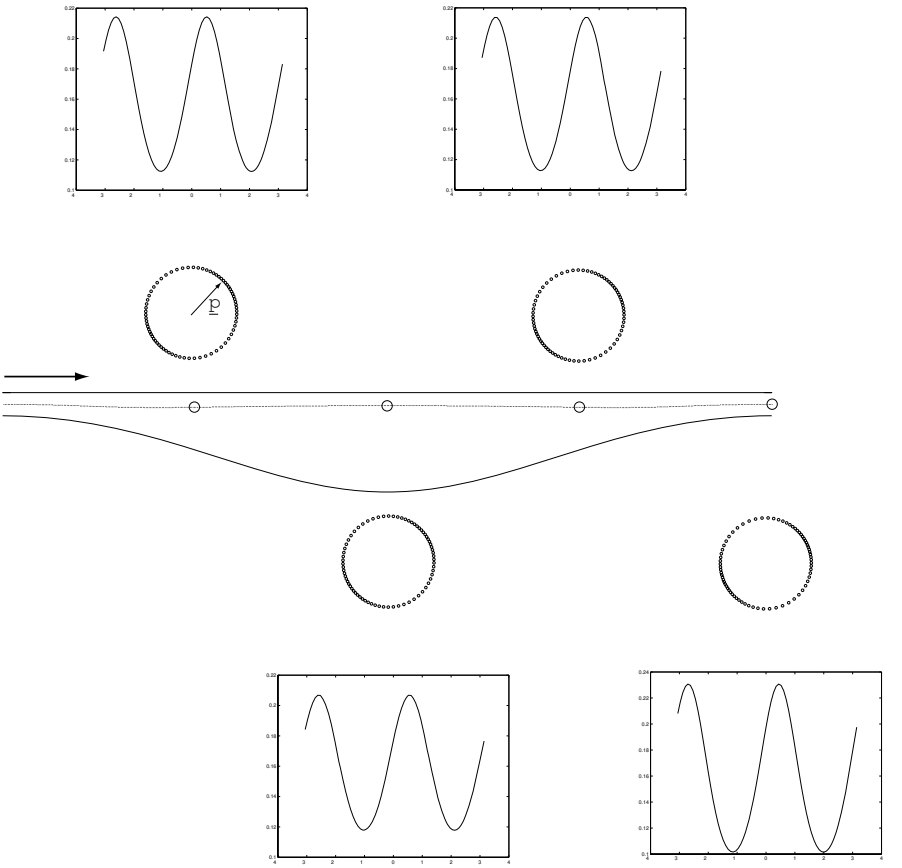


Figure 3.8. Fiber orientation distribution on a streamline located in the main recirculation of a two-eccentric rotating cylinders device

4 Conclusions

In this work we have analyzed the application of a deterministic approximation of the diffusion term in the Fokker-Planck equation using smooth particles for computing its steady solution in a steady recirculating flow. The main idea of this approach lies in the introduction of the Fokker-Planck diffusion term into the advection one, which allows to proceed in a Lagrangian deterministic manner without a mesh support requirement. In spite of its simplicity, and from the first numerical results here shown, this technique seems to be very accurate.

References

[BIR80] R.B. Bird, P.J. Dotson, N.L. Johson, Polymer solution rheology based on a finitely extensible bead-spring chain model, Journal of Non-Newtonian Fluid Mechanics, **7** 213-235 (1980).

[BIR87] R.B. Bird, C.F. Curtiss, R.C. Armstrong and O. Hassager, Dynamics of Polymeric Liquids, Vol. 2, Kinetic Theory, 2nd ed., Wiley, New York, 1987.

[CHA97] C.V. Chaubal, A. Srinivasan, O. Egecioglu and L.G. Leal, Smoothed particle hydrodynamics techniques for the solution of kinetic theory problems. Part 1: Method. Journal of Non-Newtonian Fluid Mechanics, **70**, 125–154 (1997).

[CHI03] F. Chinesta, G. Chaidron and A. Poitou, On the solution of the Fokker-Planck equations in steady recirculating flows involving short fiber suspensions, Journal of Non-Newtonian Fluid Mechanics, **113/2-3** 97-125 (2003).

Extended Meshfree Method for Elastic and Inelastic Media

Jiun-Shyan Chen* and Dongdong Wang**

Department of Civil and Environmental Engineering, University of California, Los Angeles, CA 90095-1593, USA.

Abstract In this paper, an extended meshfree method [9] for solving elastic boundary value problems is summarized, and its extension to the elasto-plasticity problem is presented. In extended meshfree method, the solution is decomposed into particular solution and homogeneous solution. The particular solution without satisfying boundary conditions is obtained analytically, while a homogeneous problem with auxiliary boundary conditions is then solved under a Galerkin framework with moving least-squares reproducing kernel approximation. The proposed method for linear differential operator in Poisson, elasticity, and Mindlin-Reissner problems is first summarized. The extension to differential equations with nonlinear self-adjoint differential operator is then introduced, and the application to elasto-plasticity problem is presented. Numerical results of an elasto-plasticity problem demonstrate a significant accuracy gain in the solution of extended meshfree method compared to that of the conventional Galerkin meshfree approach.

1 Introduction

The naturally conforming properties of Galerkin meshfree methods offer tremendous flexibility for approximation of solution with arbitrary locality and smoothness [1, 4, 6, 11, 18, 19]. For example, extended finite element method with enriched local basis functions was used to model crack tip characteristics [3, 10]. Hierarchical enrichment [16, 17, 22], coupling between FEM and meshfree methods [13, 15, 17], and interface enrichment [14, 24] were also developed for enhancement of local solutions. Partition of unity and generalized finite element method [1, 2, 11, 12, 19, 20, 21] provide a general framework for enrichment of solution with special basis functions to simultaneously achieve local and global solution accuracy.

Alternative to the aforementioned methods where the degrees of freedom associated with the global and enriched solutions are obtained concurrently,

* jschen@seas.ucla.edu
** ddwang@seas.ucla.edu

an extended meshfree method [9] has been proposed in which the solution is solved in 2 sequential steps. In this approach, the solution is decomposed into particular solution and homogeneous solution. The particular solution is solved by satisfying the differential equation containing the source term in an infinite domain without the imposition of boundary conditions. This particular solution can often be obtained analytically. When such particular solution in an infinite domain is available, the original problem can be reduced to a homogeneous problem with auxiliary boundary conditions (corrected by the particular solution) that is then solved numerically. It has been shown that if a linearly complete approximation function in conjunction with a stabilized conforming nodal integration for domain integration of the weak form are employed in the Galerkin approximation of PDE's, linear exactness and bending exactness [7, 8, 23] can be achieved in the numerical solution of potential and plate problems, respectively. These properties are particularly useful for solving homogeneous solution in solid and structural problems, where substantially higher solution accuracy can be obtained compared to the conventional Galerkin meshfree formulation.

In this paper, we first review the basic concept and solution procedures of extended meshfree method in Section 2. In this section, a general and straightforward approach is presented for constructing the particular solution by employing a fundamental solution of the differential operator without satisfying the boundary conditions. Consequently, only a homogeneous governing equation with auxiliary boundary conditions is solved by a Galerkin meshfree method with stabilized conforming nodal integration. The specific extended meshfree formulations for problems with linear differential operator such as Poisson, elasticity, and Mindlin-Reissner problems are discussed in Section 3. In section 4, we first discuss extended meshfree method for PDE's with nonlinear self-adjoint differential operator. The application to elasto-plasticity problem and some comments on numerical procedures are then presented. Several numerical examples are demonstrated in Sections 3 and 4. Concluding remarks are given in Section 5.

2 Review of Extended Meshfree Method

2.1 Moving Least-Square Reproducing Kernel (MLS/RK) Approximation

The MLS/RK approximation [18] has been widely used in the meshfree approximation of the unknown variables in differential equations. The problem domain Ω is first discretized into a set of poins $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{NP}\}$, where $\mathbf{x}_I$ is the location of node I and NP denotes the total number of points. The unknown variable $u(\mathbf{x})$ of a differential equation is approximated by:

$$u^h(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x}) d_I \tag{2.1}$$

where $u^h(\mathbf{x})$ is the approximation of $u(\mathbf{x})$, Ψ_I and d_I are the shape functions and their associated coefficients, respectively. In MLS/RK approximation [18], a shape function $\Psi_I(\mathbf{x})$ takes the form:

$$\Psi_I(\mathbf{x}) = \left(\sum_{i+j=0}^{n} (x_1 - x_{1I})^i (x_2 - x_{2I})^j \bar{b}_{ij}(x) \right) \phi_a(\mathbf{x} - \mathbf{x}_I) \tag{2.2}$$

where $\phi_a(\mathbf{x}-\mathbf{x}_I)$ is a kernel function that defines the smoothness and locality of the approximation with a compact support $\omega_I = \text{supp}(\phi_a(\mathbf{x}-\mathbf{x}_I))$, where a is the radius of ω_I, and $\cup_{I=1}^{NP} \omega_I \supset \bar{\Omega}$. The coefficient vector $\bar{\mathbf{b}}(\mathbf{x})$ in Eq. (2.2) is obtained by satisfying the following n-th order reproducing conditions:

$$\sum_{I=1}^{NP} \Psi_I(\mathbf{x}) x_{I1}^{\alpha} x_{I2}^{\beta} = x_1^{\alpha} x_2^{\beta}, \ \alpha + \beta = 0, 1, 2, \ldots, n \tag{2.3}$$

Upon solving $\bar{b}_{ij}(x)$ from Eq. (2.3), the MLS/RK shape functions is obtained:

$$\Psi_I(\mathbf{x}) = \{\mathbf{H}\}^T(\mathbf{0})\{\mathbf{M}\}^{-1}(\mathbf{x})\{\mathbf{H}\}(\mathbf{x} - \mathbf{x}_I)\phi_a(\mathbf{x} - \mathbf{x}_I) \tag{2.4}$$

where

$$\{\mathbf{H}\}^T(\mathbf{x} - \mathbf{x}_I) = \left\{ 1, x_1 - x_{I1}, x_2 - x_{I2}, (x_1 - x_{I1})^2, \ldots \ldots, (x_2 - x_{I2})^n \right\} \tag{2.5}$$

$$\{\mathbf{M}\}(\mathbf{x}) = \sum_{I=1}^{NP} \{\mathbf{H}\}(\mathbf{x} - \mathbf{x}_I)\{\mathbf{H}\}^T(\mathbf{x} - \mathbf{x}_I)\phi_{a_I}(\mathbf{x} - \mathbf{x}_I) \tag{2.6}$$

The reproducing kernel particle method (RKPM) is formulated by introducing the above approximation of unknown into the weak form of the differential equation $\mathcal{L}(u) + b = 0$ where $\mathcal{L}$ is the differential operator and b is the source term. The order of the solution is mainly dependent upon the form of source term and the differential operator. The following problem introduced in [9] is solved to illustrate the property of the RKPM using a stabilized conforming nodal integration for domain integration [7], denoted as SC-RKPM, in representing the behavior of a localized high-gradient source term.

$$\begin{aligned} &u_{,xx} + b(x) = 0 \text{ in } (0,1) \\ &b(x) = \begin{cases} \left(\frac{2}{\alpha^2} - 4[\frac{(x-0.5)}{\alpha}]^2\right) \exp\left(-[\frac{(x-0.5)}{\alpha}]^2\right) & \text{for } 0.42 \le x \le 0.58 \\ 0 \text{ otherwise} \end{cases} \\ &u(0) = 0, \ u(1) = 1 \end{aligned} \tag{2.7}$$

A comparison of the SC-RKPM solution using linear basis function with a 20-node discretization and exact solutions in Fig. 1 shows large differences in the vicinity of the localized source term. Figures 2 (a) and (b) demonstrate the effects of discretization with nodal distance (*h= 0.0625, 0.0313, 0.0156*) and the source term localization (α *= 0.5, 0.3, 0.1*) on the error of solution

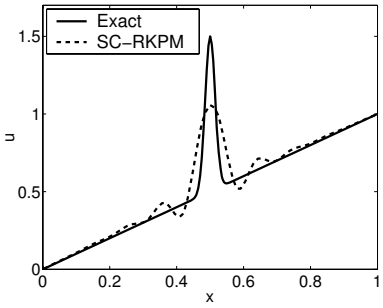
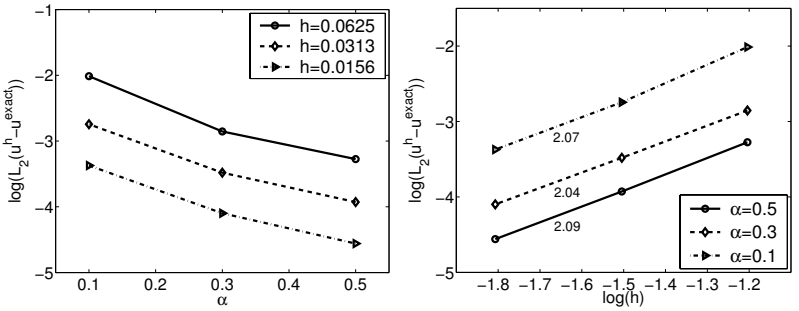


Figure 2.1. Solution comparison for problem (7)

accuracy using SC-RKPM. The results show that the L_2 error norm of u increases as the source term becomes more localized in all three discretizations. It is also shown that the level of localization α in the source term affects the constant of error in the numerical solution.



(a) (b)

Figure 2.2. Effects of discretization and source term localization on L_2 error norm of u in problem (2.7) using SC-RKPM

2.2 Extended Meshfree Method for Boundary Value Problems with Linear Differential Operator

Consider the following multi-dimensional boundary value problem (BVP):

$$\begin{aligned} &\mathcal{L}\left(\mathbf{u}(\mathbf{x})\right) + \mathbf{b}(\mathbf{x}) = \mathbf{0} \text{ in } \Omega \\ &B_u\left(\mathbf{u}(\mathbf{x})\right) = \mathbf{0} \text{ on } \partial_u\Omega \\ &B_h\left(\mathbf{u}(\mathbf{x})\right) = \mathbf{0} \text{ on } \partial_h\Omega \end{aligned} \tag{2.8}$$

where $\mathcal{L}$ is the linear differential tensor operator, $\mathbf{b}(\mathbf{x})$ is the source term tensor, B_u and B_h are the tensor operators of essential and natural boundary

conditions, respectively. Let $\boldsymbol{g}(\mathbf{x}, \bar{\mathbf{x}})$ be the fundamental solution tensor that satisfies the PDE in an *infinite domain* (i.e., without considering boundary conditions):

$$\mathcal{L}\left(\boldsymbol{g}(\mathbf{x}, \bar{\mathbf{x}})\right) + \boldsymbol{\delta}(\mathbf{x} - \bar{\mathbf{x}}) = 0 \text{ in } \mathbb{R}^3 \tag{2.9}$$

or in a matrix form

$$[\mathbf{L}]\left\{\begin{array}{c} g_{11} \\ g_{21} \\ g_{31} \end{array}\right\} + \left\{\begin{array}{c} \delta \\ 0 \\ 0 \end{array}\right\} = \mathbf{0}, \;\; [\mathbf{L}]\left\{\begin{array}{c} g_{12} \\ g_{22} \\ g_{32} \end{array}\right\} + \left\{\begin{array}{c} 0 \\ \delta \\ 0 \end{array}\right\} = \mathbf{0}, \;\; [\mathbf{L}]\left\{\begin{array}{c} g_{13} \\ g_{23} \\ g_{33} \end{array}\right\} + \left\{\begin{array}{c} 0 \\ 0 \\ \delta \end{array}\right\} = \mathbf{0} \tag{2.10}$$

where $\delta = \delta(\mathbf{x} - \bar{\mathbf{x}})$, $[\mathbf{L}]$ is the 3×3 matrix corresponding to matrix expression of $\mathcal{L}\left(\mathbf{g}(\mathbf{x}, \bar{\mathbf{x}})\right)$, and $\delta(\mathbf{x} - \bar{\mathbf{x}})$ is the delta function. Then a particular solution is obtained by the integral:

$$\mathbf{u}^p(\mathbf{x}) = \int_{\Omega} \boldsymbol{g}(\mathbf{x}, \mathbf{y}) \cdot \mathbf{b}(\mathbf{y}) d\mathbf{y} \tag{2.11}$$

where "$\cdot$" is a inner product. From Eqns. (2.9) and (2.11), it can be easily shown that the particular solution $\mathbf{u}^p$ satisfies

$$\mathcal{L}\left(\mathbf{u}^p(\mathbf{x})\right) + \mathbf{b}(\mathbf{x}) = \mathbf{0} \text{ in } \Omega \tag{2.12}$$

Note that in general $\mathbf{u}^p(\mathbf{x})$ *does not satisfy boundary conditions, i.e.,* $B_u(\mathbf{u}^p(\mathbf{x})) \neq \mathbf{0}$, $B_h(\mathbf{u}^p(\mathbf{x})) \neq \mathbf{0}$. The expressions of fundamental solutions and particular solutions for Poisson equation, elasticity, and Mindlin-Reissner plate problems are listed in Table 1. Next, the solution of the given problem is decomposed into

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}^p(\mathbf{x}) + \mathbf{u}^0(\mathbf{x}) \tag{2.13}$$

If $\mathcal{L}$ is the linear differential operator, we have the following condition:

$$\mathcal{L}\left(\mathbf{u}(\mathbf{x})\right) + \mathbf{b}(\mathbf{x}) = \mathcal{L}\left(\mathbf{u}^p(\mathbf{x})\right) + \mathcal{L}\left(\mathbf{u}^0(\mathbf{x})\right) + \mathbf{b}(\mathbf{x}) = \mathcal{L}\left(\mathbf{u}^0(\mathbf{x})\right) = \mathbf{0} \tag{2.14}$$

This leads to the following BVP for the homogeneous solution $\mathbf{u}^0(\mathbf{x})$ (assuming B_u and B_h are linear operators):

$$\begin{array}{l} \mathcal{L}\left(\mathbf{u}^0(\mathbf{x})\right) = \mathbf{0} \text{ in } \Omega \\ B_u\left(\mathbf{u}^0(\mathbf{x})\right) = -B_u\left(\mathbf{u}^p(\mathbf{x})\right) \text{ on } \partial_u\Omega \\ B_h\left(\mathbf{u}^0(\mathbf{x})\right) = -B_h\left(\mathbf{u}^p(\mathbf{x})\right) \text{ on } \partial_h\Omega \end{array} \tag{2.15}$$

Here the homogeneous solution $\mathbf{u}^0(\mathbf{x})$ is obtained from the above homogeneous equation with boundary conditions modified by $\mathbf{u}^p(\mathbf{x})$. A few examples of fundamental solutions [5, 25] for linear differential operator are given in Table 1.

Table 2.1. Particular Solutions of Several Boundary Value Problems

	Poisson Equation	Elasticity	Mindlin-Reissner Plate
Tensor Form $\mathcal{L}(\mathbf{u}) + \mathbf{b} = \mathbf{0}$	$\nabla^2 u + b = 0$	$\nabla \cdot (\mathbf{C} : \nabla^s \mathbf{u}) + \mathbf{b} = \mathbf{0}$ $(\nabla^s \mathbf{u}) = (\nabla \otimes \mathbf{u} + \mathbf{u} \otimes \nabla)/2$	$P_1(\nabla \cdot \nabla)\boldsymbol{\theta} + P_2 \nabla(\nabla \cdot \boldsymbol{\theta}) + Q(\nabla w - \boldsymbol{\theta}) = \mathbf{0}$ $Q(\nabla \cdot \nabla w - \nabla \cdot \boldsymbol{\theta}) + q = 0$
Matrix Form $[\mathbf{L}]\{\mathbf{u}\} + \{\mathbf{b}\} = \mathbf{0}$	$[\mathbf{L}] = \nabla^2$ $\{\mathbf{u}\} = u$ $\{\mathbf{b}\} = b$	$[\mathbf{L}] = \left[\hat{\nabla}\right]^T [\mathbf{D}] \left[\hat{\nabla}\right]$ $\{\mathbf{u}\} = \{ u_1, u_2, u_3 \}^T$ $\{\mathbf{b}\} = \{ b_1, b_2, b_3 \}^T$	$[\mathbf{L}] = \begin{Bmatrix} [\bar{\nabla}]^T [\mathbf{G}] + [\mathbf{D}^s][\mathbf{N}] \\ [\bar{\nabla}]^T [\mathbf{D}^s][\mathbf{N}] \end{Bmatrix}$ $\{\mathbf{u}\} = \{ \theta_1, \theta_2, w \}^T$ $\{\mathbf{b}\} = \{ 0, 0, q \}^T$
Fund. Sol. $\boldsymbol{g}(\mathbf{x}, \bar{\mathbf{x}})$	$g(\mathbf{x}, \bar{\mathbf{x}}) = -\frac{1}{2\pi} \ln r$ $r = \sqrt{\sum_{i=1}^{n} (x_i - \bar{x}_i)^2}$	$g_{ij}(\mathbf{x}, \bar{\mathbf{x}}) = \frac{1}{16\pi\mu(1-\nu)r} \times [(3-4\nu)\delta_{ij} + r_{,i} r_{,j}]$ $r_{,j} = \frac{\partial r}{\partial x_j}$	$g_{13}(\mathbf{x}, \bar{\mathbf{x}}) = \frac{1}{8\pi D} r(2 \ln r + 1) \cos(\mathbf{r}, \mathbf{x}_1)$ $g_{23}(\mathbf{x}, \bar{\mathbf{x}}) = \frac{1}{8\pi D} r(2 \ln r + 1) \sin(\mathbf{r}, \mathbf{x}_1)$ $g_{33}(\mathbf{x}, \bar{\mathbf{x}}) = \frac{1}{8\pi D} r^2 \ln r - \frac{1}{2\pi Q} \ln r$ $g_{ij} = 0 \quad for \quad j \neq 3$
Part.Sol. $\mathbf{u}^p(\mathbf{x})$	$u^p(\mathbf{x}) = \int_\Omega g(\mathbf{x}, \mathbf{y}) b(\mathbf{y}) d\mathbf{y}$	$u_i^p(\mathbf{x}) = \int_\Omega g_{ij}(\mathbf{x}, \mathbf{y}) b_j(\mathbf{y}) d\mathbf{y}$	$u_i^p(\mathbf{x}) = \int_\Omega g_{ij}(\mathbf{x}, \mathbf{y}) b_j(\mathbf{y}) d\mathbf{y}$

Note: P_1, P_2, Q, D, $\left[\hat{\nabla}\right]$, $[\bar{\nabla}]$, $[\mathbf{D}^s]$, $[\mathbf{G}]$, $[\mathbf{N}]$ in Table 1 are defined in Appendix.

2.3 A Stabilized Conforming Nodal Integration for Homogeneous Solution

The variable $\mathbf{u}^0(\mathbf{x})$ satisfying Eq. (2.15) without a source term involves a lower order behavior compared with the original problem and can be effectively solved numerically. It has been studied by Chen et al. [7, 8] that the necessary conditions for achieving linear exactness in the Galerkin approximation of second order differential equations are: (1) shape functions Ψ_I for approximation of the unknown $u_i^h(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x}) d_{iI}$ are linearly complete, and (2) integration of the weak form meets the following integration constraint:

$$\underset{\Omega}{Int}(\nabla \Psi_I) = \int_{\partial_h \Omega} \Psi_I \mathbf{n} \, d\Gamma \text{ for } \{I : supp(\Psi_I) \cap \partial_u \Omega = \emptyset\} \tag{2.16}$$

where $\underset{\Omega}{Int}(\cdot)$ denotes numerical integration over domain Ω and $\boldsymbol{n}$ is the surface normal on $\partial_h \Omega$. A stabilized conforming nodal integration (SCNI) has been proposed to stabilize the nodal integration of the weak form and fulfill linear exactness in the Galerkin approximation of second order differential equation

[7]. In this approach, a smoothed nodal gradient of u at point $\mathbf{x}_L$ is computed as:

$$\tilde{\nabla} u^h(\mathbf{x}_L) = \frac{1}{A_L} \int_{\Omega_L} \nabla u^h dA = \frac{1}{A_L} \int_{\Gamma_L} \mathbf{n} u^h d\Gamma = \sum_I \tilde{\nabla} \Psi_I(\mathbf{x}_L) d_I \tag{2.17}$$

where

$$\tilde{\nabla} \Psi_I(\mathbf{x}_L) = \frac{1}{A_L} \int_{\Gamma_L} \Psi_I(\mathbf{x}) \mathbf{n}(\mathbf{x}) d\Gamma \tag{2.18}$$

Here Ω_L and Γ_L are the representative domain and boundary associated with the node L as shown in Fig. 3, and A_L is the area (or volume) of Ω_L. It can be easily shown that this smoothed gradient meets the integration constraints for nodal integration:

$$\underset{\Omega}{Int}(\tilde{\nabla} \Psi_I) = \sum_{L=1}^{NP} \tilde{\nabla} \Psi_I(\mathbf{x}_L) A_L = \int_{\partial_h \Omega} \Psi_I \mathbf{n} \, d\Gamma \text{ for } \{I : supp(\Psi_I) \cap \partial_u \Omega = \emptyset\} \tag{2.19}$$

This SCNI method will be employed in the integration of weak form of the homogeneous solution for Poisson, elasticity, and Mindlin-Reissner problems in the next Section.

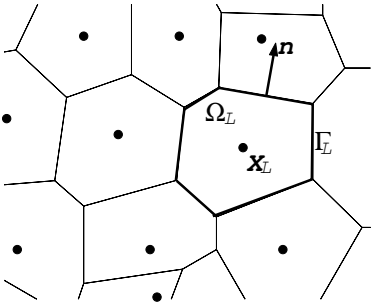


Figure 2.3. Nodal representative domain obtained by Voronoi diagram

3 Extended Meshfree Method for Elastic Boundary Value Problems

In this section we discuss the extended meshfree solution procedure when the particular solutions are obtained in Section 2. The construction of discrete equations for the homogeneous solution of Poisson, elasticity, and Mindlin-Reissner plate problems is also summarized. To distinguish from the tensor notation, matrix and vector are denoted by "$[\cdot]$" and "$\{\cdot\}$" symbols, respectively, in the following discussion.

3.1 Poisson Problem

Consider a Poisson problem in Table 1 with boundary conditions $u = g$ on $\partial_u\Omega$ and $\partial u/\partial n = h$ on $\partial_h\Omega$. The weak form of the homogeneous solution of Poisson problem is:

$$\begin{aligned} &\int_{\Omega} \nabla\delta u^0 \cdot \nabla u^0 d\Omega = \int_{\partial_h\Omega} \delta u^0 \left(h - \tfrac{\partial u^p}{\partial n}\right) d\Gamma \\ &u^0 = g - u^p \quad \text{on} \quad \partial_u\Omega \end{aligned} \tag{3.1}$$

Introducing a nodal integration of weak form with assumed gradient field (2.17) into Eq. (3.1) yields:

$$\sum_{L=1}^{NP} \tilde{\nabla}\delta u^0(\mathbf{x}_L) \cdot \tilde{\nabla} u^0(\mathbf{x}_L) A_L = \sum_{K=1}^{NB_{int}} \delta u^0(\bar{\mathbf{x}}_K) \left(h(\bar{\mathbf{x}}_K) - \frac{\partial u^p}{\partial n}(\bar{\mathbf{x}}_K)\right) \varpi_K \tag{3.2}$$

where NP and NB_{int} are the numbers of nodal points and boundary integration points, respectively, and $\bar{\mathbf{x}}_L$ and ϖ_K are the integration points and weights for boundary integration, respectively, which are consistent with the boundary integration for gradient smoothing term $\int_{\Gamma_L} \Psi_I(\mathbf{x})\mathbf{n}(\mathbf{x})d\Gamma$ in Eq. (2.18). By substituting the meshfree approximation for $u^{0^h}(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x}) d_I^0$ into Eq. (3.2), we obtain the following discrete equation for homogeneous solution:

$$[\mathbf{K}]\left\{\mathbf{d}^0\right\} = \left\{\mathbf{f}^0\right\} \tag{3.3}$$

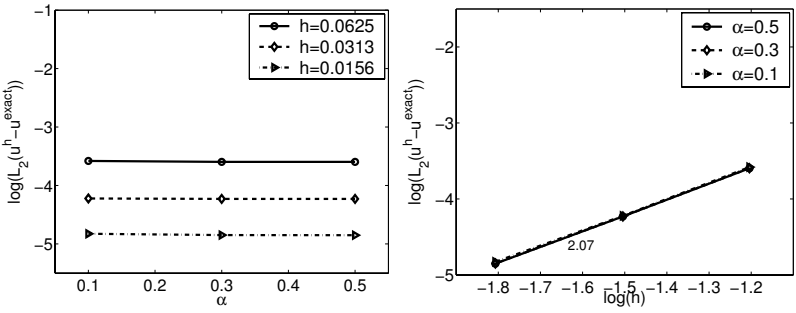
$$K_{IJ} = \sum_{L=1}^{NP} \left\{\tilde{\mathbf{b}}_I(\mathbf{x}_L)\right\}^T \left\{\tilde{\mathbf{b}}_J(\mathbf{x}_L)\right\} A_L \tag{3.4}$$

$$\left\{\tilde{\mathbf{b}}_I(\mathbf{x}_L)\right\} = \begin{Bmatrix} \tilde{b}_{1I}(\mathbf{x}_L) \\ \tilde{b}_{2I}(\mathbf{x}_L) \end{Bmatrix} \tag{3.5}$$

$$\tilde{b}_{iI}(\mathbf{x}_L) = \frac{1}{A_L} \int_{\Gamma_L} \Psi_I(\mathbf{x}) n_i(\mathbf{x}) d\Gamma \tag{3.6}$$

$$f_I^0 = \sum_{K=1}^{NBint} \Psi_I(\bar{\mathbf{x}}_K) \left(h(\bar{\mathbf{x}}_K) - \frac{\partial u^p}{\partial n}(\bar{\mathbf{x}}_K)\right) \varpi_K \tag{3.7}$$

Problem (2.7) is revisited using the aforementioned Extended-RKPM. The particular solution of this problem in Table 1 is computed using three integration zones with 6-point quadrature rule in the small domain where $b(x) \neq 0$, and the homogeneous solution is obtained by RKPM with SCNI. The comparison of the results in Fig. 4 shows a significant improvement in solution accuracy using the Extended-RKPM compared to SC-RKPM in Fig. 3. Note that since the particular solution is obtained analytically, the solution error is independent to the source term localization parameter α.



(a) (b)

Figure 3.4. Effects of discretization and source term localization on L_2 error norm of u in problem (2.7) using Extend-RKPM

3.2 Elasticity

Consider an elasticity problem in Table 1 with boundary conditions $u_i = g_i$ on $\partial_u \Omega$ and $(\mathbf{C} : (\nabla^s \mathbf{u})) \cdot \mathbf{n} = \mathbf{h}$ on $\partial_h \Omega$. The weak form of the homogeneous solution of elasticity problem is:

$$\begin{aligned} &\int_{\Omega} \left(\nabla^s \delta \mathbf{u}^0\right) : \mathbf{C} : \left(\nabla^s \mathbf{u}^0\right) d\Omega = \int_{\partial_h \Omega} \delta \mathbf{u}^0 \cdot \left(\mathbf{h} - (\mathbf{C} : (\nabla^s \mathbf{u}^p)) \cdot \mathbf{n}\right) d\Gamma \\ &u_i^0 = g_i - u_i^p \quad \text{on} \quad \partial_u \Omega \end{aligned} \tag{3.8}$$

Similar to the smoothing of ∇u^0 in Poisson problem, the strain corresponding to $\mathbf{u}^0$ is smoothed in elasticity. Let

$$\varepsilon_{ij}^0 = \left(\nabla^s \mathbf{u}^0\right)_{ij} = \frac{1}{2}\left(u_{i,j}^0 + u_{j,i}^0\right) \tag{3.9}$$

The smoothed strain of $\mathbf{u}^0$ at point $\mathbf{x}_L$ is obtained by

$$\begin{aligned} \tilde{\varepsilon}_{ij}^0(\mathbf{x}_L) &= \tfrac{1}{A_L} \int_{\Omega_L} \varepsilon_{ij}^0 dA = \tfrac{1}{2A_L} \int_{\Omega_L} \left(u_{i,j}^0 + u_{j,i}^0\right) dA \\ &= \tfrac{1}{2A_L} \int_{\Gamma_L} \left(u_i^0 n_j + u_j^0 n_i\right) d\Gamma \end{aligned} \tag{3.10}$$

Defining a smoothed strain vector $\left\{\tilde{\boldsymbol{\varepsilon}}^0\right\} = \left\{\tilde{\varepsilon}_{11}^0, \tilde{\varepsilon}_{22}^0, 2\tilde{\varepsilon}_{12}^0\right\}^T$ and introducing MLS/RK approximation for $u_i^{0^h}(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x}) d_{iI}^0$ in Eq. (3.10), we have

$$\left\{\tilde{\boldsymbol{\varepsilon}}^{0^h}(\mathbf{x}_L)\right\} = \sum_{I=1}^{NP} \left[\tilde{\mathbf{B}}_{\mathrm{I}}(\mathbf{x}_L)\right] \left\{\mathbf{d}_I^0\right\} \tag{3.11}$$

where

$$\left[\tilde{\mathbf{B}}_I(\mathbf{x}_L)\right] = \begin{bmatrix} \tilde{b}_{I1}(\mathbf{x}_L) & 0 \\ 0 & \tilde{b}_{I2}(\mathbf{x}_L) \\ \tilde{b}_{I2}(\mathbf{x}_L) & \tilde{b}_{I2}(\mathbf{x}_L) \end{bmatrix}, \quad \left\{\mathbf{d}_I^0\right\} = \begin{Bmatrix} d_{1I}^0 \\ d_{2I}^0 \end{Bmatrix} \tag{3.12}$$

and $\tilde{b}_{iI}(\mathbf{x}_L)$ is defined in Eq. (3.6). Introducing a nodal integration to the weak form of elasticity in Eq. (3.8) with smoothed strain field defined in Eq. (3.10) yields the following stiffness matrix and force vector for solving the homogeneous solution $[\mathbf{K}]\left\{\mathbf{d}^0\right\} = \left\{\mathbf{f}^0\right\}$:

$$[\mathbf{K}_{IJ}] = \sum_{L=1}^{NP} \left[\tilde{\mathbf{B}}_I(\mathbf{x}_L)\right]^T [\mathbf{D}] \left[\tilde{\mathbf{B}}_J(\mathbf{x}_L)\right] A_L \tag{3.13}$$

$$\left\{\mathbf{f}_I^0\right\} = \sum_{K=1}^{NB_{int}} \Psi_I(\bar{\mathbf{x}}_K) \left\{ \begin{array}{l} h_1(\bar{\mathbf{x}}_K) - \sigma_{1j}^p(\bar{\mathbf{x}}_K) n_j(\bar{\mathbf{x}}_K) \\ h_2(\bar{\mathbf{x}}_K) - \sigma_{2j}^p(\bar{\mathbf{x}}_K) n_j(\bar{\mathbf{x}}_K) \end{array} \right\} \varpi_K \tag{3.14}$$

where $\boldsymbol{\sigma}^p = \mathbf{C} : \nabla^s \mathbf{u}^p$, and $[\mathbf{D}]$ is the matrix form of the elasticity tensor $\mathbf{C}$.

A cantilever beam subjected to a tip shear force shown in Fig. 5(a) is analyzed using SC-RKPM (RKPM with stabilized conforming nodal integration), G-RKPM (RKPM with Gauss integration) and Extended-RKPM. The geometry and material properties are; L=10, D=2, E=21MPa, ν=0.3. The total shear load is 1, which is distributed in a parabolic function over the depth of the free end. Three discretizations as shown in Fig. 5(b) are employed to compute error in the numerical solution. The SCNI is employed for the total solution in SC-RKPM and the homogeneous solution of Extended-RKMP as discussed above. In Extended-RKPM, the shear force is treated as a source term with distributed load, and the fundamental solution in Table 1 is used to calculate the particular solution. Figure 5(c) shows that the Extended-RKPM greatly reduces the L_2 error norm of $\mathbf{u}$ compared to those of SC-RKPM and G-RKPM.

3.3 Mindlin-Reissner Plate Problem

Consider a Mindlin-Reissner plate problem in Table 1 with boundary conditions $\theta_\alpha = \bar{\theta}_\alpha$ $(\alpha = 1, 2)$, $w = \bar{w}$ on $\partial_u \Omega$ and $\left(\mathbf{C}^b : (\nabla^s \boldsymbol{\theta})\right) \cdot \mathbf{n} = \bar{\mathbf{m}}$ on $\partial_h \Omega$. The weak form of the homogeneous part of Mindlin-Reissner problem given in Table 1 can be expressed as

$$\begin{array}{l} \int_\Omega \nabla^s \delta\boldsymbol{\theta}^0 : \mathbf{C}^b : \nabla^s \boldsymbol{\theta}^0 d\Omega + \int_\Omega \left(\nabla \delta w^0 - \delta\boldsymbol{\theta}^0\right) : \mathbf{C}^s : \left(\nabla w^0 - \boldsymbol{\theta}^0\right) d\Omega \\ + \int_{\partial_h \Omega} \delta\boldsymbol{\theta}^0 \cdot \left(\bar{\mathbf{m}} - \left(\mathbf{C}^b : (\nabla^s \boldsymbol{\theta}^p)\right) \cdot \mathbf{n}\right) d\Gamma = 0 \\ \theta_\alpha = \bar{\theta}_\alpha, \quad w = \bar{w}, \quad \text{on} \quad \partial_u \Omega \end{array} \tag{3.15}$$

where $\mathbf{C}^b$ and $\mathbf{C}^s$ are the elasticity moduli for bending and shear, respectively. To meet integration constraint and provide stability to the nodally integrated weak form of Eq. (3.15), a curvature smoothing at the nodal point is introduced [24]. Let curvature of the homogeneous solution be denoted as

$$k_{\alpha\beta}^0 = \left(\nabla^s \theta^0\right)_{\alpha\beta} = \frac{1}{2}\left(\theta_{\alpha,\beta}^0 + \theta_{\beta,\alpha}^0\right) \tag{3.16}$$

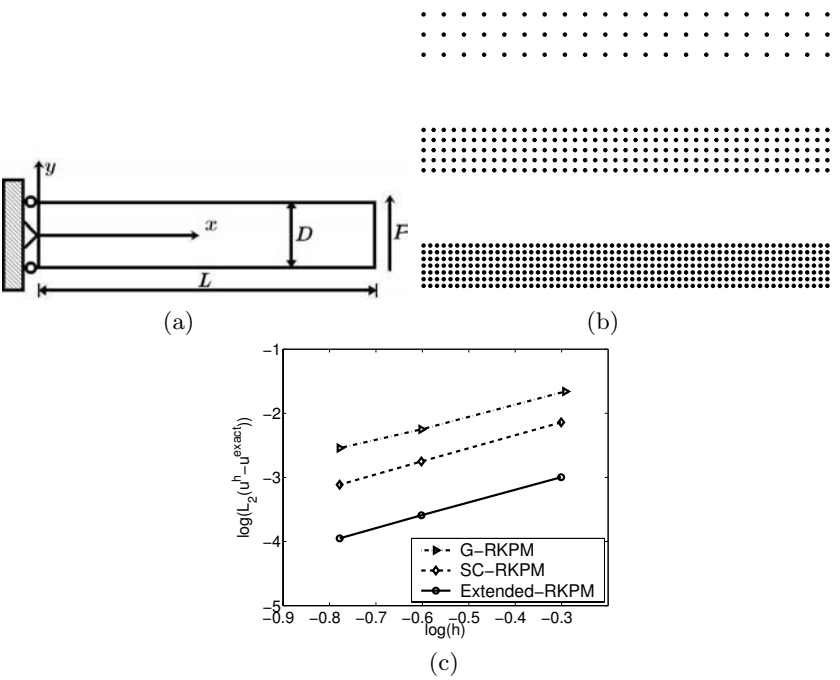


Figure 3.5. Beam problem solved by various methods: (a) problem description; (b) discretizations; (c) L_2 error norm of u

The smoothed curvature at point $\mathbf{x}_L$ is obtained by

$$\begin{aligned}\tilde{\kappa}^0_{\alpha\beta}(\mathbf{x}_L) &= \tfrac{1}{A_L}\int_{\Omega_L}\kappa^0_{\alpha\beta}(\mathbf{x})dA = \tfrac{1}{2A_L}\int_{\Omega_L}(\theta^0_{\alpha,\beta}+\theta^0_{\beta,\alpha})dA \\ &= \tfrac{1}{2A_L}\int_{\Gamma_L}(\theta^0_\alpha n_\beta+\theta^0_\beta n_\alpha)d\Gamma\end{aligned} \tag{3.17}$$

Introducing MLS/RK to the approximation of $\theta^{0^h}_\alpha(\mathbf{x}) = \sum_{I=1}^{NP}\Psi_I(\mathbf{x})\theta^0_{\alpha I}$ and $w^{0^h}(\mathbf{x}) = \sum_{I=1}^{NP}\Psi_I(\mathbf{x})w^0_I$ in smoothed curvature $\left\{\tilde{\boldsymbol{\kappa}}^0\right\} = \left\{\tilde{\kappa}^0_{11}, \tilde{\kappa}^0_{22}, 2\tilde{\kappa}^0_{12}\right\}^T$ and shear strain $\left\{\boldsymbol{\gamma}^0\right\} = \left\{\, w^0_{,1} - \theta^0_1,\, w^0_{,2} - \theta^0_2 \,\right\}^T$, we have

$$\left\{\tilde{\boldsymbol{\kappa}}^{0^h}\right\} = \sum_{I=1}^{NP}\left[\tilde{\mathbf{B}}^b_I\right]\left\{\mathbf{d}^0_I\right\}, \quad \left\{\boldsymbol{\gamma}^{0^h}\right\} = \sum_{I=1}^{NP}\left[\mathbf{B}^s_I\right]\left\{\mathbf{d}^0_I\right\} \tag{3.18}$$

$$\left[\tilde{\mathbf{B}}^b_I\right] = \begin{bmatrix} 0 & \tilde{b}_{I1} & 0 \\ 0 & 0 & \tilde{b}_{I2} \\ 0 & \tilde{b}_{I2} & \tilde{b}_{I1} \end{bmatrix}, \quad \left[\mathbf{B}^s_I\right] = \begin{bmatrix} \Psi_{I,x} & -\Psi_I & 0 \\ \Psi_{I,y} & 0 & -\Psi_I \end{bmatrix}, \quad \left\{\mathbf{d}^0_I\right\} = \begin{Bmatrix} w^0_I \\ \theta^0_{1I} \\ \theta^0_{2I} \end{Bmatrix} \tag{3.19}$$

Introducing a nodal integration to weak form in Eq. (3.15) with smoothed curvature in Eq. (3.18) yields the following stiffness matrix and force vector for solving the homogeneous solution $[\mathbf{K}]\left\{\mathbf{d}^0\right\} = \left\{\mathbf{f}^0\right\}$:

$$[\mathbf{K}] = \left[\mathbf{K}^b\right] + [\mathbf{K}^s] \tag{3.20}$$

$$\begin{aligned} \left[\mathbf{K}^b_{IJ}\right] &= \sum_{L=1}^{NP} \left[\tilde{\mathbf{B}}^b_I(\mathbf{x}_L)\right]^T \left[\mathbf{D}^b\right] \left[\tilde{\mathbf{B}}^b_J(\mathbf{x}_L)\right] A_L \\ [\mathbf{K}^s_{IJ}] &= \sum_{L=1}^{NP} \left[\mathbf{B}^s_I(\mathbf{x}_L)\right]^T [\mathbf{D}^s] \left[\mathbf{B}^s_J(\mathbf{x}_L)\right] A_L \end{aligned} \tag{3.21}$$

$$\left\{\mathbf{f}^0_I\right\} = \sum_{K=1}^{NB_{int}} \Psi_I(\bar{\mathbf{x}}_K) \begin{Bmatrix} 0 \\ m^p_{1\alpha}(\bar{\mathbf{x}}_K) n_\alpha(\bar{\mathbf{x}}_K) - \bar{m}_1(\bar{\mathbf{x}}_K) \\ m^p_{2\alpha}(\bar{\mathbf{x}}_K) n_\alpha(\bar{\mathbf{x}}_K) - \bar{m}_2(\bar{\mathbf{x}}_K) \end{Bmatrix} \varpi_K \tag{3.22}$$

where $\mathbf{m}^p = \mathbf{C}^b : \nabla^s \boldsymbol{\theta}^p$, $\left[\mathbf{D}^b\right]$ and $[\mathbf{D}^s]$ are the matrix forms of $\mathbf{C}^b$ and $\mathbf{C}^s$ tensors, respectively.

A clamped circular plate subjected to a unit concentrated load P at the plate centroid shown in Fig. 6(a) is analyzed. Since the source term is due to a point load, the particular solution is the fundamental solution. RKPM with SCNI (SC-RKPM), RKPM with Gauss integration (G-RKPM) and Extended-RKPM discussed above using three discretizations shown in Fig. 6(b) are compared in Fig. 6 (c). A substantial improvement of the solution accuracy in Extended-RKPM over G-RKPM and SC-RKPM is achieved.

4 Extended Meshfree Method for Inelastic Boundary Value Problems

4.1 Self-Adjoint Operator

Here we consider BVP in Eq. (2.8), with $\mathcal{L}$ the differential operator for inelastic media which, in general, is a nonlinear operator, i.e., $\mathcal{L}\left(\mathbf{u}^0 + \mathbf{u}^p\right) \neq \mathcal{L}\left(\mathbf{u}^0\right) + \mathcal{L}\left(\mathbf{u}^p\right)$. To start, express the weak form of BVP as

$$\int_\Omega \mathbf{v} \cdot \mathcal{L}\left(\mathbf{u}\right) d\Omega + \int_\Omega \mathbf{v} \cdot \mathbf{b}\; d\Omega = 0 \tag{4.1}$$

where $\mathbf{v}$ is the weight function and $\mathbf{v} = \mathbf{0}$ on essential boundary $\partial_u \Omega$. If $\mathcal{L}$ is self-adjoint, then we have

$$\int_\Omega \mathbf{v} \cdot \mathcal{L}\left(\mathbf{u}\right) d\Omega = \int_\Omega \mathbf{u} \cdot \mathcal{L}\left(\mathbf{v}\right) d\Omega + \int_\Omega \sum_i \mathcal{B}_i\left(\mathbf{u}, \mathbf{v}\right) d\Gamma \tag{4.2}$$

where the 2^{nd} term on the right hand side of Eq. (4.2) is a boundary term resulting from integration by parts, and $\mathcal{B}_i$ are the corresponding boundary operators. Thus Eq. (4.1) can be expressed as

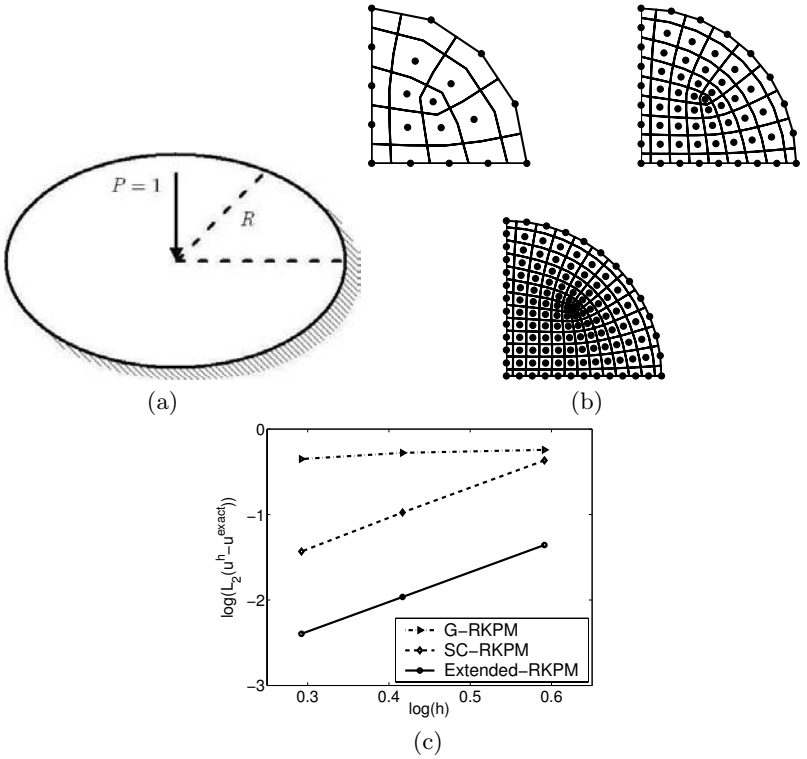


Figure 3.6. Mindlin-Reissner plate solved by various methods:(a) problem description; (b) discretizations; (c) L_2 error norm of $\mathbf{u}$

$$\int_{\Omega} \mathbf{u} \cdot \mathcal{L}(\mathbf{v}) d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathbf{b} d\Omega + \int_{\partial\Omega} \sum_i \mathcal{B}_i(\mathbf{u}, \mathbf{v}) d\Gamma = 0 \tag{4.3}$$

By deposition of unknown $\mathbf{u} = \mathbf{u}^p + \mathbf{u}^0$, we have

$$\int_{\Omega} \mathbf{u}^0 \cdot \mathcal{L}(\mathbf{v}) d\Omega + \int_{\Omega} \mathbf{u}^p \cdot \mathcal{L}(\mathbf{v}) d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathbf{b}\, d\Omega + \int_{\partial\Omega} \sum_i \mathcal{B}_i(\mathbf{u}, \mathbf{v}) d\Gamma = 0 \tag{4.4}$$

Since $\mathcal{L}$ is a self-adjoint differential operator, the first 2 terms on the left hand side of Eq. (4.4) can be transformed according to Eq. (4.2) to yield

$$\begin{aligned} &\int_{\Omega} \mathbf{v} \cdot \mathcal{L}\left(\mathbf{u}^0\right) d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathcal{L}(\mathbf{u}^p) d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathbf{b} d\Omega \\ &+ \int_{\partial\Omega} \sum_i (\mathcal{B}_i(\mathbf{v}, \mathbf{u}^0) + \mathcal{B}_i(\mathbf{v}, \mathbf{u}^p) + \mathcal{B}_i(\mathbf{u}, \mathbf{v})) d\Gamma \ = 0 \end{aligned} \tag{4.5}$$

If a particular solution for $\mathcal{L}(\mathbf{u}^p) + \mathbf{b} = \mathbf{0}$ exists, Eq. (4.5) is reduced to the following problem:

$$\int_{\Omega} \mathbf{v} \cdot \mathcal{L}(\mathbf{u}^0) d\Omega + \int_{\partial\Omega} \sum_i (\mathcal{B}_i(\mathbf{u}, \mathbf{v}) + \mathcal{B}_i(\mathbf{v}, \mathbf{u}^0) + \mathcal{B}_i(\mathbf{v}, \mathbf{u}^p)) d\Gamma = 0 \tag{4.6}$$

Equation (4.6) leads to the following strong form:

$$\mathcal{L}(\mathbf{u}^0) = \mathbf{0} \tag{4.7}$$

with natural boundary conditions:

$$\begin{aligned} &\sum_i (\mathcal{B}_i(\mathbf{u}^0 + \mathbf{u}^p, \mathbf{v}) + \mathcal{B}_i(\mathbf{v}, \mathbf{u}^0) + \mathcal{B}_i(\mathbf{v}, \mathbf{u}^p)) = \mathbf{0} \\ &\text{for arbitrary } \mathbf{v} \text{ with } \mathbf{v} = \mathbf{0} \text{ on } \partial_u \Omega \end{aligned} \tag{4.8}$$

For a given $\mathbf{u}^p$, Eq. (4.8) gives the natural boundary conditions for the homogeneous solution $\mathbf{u}^0$. Further, Eq. (4.5) provides the weak form for solving $\mathbf{u}^0$.

4.2 Elasto-plasticity Problem

Consider the following elasto-plasticity problem:

$$\begin{aligned} &\nabla \cdot \dot{\boldsymbol{\sigma}} + \dot{\mathbf{b}} = \mathbf{0} \quad \text{in} \quad \Omega_x \\ &\dot{\boldsymbol{\sigma}} = \mathbf{C}^e : \dot{\boldsymbol{\varepsilon}}^e = \mathbf{C}^* : \dot{\boldsymbol{\varepsilon}} \\ &\dot{\boldsymbol{\varepsilon}} = \nabla^s \dot{\mathbf{u}} = \dot{\boldsymbol{\varepsilon}}^e + \dot{\boldsymbol{\varepsilon}}^* \\ &\mathbf{u} = \hat{\mathbf{u}} \quad \text{on} \quad \partial_u \Omega_x, \\ &\dot{\boldsymbol{\sigma}} \cdot \mathbf{n} = \dot{\mathbf{h}} \quad \text{on} \quad \partial_h \Omega_x \end{aligned} \tag{4.9}$$

where Ω_x is the deformed configuration, $\mathbf{C}^e$ and $\mathbf{C}^*$ are elastic and elasto-plastic moduli, respectively, $\dot{\boldsymbol{\varepsilon}}^e$ and $\dot{\boldsymbol{\varepsilon}}^*$ are elastic and plastic strain rates, respectively, ∇^s is the symmetric gradient operator, i.e., $(\nabla^s \dot{\mathbf{u}})_{ij} = \dot{u}_{(i,j)} = (\dot{u}_{i,j} + \dot{u}_{j,i})/2$, $\dot{\boldsymbol{\sigma}}$ is the Cauchy stress rate, $\mathbf{n}$ is the normal unit vector of traction surface, and $\dot{\mathbf{h}}$ is the surface traction rate. For elasto-plastic materials that obey associative flow rule, we have:

$$\dot{\boldsymbol{\varepsilon}}^* = \dot{\gamma} \frac{\partial f}{\partial \boldsymbol{\sigma}} \tag{4.10}$$

where f is the yield function and $\dot{\gamma}$ denotes the plastic consistency parameter.

To employ the extended meshfree method discussed earlier, Eq. (4.9) is recast into the following form:

$$\underbrace{\nabla \cdot (\mathbf{C}^e : \nabla^s \dot{\mathbf{u}})}_{\mathcal{L}(\dot{\mathbf{u}})} + \underbrace{(\dot{\mathbf{b}} - \nabla \cdot (\mathbf{C}^e : \dot{\boldsymbol{\varepsilon}}^*)}_{\text{source}} = \mathbf{0} \tag{4.11}$$

Here the nonlinearity is included in the source term. The solution is solved as follows:

1. Particular solution

$$\dot{u}_i^p(\mathbf{x}) = \int_{\Omega_x} g_{ij}(\mathbf{x},\mathbf{y}) \left(\dot{b}_j(\mathbf{y}) - \frac{\partial (C^e_{jmkl} \dot{\varepsilon}^*_{kl}(\mathbf{y}))}{\partial y_m} \right) d\mathbf{y} \tag{4.12}$$

   where g_{ij} is the fundamental solution associated with the differential operator $\mathcal{L}(\dot{\mathbf{u}}) = \nabla \cdot (\mathbf{C}^e : \nabla^s \dot{\mathbf{u}})$, and the particular solution $\dot{u}_i^p$ is computed at every time step. The integration of Eq. (4.12) is performed over the deformed configuration Ω_x.
2. Homogeneous problem:

$$\begin{aligned} &\nabla \cdot (\mathbf{C}^e : \nabla^s \dot{\mathbf{u}}^0) = \mathbf{0} \quad \text{in } \Omega \\ &\mathbf{u}^0 = \hat{\mathbf{u}} - \mathbf{u}^p \ \text{ on } \ \partial_u \Omega, \\ &\dot{\boldsymbol{\sigma}}^0 \cdot \mathbf{n} = \dot{\mathbf{h}} - (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p) \cdot \mathbf{n} \ \text{ on } \ \partial_h \Omega \end{aligned} \tag{4.13}$$

   where $\dot{\boldsymbol{\sigma}}^0 = (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0) \cdot \mathbf{n}$ is the Cauchy stress rate associated with the homogeneous solution $\dot{\mathbf{u}}^0$.

Note that the homogeneous solution in Eq. (4.13) is almost identical to the homogeneous solution of elasticity, except for the corrected natural boundary condition involving elasto-plastic moduli. Thus the meshfree approximation and SCNI for domain integration used in elasticity are directly applicable to this case.

Remark 4.1. An alternative approach for solving problem (4.9) is to consider the following strong form:

$$\underbrace{\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}})}_{\mathcal{L}(\dot{\mathbf{u}})} + \dot{\mathbf{b}} = \mathbf{0} \tag{4.14}$$

Here, the elasto-plastic moduli $\mathbf{C}^*$ reside in the differential operator. We consider the weak form of Eq. (4.14)

$$\int_\Omega \mathbf{v} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}))\, d\Omega + \int_\Omega \mathbf{v} \cdot \dot{\mathbf{b}}\, d\Omega = 0 \tag{4.15}$$

where the weight function $\mathbf{v} = \mathbf{0}$ on $\partial\Omega_u$. Via integration by parts and divergence theorem, we have

$$\begin{aligned} &\int_{\partial_h \Omega} \mathbf{v} \cdot \dot{\mathbf{h}} d\Gamma - \int_{\partial_h \Omega} \dot{\mathbf{u}} \cdot (\nabla^s \mathbf{v} : \mathbf{C}^* \cdot \mathbf{n}) d\Gamma + \int_\Omega \dot{\mathbf{u}} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \mathbf{v})) d\Omega \\ &+ \int_\Omega \mathbf{v} \cdot \dot{\mathbf{b}} d\Omega = 0 \end{aligned} \tag{4.16}$$

Note that we have used the properties $C^*_{ijkl} = C^*_{klij} = C^*_{jikl} = C^*_{ijlk}$. Next, by the decomposition of $\dot{\mathbf{u}} = \dot{\mathbf{u}}^p + \dot{\mathbf{u}}^0$, we have

$$\begin{aligned}&\int_{\Omega} \dot{\mathbf{u}} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \mathbf{v}))d\Omega \\ &= \int_{\Omega} \dot{\mathbf{u}}^p \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \mathbf{v}))d\Omega + \int_{\Omega} \dot{\mathbf{u}}^0 \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \mathbf{v}))d\Omega\end{aligned} \tag{4.17}$$

Further applying integration by parts, the following equation can be obtained

$$\begin{aligned}&\int_{\Omega} \mathbf{v} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p))d\Omega + \int_{\Omega} \mathbf{v} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0))d\Omega + \int_{\partial_h \Omega} \mathbf{v} \cdot \dot{\mathbf{h}} d\Omega \\ &- \int_{\partial_h \Omega} \mathbf{v} \cdot ((\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p) \cdot \mathbf{n})d\Gamma + \int_{\Omega} \mathbf{v} \cdot \dot{\mathbf{b}} d\Omega = 0\end{aligned} \tag{4.18}$$

In the typical incremental solution procedure, $\mathbf{C}^*$ is formed with solution of the previous time step. Thus if the fundamental solution of differential operator in Eq. (4.14) with an explicit $\mathbf{C}^*$ exists, Eq. (4.18) can be reduced to

$$\int_{\partial_h \Omega} \mathbf{v} \cdot (\dot{\mathbf{h}} - (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p) \cdot \mathbf{n})d\Gamma + \int_{\Omega} \mathbf{v} \cdot (\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0))d\Omega = 0 \tag{4.19}$$

This leads to the following strong form and natural boundary conditions for the homogeneous solution:

$$\begin{aligned}&\nabla \cdot (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0) = \mathbf{0} \quad \text{in } \Omega \\ &(\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0) \cdot \mathbf{n} = \dot{\mathbf{h}} - (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p) \cdot \mathbf{n} \quad \text{on } \partial_h \Omega\end{aligned} \tag{4.20}$$

The corresponding weak form of Eq. (4.20) is

$$\int_{\Omega} \nabla^s \mathbf{v} : \mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^0 d\Omega = \int_{\partial_h \Omega} \mathbf{v} \cdot (\dot{\mathbf{h}} - (\mathbf{C}^* : \nabla^s \dot{\mathbf{u}}^p) \cdot \mathbf{n})d\Gamma \tag{4.21}$$

4.3 Some Remarks on Consideration of Large Deformation

In the case where the materials undergo large deformation, the undeformed configuration Ω_X and deformed configuration Ω_x of the material domain need to be distinguished. Considering a material point $\mathbf{X} \in \Omega_X$ under deformation is moved to $\mathbf{x} \in \Omega_x$, where Ω_X is the undeformed configuration and Ω_x is the corresponding deformed configuration, and $\mathbf{x} = \mathbf{X} + \mathbf{u}$, where $\mathbf{u}$ is the displacement. For a general nonlinear problem, incremental solution procedures are required, and the incremental equation of Eq. (4.11) is solved:

$$\underbrace{\nabla \cdot (\mathbf{C}^e : \nabla^s \Delta\mathbf{u})}_{\mathcal{L}(\Delta\mathbf{u})} + \underbrace{(\Delta\mathbf{b} - \nabla \cdot (\mathbf{C}^e : \Delta\varepsilon^*)}_{\text{source}} = \Delta\mathbf{r} \tag{4.22}$$

where "Δ" denote a finite increment quantity, and $\Delta\mathbf{r}$ is the residual. The increment of particular solution Δu_i^p is obtained by

$$\Delta u_i^p(x) = \int_{\Omega_x} g_{ij}(\mathbf{x},\mathbf{y}) \left(\Delta b_j - \left(C^e_{jmkl} \Delta \varepsilon^*_{kl} \right)_{,m} - \Delta r_j \right) (\mathbf{y}) d\mathbf{y} \tag{4.23}$$

where $g_{ij}(\mathbf{x},\mathbf{y})$ denotes the fundamental solution of the differential operator $\mathcal{L}(\Delta\mathbf{u}) = \nabla \cdot (\mathbf{C}^e : \nabla^s \Delta\mathbf{u})$. With the consideration of large deformation, the Galerkin weak form of the homogeneous problem of Eq. (4.22) is transformed to the undeformed configuration Ω_X, and a Lagrangian kernel [6] is used to approximate the homogeneous solution following 6:

$$\Delta u_i^0(\mathbf{X}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{X}) \Delta d_{iI}^0 \tag{4.24}$$

where $\Psi_I(\mathbf{X})$ is the MLS/RK shape function using Lagrangian kernel. Note that the solution of homogeneous solution in Eq. (4.22) requires the particular solution in Eq. (4.23). Since the transformation of weak form of the homogeneous solution in Eq. (4.22) to the undeformed configuration requires the particular solution to be evaluated in the unformed configuration, a mapping from $\mathbf{x}$ to $\mathbf{X}$ is needed. A stabilized conforming nodal integration (SCNI) that satisfies integration constraints in the Lagrangian Galerkin approximation of the homogeneous solution is employed in the domain integration of weak form in the undeformed configuration [8]. In SCNI, to meet integration constraints using nodal integration in Lagrangian Gakerkin formulation, the displacement spatial gradient $\Delta u_{i,j} = \partial u_i / \partial x_j$ at a nodal point X_L in the undeformed configuration is smoothed as:

$$\Delta \tilde{u}^0_{i,j}(\mathbf{X}_L) = \Delta \tilde{F}^0_{ik}(\mathbf{X}_L) \tilde{F}^{-1}_{kj}(\mathbf{X}_L) \tag{4.25}$$

where

$$\Delta \tilde{F}^0_{ij}(\mathbf{X}_L) = \frac{1}{A^X_L} \int\limits_{\Omega^X_L} \Delta F^0_{ij} d\Gamma = \frac{1}{A^X_L} \int\limits_{\Omega^X_L} \frac{\partial \Delta u^0_i}{\partial X_j} d\Gamma = \frac{1}{A^X_L} \int\limits_{\Gamma^X_L} (\Delta u^0_i N_j) d\Gamma \tag{4.26}$$

$$\tilde{F}_{ij}(\mathbf{X}_L) = \frac{1}{A^X_L} \int\limits_{\Omega^X_L} F_{ij} d\Gamma = \frac{1}{A^X_L} \int\limits_{\Omega^X_L} \frac{\partial u_i}{\partial X_j} d\Gamma + \delta_{ij} = \frac{1}{A^X_L} \int\limits_{\Gamma^X_L} (u_i N_j) d\Gamma + \delta_{ij} \tag{4.27}$$

Here N_i is the surface normal in the undeformed configuration, Ω^X_L and Γ^X_L are the nodal representative domain and boundary of the Voronoi cell associatd with node L in the undeformed configuration, respectively, and A^X_L is the area (volume) of Ω^X_L. The other details of Lagrangian meshfree method using SCNI for nonlinear problems can be found in [8]. Due to the absence of source term in the homogeneous problem, the Lagrangian meshfree formulation for solving the homogeneous solution can achieve much greater solution accuracy.

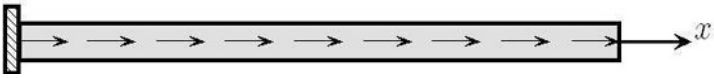


Figure 4.7. One dimensional rod subjected body force

4.4 Numerical Example

A one dimensional elasto-plastic bar subjected to a body force as shown in Fig. 7 is analyzed. In this example, the following isotropic hardening elasto-plasticity model is employed, where the yield function is expressed as:

$$f = |\sigma| - (\sigma_y + K\bar{\varepsilon}^p) \tag{4.28}$$

Here, σ_y is the initial yield stress and K is hardening parameter. The material

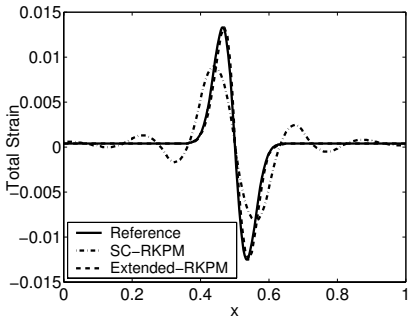


Figure 4.8. Total strain comparison of elasto-plasticity problem

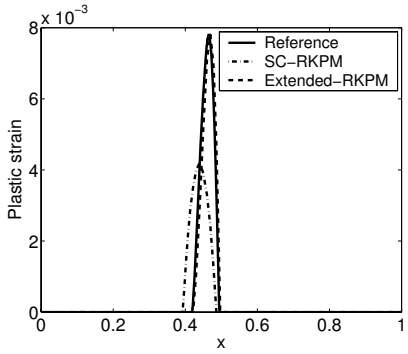


Figure 4.9. Plastic strain comparison of elasto-plasticity problem

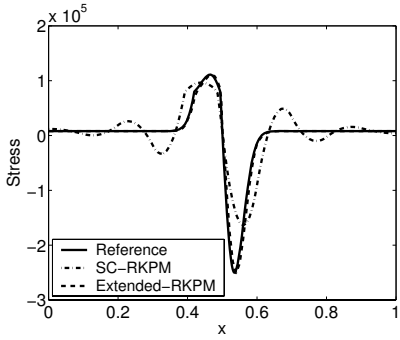


Figure 4.10. Stress comparison of elasto-plasticity problem

properties are: Young's modulus $E = 2.0 \times 10^7$, $K = 0.2E$ and $\sigma_y = 8.0 \times 10^4$. The body force function given in Eq. (2.7) is used, and the rod is fixed at the left end. Both SC-RKPM and the proposed Extended-RKPM are employed. The rod is discretized by only 8 nodes using both methods. For comparison purpose, a refined model with 51-node discretization is used to compute a reference solution. The numerical results for total strain, plastic strain and stress distributions are shown in Figs. 8, 9, and 10, respectively. A much enhanced solution is observed in the proposed new approach.

5 Conclusions

An extended meshfree method for elastic and inelastic media has been presented. In this approach, the total solution has been expressed by a combination of particular and homogeneous solutions. The particular solution is any analytical (or numerical) expression satisfying the governing differential equation containing the source term but not necessarily the boundary conditions. Thus, the problem is reduced to a homogeneous equation where the original boundary conditions are modified by the particular solution. In this paper, a general method has been presented for constructing the particular solution. The homogeneous solution has been solved numerically using Galerkin approximation. By employing moving least-square reproducing kernel (MLS/RK) approximation with linear completeness, as well as a stabilized conforming nodal integration (SCNI) for domain integration of the weak form, a linear exactness in solid continuum and a bending exactness in plate bending can be achieved in the homogeneous counterpart of these problems. Compared to the conventional Galerkin meshfree method, this new approach significantly reduces the constant in the error norms, and examples have been demonstrated in Poisson, elasticity, and Mindlin-Reissner plate problems.

The proposed method for differential equations with nonlinear differential operators has also been presented. It has been shown that if the differential operator is self-adjoint, a homogeneous problem can be obtained with standard integration by parts procedures. A Lagrangian MLS/RK approximation and the corresponding SCNI for Lagrangian weak form of large deformation problems have also been discussed. Using elasto-plasticity as a model problem, a simplified approach to obtain particular solution and to construct homogeneous problem has been presented. In this approach, the nonlinearity resides in the source term, and the differential operator takes the form of linear elasticity. Thus the fundamental solution is identical to that of linear elasticity, and the particular solution is obtained by integration of fundamental solution and the nonlinear source term at every incremental step in nonlinear calculation. The corresponding homogeneous problem has been constructed with boundary conditions corrected by the particular solution in every incremental step, and it has been solved using MLS/RK approximation with SCNI domain integration of the incremental weak form. The numerical results demonstrate a

substantially higher solution accuracy in extended meshfree method compared to that of the conventional Galerkin meshfree method.

6 Appendix

The parameters and matrices used Table 1 are defined as follows:

$$P_1 = \frac{Et^3}{24(1+\nu)},\ P_1 = \frac{Et^3}{24(1-\nu)},\ Q = \frac{5Et}{12\,(1+\nu)},\ D = \frac{Et^3}{12(1-\nu^2)} \tag{6.1}$$

$$\left[\hat{\nabla}\right] = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} & 0 \\ \frac{\partial}{\partial x_3} & 0 & \frac{\partial}{\partial x_1} \\ 0 & \frac{\partial}{\partial x_3} & \frac{\partial}{\partial x_2} \end{bmatrix},\ \left[\bar{\nabla}\right] = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix},\ \nabla = \begin{Bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \end{Bmatrix} \tag{6.2}$$

$$[\mathbf{N}] = \begin{bmatrix} -1 & 0 & \frac{\partial}{\partial x_1} \\ 0 & -1 & \frac{\partial}{\partial x_2} \end{bmatrix},\ [\mathbf{G}] = \left[-\mathbf{D}^b \left[\bar{\nabla}\right] \quad \{\mathbf{0}\} \right] \tag{6.3}$$

$$\left[\mathbf{D}^b\right] = \frac{Et^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix},\ [\mathbf{D}^s] = \frac{5}{6} t\mu \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.4}$$

Acknowledgement. The support of this work by NSF grant CMS-0296112 to UCLA is greatly acknowledged.

References

[1] I. Babuska, U. Banerjee and J.E. Osborn: Survey of meshless and generalized finite element methods: a unified approach. Acta Numerica, 1–125(2003).
[2] I. Babuska and J. M. Melenk: The partition of unity finite element method. Int. J. Numer. Meth. Eng., **40**,727–758(1997).
[3] T. Belytschko, N. Moes, S. Usui and C. Parimi: Arbitrary discontinuities in finite elements.Int. J. Numer. Meth. Eng., **50**, 993–1013(2001).
[4] T. Belytschko, Y.Y. Lu and L. Gu:Element-free Galerkin methods.Int. J. Numer. Meth. Eng., **37**,229–256(1994).
[5] C.A. Brebbia, J. Dominguez:Boundary elements: an introductory course. Computational mechanics publications, McGraw-Hill book company (1989).

[6] J.S. Chen, C. Pan, C.T. Wu and W.K. Liu: Reproducing kernel particle methods for large deformation analysis of nonlinear structures. Comput Meth. Appl. Mech. Engng., **139**, 195–227(1996).
[7] J.S. Chen, C.T. Wu, S. Yoon, and Y. You: A stabilized conforming nodal integration for Galerkin meshfree methods. Int. J. Numer. Meth. Eng., **50**, 435–466(2001).
[8] J.S. Chen, C.T. Wu and S. Yoon: Nonlinear version of stabilized conforming nodal integration for Galerkin meshfree methods. Int. J. Numer. Meth. Eng., **53**,2587–2615(2002).
[9] J.S. Chen, D. Wang and S.B. Dong: An extended meshfree method for boundary valu problems.Comput. Meth. Appl. Mech. Engng., **193**, 1085–1103(2004).
[10] J. Dolbow, N. Moes and T. Belytschko: Discontinuous enrichment in finite elements with a partition of unity method. Finite Elements Anal. Design,**36**,235–260(2000).
[11] C.A.M. Duarte and J.T. Oden: A h-p adaptive method using clouds. Comput.Meth. Appl. Mech. Engng., **139**,237–262(1996).
[12] C. A. M. Duarte, I. Babuska, and J. T. Oden: Generalized finite element method for three dimensional structural problems. Computer and Structures, **77**, 219–232(2000).
[13] A. Huerta and S. Fernández-Méndez: Enrichment and coupling of the finite element and meshless methods. Int. J. Numer. Meth. Eng., **48**, 1615–1636(2000).
[14] Y. Krongauz and T. Belytschko: EFG Approximation with discontinuous derivatives, Int. J. Numer. Meth. Eng., **41**, 1215–1233(1998).
[15] Y. Krongauz and T. Belytschko: Enforcement of essential boundary conditions in meshless approximations using finite elements. Comput. Meth. Appl. Mech. Engng., **131**, 133–145(1996).
[16] S. Li and W. K. Liu: Reproducing kernel hierarchical partition of unity part I: formulation and theory. Int. J. Numer. Meth., Eng., **45**,251–288 (1999).
[17] W.K. Liu, R.A. Uras and Y. Chen: Enrichment of the finite element method with the reproducing kernel particle method. Journal of Applied Mechanics, **64**, 861–870(1997).
[18] W.K. Liu, S. Jun, Y.F. Zhang: Reproducing kernel particle methods. Int. J. Numer. Meth. Fluids, **20** 1081–1106(1995).
[19] J.M. Melenk and I. Babuska: The partition of unity finite element method: basic theory and applications. Comput. Meth. Appl. Mech. Engng.,**139**, 289–314(1996).
[20] T. Strouboulis, I. Babuska, and K. Copps: The design and analysis of the generalized finite element method, Comput. Meth. Appl. Mech. Engng.,**181**, 43–69(2001).
[21] T. Strouboulis, K. Copps, and I. Babuska: The generalized finite element method. Comput. Meth. Appl. Mech. Engng., **190**, 4081–4193(2001).
[22] G.J. Wagner and W.K. Liu: Hierarchical enrichment for bridging scales and meshfree boundary conditions. Int. J. Numer. Meth. Eng., **50**, 507–524(2001).
[23] D. Wang and J.S. Chen: Locking-free stabilized conforming nodal integration for meshfree Mindlin-Reissner plate formulation. Comput. Meth. Appl. Mech. Engng., **193**, 1065–1083(2004).
[24] D. Wang, J.S. Chen and L. Sun: Homogenization of magnetostrictive particle-filled elastomers using an interface-enriched reproducing kernel particle method. Finite Element Anal. Design, **39**, 765–782(2003).

[25] Y. Wang: Boundary elements in engineering. China Hydraulic and Hydroelectric Press, Beijing (1995).

Meshfree Petrov-Galerkin Methods for the Incompressible Navier-Stokes Equations

Thomas-Peter Fries* and Hermann Georg Matthies**

Institute of Scientific Computing, Technical University of Braunschweig, D–38092 Braunschweig, Germany.

Abstract Meshfree stabilised methods are employed and compared for the solution of the incompressible Navier-Stokes equations in Eulerian formulation. These Petrov-Galerkin methods are standard tools in the FEM context, and can be used for meshfree methods as well. However, the choice of the stabilisation parameter has to be reconsidered. We find that reliable and successful approximation with standard formulas for the stabilisation parameter can only be expected for shape functions with small supports or dilatation parameters.

1 Introduction

Meshfree methods (MMs) for the incompressible Navier-Stokes equations are frequently used in Lagrangian formulations, i.e. as classical particle methods —with meshfree methods we refer here to methods based on the well-known Moving Least Squares (MLS) concept or the Reproducing Kernel Particle Method (RKPM) [6, 18, 19]. Here the particles (nodes) move with certain velocities through the domain. In this paper however, meshfree methods for fluid problems are analysed in an Eulerian formulation, where the particles are fixed throughout the calculation. This formulation has a number of advantages.

Eulerian meshfree methods do not have problems with boundary conditions in general, and particularly not with outflow boundary conditions, which are difficult to handle in Lagrangian approaches as particles leave the domain. Lagrangian methods also have problems with particle clustering, jeopardising the stability of meshfree methods, and adaptive refinement is difficult due to the large movement of the particles. Eulerian methods, in contrast, do not show these problems. However, the advantages of Eulerian methods come at a price: they require stabilisation in convection-dominated regimes, a frequently occuring situation in fluid mechanics [1, 10, 16].

* t.fries@tu-bs.de

** wire@tu-bs.de

In order for the implementation to be particularly convenient, we wish to employ equal-order interpolations for velocities and pressure. For the incompressible Navier-Stokes equations, this makes another stabilisation necessary to circumvent the Babuška-Brezzi condition [14, 23].

We review standard stabilisation schemes that are frequently used in the finite element (FEM) context [3]. We find that the structure of the stabilising terms can be used for MMs as well, leading to Petrov-Galerkin methods —in contrast to Bubnov-Galerkin methods the test functions are chosen differently from the shape functions. However, the stabilisation parameter τ which weighs the stabilisation terms requires special attention. Using the same formulas for τ as in the FEM context is not justified in general. We show that the standard formulas can only be expected to give satisfactory results for small dilatation parameters of the meshfree shape functions. For a more detailed discussion of stabilisation —particularly with regard to MMs— the interested reader is referred to [7].

The proposed standard stabilisations are applicable to the weak form of partial differential equations. It therefore appears natural that the employed MMs also solve the weak form rather than strong form, i.e. Galerkin conditions are used here rather than collocation.

The plan of the paper is as follows: In section 2 we review the standard stabilisation schemes, first for a scalar partial differential equation, and subsequently for the incompressible Navier-Stokes equation. In section 3 the stabilisation parameter is determined, by first recalling the situation of a one-dimensional advection-diffusion equation with linear finite elements, and subsequently proposing a new method first in one dimension, investigating the connection with the dilatation parameter, and proposing also a multi-dimensional variant. Numerical examples which show that the proposed stabilisation works are presented in section 4, ending with conclusions.

2 Stabilisation Schemes

The Streamline-Upwind Petrov-Galerkin (SUPG), the Pressure-Stabilising Petrov-Galerkin (PSPG), and the Galerkin Least-Squares (GLS) stabilisation methods are briefly recalled. The SUPG method stabilises oscillations in convection-dominated regimes, arising from the non-self-adjointness of the convective operator of a differential equation [1, 13]. PSPG stabilisation allows convenient equal-order interpolations in variational formulations with constraints, as is the case for the incompressible Navier-Stokes equations [15, 23]. This is accomplished by circumventing the Babuška-Brezzi condition, the relevant stability criterion of this class of problems. GLS methods stabilise oscillations in convection-dominated regimes as well as allowing equal-order interpolations [14, 16].

All these stabilisation methods are consistent Petrov-Galerkin methods resulting from a modification of the Bubnov-Galerkin test functions with suit-

able perturbations. The fundamental idea is to stabilise through a product of a perturbation and residual, weighted with a stabilisation parameter τ. The determination of a suitable τ will be the subject of the next section.

The stabilisation schemes are described first for general scalar partial differential equations, and subsequently for the incompressible Navier-Stokes equations.

2.1 Scalar Differential Equation

A general scalar differential equation may be written in strong form as

$$\mathcal{L}u\,(x) = f(x)\,, \quad x \in \Omega \subset \mathbb{R}^n$$

with suitable boundary conditions, where $\mathcal{L}$ is some differential operator, u the solution, and f the loading or excitation. Following the method of weighted residuals, the problem is discretised with an ansatz of $\tilde{u}\,(x) = \boldsymbol{N}^T\,(x)\,\boldsymbol{u} = \sum N_I\,(x)\,u_I$ —where $\boldsymbol{N}$ are either meshfree or mesh-based shape functions, and the resulting residual error is required to be orthogonal to the test functions;

$$\forall \boldsymbol{w}^\star : \quad \int_\Omega \boldsymbol{w}^\star\,(\mathcal{L}\tilde{u} - f)\;d\Omega = \boldsymbol{0}.$$

Choosing $\boldsymbol{w}^\star = \boldsymbol{N}$ leads to a Bubnov-Galerkin weighted residual method, whereas if any $\boldsymbol{w}^\star \neq \boldsymbol{N}$ —this holds for all stabilisation methods considered in this paper— the procedure is denoted as a Petrov-Galerkin method. SUPG and GLS stabilisation are defined as follows:

$$\begin{aligned} \text{SUPG} : \boldsymbol{w}^\star &= \boldsymbol{w} + \tau\mathcal{L}_{adv}\boldsymbol{w} \\ \text{GLS} : \boldsymbol{w}^\star &= \boldsymbol{w} + \tau\mathcal{L}\boldsymbol{w}, \end{aligned}$$

where $\boldsymbol{w}$ is the Bubnov-Galerkin weighting function, $\mathcal{L}_{adv}$ is the advection part of the whole operator $\mathcal{L}$, and τ is the stabilisation parameter. As there is no pressure as Lagrange multiplier, PSPG stabilisation is obviously not possible for this scalar problem.

2.2 Incompressible Navier-Stokes Equations

The incompressible Navier-Stokes equations for a Newtonian fluid are in strong form

$$\begin{aligned} \varrho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \boldsymbol{f}\right) - \nabla\cdot\boldsymbol{\sigma} &= \boldsymbol{0} \\ \nabla\cdot\boldsymbol{u} &= 0, \end{aligned}$$

$$\text{with } \boldsymbol{\sigma} = -p\boldsymbol{I} + \mu\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right),$$

where $\boldsymbol{u} = (u, v)$ is the velocity vector, p the pressure, $\boldsymbol{I}$ the identity tensor, μ the dynamic viscosity and ϱ the density. Assume that suitable boundary conditions are given. With an ansatz of $\tilde{\boldsymbol{u}}(\boldsymbol{x}) = \boldsymbol{N}^T(\boldsymbol{x})\,\boldsymbol{u}$ and $\tilde{p}(\boldsymbol{x}) = \boldsymbol{M}^T(\boldsymbol{x})\,\boldsymbol{p}$, the discretised stationary weak problem is

$$\int_\Omega \boldsymbol{w}^\star \cdot \left[\varrho\left(\tilde{\boldsymbol{u}} \cdot \nabla\tilde{\boldsymbol{u}} - \boldsymbol{f}\right) - \nabla \cdot \boldsymbol{\sigma}\left(\tilde{\boldsymbol{u}}, \tilde{p}\right)\right] d\Omega + \int_\Omega q\left[\nabla \cdot \tilde{\boldsymbol{u}}\right] = \boldsymbol{0}.$$

The Bubnov-Galerkin choice for the weighting functions $\boldsymbol{w}^\star$ and q leads to oscillations in convection-dominated regimes, and equal-order interpolations $\boldsymbol{M} = \boldsymbol{N}$ violate the Babuška-Brezzi condition [14, 23]. Both situations require stabilisation:

$$\begin{aligned}
\text{SUPG}: \quad & \boldsymbol{w}^\star = \boldsymbol{w} + \tau\tilde{\boldsymbol{u}} \cdot \nabla\boldsymbol{w}, \\
\text{PSPG}: \quad & \boldsymbol{w}^\star = \boldsymbol{w} + \tfrac{\tau}{\varrho}\nabla q, \\
\text{SUPG/PSPG}: \quad & \boldsymbol{w}^\star = \boldsymbol{w} + \tau\tilde{\boldsymbol{u}} \cdot \nabla\boldsymbol{w} + \tfrac{\tau}{\varrho}\nabla q, \\
\text{GLS}: \quad & \boldsymbol{w}^\star = \boldsymbol{w} + \tfrac{\tau}{\varrho}\left[\varrho\left(\tilde{\boldsymbol{u}} \cdot \nabla\boldsymbol{w}\right) - \nabla \cdot \boldsymbol{\sigma}\left(\boldsymbol{w}, q\right)\right],
\end{aligned}$$

where $\boldsymbol{w}$ and q are the usual Bubnov-Galerkin weighting functions.

One may choose individual τ for each stabilisation, however, in practice, we have $\tau_{\text{SUPG}} = \tau_{\text{PSPG}} = \tau_{\text{GLS}}$, which can be justified with numerical experiments and mathematical analysis. SUPG stabilises oscillations in convection-dominated regimes [1], and PSPG enables equal-order interpolations [23], whereas GLS stabilises both aspects [14, 16]. In the following SUPG/PSPG and GLS stabilisation are used respectively. The difference between these two formulations is in the modification of the test functions

$$\begin{aligned}
\text{SUPG/PSPG}: \quad & \tfrac{1}{\varrho}\nabla q \\
\text{GLS}: \quad & -\tfrac{1}{\varrho}\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{w}, q) = \tfrac{1}{\varrho}\nabla q - \tfrac{1}{\varrho}\nabla \cdot \left[\mu\left(\nabla\boldsymbol{w} + (\nabla\boldsymbol{w})^T\right)\right];
\end{aligned}$$

i.e. there are additional terms in the modification of the test functions in the GLS stabilisation, which result from the diffusion part.

3 The Stabilisation Parameter

Each of the stabilisation methods described in the previous section consists of two ingredients: The structure of the perturbation and the stabilisation parameter τ. It can easily be shown that the same arguments for the structure of the stabilisation schemes hold both for meshfree and mesh-based methods [10]. However, this is in general not true for the stabilisation parameter τ itself.

In the finite element context, there are several suggestions for the determination of τ in the literature, i.e. with the help of element matrix and vector norms [24], the Green's function of the element [12], mathematical error analysis [4, 5, 16], or model equations [2, 9, 17].

From mathematical analysis in the finite element context, one can find the following design criteria for the stabilisation parameter: $\tau > 0$ in general, $\tau = O\left(h^2/\mu\right)$ for low Peclet numbers $Pe = |\boldsymbol{c}|\, h/(2K)$, and $\tau = O\left(h/|\boldsymbol{c}|\right)$ for high Peclet numbers, where h is a measure of the node distribution, and μ and $|\boldsymbol{c}|$ are measures of the diffusion and convection respectively. A number of formulas that fulfil these basic requirements for the stabilisation parameter are available in the finite element context, see e.g. [11, 20].

The question of an 'optimal' stabilisation parameter τ requires an optimality criterion of the resulting approximation. Often the one-dimensional advection-diffusion equation is taken as a model equation. There, the exact solution is known, and enables one to calculate stabilisation parameters that fulfil any desired optimality criterion. An optimality criterion that has proven to be particularly useful is the one that obtains the *nodally exact solution* of the model equation. It can be shown that for linear FEM and a regular node distribution, the 'coth-formula'

$$\tau = \frac{\Delta x}{2c}\left(\coth\left(Pe\right) - \frac{1}{Pe}\right), \quad Pe = \frac{c\Delta x}{2K}, \quad \Delta x = x_i - x_{i-1} = \text{const}$$

fulfils this criterion and leads to nodally exact approximations. This formula has been generalised straightforward to multi-dimensions and is —together with similar versions— frequently used in practice for the successful stabilisation of arbitrary problems with linear FEM; and this although it is derived only from the special case of the one-dimensional advection-diffusion equation. It has been shown in [4, 5], that straightforward use of this formula for higher-order FEM is not justified in general, and requires some modifications. It may thus be presumed that using these standard formulas derived in the mesh-based context of the linear FEM is also not suitable for MMs in general.

The standard way to obtain the coth-formula is to analytically solve the resulting *difference* equations in the system of equations emanating from the weak form of the model equation, discretised with linear FEM. Then, this solution is equated with the analytical solution of the *differential* equation [2, 9, 17]. In the following, we present a new approach which does not require the analytical solution of difference equations. We find this approach particularly useful to determine nodally exact solutions of the one-dimensional advection-diffusion equation with arbitrary (not only linear) finite element interpolations and also with MMs.

3.1 One-Dimensional Advection-Diffusion Equation

The strong form of the one-dimensional advection-diffusion equation is

$$c\frac{\partial u}{\partial x} - K\frac{\partial^2 u}{\partial x^2} = 0, \tag{3.1}$$

with suitable boundary conditions. A scalar quantity $u\left(x\right)$ is advected with the velocity c and thereby experiences diffusion dependent on K. The exact

solution of this problem is known as $u^{(\mathrm{ex})}(x) = C_1 e^{\gamma \cdot x} + C_2$, with $\gamma = c/K$. Particles (nodes) are introduced at the positions $x_1, x_2, \ldots, x_n$ inside the domain. Discretisation of the SUPG stabilised weak form with $\tilde{u}(x) = \boldsymbol{N}^T(x)\,\boldsymbol{u}$ gives

$$\int_\Omega \left(\boldsymbol{w} + \tau c \cdot \partial_x \boldsymbol{w}\right)\left(c \cdot \partial_x \widetilde{u} - K\,\partial_x^2 \widetilde{u}\right)\,d\Omega = \boldsymbol{0},$$

where $\partial_x = \partial/\partial x$.

Let us extract one equation —say equation no. I— of this system of equations,

$$\left[\int_\Omega \left(w_I + \tau_I c \cdot \partial_x w_I\right)\left(c \cdot \partial_x \boldsymbol{N}^T - K\,\partial_x^2 \boldsymbol{N}^T\right)\,d\Omega\right]\boldsymbol{u} = \boldsymbol{0}. \tag{3.2}$$

This equation corresponds to node I at x_I with the test function w_I. There is one τ_I for each equation/node. Consequently, one may call this stabilisation *nodal stabilisation*, in contrast to element stabilisation —where stabilisation parameters τ_e for each element are used— which is standard in the FEM.

The τ_I-values of each equation are computed such that the nodally exact solution is obtained. This can be done by introducing the exact solution into the vector $\boldsymbol{u}$. We have $u^{(\mathrm{ex})}(x_j) = u_j^{(ex)} = C_1 e^{\gamma \cdot x_j} + C_2$, and according to the ansatz $\tilde{u}(x_j) = \tilde{u}_j = \sum N_i(x_j) u_i$. Nodal exactness means

$$\begin{aligned} \tilde{u}_j &= u_j^{(ex)}, \\ \sum N_i(x_j) u_i &= C_1 e^{\gamma \cdot x_j} + C_2, \\ \boldsymbol{D}\boldsymbol{u} &= \boldsymbol{u}^{(ex)}, \end{aligned}$$

where $\boldsymbol{D} = D_{ij} = N_i(x_j)$ is a $n \times n$ matrix of the n shape functions evaluated at the n nodal positions. $\boldsymbol{D}$ is a sparse matrix if the shape functions are non-zero only in small parts of the domain Ω. In the FEM the shape functions are non-zero only on local supports, specified indirectly with help of the mesh, whereas the supports of MMs are defined with help of the dilatation parameter ρ [6]. For shape functions with Kronecker-δ property, $N_i(x_j) = \delta_{ij}$ and thus $\boldsymbol{D} = \boldsymbol{I}$.

Rearranging Eq.(3.2) for τ_I and replacing $\boldsymbol{u}$ with $\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}$ results in

$$\begin{aligned} \tau_I &= -\frac{\left[\int_\Omega (w_I)\left(c \cdot \partial_x \boldsymbol{N}^T - K\,\partial_x^2 \boldsymbol{N}^T\right)\,d\Omega\right]\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega \left(c \cdot \partial_x w_I \partial_x\right)\left(c \cdot \partial_x \boldsymbol{N}^T - K\,\partial_x^2 \boldsymbol{N}^T\right)\,d\Omega\right]\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}} \\ &= -\frac{\left[\int_\Omega (w_I)\left(c \cdot \partial_x \boldsymbol{N}^T\boldsymbol{D}^{-1} - K\,\partial_x^2 \boldsymbol{N}^T\boldsymbol{D}^{-1}\right)\,d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega \left(c \cdot \partial_x w_I\right)\left(c \cdot \partial_x \boldsymbol{N}^T\boldsymbol{D}^{-1} - K\,\partial_x^2 \boldsymbol{N}^T\boldsymbol{D}^{-1}\right)\,d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}. \end{aligned} \tag{3.3}$$

In what follows, this result will be interpreted.

3.2 Linear FEM

In the case of linear finite element shape functions, a number of simplifications for Eq.(3.3) is possible. Due to the Kronecker-δ property of the nodal finite

element shape functions, we have $\boldsymbol{D} = \boldsymbol{D}^{-1} = \boldsymbol{I}$. Partial integration is applied to the diffusion term in the nominator, whereas this term cancels out in the denominator. It remains for τ_I (for constant c and K):

$$\begin{aligned}\tau_I &= -\frac{\left[c\int_\Omega w_I\,\partial_x \boldsymbol{N}^T d\Omega + K\int_\Omega \partial_x w_I\,\partial_x \boldsymbol{N}^T d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[c^2\int_\Omega \partial_x w_I\,\partial_x \boldsymbol{N}^T d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}} \\ &= -\frac{\left[\int_\Omega w_I\,\partial_x \boldsymbol{N}^T d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[c\int_\Omega \partial_x w_I\,\partial_x \boldsymbol{N}^T d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}} - \frac{K}{c^2}\,.\end{aligned}$$

The integral expressions can be evaluated explicitely for the case of linear shape and test functions and a regular node distribution as

$$\begin{aligned}\int_\Omega w_I\,\partial_x \boldsymbol{N}^T\,d\Omega &= \tfrac{1}{2}\left[-1,\,0,\,1\right]; \\ \int_\Omega \partial_x w_I \partial_x \boldsymbol{N}^T\,d\Omega &= \tfrac{1}{\Delta x}\left[-1,\,2,\,-1\right].\end{aligned}$$

The scalar product of these expressions with $\boldsymbol{u}^{(\mathrm{ex})} = C_1 e^{\gamma\cdot\boldsymbol{x}} + C_2$ gives

$$\begin{aligned}\tau_I &= \frac{\Delta x}{2c}\frac{(E_{I+1} - E_{I-1})}{(E_{I-1} - 2E_I + E_{I+1})} - \frac{K}{c^2} \\ &= \frac{\Delta x}{2c}\frac{\sinh(\gamma\Delta x)}{\cosh(\gamma\Delta x) - 1} - \frac{K}{c^2} \;=\; \frac{\Delta x}{2c}\left(\coth(Pe) - \frac{1}{Pe}\right), \qquad (3.4)\end{aligned}$$

with $E_J = C_1 e^{\gamma\cdot x_J} + C_2$ and $Pe = \gamma\cdot\Delta x/2 = c\cdot\Delta x/(2K)$. With this definition of the stabilisation parameter one obtains the nodally exact solution for the one-dimensional advection-diffusion equation, approximated with linear FEM and a regular node distribution. Using standard element stabilisation instead of nodal stabilisation with $\tau_e = \tau_I$ leads to the same result. This formula for τ has often be called 'optimal' in the literature [1, 2, 9, 17]. It has a *local* character as it is independent of the boundary conditions and only relies on the relative positions of the neighbouring nodes x_{I-1} and x_{I+1}.

3.3 Meshfree Methods

For meshfree methods, Eq.(3.3), can not be simplified in general. This result is interpreted as follows. Let us rewrite the expression for τ_I as

$$\tau_I = -\frac{\left[\int_\Omega f_1(w_I) g\left(\boldsymbol{N}^T\boldsymbol{D}^{-1}\right)\,d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega f_2(w_I) g\left(\boldsymbol{N}^T\boldsymbol{D}^{-1}\right)\,d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}},$$

where f_1, f_2 and g are linear functions of the test and shape functions respectively. The meshfree test and shape functions $\boldsymbol{w}$ and $\boldsymbol{N}$ have local supports. However, the term $\boldsymbol{N}^T\boldsymbol{D}^{-1}$ can be interpreted as the 'globalised' meshfree shape functions having Kronecker-δ property. This may be gleaned from Figure 3.1, where local shape functions $\boldsymbol{N}^T$ without and transformed global shape

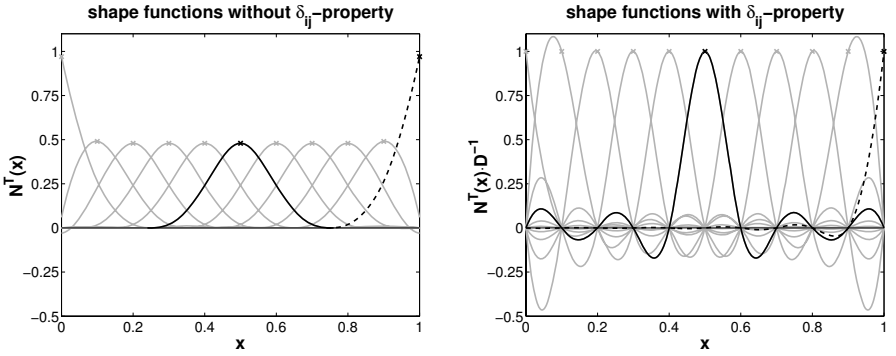


Figure 3.1. Local shape functions without and with Kronecker-δ property.

functions $\boldsymbol{N}^T\boldsymbol{D}^{-1}$ with Kronecker-δ property are depicted. It is well known [18], that locally defined meshfree shape functions with Kronecker-δ property require singular test functions in the MLS procedure. This however has a number of severe numerical disadvantages and is only rarely used in practice.

Consequently, the vector $\int_\Omega f_i(w_I) g\left(\boldsymbol{N}^T\boldsymbol{D}^{-1}\right) d\Omega$ is a full vector, which is in contrast to shape functions having Kronecker-δ property. In the latter case, $g\left(\boldsymbol{N}^T\boldsymbol{D}^{-1}\right) = g\left(\boldsymbol{N}^T\right)$, and the vector is sparse. Evaluating the scalar product with $\boldsymbol{u}^{(\mathrm{ex})}$ shows the important difference: Shape functions without Kronecker-δ property have non-zero entries in the scalar-product for *all* components of the vector $\boldsymbol{u}^{(\mathrm{ex})}$, whereas, in contrast, shape functions with Kronecker-δ property only have non-zero entries for the *neighbouring* nodes. This may be seen symbolically from Figure 3.2, where it is clear that the nodally exact τ_I for shape functions without Kronecker-δ property can only be obtained with a global criterion, because all entries of $\boldsymbol{u}^{(\mathrm{ex})}$ have an influence on the result.

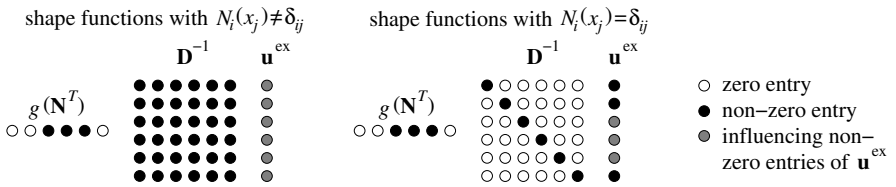


Figure 3.2. Evaluating the scalar products for τ.

Keeping in mind that $\boldsymbol{u}^{(\mathrm{ex})}$ is an exponential function, the scalar product will depend more and more on the last entry of this vector as the convection-diffusion ratio $\gamma = c/K$ grows, because then

$$u^{(\mathrm{ex})}(x_n) = u_n^{(\mathrm{ex})} \gg u_i^{(\mathrm{ex})} = u^{(\mathrm{ex})}(x_i) \qquad \forall i \neq n$$

The last component of $\boldsymbol{u}^{(\mathrm{ex})}$ is $u_n^{(\mathrm{ex})}$, and belongs to node n with the largest x-value, i.e. the global downstream node. We conclude that the stabilisation parameter τ_I, leading to nodally exact solutions has a *global* character, as it depends on all particle positions and for convection-dominated cases most importantly on the global downstream node. This is in contrast to shape functions with Kronecker-δ property, whose stabilisation relies on the neighbouring nodes only. Therefore, it can not be expected, that using the simple coth-formula —or other alternative similar versions derived as a local stabilisation criterion for linear FEM— is successful also for MMs.

3.4 Small Dilatation Parameters

Meshfree shape functions are constructed with help of the node distribution and the definition of supports [6]. The support size is defined by the dilatation parameter ρ. It is a well known fact, that MLS shape functions in one

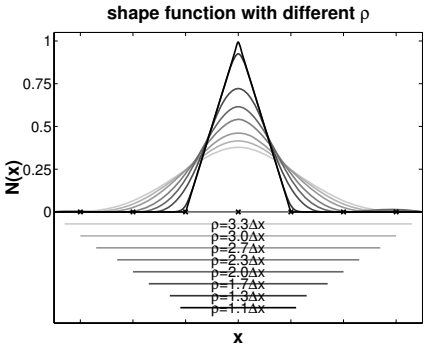


Figure 3.3. Meshfree shape function with varying dilatation parameters ρ.

dimension with first order consistency become more and more equal to the standard nodal linear shape functions of the FEM as the dilatation parameter ρ approaches Δx. This is also shown in Figure 3.3. Hence it may be concluded that when $\rho \longrightarrow \Delta x$, the coth-formula becomes more and more suited also for MMs. Hence

$$\rho \longrightarrow \Delta x \Rightarrow \tau_I^{(\mathrm{MM})} \longrightarrow \tau_I^{(\text{lin. FEM})} = \frac{\Delta x}{2c}\left(\coth\left(Pe\right) - \frac{1}{Pe}\right)$$

A stability criterion of the MLS requires $\rho > \Delta x$ [6]. Thus, one can never reach the limit $\rho = \Delta x$, where the coth-formula gives the nodally exact solution. We propose however, that for reasonable advection-diffusion ratios and 'small' dilatation parameters a successful stabilisation with standard formulas —derived for mesh-based methods— can be obtained. We suggest dilatation parameters of $1.3\Delta x \leq \rho \leq 1.7\Delta x$. For smaller ρ, the condition number of the MLS system of equations which has to be solved at every integration point

may be too large to allow a sufficiently accurate solution, and for larger ρ the stabilisation may not be reliable. The numerical results in section 4 confirm this assumption.

3.5 Stabilisation Parameter in Multi-Dimensions

In the FEM, i.e. in the mesh-based context, the generalisation of the τ-formulas derived from the one-dimensional advection-diffusion equation to multi-dimensions is straightforward [1]. The one-dimensional parameters Δx and c are replaced with the element length h_e and the advection $|\boldsymbol{c}|$. Assuming small dilatation parameters, the same generalisation is proposed for meshfree methods. Hence τ_I in multi-dimensions may be computed with

$$\tau_I = \frac{h_\rho}{2\,|\boldsymbol{c}|}\left(\coth\left(Pe_\rho\right) - \frac{1}{Pe_\rho}\right) \qquad \text{with } Pe_\rho = \frac{|\boldsymbol{c}|\,h_\rho}{2K},$$

or any other of the alternative version for τ. Here h_ρ is the 'support length', analogously to the 'element length' h_e in the mesh-based context. Figure 3.4

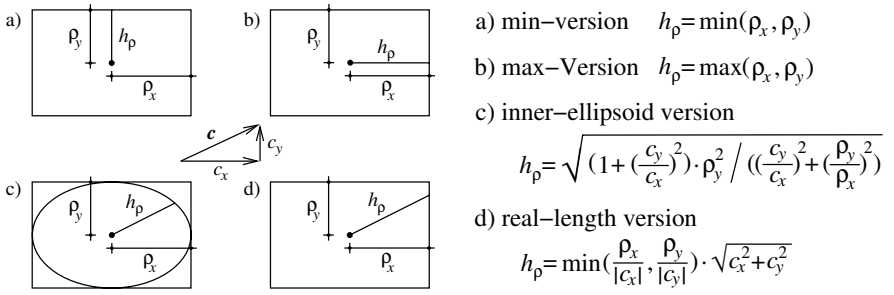


Figure 3.4. Different versions to compute the support length.

shows several possibilities to interpret h_ρ in case of rectangular supports. The support lengths for circular and ellipsoid supports can be directly read of from these formulas.

In case of the incompressible Navier-Stokes equations, the advection coefficients c_x and c_y are replaced by the velocities u and v. In the numerical experiments, we find that particularly the min-version works very successful also for large aspect ratios ($\rho_x/\rho_y \gg 1$, or $\rho_y/\rho_x \gg 1$). See Mittal [21] for an interesting parallel for high aspect *elements*: He also finds that the minimal edge length works better than other versions for h_e.

The inner-ellipsoid-version and the real-length-version are dependent on the streamline direction of the flow inside the support. In case of the incompressible Navier-Stokes equations, this introduces some disadvantages: A representative streamline-direction has to be found for the whole support, the

streamline direction changes with each iteration step, and the non-linearity introduced by $\tau = f(u, v, h_\rho)$ is more complex as compared with the min and max version.

4 Numerical Results

All results are computed with standard MLS shape functions with first order consistency.

4.1 1D Advection-Diffusion Equation

The one-dimensional advection-diffusion Eq.(3.1) is solved with 21 particles. The advection-diffusion ratio is $\gamma = c/K = 100$. Figure 4.5a) shows the unstabilised results for two different dilatation parameters $\rho = 1.3\Delta x$ ('small') and $\rho = 3.3\Delta x$ ('large'). It may be seen that higher dilatation parameters lead to more oscillations, simply due to their higher Peclet number, $Pe_\rho = c\rho/(2K)$. Clearly, for both cases, stabilisation is required. Figure 4.5b) shows the nodally

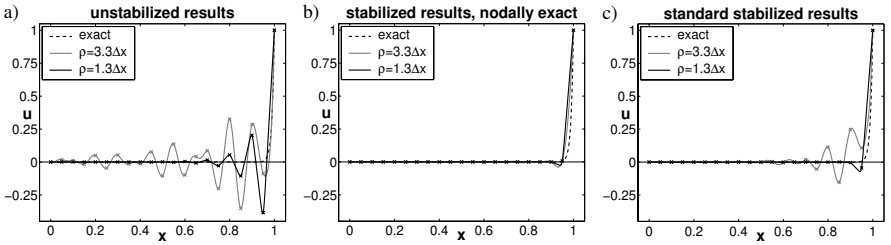


Figure 4.5. Results for the 1D advection-diffusion equation; a) without any stabilisation, b) with global stabilisation Eq.(3.3), c) with local coth-formula Eq.(3.4).

exact result, which can be obtained with the global stabilisation criterion for τ_I derived in subsection 3.1. In figure 4.5c) it can be seen, that standard formulas for τ_I, like the coth-formula, only lead to successful stabilisation when the dilatation parameter is small, which confirms our conjecture in section 3.4.

Comparing figure 4.5b) and c) shows that for small dilatation parameters, the result of the complicated global criterion and the coth-criterion gives almost the same result. This, however, is not the case for the large dilatation parameter of $\rho = 3.3\Delta x$, where pronounced oscillations remain in the solution. These oscillations are clearly not a problem of the high gradient itself that could not be captured by shape functions with such a large dilatation parameter, but result from the use of unsuited stabilisation parameters.

4.2 Incompressible Navier-Stokes Equations

As a first numerical test case we consider the driven cavity. This is a standard test case with benchmark solutions given in [8] for a variety of Reynolds numbers. Here the problem is solved with $Re = 1000$. For a problem statement see Figure 4.6, showing also streamlines and pressure distribution for $Re = 1000$. Figure 4.7a) and b) show velocity profiles for u and v at $y = 0.95$, i.e. near the

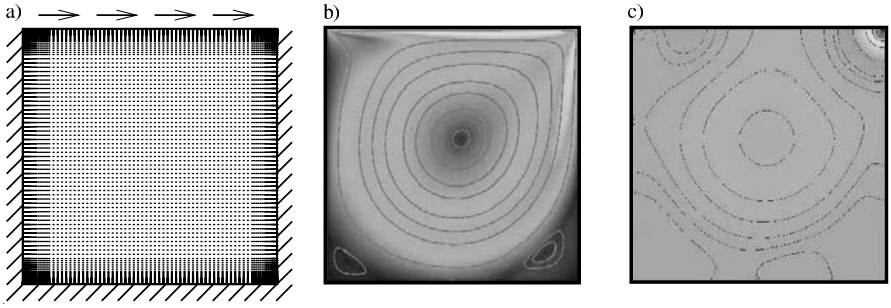


Figure 4.6. a) Data of the driven cavity test case with irregular particle distribution. Velocity b) and pressure c) fields for $Re = 1000$. See Color Plate 1 on page 293.

tangential flow boundary, where most of the oscillations occur. 21×21 regularly distributed particles are used. The dilatation parameter of all supports is $\rho = 1.3\Delta x$. Solutions for dilatation parameters $\rho > 2.7\Delta x$ converged either not at all or only very badly, underlining the need for small dilatation parameters, when standard formulas for τ_I are used. One can clearly see that the oscillations apparent in the unstabilised result are smoothed out successfully. This leads to a superior overall solution, even in those parts of the domain where no oscillations are apparent in the unstabilised case. The next results are computed with 101×101 particles and $\rho = 1.3\Delta x$. With such a large number of particles, stabilisation is not needed at all, i.e. the unstabilised solution is already free of oscillations. The centre velocity profiles in Figure 4.7c) and d) show that stabilisation does not degrade the accuracy when it is not needed. It is interesting that unstabilised and SUPG stabilised results are indistinguishable, whereas GLS stabilised results are slightly more diffusive. This could be confirmed with a number of additional computations.

Figure 4.7e) and f) show a comparison of the benchmark solution with the meshfree solution (with $\rho = 1.3\Delta x$), and the solution from the P1/P1 triangular element with the same number of unknowns. For both methods, SUPG/PSPG stabilisation and an irregular node distribution as shown in Figure 4.6a) has been used. The supports of the nodes are anisotropic with respect to the distance to the neighbouring nodes —$\rho_{x,i} = c \cdot \min(|x_j - x_i|)$ and $\rho_{y,i} = c \cdot \min(|y_j - y_i|)$— with $c = 1.6$. The min-version for a support

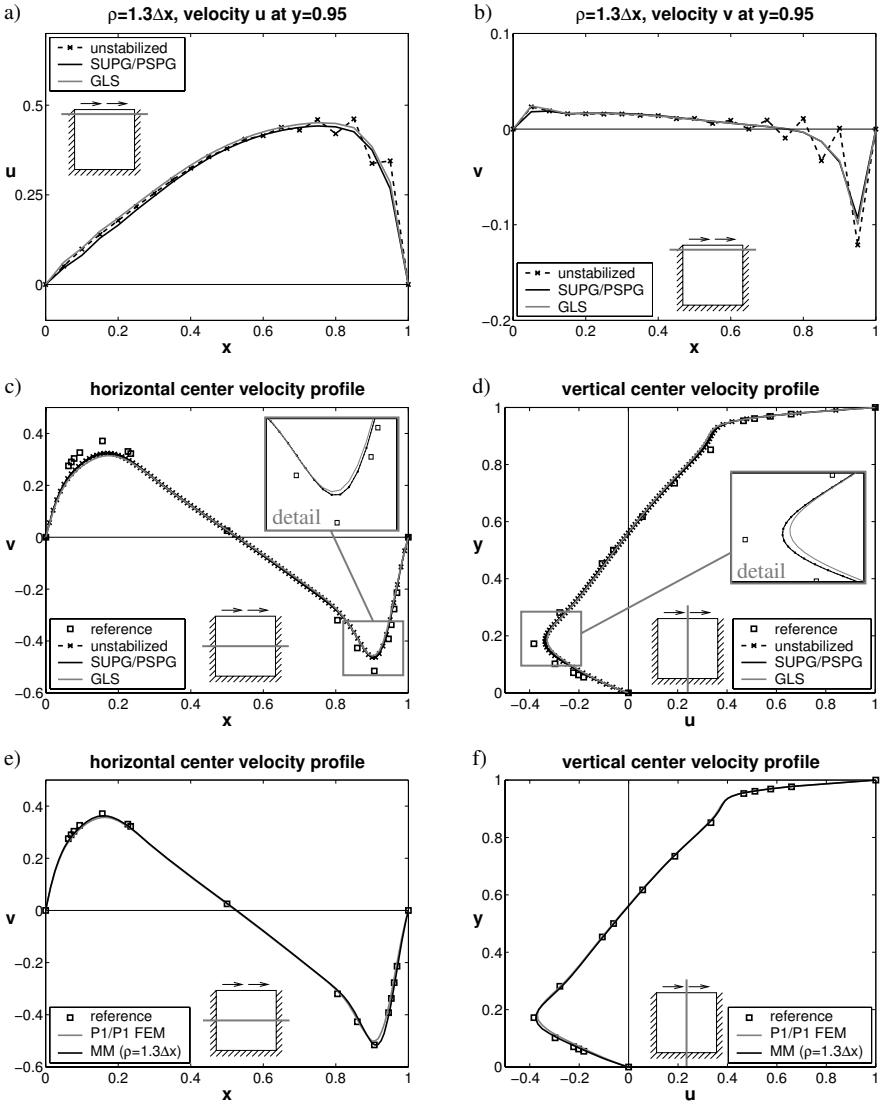


Figure 4.7. Velocity profiles of the driven cavity with different particle distributions.

length h_ρ performs best compared to the other h_ρ-versions. The convergence towards the benchmark solution may clearly be seen, indeed the meshfree solution is more accurate than the P1/P1 element. Comparing the results for the regular 101×101 mesh with the irregular 96×96 mesh, one can see an improvement in the solution for the anisotropic supports. Hence, similar to the use of high-aspect ratio elements in mesh-based methods in order to resolve

boundary layers successfully, high-aspect anisotropic supports should be used in the meshfree context analogously.

The next test case is the 'steady-state' solution for flow past a cylinder at $Re = 100$, as presented in [22]. Instationary computations at this Reynolds number lead to periodic flow patterns known as the Kármán vortex street, but this is not considered here. A problem statement is given in Figure 4.8a). Slip boundary conditions are applied at the upper and lower boundary, no-slip

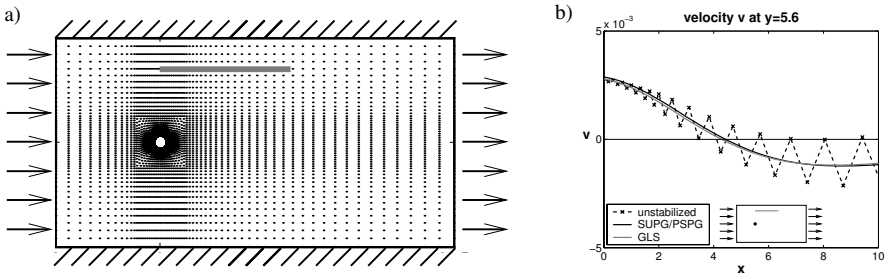


Figure 4.8. a) Problem statement of flow past a cylinder with irregular particle distribution, b) velocity profile for v along the marked line.

boundary conditions are applied at the cylinder surface. The supports of the meshfree shape functions are anisotropic as defined above for the irregular driven cavity test case. Figure 4.8b) depicts the oscillatory unstabilised velocity profile for v along the marked line in Figure 4.8a). Both SUPG/PSPG and GLS stabilisation suppress the oscillations successfully.

5 Conclusion

We employ SUPG/PSPG and GLS stabilisation for the incompressible Navier-Stokes equations in Eulerian formulation and propose a stabilisation parameter τ in MMs. We find that only *small* dilatation parameters of the meshfree shape functions enable the successful use of standard formulas for τ. Generalisation of the stabilisation parameter to multi-dimensions is as straightforward as it is for mesh-based methods. This holds also for irregular particle distributions and anisotropic supports. It is observed that the stabilisation methods smooth out oscillations successfully and maintain higher-order accuracy. The GLS stabilisation introduces slightly more artificial diffusion than SUPG/PSPG stabilisation.

References

1. Brooks, A.N.; Hughes, T.J.R.: Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **32**, 199–259, 1982.
2. Christie, I.; Griffiths, D.F.; Mitchell, A.R.; Zienkiewicz, O.C.: Finite element methods for second order differential equations with significant first derivatives. *Internat. J. Numer. Methods Engrg.*, **10**, 1389–1396, 1976.
3. Donea, J.; Huerta, A.: *Finite Element Methods for Flow Problems.* John Wiley & Sons, Chichester, 2003.
4. Franca, L.P.; Frey, S.L.: Stabilized finite element methods: II. The incompressible Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **99**, 209–233, 1992.
5. Franca, L.P.; Frey, S.L.; Hughes, T.J.R.: Stabilized finite element methods: I. Application to the advective-diffusive model. *Comp. Methods Appl. Mech. Engrg.*, **95**, 253–276, 1992.
6. Fries, T.P.; Matthies, H.G.: Classification and Overview of Meshfree Methods. Informatikbericht-Nr. 2003-03, Technical University Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2003/418/), Brunswick, 2003.
7. Fries, T.P.; Matthies, H.G.: A Review of Petrov-Galerkin Stabilization Approaches and an Extension to Meshfree Methods. Informatikbericht-Nr. 2004-01, Technical University of Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2004/549/), Brunswick, 2004.
8. Ghia, U.; Ghia, K.N.; Shin, C.T.: High-Re solutions for incompressible flow using the Navier-Stokes equations and a multi-grid method. *J. Comput. Phys.*, **48**, 387–411, 1982.
9. Heinrich, J.C.; Huyakorn, P.S.; Zienkiewicz, O.C.; Mitchell, A.R.: An 'upwind' finite element scheme for two-dimensional convective transport equation. *Internat. J. Numer. Methods Engrg.*, **11**, 131–143, 1977.
10. Huerta, A.; Fernández-Méndez, S.M.: Time accurate consistently stabilized mesh-free methods for convection-dominated problems. *Internat. J. Numer. Methods Engrg.*, **50**, 1–18, 2001.
11. Hughes, T.J.R.: A simple scheme for developing 'upwind' finite elements. *Internat. J. Numer. Methods Engrg.*, **12**, 1359–1365, 1978.
12. Hughes, T.J.R.: Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Comp. Methods Appl. Mech. Engrg.*, **127**, 387–401, 1995.
13. Hughes, T.J.R.; Brooks, A.N.: A multidimensional upwind scheme with no crosswind diffusion. In *ASME Monograph AMD-34.* (Hughes, T.J.R., Ed.), Vol. 34, ASME, New York, NY, 1979.
14. Hughes, T.J.R.; Franca, L.P.: A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces. *Comp. Methods Appl. Mech. Engrg.*, **65**, 85–96, 1987.
15. Hughes, T.J.R.; Franca, L.P.; Balestra, M.: A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Comp. Methods Appl. Mech. Engrg.*, **59**, 85–99, 1986.

16. Hughes, T.J.R.; Franca, L.P.; Hulbert, G.M.: A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Least-squares method for advective-diffusive equations. *Comp. Methods Appl. Mech. Engrg.*, **73**, 173–189, 1989.
17. Kelly, D.W.; Nakazawa, S.; Zienkiewicz, O.C.: A note on upwinding and anisotriopic balancing dissipation in finite element approximations to convective diffusion problems. *Internat. J. Numer. Methods Engrg.*, **15**, 1705–1711, 1980.
18. Lancaster, P.; Salkauskas, K.: Surfaces Generated by Moving Least Squares Methods. *Math. Comput.*, **37**, 141–158, 1981.
19. Liu, W.K.; Jun, S.; Zhang, Y.F.: Reproducing Kernel Particle Methods. *Int. J. Numer. Methods Fluids*, **20**, 1081–1106, 1995.
20. Shakib, F.; Hughes, T.J.R.; Johan, Z.: A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **89**, 141–219, 1991.
21. S.Mittal: On the performance of high aspect ratio elements for incompressible flows. *Comp. Methods Appl. Mech. Engrg.*, **188**, 269–287, 2000.
22. Tezduyar, T.E.: Stabilized Finite Element Formulations for Incompressible Flow Computations. In *Advances in Applied Mechanics.* (Hutchinson, J.W.; Wu, T.Y., Eds.), Vol. 28, Academic Press, New York, NY, 1992.
23. Tezduyar, T.E.; Mittal, S.; Ray, S.E.; Shih, R.: Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements. *Comp. Methods Appl. Mech. Engrg.*, **95**, 221–242, 1992.
24. Tezduyar, T.E.; Osawa, Y.: Finite element stabilization parameters computed from element matrices and vectors. *Comp. Methods Appl. Mech. Engrg.*, **190**, 411–430, 2000.

The α-shape Based Natural Element Method in Solid and Fluid Mechanics

D. González, I. Alfaro, E. Cueto, M. Doblaré, and F. Chinesta

[1] Group of Structural Mechanics and Material Modelling. Aragón Institute of Engineering Research ($I3A$). University of Zaragoza. María de Luna, 3. E-50018 Zaragoza, Spain.
[2] Laboratoire de mécanique des systèmes et des procédés. LMSP UMR 8106 CNRS-ENSAM-ESEM. 151 Bd. de l'Hôpital, 75013 Paris, France.

Abstract The Natural Element method (also known as Natural Neighbour Galerkin method) is a Galerkin method based on the use of Natural Neighbour interpolation to construct the trial and test functions. Unlike many other meshless methods, it has some important characteristics, such as interpolant shape functions, easy imposition of essential boundary conditions and linear precision along convex boundaries.

The natural neighbour interpolation scheme is based on the construction of a Delaunay triangulation of the given set of points. This geometrical link provides the NEM some other interesting properties. One of them is the ability of constructing models without any explicit (CAD) boundary description. Instead, by invoking the concept of α-shape of the cloud of points, the method is able to accurately extract the geometry described by the nodes as it evolves, thus avoiding complex geometrical checks in the formation of holes or waves in the domain, without any loss in mass conservation requirements. It has been also proved how the use of α-shapes ensures the strictly interpolant character of the shape functions along any type of boundary. In this work we review the main characteristics of the method in its application to Solid and Fluid Mechanics, including the study of mixed natural neighbour approximation, simulation of nearly incompressible media and some industrial applications.

1 Introduction

In this work we analyse the main features of one member of the wide family of meshless methods. Natural Neighbour Galerkin methods (also known as Natural Element methods, NEM) are based on the use of any natural neighbour-based interpolation scheme. The most extended of these schemes is often referred to as Sibson interpolation [18]. Among the most remarkable properties of the resulting methods is the strictly interpolant character of the resulting approximation, leading to an easy imposition of essential (Dirichlet) boundary conditions. Also, the fact that the computation of the shape functions is made —alt least, formally— upon the Delaunay triangulation of

the cloud of points, confers the method with a sound geometrical basis. This allows, form instance, to couple the method with some geometrical methods, such as α-shapes [12]. This will allow to build a method in which the geometry of the domain is extracted as it evolves, without any geometrical description of the boundary. Thus, boundary evolution, with holes or waves formation, is naturally handled by the method in a very efficient way.

The resulting method appears to be specially well suited to be used in an updated Lagrangian framework, since it has been proved that the shape of Delaunay triangles does not affect the quality of the results [19]. In this work, we review the main characteristics of the method when applied to Fluid and Solid Mechanics with large strains. In section 2 we revisit the basics of natural neighbour interpolation. In section 3 we review the governing equations for incompressible media, together with the conditions that ensure stable approximations (LBB condition) and suitable enriched NE approximations. In section 4 we review the α-shape based Natural Element Method, firstly proposed in [9] and its potential applications in free surface flows. Finally, in section 5 two examples corresponding to Solid and Fluid Mechanics are presented.

2 Natural Neighbour Galerkin Methods

2.1 Natural Neighbour Interpolation

Natural neighbour interpolation is a method to interpolate multivariate data first established by Sibson [18]. Recently, it has been generalised in [5] and [15]. It relies in the concepts of Voronoi diagram [22] of the given cloud of points and its dual structure, the Delaunay triangulation [11]. The Delaunay triangulation (tetrahedrisation) of a cloud of points $N = \{n_1, ..., n_N\} \in \mathbb{R}^2$ ($\mathbb{R}^3$) is the decomposition of the convex hull of the points into k-simplexes (where k represents the dimension of the simplex, that is, $k = 2$ for a triangle and $k = 3$ for a tetrahedron) such that the *empty circumcircle* criterion holds. That is, the circumcircle (circumsphere) of each simplex contains no other point of the cloud N. Fig. 2.1 represents the Delaunay triangulation of a cloud of points.

For a given node n_I, the associated Voronoi cell is composed by all of the points which are closer to the node n_I than to any other node. Formally,

$$T_I = \{\boldsymbol{x} \in \mathbb{R}^n : d(\boldsymbol{x}, \boldsymbol{x}_I) < d(\boldsymbol{x}, \boldsymbol{x}_J) \; \forall \; J \neq I\} \tag{2.1}$$

where T_I is the Voronoi cell and $d(\cdot, \cdot)$ represents the Euclidean distance. In the problems considered in this paper, $n = 2, 3$.

For the definition of Sibson interpolation it is necessary to previously introduce the concept of *second order* Voronoi cell. It is defined as the locus of the points that have the node n_I as the closest node and the node n_J as the second closest node:

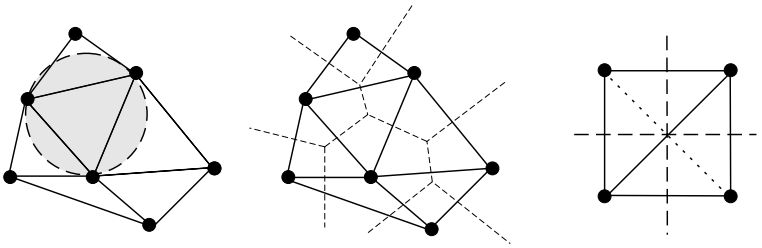


Figure 2.1. Delaunay triangulation and Voronoi diagram of a cloud of points. On the right, an example of a degenerate distribution of nodes, with the two possible triangulations depicted. In this last case, four points lie in the same circumcircle and thus no single triangulation exists.

$$T_{IJ} = \{\boldsymbol{x} \in \mathbb{R}^n : d(\boldsymbol{x}, \boldsymbol{x}_I) < d(\boldsymbol{x}, \boldsymbol{x}_J) < d(\boldsymbol{x}, \boldsymbol{x}_K) \; \forall \; J \neq I \neq K\} \tag{2.2}$$

If a new node is added to a given cloud of points, the Voronoi cells will be modified by the presence of the new point. Sibson [18] defined the natural neighbour coordinates of a point $\boldsymbol{x}$ with respect to one of its neighbours I as the ratio of the cell T_I that is transferred to T_x, when adding $\boldsymbol{x}$ to the initial cloud of points, to the total area of T_x. In other words, being $\kappa(\boldsymbol{x})$ and $\kappa_I(\boldsymbol{x})$ the Lebesgue measures of T_x and T_{xI} respectively, the natural neighbour coordinates of $\boldsymbol{x}$ with respect to the node I is defined as

$$\phi_I^{sib}(\boldsymbol{x}) = \frac{\kappa_I(\boldsymbol{x})}{\kappa(\boldsymbol{x})}. \tag{2.3}$$

The resultant shape function depends obviously of the relative position of the nodes. An example for a node surrounded by other eight on a regular lattice is depicted in Fig. 2.3.

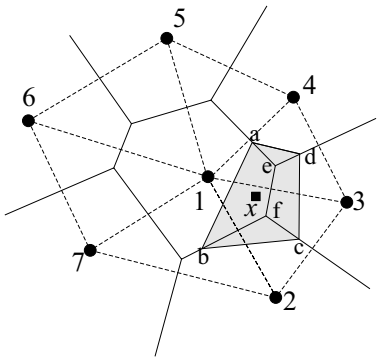


Figure 2.2. Definition of the Natural Neighbour coordinates of a point $\boldsymbol{x}$.

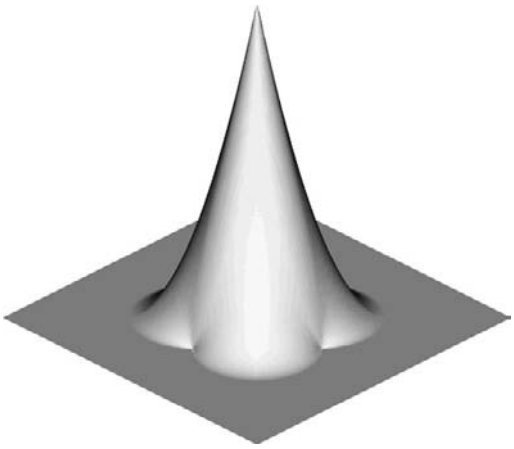


Figure 2.3. Natural Element shape function (courtesy N. Sukumar).

The resultant shape function has some remarkable properties (see [19] or [10] for more in-deep explanations and rigorous proofs of this behaviour). Firstly, it is smooth (C^1 at least) everywhere except at the nodes, as can be seen in Fig. 2.3. Natural neighbour (Sibson) shape functions posses linear completeness [19] and form a partition of unity. Therefore, it is possible to enrich natural neighbour interpolants in order to increase the order of the polinomial of the interpolation, as proposed in [3].

3 Natural Element Methods for Incompressible Media

The governing equations for incompressible Solid or Fluid Mechanics are:

1. Equilibrium equations (balance of linear momentum in the absence of inertial and body forces):

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0} \ \text{ in } \ \Omega. \tag{3.1}$$

2. Incompressibility of the medium:

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \ \text{ in } \Omega, \tag{3.2}$$

where $\boldsymbol{\sigma}$ represents the Cauchy stress tensor and $\boldsymbol{u}$ the displacement vector if we deal with Solid Mechanics and usually the velocity vector if it is the case with Fluid Mechanics. Boundary conditions are of the type

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \overline{\boldsymbol{t}} \ \text{ on } \ \Gamma_t \tag{3.3}$$

$$\boldsymbol{u} = \overline{\boldsymbol{u}} \ \text{ on } \ \Gamma_u. \tag{3.4}$$

in which $\boldsymbol{n}$ is the outward normal on the boundary $\Gamma = \Gamma_u \cup \Gamma_t$, with $\Gamma_u \cap \Gamma_t = \emptyset$.

If we assume small displacements and strains, the constitutive equations can be expressed, both for an isotropic elastic solid and a newtonian fluid, as

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + 2\mu\nabla_s \boldsymbol{u} \tag{3.5}$$

$$0 = \nabla \cdot \boldsymbol{u} - \frac{p}{\lambda} \tag{3.6}$$

where p represents the hydrostatic pressure and ∇_s the symmetric part of the gradient operator, thus leading to the Cauchy small strains tensor or to the strain rate tensor in Solid or Fluid Mechanics, respectively. The Lamé parameters λ and μ are expressed in terms of the Young's modulus E and the Poisson's ratio ν as

$$\mu = \frac{E}{2(1+\nu)}, \quad \lambda = \frac{2\mu\nu}{1-2\nu} \tag{3.7}$$

As ν approaches 0.5, it is clear that λ becomes unbounded, so that the equation (3.6) represents the incompressibility restraint $\nabla \cdot \boldsymbol{u} = 0$.

The variational (weak) formulation is usually stated as:

Find $\boldsymbol{u} \in \mathcal{U}$ *such that*

$$\int_{\Omega(t)} \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}^* d\Omega = \int_{\Gamma_t} \bar{\boldsymbol{t}} \cdot \boldsymbol{u}^* d\Gamma \;\; \forall \boldsymbol{u}^* \in \mathcal{V} \tag{3.8}$$

$$\int_{\Omega(t)} (-\nabla \boldsymbol{u} + \frac{1}{\lambda} p) p^* d\Omega = 0 \;\; \forall p^* \in L_2(\Omega(t)), \tag{3.9}$$

where $\mathcal{U} = \{\boldsymbol{u} | \boldsymbol{u} \in \left(H^1(\Omega)\right)^2, \; \boldsymbol{u}|_{\Gamma_u} = \bar{\boldsymbol{u}}\}$, $\mathcal{V} = \{\boldsymbol{u}^* | \boldsymbol{u}^* \in \left(H^1(\Omega)\right)^2, \; \boldsymbol{u}|_{\Gamma_u} = \boldsymbol{0}\}$, and Γ_u and Γ_t are the portions of the boundary of the domain Ω with prescribed displacements (velocities) and tractions, respectively. $\overline{\boldsymbol{t}}$ and $\overline{\boldsymbol{u}}$ represent such tractions and displacements. As usual, H^1 and L_2 are the Sobolev and Lebesgue functional spaces, respectively.

If we approximate the displacement (velocities) and pressures by employing a finite-dimensional set of basis functions, we arrive to a discrete form of the previous equations (Bubnov-Galerkin method)

$$\boldsymbol{u}^h(\boldsymbol{x}) = \sum_{I=1}^{n} \phi_I(\boldsymbol{x}) \boldsymbol{u}_I \tag{3.10}$$

$$p^h(\boldsymbol{x}) = \sum_{I=1}^{n} \psi_I(\boldsymbol{x}) p_I. \tag{3.11}$$

The functions $\psi_I(\boldsymbol{x})$ and $\phi_I(\boldsymbol{x})$ in this work represent some form of natural neighbour interpolation, as presented before. This leads to the following system of algebraic equations:

$$\begin{pmatrix} \overline{\boldsymbol{K}} & \boldsymbol{G} \\ \boldsymbol{G}^T & \boldsymbol{M} \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{p} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{0} \end{pmatrix} \tag{3.12}$$

where

$$\overline{\boldsymbol{K}}_{IJ} = \int_{\Omega} \boldsymbol{B}_I^T \overline{\boldsymbol{C}} \boldsymbol{B}_J d\Omega \tag{3.13}$$

$$\boldsymbol{G}_{IJ} = -\int_{\Omega} \tilde{\boldsymbol{B}}_I^T \psi_J d\Omega \tag{3.14}$$

$$\boldsymbol{M}_{IJ} = -\frac{1}{\lambda} \int_{\Omega} \psi_I \psi_J d\Omega \tag{3.15}$$

$$\boldsymbol{f}_I = \int_{\Gamma_t} \phi_I \overline{\boldsymbol{t}} d\Gamma \tag{3.16}$$

and

$$\tilde{\boldsymbol{B}}_I = \begin{bmatrix} \phi_{I,1}(\boldsymbol{x}) & \phi_{I,2}(\boldsymbol{x}) \end{bmatrix} \tag{3.17}$$

$$\boldsymbol{B}_I = \begin{pmatrix} \phi_{I,1}(\boldsymbol{x}) & 0 \\ 0 & \phi_{I,2}(\boldsymbol{x}) \\ \phi_{I,2}(\boldsymbol{x}) & \phi_{I,1}(\boldsymbol{x}) \end{pmatrix} \tag{3.18}$$

$$\overline{C}_{IJKL} = \mu(\delta_{IK}\delta_{JL} + \delta_{IL}\delta_{JK}) \tag{3.19}$$

It must be noted that, if we consider totally incompressible situations, $\boldsymbol{M} = \boldsymbol{0}$.

As it is well known, not all of the displacement-pressure approximations constructed in this way lead to stable and convergent results [4]. The conditions to be fulfilled by the chosen approximation are determined by the *inf–sup* or Ladyzhenskaya-Babuška-Brezzi (LBB) condition [2] [7], together with the elipticity condition of the resultant formulation. The LBB condition may be written as:

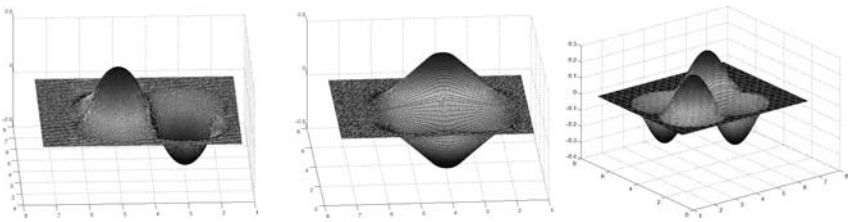
$$\inf_{p^h \in \mathcal{P}_h} \sup_{\boldsymbol{u}^h \in \mathcal{U}_h} \frac{\int_{\Omega} p^h \mathrm{div} \boldsymbol{u}^h d\Omega}{||p^h||_0 ||\boldsymbol{u}^h||_1} = \gamma_h \geq \gamma > 0 \tag{3.20}$$

where γ is a positive constant independent of the mesh size, h. $\mathcal{P}_h$ and $\mathcal{U}_h$ represent the pressure and displacement approximation spaces.

This condition is rarely proved analytically. Instead, its fulfillment is usually checked numerically. In [14], the authors have tested some mixed formulations arising from the Partition of Unity enrichment [3] of the displacement (or velocity) field. If the resultant approximation is able to reproduce the xy monomial, together with the usual NE linear consistency, it seems to verify the LBB condition. The resultant shape functions are depicted in Fig. 3.4.

In order to verify the fulfilment of the LBB condition, Bathe [4] proposed a numerical test based on the use of a finite, small, set of meshes. To evaluate expression (3.20), an equivalent discrete form is developed:

$$\inf_{\boldsymbol{W}_h} \sup_{\boldsymbol{U}_h} \frac{\boldsymbol{W}_h^T \boldsymbol{G}_h \boldsymbol{U}_h}{\sqrt{\boldsymbol{W}_h^T \boldsymbol{G}_h \boldsymbol{W}_h} \cdot \sqrt{\boldsymbol{U}_h^T \boldsymbol{S}_h \boldsymbol{U}_h}} = \gamma_h \geq \gamma > 0 \tag{3.21}$$



(a) (b) (c)

Figure 3.4. Plot of the shape functions ϕx (a), ϕy (b) and ϕxy (c).

where $\boldsymbol{W}_h$ and $\boldsymbol{U}_h$ are vectors of nodal values corresponding to pressures and displacements, respectively. $\boldsymbol{G}_h$ and $\boldsymbol{S}_h$ are the matrices associated with the norms

$$||p_h||_0^2 = \boldsymbol{W}_h^T \boldsymbol{G}_h \boldsymbol{W}_h \text{ and } ||\boldsymbol{u}_h||_1^2 = \boldsymbol{U}_h^T \boldsymbol{S}_h \boldsymbol{U}_h \tag{3.22}$$

$\boldsymbol{G}_h$ and $\boldsymbol{S}_h$ are positive semidefinite and positive definite [8], respectively. It is then demonstrated that the first non-zero eigenvalue, λ_k, of the problem

$$\boldsymbol{G}_h \boldsymbol{\phi}_h = \lambda \boldsymbol{S}_h \boldsymbol{\phi}_h \tag{3.23}$$

is related to the searched value γ_h through the expression

$$\gamma_h = \sqrt{\lambda_k}. \tag{3.24}$$

If the constructed approximation for a given problem consists of n_p pressure degrees of freedom and n_u displacement degrees of freedom, the number of spurious pressure modes is given by

$$k_{pm} = k - (n_u - n_p - 1) \tag{3.25}$$

Bathe [4] proposed the use of sequences composed of three or more meshes in order to test a given approximation. If the γ_h value is not bounded away from 0, one can say that the LBB condition is not satisfied. The test is only valid for the given geometry and problem considered, but it can be assured that if the test is not passed, the approximation will not verify the LBB condition.

In order to test the ability of the proposed formulations to pass the inf-sup condition, we have checked a sequence of three meshes, composed by 3×3, 4×4 and 5×5 nodes, regularly and irregularly distributed over a square of side unity (see Fig. 3.5).

Results for these discretisations are shown in Fig. 3.6. Results for the test applied over the 3/1 triangular Finite Element are also shown. It can be seen how the Sibson approximation, enriched with $\{1, x, y, xy\}$, for the displacements gives good results, both using Thiessen ($\mathcal{C}^{-1}$) and Sibson interpolations

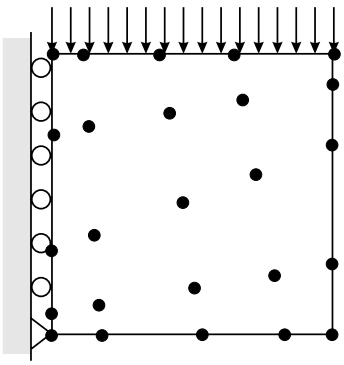


Figure 3.5. Geometry of the problem used to evaluate the numerical inf-sup test.

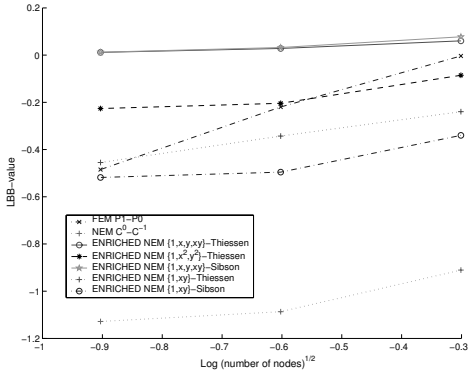


Figure 3.6. Inf-sup test for the proposed approximations.

for the pressure. However, the enrichment with the monomials $\{1, x^2, y^2\}$, initially proposed in order to avoid rank deficiency in the resultant stiffness matrix, gives rise to spurious pressure modes. These same spurious modes can be obtained for certain configurations in the Sibson-Thiessen element, in a way similar to the bilinear displacement–constant pressure quadrilateral Finite Element [4].

Although results for a unique problem are not extrapolable, the proposed formulation for the $\{1, x, y, xy\}$ enrichment seems to be stable and adequate for a wide variety of problems. Also, enrichment with the set $\{1, xy\}$ seems to give stable approximations. In this case, the resulting approximation closely resembles the MINI element [1] (linear triangular finite element with bubble and discontinuous approximation for pressure).

The most usual Sibson-Thiessen mixed approximation closely resembles the bilinear displacement-constant pressure finite element, which presents spurious modes under certain boundary conditions. However, for most practical applications, it has shown a good behaviour and has never presented locking. An example is presented in section 5.2 in which this approximation is used.

4 The α-shape based Natural Element Method

In most Lagrangian Fluid Mechanics simulations the tracking of the free surfaces requires a special treatment, since the possibility of development of holes, waves, etc. exists. Traditionally, this task has been accomplished by discretising the boundary and performing complex geometrical checks. This may include the checking for new free surface boundary segments, if holes are developing, or boundary segments deletion, if the free surface disappears.

In this work we have chosen a different approach, based on the geometrical concept of α-shapes [13]. In addition, the authors have shown that the

construction of natural neighbour-based interpolants over an α-shape of the domain leads to linearly interpolant approximations along the whole boundary, thus solving one of the biggest problems of meshless methods. The problem can be formulated briefly as follows: can the cloud of points itself contain enough information about the geometry of the domain over which it is defined? If the density of the cloud is enough, the answer is affirmative.

An α-shape is a polytope that is not necessarily convex nor connected, being triangulated by a subset of the Delaunay triangulation of the points. Thus, the empty circumcircle criterion holds. Let N be a finite set of points in $\mathbb{R}^3$ and α a real number, with $0 \leq \alpha < \infty$. A k-simplex σ_T with $0 \leq k \leq 3$ is defined as the convex hull of a subset $T \subseteq N$ of size $\mid T \mid = k + 1$. Let b be an α-ball, that is, an open ball of radius α. A k-simplex σ_T is said to be α-exposed if there exist an empty α-ball b with $T = \partial b \bigcap N$ where ∂ means the boundary of the ball. In other words, a k-simplex is said to be α-exposed if an α-ball that passes through its defining points contains no other point of the set N.

Following this, we can define the family of sets $F_{k,\alpha}$ as the sets of α-exposed k-simplexes for the given set N. This allows us to define an α-shape of the set N as the polytope whose boundary consists on the triangles in $F_{2,\alpha}$, the edges in $F_{1,\alpha}$ and the vertices or nodes in $F_{0,\alpha}$. As remarked before, an α-shape is a polytope that can be triangulated by a subset of the Delaunay triangulation or tetrahedrization, that is, by an α-complex.

In the case of non-uniform nodal distributions, the possibility of using density-scaled α-shapes [20] has also been studied in [9]. The resultant geometry of the domain ranges from the cloud of points itself for $\alpha = 0$ to the convex hull of the cloud for $\alpha = \infty$. The parameter α can be seen as a measure of the level of detail up to which the domain is represented. If there exists a sufficiently dense nodal sampling it would be easy to find an α value that gives an accurate geometry definition. See [9, 17] for a deeper discussion on how the value of α can affect the results.

In this approach, the geometry of the domain is extracted at each time step, with no need of complex geometrical checks. See example 5.2 for an application of the method in an extrusion flow.

5 Numerical Examples

5.1 Cantilever Beam under Bending

In this section we study the problem of a two-dimensional beam subjected to a parabolically distributed load at its end and fixed at the other side, as shown in Fig. 5.7. We consider a discretisation composed of 85 nodes, as shown in Fig. 5.8.

Material characteristics are: Young's modulus 1.0 and variable Poisson's ratio, ranging from 0.4 to 0.4999999. To test the performance of the proposed

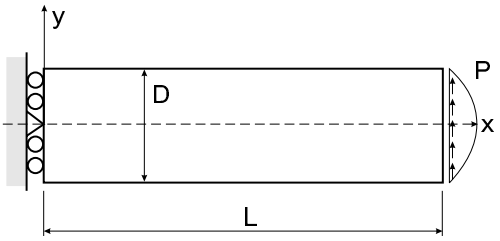


Figure 5.7. Geometry of the cantilever beam under bending.

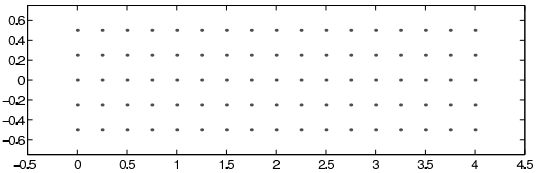


Figure 5.8. Cloud of points for the simulation of a beam under bending.

formulations, we have compared the tip displacement to the theoretical one [21]. In this case, $L = 4.0$ and $D = 1.0$.

In table 5.1, normalised end displacements ($u_y(L, 0)$) are presented. These results include those of the FEM, obtained by using the same Delaunay triangles for the construction of the approximation. It can be seen how the displacement-based NEM presents severe locking as the Poisson's ratio increases, as expected. However, all the other constructed approximations present good agreement with the analytical result. In particular, we would like to highlight the good behaviour of the Sibson-Thiessen approximation, previously used by the authors in Fluid Mechanics simulations in [17]. Although it is possible to impose certain boundary conditions so as to generate a spurious pressure mode (in a similar way than to the 4/1 Finite Element [4]), no spurious pressure modes have been observed throughout our computations.

Poisson's Ratio	0.4	0.4999	0.4999999
FEM-displacement-based	93.74	18.73	17.75
FEM-3/3	96.38	94.45	94.45
FEM-3/1	100.61	101.52	101.52
NEM-displacement-based	94.36	19.09	19.58
NEM-Sibson-Sibson	96.78	94.93	94.94
NEM-Sibson-Thiessen	99.28	99.07	99.07
NEM Sibson $\times \{1, x^2, y^2\}$-Thiessen	100.38	100.33	99.34
NEM Sibson $\times \{1, x, y, xy\}$-Sibson	99.32	100.52	100.52
NEM Sibson $\times \{1, x, y, xy\}$-Thiessen	99.4	100.7	100.7

Table 5.1. Results for the beam under bending problem, expressed as % of the theoretical result at the beam end.

Results are even better for the enriched Sibson approximation. None of the implemented enrichments showed spurious pressure modes for this problem and the obtained accuracy is remarkable. However, the use of enriched approximations is not so intuitive, since the nodal parameters in the discrete system of equations do not represent the nodal displacements.

5.2 Simulation of Extrusion Processes

In this section we deal with the simulation of extrusion processes. We consider a benchmark example of an extrusion die whose geometry is shown in Fig. 5.9. For the simulation, only the region of the metal nearest to the die was considered by enforcing appropriate displacements. The cloud of points is composed by 2989 nodes and remained unchanged throughout the simulation. The extruded metal was simulated as viscoplastic, with constitutive equations as follows:

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + 2\mu(\boldsymbol{D})\boldsymbol{D} \tag{5.1}$$

If we assume a Norton-Hoff plasticity model (viscoplastic flow with null yield stress), viscosity is a function of the second invariant of the strain rate tensor, namely

$$\mu(\boldsymbol{D}) = \mu_0\Big(\sqrt{2\boldsymbol{D}:\boldsymbol{D}}\Big)^{n-1} \tag{5.2}$$

being μ_0 the so-called consistency coefficient and n the pseudo-plasticity coefficient. In this example, the consistency coefficient was $\mu_0 = 1.0MPa \cdot s$ and the pseudo-plasticity $n = 0.3$. Note the highly non-linear character of the resultant behaviour. Similar material behaviour was employed in [6] in the simulation of metal forging processes with Corrected Smooth Particle Hydrodynamics methods (CSPH).

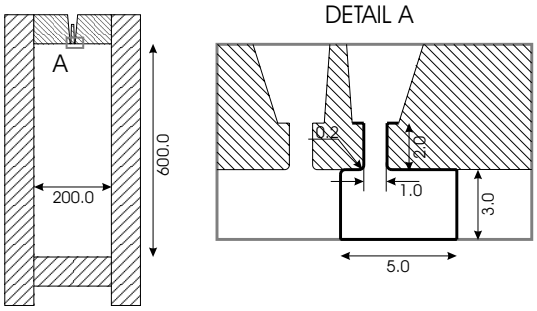


Figure 5.9. Geometry of the extrusion die and detail of the simulated region.

The mould is considered as perfectly rigid and slipping contact between metal and die was assumed. Of course, this model is a first attempt to validate the ability of natural neighbour Galerkin methods to handle such type of

processes. The *springback* of the extruded metal can not be predicted, for instance, by assuming a Norton-Hoff plasticity model like the one exposed in Eq. (5.1).

Equivalent plastic strain (or, equivalently, second invariant of the strain rate tensor, $\sqrt{2\boldsymbol{D} : \boldsymbol{D}}$) for time steps 1, 100 and 200 is depicted in Fig. 5.10. It can be noticed the accuracy obtained in the volume prediction (see Fig. 5.11). In this case, total volume error is under 0.25%, much lower than those predicted in references like [16] by using ALE techniques and a similar number of nodes. This is in spite of the high distortion of the Delaunay triangles achieved throughout the simulation (see the α-shape of the cloud of points in an intermediate time step in Fig. 5.12).

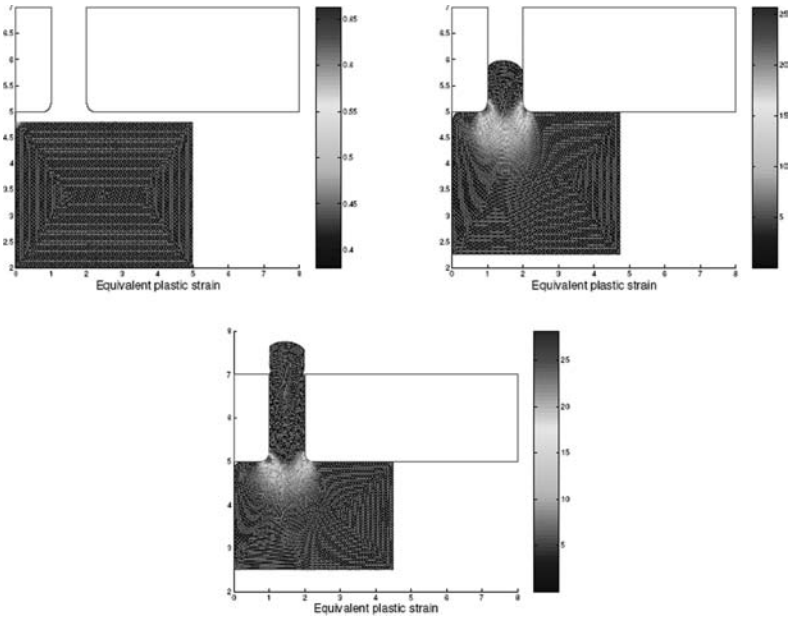
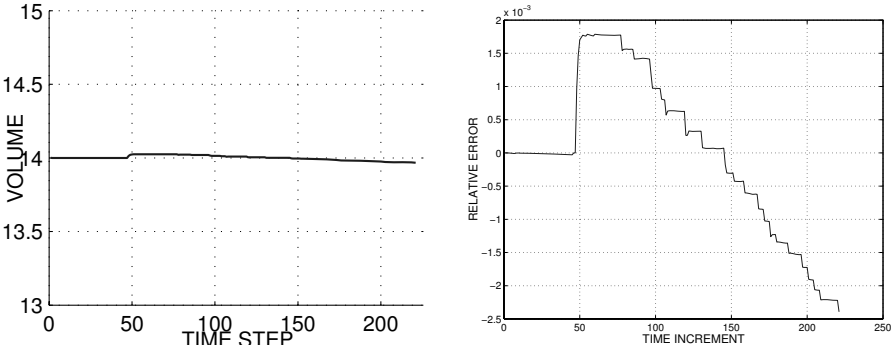


Figure 5.10. Equivalent plastic strain for time steps 1, 100 and 200. See Color Plate 2 on page 293.

In this example it can be seen how natural neighbour Galerkin methods constitute an appealing choice among numerical methods to simulate forming processes in general, and among meshless methods in particular.



(a) (b)

Figure 5.11. Volume of the computed α-shape of the extruded metal (a) and associated relative error (b).

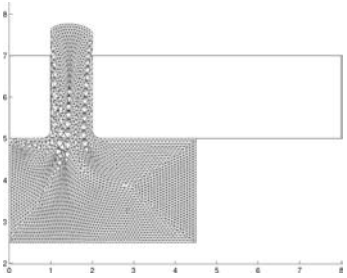


Figure 5.12. α-shape of the cloud of points at the 200^{th} time step.

6 Closing Remarks

In this paper we have briefly reviewed the characteristics of Natural Neighbour Galerkin methods. In particular, we have focused on the simulation of incompressible media and the development of stable mixed approximations through the use of Partition of Unity enrichment of Sibson interpolants.

Natural Neighbour Galerkin methods offer a sound geometrical basis for the treatment of free boundaries through the use of α-shapes. It has been shown how the use of an appropriate α-shape of the cloud of points allows us to accurately track the free surface of the domains through the simulated time interval. At the same time, the employ of α-shapes ensures linear interpolation along the boundary of the domain.

In summary, we believe that the α-shape based Natural element Method (α-NEM) is an attractive choice when large distortion of the domain appears, both in Solid and Fluid Mechanics.

References

1. D. N. Arnold, F. Brezzi, and M. Fortin. A stable finite element for the Stokes equations. *Calcolo*, 21:337–344, 1984.
2. I. Babuška. The Finite Element Method with Lagrange multipliers. *Numerische Mathematik*, 20:179–192, 1973.
3. I. Babuška and J. M. Melenk. The partition of unity finite element method: Basic theory and applications. *Comp. Meth. in Appl. Mech. and Eng.*, 4:289–314, 1996.
4. K. J. Bathe. *Finite Element procedures*. Prentice-Hall, 1996.
5. J.-D. Boissonnat and F. Cazals. Natural neighbour coordinates of points on a surface. *Computational Geometry: Theory and Applications*, 19(2–3):155–173, 2001.
6. J. Bonet and S. Kulasegaram. Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *International Journal for Numerical Methods in Engineering*, 47:1189–1214, 2000.
7. F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from lagrange multipliers. *Revue Française d'Automatique Informatique Recherche Operationelle, Analyse Numérique*, 8:129–151, 1974.
8. D. Chapelle and K. J. Bathe. The inf-sup test. *Computers and Structures*, 47(4-5):537–545, 1993.
9. E. Cueto, M. Doblaré, and L. Gracia. Imposing essential boundary conditions in the Natural Element Method by means of density-scaled α-shapes. *International Journal for Numerical Methods in Engineering*, 49-4:519–546, 2000.
10. E. Cueto, N. Sukumar, B. Calvo, J. Cegoñino, and M. Doblaré. Overview and recent advances in Natural Neighbour Galerkin methods. *Archives of Computational Methods in Engineering*, 10(4):307–387, 2003.
11. B. Delaunay. Sur la Sphère Vide. A la memoire de Georges Voronoi. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7:793–800, 1934.
12. H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, IT-29(4):551–559, 1983.
13. H. Edelsbrunner and E. Mücke. Three dimensional alpha shapes. *ACM Transactions on Graphics*, 13:43–72, 1994.
14. D. González, E. Cueto, and M. Doblaré. Volumetric locking in Natural Neighbour Galerkin methods. *International Journal for Numerical Methods in Engineering*, submitted for publication, 2003.
15. H Hiyoshi and K. Sugihara. Voronoi-based interpolation with higher continuity. In *Proceedings of the 16th Annual ACM Symposium on Computational Geometry*, pages 242–250, 2000.
16. R. W. Lewis, S. E. Navti, and C. Taylor. A mixed lagrangian-eulerian approach to modelling fluid flow during mould filling. *International Journal for Numerical Methods in Engineering*, 25:931–952, 1997.

17. M. A. Martínez, E. Cueto, M. Doblaré, and F. Chinesta. Natural Element meshless simulation of injection processes involving short fiber suspensions. *Journal of Non-Newtonian Fluid Mechanics*, 115:51–78, 2003.
18. R. Sibson. A brief description of natural neighbour interpolation. In *Interpreting Multivariate Data. V. Barnett (Editor)*, pages 21–36. John Wiley, 1981.
19. N. Sukumar, B. Moran, and T. Belytschko. The Natural Element Method in Solid Mechanics. *International Journal for Numerical Methods in Engineering*, 43(5):839–887, 1998.
20. M. Teichmann and M. Capps. Surface reconstruction with anisotropic density-scaled alpha shapes. In *Proceedings of the 1998 IEEE Visualization Conference*, 1998.
21. S. Timoshenko and J. N. Goodier. *Teoría de la Elasticidad*. Editorial Urmo, 1972.
22. G. M. Voronoi. Nouvelles Applications des Paramètres Continus à la Théorie des Formes Quadratiques. Deuxième Memoire: Recherches sur les parallélloèdres Primitifs. *J. Reine Angew. Math.*, 134:198–287, 1908.

A Particle-Partition of Unity Method Part VI: A *p*-robust Multilevel Solver

Michael Griebel[1⋆], Peter Oswald[2⋆⋆], and Marc Alexander Schweitzer[1⋆⋆⋆]

[1] Institut für Numerische Simulation, Rheinische Friedrich–Wilhelms Universität Bonn, Wegelerstraße 6, D–53115 Bonn, Germany.
[2] International University Bremen, School of Engineering and Science, Campus Ring 1, D–28759 Bremen, Germany.

Abstract In this paper we focus on the efficient multilevel solution of linear systems arising from a higher order discretization of a second order partial differential equation using a partition of unity method. We present a multilevel solver which employs a tree-based spatial multilevel sequence in conjunction with a domain decomposition type smoothing scheme. The smoother is based on an overlapping subspace splitting, where the subspaces contain all interacting local polynomials. The resulting local subspace problems are solved exactly. This leads to a computational complexity of the order $O(Np^{3d})$ per iteration. The results of our numerical experiments indicate that the convergence rate of this multilevel solver is independent of the number of points N and the approximation order p. Hence, the overall complexity of the solver is of the order $O(\log(1/\epsilon)Np^{3d})$ to reduce the initial error by a prescribed factor ϵ.

1 Introduction

The particle–partition of unity method (PUM) [7, 8, 9, 10, 11, 18] is a meshfree Galerkin method for the numerical treatment of partial differential equations (PDE). In essence, it is a generalized finite element method (GFEM) which employs piecewise rational shape functions rather than piecewise polynomial functions. The shape functions $\varphi_i\psi_i^n$ of a PUM are products of a partition of unity (PU) $\{\varphi_i\}$ and local approximation functions ψ_i^n which are usually chosen as polynomials. In contrast to other GFEM approaches [19, 20], these shape functions are linearly independent and make up a basis of the discrete function space. This allows us to construct fast multilevel solvers in a similar fashion as in the finite element method (FEM) [12, 21].

In most meshfree discretizations the two most time-consuming tasks are the assembly of the stiffness matrix and the load vector, i.e. numerical integration, and the solution of the resulting linear system. With respect to the

⋆ griebel@ins.uni-bonn.de
⋆⋆ p.oswald@iu-bremen.de
⋆⋆⋆ schweitzer@ins.uni-bonn.de

asymptotic complexity of these steps the assembly of the stiffness matrix is usually directly proportional to the number of nonzeros nnz of the matrix. The solution of the linear system, however, may only be achieved with similar complexity if an optimal iterative solver is employed. For example, the complexity of a direct solver such as LU- or Cholesky-decomposition is in general much larger. In the case of dense matrices a direct solver requires $O(\mathsf{dof}^3)$ operations and $O(\mathsf{dof}^2)$ storage. With more advanced sparse direct solvers these complexities can be reduced to some extent only. In the special case of regular meshes in two dimensions for instance a nested dissection solver requires $O(\mathsf{dof}^{3/2})$ operations and $O(\mathsf{dof}\ln(\mathsf{dof}))$ storage [5] whereas the optimal computational complexity is of the order $O(\mathsf{dof}) = O(\mathsf{nnz})$. Hence, this optimal storage and operation complexity will be lost when a direct solver is employed.

For a PUM discretization in d dimensions, the number of degrees of freedom is of the order $\mathsf{dof} = O(Np^d)$ where N denotes the number of points or particles and p is the approximation order. Here, the number of nonzeros nnz is of the order $O(Np^{2d})$. Thus, the optimal complexity for the assembly of the stiffness matrix is $O(Np^{2d})$. To allow for an efficient and scalable meshfree simulation the employed linear solver should have a similar complexity. Since there is no such optimal solver based on general algebraic methods, non-optimal (sparse) direct solvers are often employed in meshfree methods or generalized finite element methods [20]. Our goal is the development of an iterative solver with optimal complexity for the PUM; i.e., the number of iterations required to solve the linear system should be independent of the number of points N and the approximation order p, and the computational cost associated with a single iteration should be close to $O(Np^{2d})$.

In this paper we present a multilevel solver for partition of unity discretizations which employs a tree-based spatial multilevel construction and a domain decomposition type smoothing scheme. The respective subspace splitting is based on overlapping subspaces which contain all shape functions $\varphi_j\psi_j^m$ interacting on a given patch $\omega_i := \mathrm{supp}(\varphi_i)$. The computational complexity of a single iteration of this solver is $O(Np^{3d})$. Furthermore, the results of our numerical experiments indicate that the convergence rate is independent of N and p; i.e., the number of iterations required to solve the linear system within a given relative accuracy ϵ is independent of N and p. Therefore, the overall complexity of the solver is of the order $O(\log(1/\epsilon)Np^{3d})$ for any polynomial degree p which is optimal up to a factor of $O(p^d)$.

The remainder of this paper is organized as follows: In section 2 we shortly review the construction of PUM spaces and the Galerkin discretization of a linear elliptic PDE using our PUM. In section 3 we shortly review our tree-based multilevel cover construction and introduce a p-robust smoothing scheme based on domain decomposition ideas. Then, we present the results of our numerical experiments in section 4 which indicate that the convergence rate of our multilevel solver is independent of N and p. We consider scalar Poisson-type problems in two and three space dimensions and the system of

the Navier–Lamé equations in two dimensions with up to 393216 degrees of freedom. Finally, we conclude with some remarks in section 5.

2 Partition of Unity Method

In the following, we shortly review the construction of partition of unity spaces and the meshfree Galerkin discretization of an elliptic PDE, see [7, 8, 18] for details.

2.1 Construction of Partition of Unity Spaces

In a PUM, we define a global approximation u^{PU} as a weighted sum of local approximations u_i,

$$u^{\mathrm{PU}}(x) := \sum_{i=1}^{N} \varphi_i(x) u_i(x). \tag{2.1}$$

These local approximations u_i are completely independent of each other, i.e., the local supports $\omega_i := \mathrm{supp}(u_i)$, the local basis $\{\psi_i^n\}$ and the order of approximation p_i for every single $u_i := \sum_n u_i^n \psi_i^n \in V_i^{p_i}$ can be chosen independently of all other u_j. Here, the functions φ_i form a partition of unity (PU). They are used to splice the local approximations u_i together in such a way that the global approximation u^{PU} benefits from the local approximation orders p_i yet it still fulfills global regularity conditions. Hence, the global approximation space on Ω is defined as

$$V^{\mathrm{PU}} := \sum_i \varphi_i V_i^{p_i} = \sum_i \varphi_i \,\mathrm{span}\langle\{\psi_i^n\}\rangle = \mathrm{span}\langle\{\varphi_i \psi_i^n\}\rangle. \tag{2.2}$$

The starting point for any meshfree method is a collection of N independent points $P := \{x_i \in \mathbb{R}^d \,|\, x_i \in \overline{\Omega}, i = 1, \ldots, N\}$. In the PU approach we need to construct a partition of unity $\{\varphi_i\}$ on the domain of interest Ω to define an approximate solution (2.1) where the union of the supports $\mathrm{supp}(\varphi_i) = \overline{\omega_i}$ covers the domain $\overline{\Omega} \subset \bigcup_{i=1}^N \omega_i$ and $u_i \in V_i^{p_i}(\omega_i)$ is some locally defined approximation of order p_i to u on ω_i. Thus, the first (and most crucial) step in a PUM is the efficient construction of an appropriate cover $C_\Omega := \{\omega_i\}$. Throughout this paper we use a tree-based construction algorithm for d-rectangular covers C_Ω presented in [8, 18]. Here, the cover patches ω_i are products of intervals $(x_i^l - h_i^l, x_i^l + h_i^l)$ for $l = 1, \ldots, d$. With the help of weight functions W_k defined on these cover patches ω_k we can easily generate a partition of unity by *Shepard's method*, i.e., we define

$$\varphi_i(x) = \frac{W_i(x)}{\sum_{\omega_k \in C_\Omega^i} W_k(x)}, \tag{2.3}$$

where $C_i := \{\omega_j \in C_\Omega \,|\, \omega_i \cap \omega_j \neq \emptyset\}$ is the set of all geometric neighbors of a cover patch ω_i. Due to the use of d-rectangular patches ω_i, the most natural choice for a weight function W_i is a product of one-dimensional functions, i.e., $W_i(x) = \prod_{l=1}^d W_i^l(x^l) = \prod_{l=1}^d \mathcal{W}(\frac{x-x_i^l+h_i^l}{2h_i^l})$ with $\mathrm{supp}(\mathcal{W}) = [0,1]$ such that $\mathrm{supp}(W_i) = \overline{\omega_i}$. It is sufficient for this construction to choose a one-dimensional weight function $\mathcal{W}$ with the desired regularity which is non-negative. The partition of unity functions φ_i inherit the regularity of the generating weight function $\mathcal{W}$.

In general, a partition of unity $\{\varphi_i\}$ can only recover the constant function on the domain Ω. Hence, we need to improve the approximation quality to use the method for the discretization of a PDE. To this end, we multiply the partition of unity functions φ_i locally with polynomials ψ_i^n. Since we use d-rectangular patches ω_i only, a local tensor product space is the most natural choice. Here, we use products of univariate Legendre polynomials as local approximation spaces $V_i^{p_i}$, i.e., we choose

$$V_i^{p_i} = \mathrm{span}\langle\{\psi_i^n \,|\, \psi_i^n = \prod_{l=1}^d \mathcal{L}_i^{\hat{n}_l}, \|\hat{n}\|_1 = \sum_{l=1}^d \hat{n}_l \leq p_i\}\rangle,$$

where $\hat{n}$ is the multi-index of the polynomial degrees $\hat{n}_l$ of the univariate Legendre polynomials $\mathcal{L}_i^{\hat{n}_l} : [x_i^l - h_i^l, x_i^l + h_i^l] \to \mathbb{R}$, and n is the index associated with the product function $\psi_i^n = \prod_{l=1}^d \mathcal{L}_i^{\hat{n}_l}$.

In summary, we can view the construction given above as follows

$$\begin{pmatrix} \{x_i\} \\ \mathcal{W} \\ \{p_i\} \end{pmatrix} \to \begin{pmatrix} \{\omega_i\} \\ \{W_i\} \\ \{V_i^{p_i} = \mathrm{span}\langle\psi_i^n\rangle\} \end{pmatrix} \to \begin{pmatrix} \{\varphi_i\} \\ \{V_i^{p_i}\} \end{pmatrix} \to V^{\mathrm{PU}} = \sum \varphi_i V_i^{p_i},$$

where the set of points $P = \{x_i\}$, the generating weight function $\mathcal{W}$ and the local approximation orders p_i are assumed to be given. For the approximation of vector fields we employ vector-valued shape functions; i.e., we change the definition of our local approximation spaces $V_i^{p_i} = \mathrm{span}\langle\psi_i^{n,l}\rangle = \mathrm{span}\langle\psi_i^n \boldsymbol{e}_l\rangle$ where $\boldsymbol{e}_l$ denotes an appropriate unit vector but leave the scalar partition of unity functions φ_i unchanged. Throughout this paper we use a fixed polynomial degree p on all patches ω_i, i.e, $p_i = p$ for all $i = 1, \ldots, N$.

2.2 Variational Formulation and Galerkin Discretization

The imposition of essential boundary conditions within meshfree methods is more involved than in the FEM for a number of reasons and many different approaches have been proposed [18]. We use Nitsche's method [14] to enforce Dirichlet boundary conditions which leads to a non-standard weak formulation. The main advantages of this approach are that it does not require a second function (or multiplier) space and that it leads to a positive definite linear system, see [11, 18] for a more detailed discussion of Nitsche's

method in the PUM context. Here, we only state the resulting weak formulation $a(u,v) = l(v)$ of the Poisson problem

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega \subset \mathbb{R}^d, \\ u &= g_D && \text{on } \Gamma_D \subset \partial\Omega, \\ u_n &= g_N && \text{on } \Gamma_N = \partial\Omega \setminus \Gamma_D, \end{aligned} \tag{2.4}$$

and for the system of the Navier–Lamé equations

$$-\mu \boldsymbol{\Delta} u - (\lambda + \mu)\nabla(\nabla \cdot u) = f \quad \text{in} \quad \Omega \subset \mathbb{R}^d, \quad d = 2,3 \tag{2.5}$$

with suitable boundary conditions $u_D = g_D$ on $\Gamma_D \subset \partial\Omega$ and $\boldsymbol{\sigma}(u) \cdot n = g_N$ on $\Gamma_N = \partial\Omega \setminus \Gamma_D$. The parameters λ and μ are the so-called Lamé parameters of the material and are related to the Poisson ratio ν and the Young modulus E via $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$ and $\mu = \frac{E}{2(1+\nu)}$.

The bilinear form $a(u,v)$ associated with (2.4) using Nitsche's method is given by

$$a(u,v) = \int_\Omega \nabla u \nabla v + \int_{\Gamma_D} u(\beta v - v_n) - u_n v$$

and the respective right-hand side is given by

$$l(v) = \int_\Omega f v + \int_{\Gamma_D} g_D(\beta v - v_n) + \int_{\Gamma_N} g_N v.$$

The subscript n denotes the normal derivative and β is the Nitsche regularization parameter. For (2.5) we obtain the bilinear form

$$a(u,v) = \int_\Omega \boldsymbol{\sigma}(u) : \boldsymbol{\epsilon}(v) + \int_{\Gamma_D} 2\mu\beta_{\boldsymbol{\epsilon}} u \cdot v + \lambda\beta_{\text{div}}(u \cdot n)(v \cdot n) - \big((\boldsymbol{\sigma}(u)\cdot n)\cdot v + u \cdot (\boldsymbol{\sigma}(v)\cdot n)\big)$$

where $\beta_{\boldsymbol{\epsilon}}$ and β_{div} denote the two Nitsche regularization parameters involved, $\boldsymbol{\sigma}(u) := \lambda \nabla \cdot u I + 2\mu\, \boldsymbol{\epsilon}(u)$ is the symmetric stress tensor and $\boldsymbol{\epsilon}(u) := \frac{1}{2}(\partial_i u_j + \partial_j u_i)$ denotes the strain tensor associated with the displacement field $u = (u_i)$, $i = 1, \ldots, d$. The respective linear form $l(v)$ on the right-hand side is given by

$$l(v) = \int_\Omega f \cdot v + \int_{\Gamma_N} g_N \cdot v + \int_{\Gamma_D} 2\mu\beta_{\boldsymbol{\epsilon}} g_D \cdot v + \lambda\beta_{\text{div}}(g_D \cdot n)(v \cdot n) - g_D \cdot (\boldsymbol{\sigma}(v)\cdot n).$$

Note that Nitsche's method introduces regularization parameters which depend on the employed discretization space. Hence, as we refine the discretization space, the regularization parameters and therefore the bilinear form $a(\cdot,\cdot)$ and the linear form on the right-hand side $l(\cdot)$ change. Note also that the regularization parameters can be computed automatically during a simulation without much computational cost for a specific discretization space, see [11, 18] for details.

Finally, for the Galerkin discretization of (2.4) or (2.5) we have to compute the stiffness matrix

$$A = (A_{(i,n),(j,m)}), \text{ with } A_{(i,n),(j,m)} = a(\varphi_j \psi_j^m, \varphi_i \psi_i^n),$$

and the right-hand side vector

$$\hat{f} = (f_{(i,n)}), \text{ with } f_{(i,n)} = l(\varphi_i \psi_i^n).$$

The stable approximation of the respective integrals is somewhat more involved in the PUM than in the FEM. Due to the meshfree construction given above the shape functions $\varphi_i \psi_i^n$ are piecewise rational functions only, so that the respective integrands have a number of jumps within the integration domain which need to be resolved.

To estimate the computational cost associated with the Galerkin discretization, let us first assume that the shape functions are (piecewise) polynomials like in the GFEM. Then each integral associated with a particular entry $A_{(i,n),(j,m)}$ of the stiffness matrix can be computed as the sum of integrals over the elements within $\text{supp}(\varphi_i) \cap \text{supp}(\varphi_j)$. Now, all integrands[3] are polynomial functions as well and the integrals can be evaluated exactly. If the shape functions are polynomials of degree p the integrals are polynomials of degree $(p+1)^2$ and can be evaluated with $O(dp^2)$ operations. Therefore, we can assemble the stiffness matrix with $O(N(dp^2 + p^{2d}))$ operations, if we evaluate all polynomials simultaneously.

In the PUM, however, the analytical integration of the integrals is in general not possible. Hence, we need to employ a numerical integration scheme. The cost $\mathcal{C}_{\text{NI}}$ associated with the numerical integration of a single entry of the stiffness matrix is given by

$$\mathcal{C}_{\text{NI}} = O(n_{\text{IC}}\ n_{\text{IN}}\ \mathcal{C}_{\text{EI}})$$

where n_{IC} denotes the number of integration cells, n_{IN} the number of integration nodes per cell, and $\mathcal{C}_{\text{EI}}$ the cost associated with the evaluation of the integrand.[4] In our implementation we use a subdivision sparse grid integration scheme [8, 18] where n_{IC} is essentially determined by the jumps of the derivatives of φ_i; i.e., $n_{\text{IC}} = O(3^d(l+1)^d)$ is given by the order l of the employed spline weight function $\mathcal{W}$. In the following we restrict ourselves to the case of $l = 1$. Within each of these integration cells the integrands are smooth functions and the integrals can be approximated efficiently by a higher order quadrature formula. To this end, we use a sparse grid quadrature scheme [6] based on univariate Gauss–Patterson [16] rules. The number of quadrature points of such a sparse grid formula is given $n_{\text{IN}} = O(2^q q^{d-1})$ where q denotes the refinement level of the employed univariate quadrature rule; i.e., the

[3] In fact, only the integrals associated with the stiffness matrix and the mass matrix are polynomials. The exact integration of the load vector is usually not possible.

[4] Note that this bound on the computational complexity does not directly involve the required quality of the numerical integration scheme. The parameters, however, must be chosen such that the numerical integration is accurate enough.

number of quadrature points of the univariate rule is of the order $O(2^q)$. The polynomial exactness of the Gauss–Patterson rule on level q is $3 \cdot 2^{q-1} - 1$.

Let us assume that the integrands on the integration cells can be approximated accurately by a polynomial of degree $(p+1)^2$.[5] Hence, we can estimate the cost $\mathcal{C}_{\mathrm{A,NI}}$ associated with the assembly of the stiffness matrix in our implementation by

$$\mathcal{C}_{\mathrm{A,NI}} = O(Np(\ln p)^{d-1}(dp + p^d + p^{2d})).$$

Hence, our implementation is optimal up a factor of $O(p(\ln p)^{d-1})$, under the assumptions stated above. The results of our numerical experiments indicate that our numerical integration scheme gives an accurate and stable approximation of the stiffness matrix.[6]

3 Multilevel Solution of Resulting Linear System

In the following we focus on the solution of the large sparse linear system $A\tilde{u} = \hat{f}$ where $\tilde{u}$ denotes a coefficient vector and $\hat{f}$ denotes a moment vector. This solution step is a very time consuming part of any numerical simulation. The use of an inappropriate solver can drive up the compute time as well as the storage demand dramatically.

Classical direct solvers for dense matrices like Gaussian elimination or LU-decomposition have a storage requirement of $O(\mathsf{dof}^2)$ and the number of operations even scales with $O(\mathsf{dof}^3)$, where dof denotes the number of degrees of freedom. More advanced sparse direct solvers can reduce these complexities to some extent only. In the special case of regular meshes in two dimensions for instance a nested dissection solver requires $O(\mathsf{dof}^{3/2})$ operations and $O(\mathsf{dof}\ln(\mathsf{dof}))$ storage [5]. For our PUM space we have $\mathsf{dof} = O(Np^d)$ where $N = \mathrm{card}(C_\Omega)$ denotes the number of patches ω_i and p the order of approximation. The number of nonzeros entries of a PUM stiffness matrix is of the order $O(Np^{2d})$. Hence, this optimal storage and operation complexity will be lost when a direct solver is employed.

Alternatively, an iterative scheme like the Jacobi- or Gauss–Seidel method can be used. Here, we do not have a significant increase in the storage requirements, but the number of operations necessary to obtain the solution of

[5] This assumption can be justified by the structure of φ_i and our choice of α and $\mathcal{W}$.

[6] In fact, we use an adaptive version of the quadrature scheme with a dynamic stopping criterion to ensure the quality of the numerical integration also for problems with non-constant coefficients, see [8, 18]. Note that it is essential to analyze the interplay of the numerical integration error and the approximation error to be able to develop an assembly scheme for the stiffness matrix with optimal complexity and optimal approximation properties. Such an analysis would allow to determine the required tolerance of the numerical quadrature automatically and can help to minimize the computational costs associated with the assembly of the stiffness matrix.

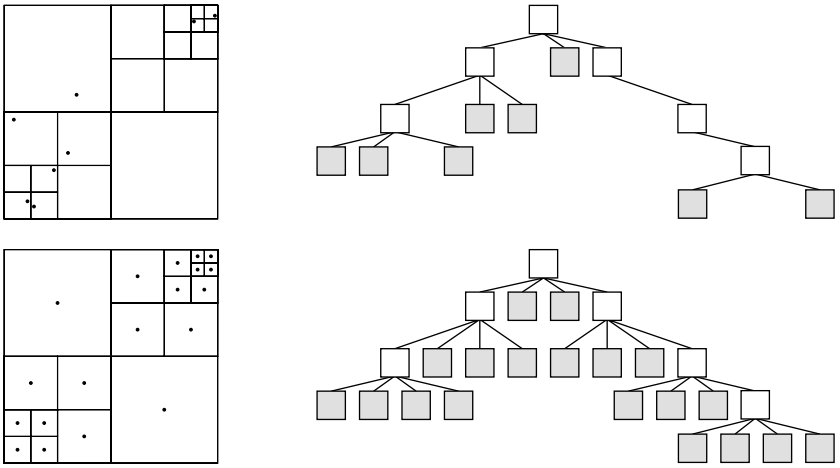


Figure 3.1. Hierarchical cover construction in two dimensions. The cell decomposition induced by the initial point set (upper left) and its corresponding tree representation (upper right, white: INNER tree nodes, gray shaded: LEAF tree nodes). Here, the leaves of the tree correspond to the given initial points x_i. The final cell decomposition with all generated points x_L (lower left) and its tree representation (lower right) after the completion of the cover construction. Now, the leaves of the tree correspond to the points $x_L \in P_J$.

the linear system up to a prescribed accuracy does not scale with the optimal complexity. A sophisticated class of iterative methods which not only show an optimal scaling in the storage demand but also in the operation count are so-called multilevel iterative solvers or multigrid methods [12, 21]. These solvers, however, are not general algebraic methods but involve a substantial amount of information about the discretization and possibly the PDE. We have developed a first multilevel solver for PUM discretizations in [9, 18]. The convergence rate of this solver is independent of the number of patches N, but not of the approximation order p. Now we present an extension of this multilevel solver which gives a convergence rate that is also independent of p. To this end, let us shortly review the multilevel construction from [9, 18].

The first step toward the design of an efficient multilevel solver is the construction of an appropriate sequence of function spaces. To this end, we have developed a hierarchical multilevel cover construction algorithm [8, 9, 18] which gives a sequence of point sets $P_k = \{x_{i,k}\}$ and covers $C_\Omega^k = \{\omega_{i,k}\}$. The algorithm is based on so-called d-binary trees (quadtrees, octrees) and ensures the covering property on all levels $k = 0, \ldots, J$, i.e., $\bigcup_{\omega_{i,k} \in C_\Omega^k} \omega_{i,k} \supset \Omega$, see Figures 3.1 and 3.2. We define the cover patches $\omega_{i,k} = \alpha\, \mathcal{C}_{i,k}$ via a scaling of the cells $\mathcal{C}_{i,k}$ of the tree decomposition by a scalar factor $\alpha \in (1, 2)$, see Figure 3.3. The overlap parameter α must be larger than 1 to obtain (at least) a $\mathcal{C}^0$

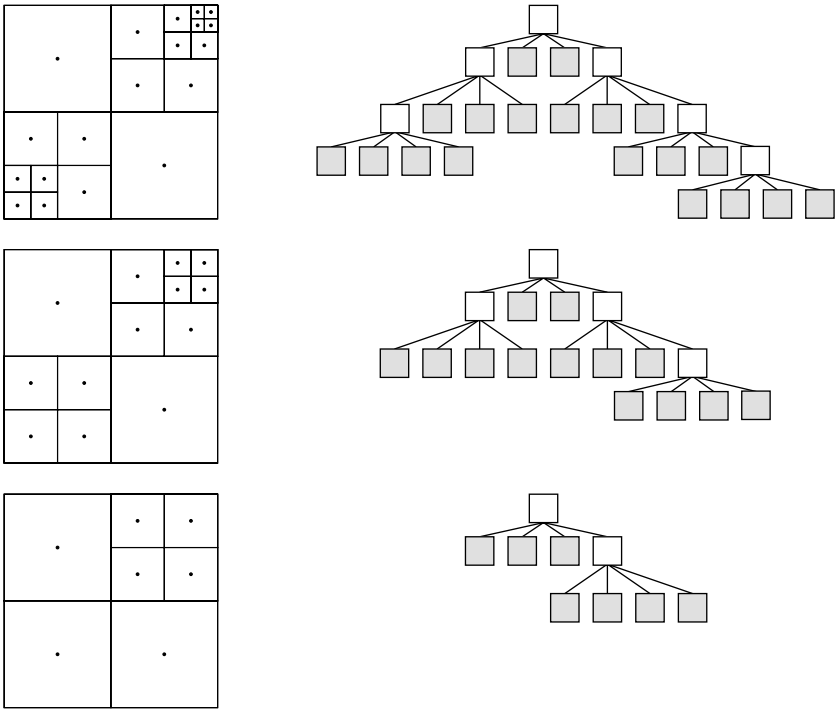


Figure 3.2. Multilevel cover sequence in two dimensions. The cell decompositions and its respective tree representation (upper right, white: INNER tree nodes, gray shaded: LEAF tree nodes) for the fine level point set $P_J = P_4$ (upper row), and two coarser level point sets P_3 (center row) and P_2 (lower row). The leaves of the respective tree correspond to the points $x_L \in P_k$.

PU (independent of the employed weight function $\mathcal{W}$) and should be smaller than 2 to ensure the linear independence of the resulting shape functions.

Note that the underlying tree data structure also allows for an efficient neighbor search for general point sets P_J and it reduces the computational effort associated with numerical integration, see [8, 9, 18] for details. Furthermore, it can be used for the approximation of the domain Ω, see Figure 3.4. Via the general PUM construction given in section 2 we then obtain the sequence of PUM function spaces V_k^{PU} associated with the sequence of covers C_Ω^k. Note that these spaces are nonnested, i.e., $V_{k-1}^{\mathrm{PU}} \not\subset V_k^{\mathrm{PU}}$, and that the shape functions $\varphi_{i,k}\psi_{i,k}^n$ are non-interpolatory. Thus, the natural injection and a direct interpolation between two successive PUM spaces V_{k-1}^{PU} and V_k^{PU} are not available. Therefore, we need to construct appropriate prolongation operators $I_{k-1}^k : V_{k-1}^{\mathrm{PU}} \to V_k^{\mathrm{PU}}$ and restriction operators $I_k^{k-1} : V_k^{\mathrm{PU}} \to V_{k-1}^{\mathrm{PU}}$ to transfer information between the PUM spaces.

To this end, we can use an L^2-projection as prolongation between two spaces V_{k-1}^{PU} and V_k^{PU}. However, a global L^2-projection is prohibitively expen-

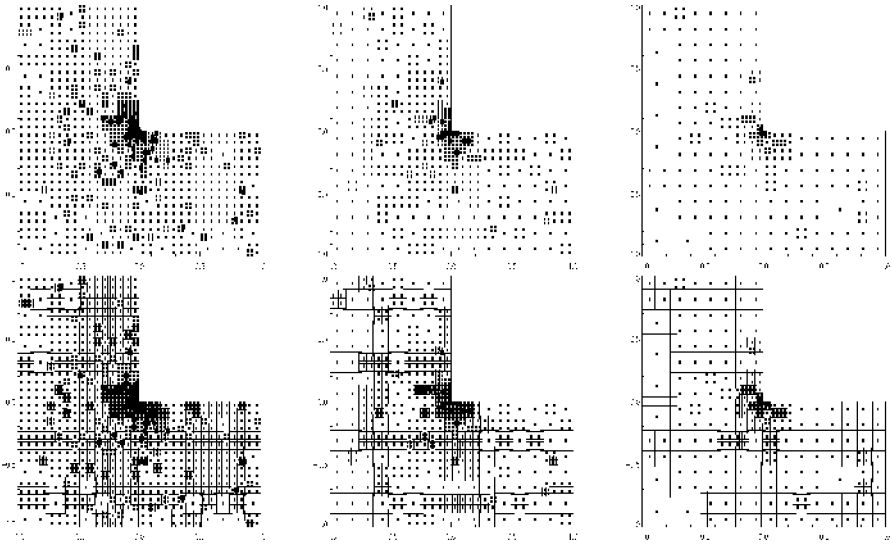


Figure 3.3. Point sets P_k (upper row) and covers C_Ω^k (lower row) for $k = 10, \ldots, 8$ for a graded initial point set.

sive. Yet, within the PUM context we can construct a very cheap prolongation operator using a localized L^2-projection approach. Based on the error estimates for the PUM [1, 2] and the geometric hierarchy of our tree construction, we have developed this cheap but qualitatively good projection. Here, we only give a short review over the construction principles, see [9, 18] for details.

The localization of the L^2-projection within our PUM consists of two steps. At first consider the basic PUM error estimate

$$\|v - v^{\mathrm{PU}}\|^2_{L^2(\Omega)} \leq C \sum_i \|v - v_i\|^2_{L^2(\omega_i \cap \Omega)}, \tag{3.1}$$

where $v^{\mathrm{PU}} := \sum_i \varphi_i \sum_n u_i^n \psi_i^n$ and $v_i := \sum_n u_i^n \psi_i^n$. From (3.1) we know that it is sufficient to control the local errors $\|v - v_i\|_{L^2(\omega_i \cap \Omega)}$ on each cover patch ω_i. Now choose $v = u_{k-1}^{\mathrm{PU}} = \sum_j \varphi_{j,k-1} u_{j,k-1} = \sum_j \varphi_{j,k-1} \sum_m u_{j,k-1}^m \psi_{j,k-1}^m$ and $v^{\mathrm{PU}} = I_{k-1}^k u_{k-1}^{\mathrm{PU}} = \sum_i \varphi_{i,k} u_{i,k} = \sum_i \varphi_{i,k} \sum_n u_{i,k}^n \psi_{i,k}^n$ so that (3.1) reads

$$\|u_{k-1}^{\mathrm{PU}} - I_{k-1}^k u_{k-1}^{\mathrm{PU}}\|^2_{L^2(\Omega)} \leq C \sum_i \|u_{k-1}^{\mathrm{PU}} - u_{i,k}\|^2_{L^2(\omega_{i,k} \cap \Omega)}. \tag{3.2}$$

Hence, we observe that we can approximate the global coarse function u_{k-1}^{PU} locally on the fine cover patches $\omega_{i,k}$ using the local basis functions $\psi_{i,k}^n$, rather than approximating u_{k-1}^{PU} by the global shape functions $\varphi_{i,k}\psi_{i,k}^n$ on the finer level k. Now in a second step we establish an upper bound for each of the

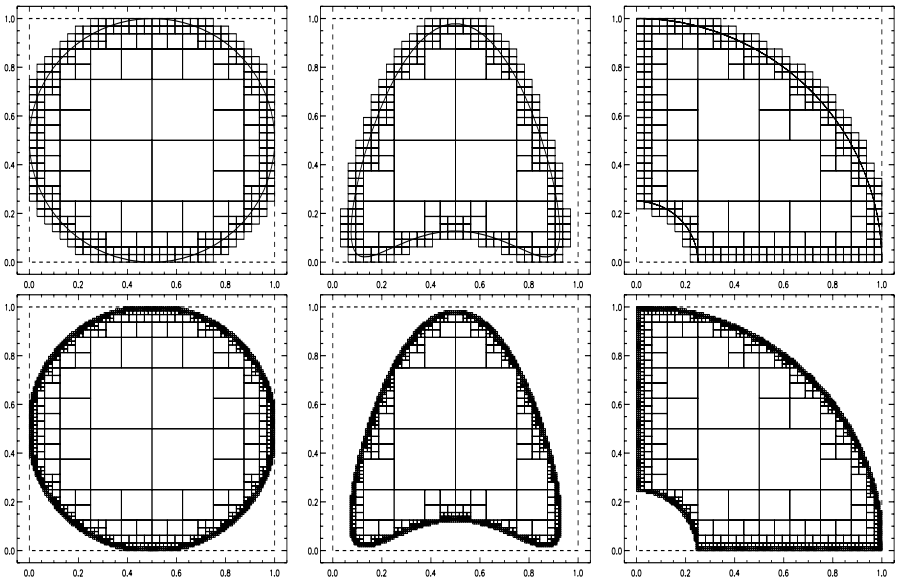


Figure 3.4. Tree-based approximation of a spherical domain (left), a smooth non-convex domain (center), and a quarter of a spherical domain with a spherical hole (right) on level $k = 5$ (upper row) and level $k = 7$ (lower row).

terms on the right-hand side of (3.2) utilizing the geometric hierarchy of our tree. Due to our tree-based cover construction we can find exactly one coarse patch $\omega_{\tilde{i},k-1}$ for every fine patch $\omega_{i,k}$ such that $\omega_{i,k} \subset \omega_{\tilde{i},k-1}$ holds. Hence, we can introduce the respective coarse local function $u_{\tilde{i},k-1}$ associated with the unique coarse patch $\omega_{\tilde{i},k-1}$ into each term $\|u_{k-1}^{\mathrm{PU}} - u_{i,k}\|_{L^2(\omega_{i,k}\cap\Omega)}$ of (3.2). Finally, we obtain the estimate

$$\begin{aligned}\|u_{k-1}^{\mathrm{PU}} - u_{i,k}\|_{L^2(\omega_{i,k}\cap\Omega)} \leq \|u_{k-1}^{\mathrm{PU}} - u_{\tilde{i},k-1}\|_{L^2(\omega_{i,k}\cap\Omega)} + \\ \|u_{\tilde{i},k-1} - u_{i,k}\|_{L^2(\omega_{i,k}\cap\Omega)}\end{aligned} \tag{3.3}$$

by the triangle inequality. This estimate allows us to approximate each coarse local function $u_{\tilde{i},k-1}$, independent of all other local components $u_{j,k-1}$ of u_{k-1}^{PU}, on the respective fine cover patch $\omega_{i,k}$ with $\omega_{i,k} \subset \omega_{\tilde{i},k-1}$ since the first term of (3.3) is small by definition of u_{k-1}^{PU}. Hence, we can set up our prolongation operators via the so-called local-to-local L^2-projection. To this end, we project each local approximation $u_{i,k-1}$ on level $k-1$ independently to the finer level k using the hierarchical condition $\omega_{i,k} \subseteq \omega_{\tilde{i},k-1}$ instead of the geometric neighbor relation $\omega_{i,k} \cap \omega_{j,k-1} \neq \emptyset$ only. The respective matrix representation of this prolongation is given by

$$I_{k-1}^k := \widetilde{\Pi}_{k-1}^k := (\widetilde{M}_k^k)^{-1}(\widetilde{M}_{k-1}^k) \qquad \text{with}$$
$$(\widetilde{M}_k^k)_{(i,n),(i,m)} := \langle \psi_{i,k}^m, \psi_{i,k}^n \rangle_{L^2(\omega_{i,k}\cap\Omega)} \qquad \text{and}$$
$$(\widetilde{M}_{k-1}^k)_{(i,n),(\tilde{i},m)} := \langle \psi_{i,k-1}^m, \psi_{i,k}^n \rangle_{L^2(\omega_{i,k}\cap\Omega)}.$$

The storage requirement of this local-to-local projection is minimal. We need to store only a single block-entry $(\widetilde{M}_k^k)_{i,i}^{-1}(\widetilde{M}_{k-1}^k)_{i,\tilde{i}}$ for each patch $\omega_{i,k}$ on level k. Moreover, the respective integrals involve only the local basis functions $\psi_{i,k-1}^m$ and $\psi_{i,k}^n$ and can be computed very efficiently. Overall, the local-to-local projection operator can be computed with $O(Np^{3d})$ operations in general (and with $O(Np^d)$ if we use orthogonal polynomials locally). Furthermore, it is exact for polynomials of degree p and therefore suitable also for higher order approximations.

In summary, we now have a sequence of stiffness matrices A_k coming from the direct Galerkin approximation of the respective bilinear form $a_k(\cdot,\cdot)$ on each level k and a sequence of high-quality transfer operators I_{k-1}^k, I_k^{k-1} based on localized L^2-projections. As the final ingredient for our multilevel solver, see Algorithm 3.1, we now need to construct a sequence of appropriate smoothing operators S_k.

ALGORITHM 3.1 (Multilevel Algorithm $M_\gamma^{\nu_1,\nu_2}(k, x_k, b_k)$)

1. If $k > 0$:
   a) For $l = 1, \ldots, \nu_1$: Set $x_k = S_k^{\text{pre}}(x_k, b_k)$.
   b) Set $d_{k-1} := I_k^{k-1}(b_k - A_k x_k)$.
   c) Set $e_{k-1} := 0$.
   d) For $i = 1, \ldots, \gamma$: $e_{k-1} = M_\gamma^{\nu_1,\nu_2}(k-1, e_{k-1}, d_{k-1})$.
   e) Set $x_k = C_k(x_k, e_{k-1}) := x_k + I_{k-1}^k e_{k-1}$.
   f) For $l = 1, \ldots, \nu_2$: Set $x_k = S_k^{\text{post}}(x_k, b_k)$.
2. Else:
   a) Set $x_k = A_k^{-1} b_k$.

Many iterative solvers such as the classical Jacobi- and Gauss–Seidel iterations, the overlapping domain decomposition methods and even multigrid methods can be interpreted in the framework of subspace correction methods (SCM) [3, 4, 13, 15, 21, 22]. Hence, let us shortly review the abstract setting of an SCM.

The general idea is as follows: First, we write the discretization space $\mathcal{V} = \sum_{j=1}^{\mathcal{N}} \mathcal{V}_j$ as the sum[7] of subspaces $\mathcal{V}_j$ with maps $P_j : \mathcal{V}_j \to \mathcal{V}$.[8] Then, we choose symmetric positive definite bilinear forms $b_j(\cdot,\cdot)$ on each $\mathcal{V}_j$ represented by operators B_j such that solutions to the systems of linear equations $B_j u_j = f_j$ on $\mathcal{V}_j$ are easily computable, and B_j^{-1} can be considered as an approximate inverse to the restriction of A to $\mathcal{V}_j$. Finally, we combine these local

[7] Note that we do not assume that the splitting is a direct sum.

[8] Actually, it is sufficient to require $\mathcal{V} = \sum_j P_j \mathcal{V}_j$, i.e., the condition $\mathcal{V}_j \subset \mathcal{V}$ is not necessary.

approximate inverses B_j^{-1} appropriately to define a global approximate inverse to A on the discretization space $\mathcal{V}$. There are essentially two approaches to the definition of an approximate inverse of A by the B_j^{-1}, the additive approach and the multiplicative approach.

In the so-called parallel subspace correction (PSC) or additive Schwarz method we set up an iterative solution process via the operator

$$M_{\mathrm{PSC}} := \mathbb{I} - \omega \sum_{j=1}^{\mathcal{N}} P_j T_j = \mathbb{I} - \omega \Big(\sum_{j=1}^{\mathcal{N}} P_j B_j^{-1} R_j \Big) A, \tag{3.4}$$

where ω is a relaxation parameter and the involved operators are defined by

$$a(u,v) = \langle Au, v \rangle_{\mathcal{V}}, \quad b_j(u_j, v_j) = \langle B_j u_j, v_j \rangle_{\mathcal{V}_j},$$
$$\langle R_j u, v_j \rangle_{\mathcal{V}} = \langle u, P_j v_j \rangle_{\mathcal{V}}, \; b_j(T_j u, v_j) = a(u, P_j v_j).$$

The iteration operator of the successive subspace correction (SSC) or multiplicative Schwarz method is given by

$$M_{\mathrm{SSC}} := \prod_{j=1}^{\mathcal{N}} \big(\mathbb{I} - P_j T_j \big) = \prod_{j=1}^{\mathcal{N}} \big(\mathbb{I} - P_j B_j^{-1} R_j A \big). \tag{3.5}$$

Note that the PSC operator (3.4) can also be interpreted as a preconditioned Richardson iteration where the preconditioner is given by

$$\mathcal{C}_{PSC} := \sum_{j=1}^{\mathcal{N}} P_j B_j^{-1} R_j. \tag{3.6}$$

Let us now restrict ourselves to the case of $B_j := A|_{\mathcal{V}_j}$ which means that we consider exact subspace solvers only. Then, we have essentially two degrees of freedom in the design of our smoothing scheme: The splitting of the discretization space and the type of the iteration, namely the additive scheme (3.4) or the multiplicative scheme (3.5). To define an appropriate splitting of our PUM space V^{PU} (we omit the level index k in the following), let us consider the specific structure of the PUM shape functions. The product structure of the shape functions $\varphi_i \psi_i^n$ implies two natural subspace definitions. For instance, we can define the subspaces $\mathcal{V}_n := \mathrm{span}_i \langle \varphi_i \psi_i^n \rangle := \{ v \in V^{\mathrm{PU}} \,|\, v = \sum_i \varphi_i v_i^n \psi_i^n \}$. These subspaces, however, contain functions with *global* support on the domain Ω, see Figure 3.5 (left), and the dimension of each subspace is of the order $O(N)$. Therefore, a direct solution of $A|_{\mathcal{V}_n}$ is not feasible. We would need to resort to fast iterative solution techniques for these subspace problems. Furthermore, we are interested in smoothing schemes S_k for Algorithm 3.1 based on our spatial multilevel construction. Hence, there is no additional benefit from the fact that the solutions to $A|_{\mathcal{V}_n}$ contain global information and the computational cost associated with the solution of the subspace problems make this splitting unsuitable for our construction.

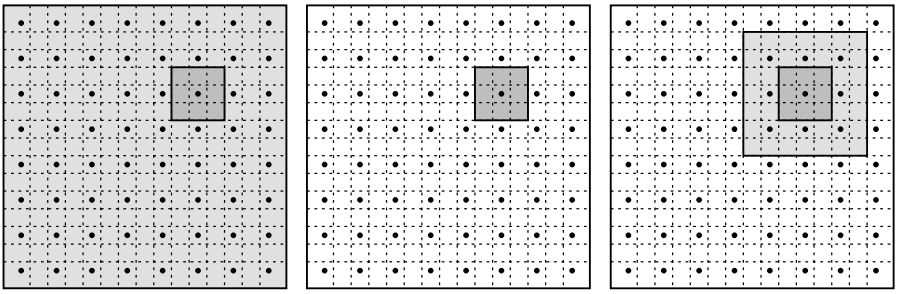


Figure 3.5. Subdomains (light gray shaded) associated with the subspaces $\mathcal{V}_n$ (left), $\mathcal{V}_i$ (center), and $\widetilde{\mathcal{V}}_l$ (right) and the support of the a single shape function $\varphi_i\psi_i^n$ (dark gray shaded) based on a cover with $\alpha = 1.5$.

A more appropriate subspace definition is given by $\mathcal{V}_i := \varphi_i V_i^p = \mathrm{span}_n\langle\varphi_i\psi_i^n\rangle := \{v \in V^{\mathrm{PU}} \,|\, v = \sum_n \varphi_i v_i^n \psi_i^n\}$. These spaces contain functions with local supports only, see Figure 3.5 (center). Furthermore, the dimension of the subspace $\mathcal{V}_i$ is given by the dimension $O(p^d)$ of the local approximation spaces V_i^p. Hence, we can compute the inverse $(A|_{\mathcal{V}_i})^{-1}$ of each of the subspace problems with acceptable complexity of $O(p^{3d})$; i.e., one iteration of a PSC or SSC iteration based on this splitting is of the order $O(Np^{3d})$.

Note that both subspace definitions lead to a *direct* splitting of our PUM function space $V^{\mathrm{PU}} = \sum_i \mathcal{V}_i = \sum_n \mathcal{V}_n$; i.e., every basis function $\varphi_i\psi_i^n$ is contained in exactly one subspace. In terms of the index pairs (i,n) we have a disjoint decomposition of the index set $\{(i,n)\}$ which induces a specific partitioning of the PUM stiffness matrix $A = (A_{(i,n),(j,m)})$. Using the subspaces $\mathcal{V}_i$ we obtain the so-called polynomial block-form. Here, a single block $A_{i,j} = (A_{(i,n),(j,m)})$ corresponds to a local discretization of the PDE on the domain $\omega_i \cap \omega_j \cap \Omega$.

Note that a PSC iteration (3.4) based on the direct splitting $V^{\mathrm{PU}} = \sum_i \mathcal{V}_i$ corresponds to the classical block-Jacobi iteration and the SSC iteration (3.5) corresponds to the block-Gauss–Seidel iteration (BGS) where we have only a small overlap between the supports of functions from different subspaces, see Figure 3.5 (center). Even though we consider a direct splitting and employ an exact solver $(A|_{\mathcal{V}_i})^{-1}$ within a specific subspace $\mathcal{V}_i$ there are still couplings between the subspaces due to the overlap of the supports via the global problem A. The quality of the PSC and SSC iterations is obviously determined by the strength of these couplings. The two parameters within our PUM which can influence the strength of the couplings between two different subspaces $\mathcal{V}_i$ and $\mathcal{V}_j$, and hence the quality of the iterations, are the overlap parameter α used in our cover construction and the polynomial degree p. Since we are interested in a smoothing scheme that is truly robust, i.e., that works with the same quality at least for a large range of parameters α and p, this direct splitting approach cannot be pursued.

One approach to overcome this problem is to consider subspace splittings $V^{\mathrm{PU}} = \sum_l \widetilde{\mathcal{V}}_l$ which are no longer direct splittings, i.e., a basis function $\varphi_i \psi_i^n$ may now belong to several subspaces $\widetilde{\mathcal{V}}_l$. Consider the subspace definition

$$\widetilde{\mathcal{V}}_l := \sum_{\omega_i \cap \omega_l \neq \emptyset} \mathcal{V}_i = \mathrm{span}_{(i,n), i \in C_l} \langle \varphi_i \psi_i^n \rangle \tag{3.7}$$

where $C_l := \{i \,|\, \omega_i \cap \omega_l \neq \emptyset\}$ denotes the neighborhood of the cover patch ω_l, see Figure 3.5 (right). The subspace $\widetilde{\mathcal{V}}_l$ contains *all* functions $\varphi_i \psi_i^n$ whose support ω_i has a non-vanishing intersection with the patch ω_l. Hence, when we solve the subspace problem $A|_{\widetilde{\mathcal{V}}_l}$ we resolve all couplings involving the basis functions $\varphi_l \psi_l^q$. Thus, for each patch ω_l there is one subspace problem $A|_{\widetilde{\mathcal{V}}_l}$ which resolves *all* couplings involving the associated basis functions $\varphi_l \psi_l^q$ independent of the overlap parameter α and the polynomial degree p. Note that this splitting is similar to the one employed in [17].

Since the subspace splitting into the $\widetilde{\mathcal{V}}_l$ is not a direct splitting it does not correspond to a simple partitioning scheme of the stiffness matrix A. Here, we rather have to assemble the (discrete) local subproblems $A_{l,l}$ from the global linear system via the so-called Galerkin products $A_{l,l} := P_l^T A P_l$ where P_l denotes the discrete extension operator which embeds the subspace $\widetilde{\mathcal{V}}_l$ in the global PUM space V^{PU}. In our case P_l is just a mask matrix, i.e., a reduced identity matrix. With the matrices A and P_l the application of the SSC iteration operator (3.5) to a linear system $A\tilde{u} = \hat{f}$ can be realized by Algorithm 3.2. In the following we refer to this iteration as a multiplicative overlapping Schwarz (MOS) smoother.

Algorithm 3.2 (Successive subspace correction method)

1. For all $l = 1, \ldots, N$:
   a) Compute local residual $\hat{f}_l := P_l^T(\hat{f} - A\tilde{u})$.
   b) Solve subspace problem $(P_l^T A\ P_l)\tilde{u}_l = A_{l,l}\tilde{u}_l = \hat{f}_l$.
   c) Update global iterate $\tilde{u} = \tilde{u} + P_l \tilde{u}_l$.

In Figure 3.6 we give the smoothing results obtained after one iteration of the BGS and the MOS smoother for $p = 1$ and $p = 5$. From these surface plots we can clearly observe that the MOS smoother gives much smoother iterates than the BGS smoother. More notably, the quality of the BGS smoother deteriorates for higher order approximations. The results for $p = 5$ are not as smooth as for $p = 1$. For the MOS smoother we find a completely different behavior. There is no deterioration in the quality for larger p. In fact it even seems that the results for $p = 5$ are better than for $p = 1$.

Note that Algorithm 3.1 corresponds to an SSC iteration based on a multilevel subspace splitting. Similarly, we can define a PSC type multilevel iteration and the associated preconditioner (3.6). For instance if apply (3.4) not only the sum of all local subspaces (3.7) on a particular level k but rather to the sum of all local subspaces $\widetilde{\mathcal{V}}_{l,k}$ on all levels k, i.e., $\sum_k \sum_l \widetilde{\mathcal{V}}_{l,k}$, together

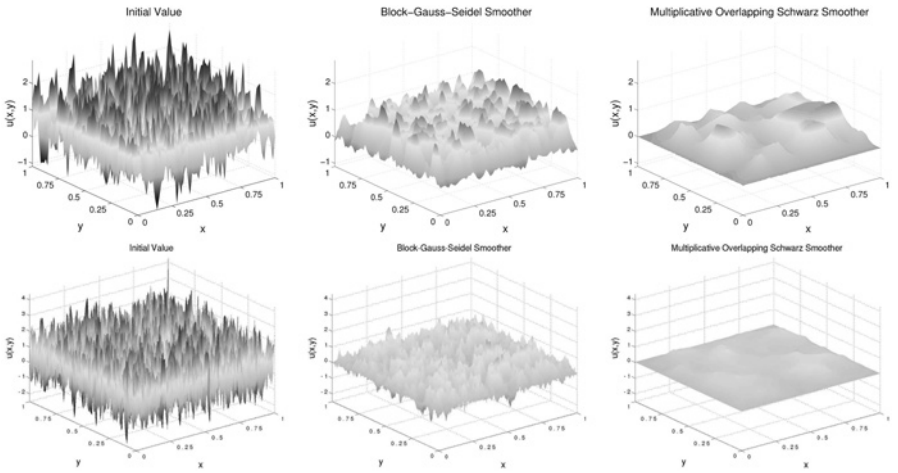


Figure 3.6. Random valued initial guess (left) and smoothing results using a block-Gauss–Seidel (center) and a multiplicative overlapping Schwarz smoother (right). Depicted are the current iterates after a single application of the smoother. The discretization was based on a uniform node arrangement on level 5 and employed polynomials of degree $p = 1$ (upper row) and $p = 5$ (lower row).

with the mask matrices $P_{l,k}$, the prolongations I_{k-1}^k and the restrictions I_k^{k-1}, we obtain a multilevel PSC iteration. The application of the associated multilevel additive Schwarz ($MAOS$) preconditioner (3.6) can be realized with Algorithm 3.3.

Algorithm 3.3 (Multilevel parallel subspace correction preconditioner)

1. Set initial value $\tilde{u}_J = 0$.
2. For all levels $k = J, \ldots, 1$:
   a) Set right-hand side $\hat{f}_{k-1} = I_k^{k-1} \hat{f}_k$.
   b) Set initial value $\tilde{u}_{k-1} = 0$.
3. For all levels $k = 0, \ldots, J$:
   a) For all $l = 1, \ldots, N_k$:
      i. Set local right-hand side $\hat{f}_{k,l} := P_{k,l}^T(\hat{f}_k)$.
      ii. Solve subspace problem $(P_{l,k}^T A_k P_{l,k})\tilde{u}_{l,k} = (A_k)_{l,l}\tilde{u}_{k,l} = \hat{f}_{k,l}$.
      iii. Update global iterate $\tilde{u}_k = \tilde{u}_k + P_{l,k}\tilde{u}_{l,k}$.
4. For all levels $k = 1, \ldots, J$:
   a) Update global iterate $\tilde{u}_k = \tilde{u}_k + I_{k-1}^k \tilde{u}_{k-1}$.

In many cases the convergence behavior of the PSC schemes and the respective SSC schemes are similar. Yet, the PSC methods have certain advantages with respect to parallelization.

4 Numerical Results

In all our experiments with our multilevel solver (see Algorithm 3.1) and the multilevel preconditioner (see Algorithm 3.3) we solve a linear system of the form

$$A\tilde{u} = \hat{f} = 0.$$

We choose a vanishing right-hand side vector $\hat{f} = 0$ and a random initial guess u_0^{PU} with $\|u_0^{\mathrm{PU}}\|_{L^2} \approx 1$. Here, we approximate the L^2-norm of the function u_0^{PU} by

$$\|u_0^{\mathrm{PU}}\|_{L^2}^2 \approx \tilde{u}_0^T M \tilde{u}_0 =: \|\tilde{u}_0\|_{L^2} = 1,$$

where $\tilde{u}_0$ denotes the random valued coefficient vector associated with the function u_0^{PU} and $M := \langle \varphi_j \psi_j^m, \varphi_i \psi_i^n \rangle_{L^2}$ is the mass matrix. The stopping criterion for the iteration is $\|\tilde{u}_r\|_{L^2} < 10^{-13}$ (or $r = 50$ for Algorithm 3.1 and $r = 250$ for Algorithm 3.3) where r denotes the number of iterations. Hence, the average convergence or error reduction rate of our multilevel iteration with respect to the L^2-norm[9] of the error is given by

$$\rho_{L^2} := \left(\frac{\tilde{u}_r^T M \tilde{u}_r}{\tilde{u}_0^T M \tilde{u}_0} \right)^{\frac{1}{r}} = \left(\frac{\|\tilde{u}_r\|_{L^2}}{\|\tilde{u}_0\|_{L^2}} \right)^{\frac{1}{r}} = \|\tilde{u}_r\|_{L^2}^{\frac{1}{r}}.$$

Furthermore, we also compute the more common convergence rates

$$\rho_{l^2} := \left(\frac{\|\tilde{u}_r\|_{l^2}}{\|\tilde{u}_0\|_{l^2}} \right)^{\frac{1}{r}} \quad \text{and} \quad \rho_R := \left(\frac{\|A\tilde{u}_r\|_{l^2}}{\|A\tilde{u}_0\|_{l^2}} \right)^{\frac{1}{r}}$$

which are based on the l^2-norm of the current coefficient vector $\tilde{u}_r$ and the respective residual vector $A\tilde{u}_r$. These convergence rates are given for multilevel iterations $M_1^{1,1}$, the so-called $V(1,1)$-cycle, based on the local-to-local L^2-projection using the non-overlapping BGS smoother and the overlapping MOS smoother. The quality of both these SSC smoothers is dependent on the ordering of the respective subspaces, i.e., of the cover patches. In all our experiments we use a space filling curve ordering scheme based on the Hilbert curve for the cover patches, see [9, 18] for details. Besides the convergence rates ρ^{BGS} and ρ^{MOS} we also give the finest discretization level J, the polynomial degree p of the local approximation spaces V_i^p, the respective number of degrees of freedom dof, and the number of nonzeros entries nnz of the stiffness matrix A_J on the finest level. In all experiments, we used a linear B-spline as the generating weight function $\mathcal{W}$, a uniform node arrangement as initial point set, and Legendre polynomials up to degree 9 as local approximation spaces in our PUM discretization.

[9] Note that even for uniform covers C_Ω we have no uniform correspondence (with respect to the number of degrees of freedom) between the L^2-norm of a function u^{PU} and the l^2-norm of its corresponding coefficient vector $\tilde{u}$ due to the use of local polynomials, just like in the p-version of the finite element method.

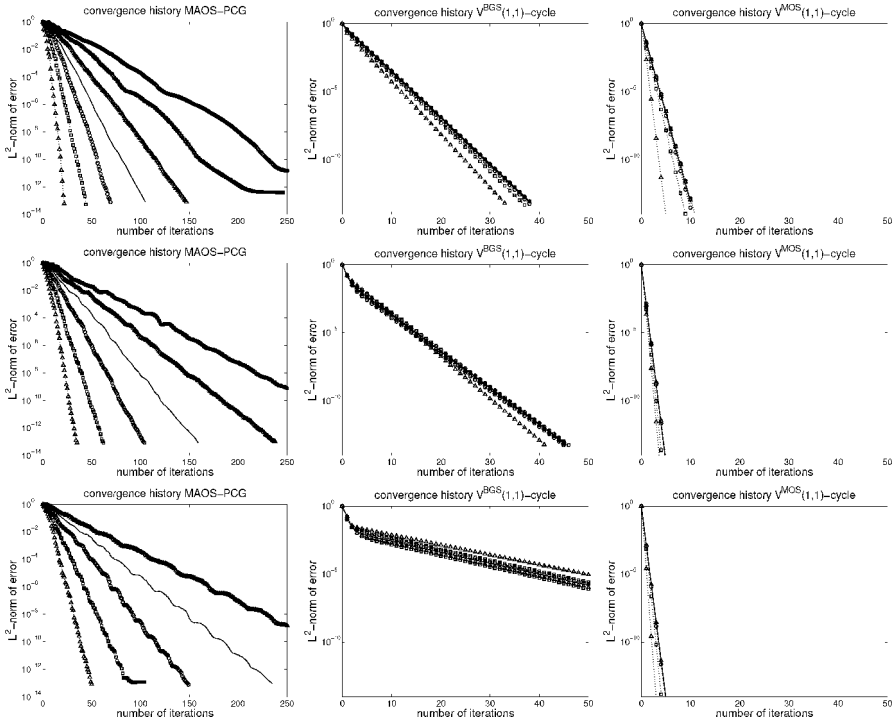


Figure 4.7. Convergence history for a preconditioned conjugate gradient solver with the $MAOS$ preconditioner (left), the $V(1,1)$-cycle using the BGS smoother (center) and the MOS smoother (right), for the Poisson problem (4.1) in two dimensions. Depicted are the results for $p = 1$ on levels $2, \ldots, 8$ (top row), $p = 3$ on levels $2, \ldots, 7$ (center row), and $p = 5$ on levels $2, \ldots, 6$ (bottom row).

In our first experiment, we considered the Poisson problem

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega = [0,1]^d \subset \mathbb{R}^d, \\ u &= g \quad \text{on } \partial\Omega, \end{aligned} \tag{4.1}$$

in two and three dimensions with $f = 0$ and Dirichlet boundary conditions $g = 0$. The results of this experiment are summarized in Figure 4.7 and in Tables 4.1 and 4.2. From the plots depicted in Figure 4.7 we clearly observe that the convergence behavior of a conjugate gradient method preconditioned with the $MAOS$ scheme (Algorithm 3.3, Figure 4.7 (left)) is dependent on the number of points N as well as the polynomial degree p. The multilevel iteration $M_1^{1,1}$ (Algorithm 3.1), i.e., the $V(1,1)$-cycle, converges independent of the number of points N for the BGS smoother (Figure 4.7 (center)) but not independent of the polynomial degree p. Only the $V(1,1)$-cycle with the MOS smoother (Figure 4.7 (right)) shows a convergence behavior that is independent of the number of points N and the polynomial degree p. Hence, in

Table 4.1. Convergence rates ρ^{BGS} obtained for a $V(1,1)$-cycle using the BGS smoother, and the rates ρ^{MOS} using the overlapping MOS smoother for the Poisson problem (4.1) in two dimensions.

J	p	dof	nnz	$\rho^{BGS}_{L^2}$	$\rho^{BGS}_{l^2}$	ρ^{BGS}_{R}	$\rho^{MOS}_{L^2}$	$\rho^{MOS}_{l^2}$	ρ^{MOS}_{R}
8	1	196608	5280804	0.261	0.256	0.220	0.039	0.037	0.020
8	2	393216	21123216	0.166	0.160	0.150	0.002	0.001	0.001
7	3	163840	14592400	0.324	0.337	0.322	0.001	0.001	0.001
6	4	61440	8122500	0.580	0.607	0.581	0.002	0.001	0.001
6	5	86016	15920100	0.692	0.736	0.695	0.001	0.001	0.001
5	6	28672	6927424	0.791	0.868	0.754	0.002	0.004	0.001
5	7	36864	11451456	0.811	0.892	0.781	0.001	0.003	0.001
5	8	46080	17892900	0.829	0.910	0.813	0.002	0.002	0.001
4	9	14080	6400900	0.850	0.926	0.830	0.001	0.002	0.001

Table 4.2. Convergence rates ρ^{BGS} obtained for a $V(1,1)$-cycle using the BGS smoother, and the rates ρ^{MOS} using the overlapping MOS smoother for the Poisson problem (4.1) in three dimensions.

J	p	dof	nnz	$\rho^{BGS}_{L^2}$	$\rho^{BGS}_{l^2}$	ρ^{BGS}_{R}	$\rho^{MOS}_{L^2}$	$\rho^{MOS}_{l^2}$	ρ^{MOS}_{R}
5	1	131072	13289344	0.319	0.312	0.266	0.034	0.032	0.020
4	2	40960	9733600	0.192	0.182	0.163	0.002	0.002	0.001
3	3	10240	4259200	0.443	0.460	0.430	< 0.001	< 0.001	< 0.001
3	4	17920	13043800	0.657	0.689	0.637	< 0.001	< 0.001	< 0.001

contrast to FEM the additive Schwarz or PSC preconditioner (Algorithm 3.3) does not yield a similar convergence behavior as the multiplicative Schwarz or SSC iteration (Algorithm 3.1).

The rates ρ^{BGS}, see Tables 4.1 and 4.2, measured for the $V(1,1)$-cycle with the BGS smoother are clearly p-dependent and deteriorate with increasing p. We measure $\rho^{BGS}_{L^2} = 0.261$ for $p = 1$ and $\rho^{BGS}_{L^2} = 0.850$ for $p = 9$. The rates ρ^{MOS} obtained with the MOS smoother, however, are almost constant for all polynomial degrees p. Furthermore, these rates are very small. We find $\rho^{MOS}_{L^2}$ to be no worse than 0.002 for $p \geq 2$; i.e., our solver converges up to machine accuracy in less than 6 iterations. Note that due to storage limitations not all experiments with increasing p could be carried out up to the same discretization level J. Here, we were limited by the number of nonzeros nnz of the stiffness matrix A_J on the finest level J. From the numbers given in Tables 4.1 and 4.2 we see that nnz on the finest level J is similar for all experiments and ranges from 5 million to 21 million entries.

In our second experiment we considered the Navier–Lamé equations

$$\begin{aligned} -\mu \boldsymbol{\Delta} u - (\lambda + \mu) \nabla (\nabla \cdot u) &= f \quad \text{in} \quad \Omega = [0,1]^2 \subset \mathbb{R}^2, \\ u &= g \quad \text{on } \partial\Omega, \end{aligned} \tag{4.2}$$

in two dimensions with $f = 0$ and Dirichlet boundary conditions $g = 0$. The measured convergence rates ρ for our multilevel solver with the BGS

Table 4.3. Convergence rates ρ^{BGS} obtained for a $V(1,1)$-cycle using the BGS smoother, and the rates ρ^{MOS} using the overlapping MOS smoother for the elasticity problem (4.2) in two dimensions.

J	p	dof	nnz	$\rho^{BGS}_{L^2}$	$\rho^{BGS}_{l^2}$	ρ^{BGS}_{R}	$\rho^{MOS}_{L^2}$	$\rho^{MOS}_{l^2}$	ρ^{MOS}_{R}
8	1	393216	21123216	0.664	0.658	0.552	0.128	0.124	0.074
7	2	196608	21013056	0.347	0.336	0.277	0.008	0.007	0.003
6	3	81920	14440000	0.671	0.689	0.655	0.010	0.011	0.009
5	4	30720	7952400	0.786	0.813	0.764	0.002	0.001	< 0.001
5	5	43008	15586704	0.791	0.831	0.778	0.002	0.002	< 0.001
4	6	14336	6635776	0.827	0.882	0.805	0.002	0.003	0.001
4	7	18432	10969344	0.857	0.908	0.831	0.002	0.002	0.001
3	8	5760	3920400	0.871	0.927	0.836	0.001	0.001	< 0.001

smoother and the MOS smoother are given in Table 4.3. Again we observe that the rates ρ^{BGS} for the BGS smoother deteriorate for larger p. The rates ρ^{MOS} obtained for the MOS smoother on the other hand are very small and stay constant e.g. $\rho^{MOS}_{L^2} \approx 0.002$ for increasing $p \geq 2$.

In summary, the results of our numerical experiments indicate that only the multilevel solver (Algorithm 3.1) with the overlapping MOS smoother (Algorithm 3.2) converges independent of N and p. Furthermore, the measured convergence rates ρ^{MOS} are very small. It took no more than 6 iterations for the solver to converge up to machine accuracy for $p \geq 2$.

Note, however, that the computational costs associated with the overlapping smoothing scheme involve a rather large and d-dependent constant. Due to the overlap of the subspace splitting, the local problems are of dimension $O(3^d p^d)$ whereas the dimension of the local problems without overlap is $O(p^d)$, see Figure 3.5 (center) and (right). Since these local problems lead to dense matrices in both cases, the storage requirement of the overlapping MOS smoother is larger by a factor of 3^{2d} than the storage demand of the non-overlapping BGS smoother. Similarly, the number of operations required by the MOS smoother is larger by a factor of 3^{3d} due to the direct solution of the larger local problems. Hence, the overall compute time of the multilevel solver with the non-optimal BGS smoother may be significantly smaller in the practical range of N and p. Furthermore, the multilevel solver with the BGS smoother is applicable to larger problems due to its smaller storage demand.

5 Concluding Remarks

We presented a multilevel solver for PUM discretizations which employs a tree-based spatial multilevel sequence in conjunction with an overlapping domain decomposition type smoothing scheme. The results of our numerical experiments indicate that the convergence rate ρ of this multilevel solver is independent of the number of points N and the approximation order p; i.e.,

the solver is robust with respect to p. The overall computational complexity of our solver is of the order $O(Np^{3d})$ to reduce the initial error by a prescribed factor which is optimal up to a factor of $O(p^d)$. Note, however, that the constants involved are rather large.

These results hold also for irregular point distributions and different cycle types. For instance, also the $V(1,0)$-cycle with the MOS smoother based on a graded point set converges independent of N and p. Furthermore, the rate of the multilevel solver with the MOS smoother seems to robust also for general domains, i.e., when the domain is not resolved on the coarsest level.

Acknowledgement. Finally, we would like to acknowledge the support of the Sonderforschungsbereich 611 „Singuläre Phänomene und Skalierung in mathematischen Modellen“ funded by the Deutsche Forschungsgemeinschaft.

References

1. I. Babuška and J. M. Melenk. The Partition of Unity Finite Element Method: Basic Theory and Applications. *Comput. Meth. Appl. Mech. Engrg.*, 139:289–314, 1996. Special Issue on Meshless Methods.
2. I. Babuška and J. M. Melenk. The Partition of Unity Method. *Int. J. Numer. Meth. Engrg.*, 40:727–758, 1997.
3. J. H. Bramble and X. Zhang. Handbook of Numerical Analysis. In P. G. Ciarlet and J. L. Lions, editors, *The Analysis of Multigrid Methods*, volume VII, pages 173–416. Elsevier, 2000.
4. W. Dahmen. Multiscale Analysis, Approximation, and Interpolation Spaces. In C. K. Chui and L. L. Schumaker, editors, *Approximation Theory VIII*, volume 2, pages 47–88. World Scientific, 1995.
5. J. A. George. Nested Dissection of a Regular Finite Element Mesh. *SIAM J. Num. Anal.*, 10:345–363, 1973.
6. T. Gerstner and M. Griebel. Numerical Integration using Sparse Grids. *Numer. Alg.*, 18:209–232, 1998.
7. M. Griebel and M. A. Schweitzer. A Particle-Partition of Unity Method for the Solution of Elliptic, Parabolic and Hyperbolic PDE. *SIAM J. Sci. Comput.*, 22(3):853–890, 2000.
8. M. Griebel and M. A. Schweitzer. A Particle-Partition of Unity Method—Part II: Efficient Cover Construction and Reliable Integration. *SIAM J. Sci. Comput.*, 23(5):1655–1682, 2002.
9. M. Griebel and M. A. Schweitzer. A Particle-Partition of Unity Method—Part III: A Multilevel Solver. *SIAM J. Sci. Comput.*, 24(2):377–409, 2002.
10. M. Griebel and M. A. Schweitzer. A Particle-Partition of Unity Method—Part IV: Parallelization. In M. Griebel and M. A. Schweitzer, editors, *Meshfree Methods for Partial Differential Equations*, volume 26 of *Lecture Notes in Computational Science and Engineering*, pages 161–192. Springer, 2002.
11. M. Griebel and M. A. Schweitzer. A Particle-Partition of Unity Method—Part V: Boundary Conditions. In S. Hildebrandt and H. Karcher, editors, *Geometric Analysis and Nonlinear Partial Differential Equations*, pages 517–540. Springer, 2002.

12. W. Hackbusch. *Multi-Grid Methods and Applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer, 1985.
13. W. Hackbusch. *Iterative Solution of Large Sparse Linear Systems of Equations*. Springer, 1994.
14. J. Nitsche. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abh. Math. Sem. Univ. Hamburg*, 36:9–15, 1970–1971.
15. P. Oswald. *Multilevel Finite Element Approximation*. Teubner Skripten zur Numerik. Teubner, 1994.
16. T. N. L. Patterson. The Optimum Addition of Points to Quadrature Formulae. *Math. Comp.*, 22:847–856, 1968.
17. L. F. Pavarino. Additive Schwarz Methods for the p-Version Finite Element Method. *Numer. Math.*, 66:493–515, 1994.
18. M. A. Schweitzer. *A Parallel Multilevel Partition of Unity Method for Elliptic Partial Differential Equations*, volume 29 of *Lecture Notes in Computational Science and Engineering*. Springer, 2003.
19. T. Strouboulis, I. Babuška, and K. Copps. The Design and Analysis of the Generalized Finite Element Method. *Comput. Meth. Appl. Mech. Engrg.*, 181(I 3):43–69, 2000.
20. T. Strouboulis, K. Copps, and I. Babuška. The Generalized Finite Element Method. *Comput. Meth. Appl. Mech. Engrg.*, 190:4081–4193, 2001.
21. J. Xu. Iterative Methods by Space Decomposition and Subspace Correction. *SIAM Review*, 34(4):581–613, 1992.
22. H. Yserentant. Old and New Convergence Proofs for Multigrid Methods. *Acta Numerica 93*, pages 285–326, 1993.

Enriched Reproducing Kernel Approximation: Reproducing Functions with Discontinuous Derivatives

Pierre Joyot[1⋆], Jean Trunzler[1,2⋆⋆], and Fransisco Chinesta[2⋆⋆⋆]

[1] LIPSI-ESTIA, Technopole Izarbel, 64210 Bidart, France.
[2] LMSP, 151 Bd. de l'Hôpital, 75013 Paris, France.

Abstract In this paper we propose a new approximation technique within the context of meshless methods able to reproduce functions with discontinuous derivatives. This approach involves some concepts of the reproducing kernel particle method (RKPM), which have been extended in order to reproduce functions with discontinuous derivatives. This strategy will be referred as Enriched Reproducing Kernel Particle Approximation (E-RKPA). The accuracy of the proposed technique will be compared with standard RKP approximations (which only reproduces polynomials).

1 Introduction

Meshless methods are an appealing choice to develop functional approximations (with different degrees of consistency and continuity) without a mesh support. Thus, this kind of techniques seem to be specially appropriated for treating 3D problems involving large displacements, due to the fact that the approximation is constructed only from the cloud of nodes whose position evolve during the material deformation. In this manner neither remeshing nor fields projections are a priori required.

In order to adapt the approximation for introducing some known information associated to the searched solution different possibilities exist. The first one lies in the enrichment of standard approximations in the framework of the partition of unity (PU), originally proposed by Babuška and Melenk [3], and which is at the origin of the partition of unity finite elements (PUFEM) [12]. Generalized finite elements (G-FEM) [15] or extended finite elements (X-FEM) [13, 16, 17, 9] are two members of this family. In the extended finite elements the approximation is locally enhanced by introducing appropriate functions describing the known behavior of the problem solution.

⋆ p.joyot@estia.fr
⋆⋆ j.trunzler@estia.fr
⋆⋆⋆ francisco.chinesta@paris.ensam.fr

The second possibility lies in the introduction of some information related to the problem solution within the approximation functional basis. For this purpose, different reproduction conditions are enforced in the construction of the approximation functions. This approach has been widely used in the context of the moving least squares approximations used in the diffuse meshless techniques [2] as well as in the element free Galerkin method [4]. Thus, all the known information related to the problem solution can be introduced in the functional approximation. Very accurate results were obtained for example in fracture mechanics by introducing the crack tip behavior into the approximation basis [6].

In this work we propose a numerical strategy, based on the reproducing kernel particle techniques, able to construct approximation functions with discontinuous derivatives on fixed or moving interfaces. This problem was treated in the context of the partition of unity by Kronggauz et al. [10]. In our approach the size of the discrete system of equations remains unchanged because no additional degrees of freedom are introduced related to the enrichment. However, the fact of enriching the approximation implies, as shown later, a bigger moment matrix, with worse properties, but in some cases this enrichment is only local, and in any case, the moment matrix has a low dimension.

The starting point of our development is the reproducing kernel particle approximation (RKPA). The RKP approximation was introduced by Liu et al. [11] for enforcing some degree of consistency to standard smooth particle approximations, i.e. they proved that starting from a SPH (smooth particle hydrodynamics) approximation [7] it is possible to enhance the kernel function for reproducing a certain degree of polynomials. We have extended or generalized this procedure in order to reproduce any function, and more concretely, functions involving discontinuous derivatives.

Moreover, the strategy here described, can be easily coupled with a level set description of the interface location [14] (recently introduced in the framework of the extended finite elements [17]), which allows to capture its position when it evolves in time. Thus, both the enrichment and the interface tracking are made in a transparent way for the user.

This paper consists of 6 sections, in which we summarize the reproducing kernel approximation (section 2) which allows to treat in section 3 the reproduction of a function with discontinuous derivatives. In sections 4 and 5 we describe and illustrate the properties of the resulting approximation. Finally, in section 6 we focus on a test problem which allows to conclude about the accuracy and potentiality of the proposed strategy.

2 Enriched Reproducing Kernel Particle Approximation

Let Ω be the domain where the problem is defined. The points within this domain will be noted by x or s. For the sake of simplicity, from now on, we will focus in the 1D case, but all the results have a direct 2D or 3D counterpart.

2.1 Reproduction Conditions

The approximation $f^a(x)$ of $f(x)$ is built from the integral convolution integral

$$f^a(x) = \int_\Omega \Phi(x-s,h) f(s) d\Omega, \tag{2.1}$$

where $\Phi(x-s,h)$ is the kernel function and h a parameter defining the size of the approximation support.
The main aim of this work is to enforce the reproduction of a general function that we can write in the form of a polynomial plus another function noted by $f^e(x)$:

$$f^a(x) = a_0 + a_1 x + \ldots + a_n x^n + f^e(x). \tag{2.2}$$

In the following paragraphs we analyze the required properties of the kernel function $\Phi(x-s,h)$ for reproducing a function expressed by (2.2).
From Eq. (2.1), the reproduction of a constant function a_0 is given by

$$\int_\Omega \Phi(x-s,h) a_0 d\Omega = a_0, \tag{2.3}$$

which implies

$$\int_\Omega \Phi(x-s,h) d\Omega = 1, \tag{2.4}$$

which constitutes the partition of unity property.
Now, the required condition to reproduce a linear function $f^a(x) = a_0 + a_1 x$ is

$$\int_\Omega \Phi(x-s,h)(a_0 + a_1 s) d\Omega = a_0 + a_1 x. \tag{2.5}$$

By using the partition of unity (2.4), Eq. (2.5) can be rewritten as

$$\begin{cases} \int_\Omega \Phi(x-s,h) d\Omega = 1 \\ \int_\Omega \Phi(x-s,h) s d\Omega = x \end{cases}, \tag{2.6}$$

which implies the linear consistency of the approximation. Repeating this reasoning, we can write the n-order consistency as

$$\begin{cases} \int_\Omega \Phi(x-s,h) d\Omega = 1 \\ \int_\Omega \Phi(x-s,h) s d\Omega = x \\ \vdots \\ \int_\Omega \Phi(x-s,h) s^n d\Omega = x^n \end{cases} \tag{2.7}$$

and consequently, the reproduction of the function given by (2.2) implies

$$\int_\Omega \Phi(x-s,h)(a_0 + a_1 s + \ldots + a_n s^n + f^e(s)) d\Omega \; =$$

$$= a_0 + a_1 x + \ldots + a_n x^n + f^e(x), \tag{2.8}$$

from which it results

$$\begin{cases} \int_\Omega \Phi(x-s,h)d\Omega = 1 \\ \int_\Omega \Phi(x-s,h)s d\Omega = x \\ \vdots \\ \int_\Omega \Phi(x-s,h)s^n d\Omega = x^n \\ \int_\Omega \Phi(x-s,h)f^e(s)d\Omega = f^e(x) \end{cases}. \tag{2.9}$$

Chen et al. [5] applied the procedure proposed by Liu et al. [11] for enforcing n-order consistency, however their procedure does not allow to enforce directly the reproduction condition associated with $f^e(x)$.

2.2 The Moment Matrix

We will note by $f^r(x)$ the approximation function verifying the conditions (2.9). Usually a cubic spline is considered as kernel function, and consequently the conditions given by Eq. (2.9) are not satisfied. Liu et al. [11] propose the introduction of a correction function $C(x, x-s)$ for satisfying the reproduction conditions. In our case we consider the more general form $C(x, s, x-s)$ whose pertinence will be discussed later. Thus $f^r(x)$ will be expressed by

$$f^r(x) = \int_\Omega C(x,s,x-s)\Phi(x-s,h)f(s)d\Omega, \tag{2.10}$$

where $C(x, s, x-s)$ is assumed to have the following form

$$C(x,s,x-s) = \boldsymbol{H}^T(x,s,x-s)\boldsymbol{b}(x), \tag{2.11}$$

where $\boldsymbol{H}^T(x, s, x-s)$ represents the vector containing the functions considered in the approximation basis, and $\boldsymbol{b}(x)$ is a vector containing unknown functions that will be determined for satisfying the reproduction conditions. Thus, Eq. (2.9) can be rewritten as

$$\begin{cases} \int_\Omega \boldsymbol{H}^T(x,s,x-s)\boldsymbol{b}(x)\Phi(x-s,h)d\Omega = 1 \\ \int_\Omega \boldsymbol{H}^T(x,s,x-s)\boldsymbol{b}(x)\Phi(x-s,h)s d\Omega = x \\ \vdots \\ \int_\Omega \boldsymbol{H}^T(x,s,x-s)\boldsymbol{b}(x)\Phi(x-s,h)s^n d\Omega = x^n \\ \int_\Omega \boldsymbol{H}^T(x,s,x-s)\boldsymbol{b}(x)\Phi(x-s,h)f^e(s)d\Omega = f^e(x) \end{cases}. \tag{2.12}$$

Remark: Usual RKP approximations consider $\boldsymbol{H}^T(x-s)$, but we retain the more general form $\boldsymbol{H}^T(x, s, x-s)$ in order to reproduce more general functions.

In fact, the reproduction conditions must be enforced in a discrete form. For this purpose we consider NP points (also refereed as nodes) which allow to compute the discrete form of Eq. (2.12), i.e.

$$\begin{cases} \sum_{I=1}^{NP} \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{b}(x)\Phi(x - x_I, h)\Delta x_I = 1 \\ \sum_{I=1}^{NP} \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{b}(x)\Phi(x - x_I, h)x_I\Delta x_I = x \\ \vdots \\ \sum_{I=1}^{NP} \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{b}(x)\Phi(x - x_I, h)x_I^n\Delta x_I = x^n \\ \sum_{I=1}^{NP} \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{b}(x)\Phi(x - x_I, h)f^e(x_I)\Delta x_I = f^e(x) \end{cases}, \tag{2.13}$$

that in a matrix form results

$$\left[\sum_{I=1}^{NP} \boldsymbol{R}(x_I)\boldsymbol{H}^T(x, x_I, x - x_I)\Phi(x - x_I, h)\Delta x_I\right]\boldsymbol{b}(x) = \boldsymbol{R}(x), \tag{2.14}$$

where $\boldsymbol{R}(x)$ is the reproduction vector

$$\boldsymbol{R}^T(x) = [1, x, \ldots, x^n, f^e(x)]\,. \tag{2.15}$$

Eq. (2.14) allows the computation of vector $\boldsymbol{b}(x)$,

$$\boldsymbol{b}(x) = \boldsymbol{M}(x)^{-1}\boldsymbol{R}(x), \tag{2.16}$$

where the moment matrix $\boldsymbol{M}(x)$ is defined by

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \boldsymbol{R}(x_I)\boldsymbol{H}^T(x, x_I, x - x_I)\Phi(x - x_I, h)\Delta x_I. \tag{2.17}$$

This moment matrix differs of the usual moment matrix proposed in [11], and in fact it becomes non symmetric.

2.3 Discrete Form of the Approximation Function

The discrete form $f^r(x)$ of $f^a(x)$ derives from Eqs. (2.10), (2.11) and (2.16)

$$\begin{aligned} f^r(x) &\cong \sum_{I=1}^{NP} \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{M}(x)^{-1}\boldsymbol{R}(x)\Phi(x - x_I, h)f(x_I)\Delta x_I \\ &= \sum_{I=1}^{NP} \psi_I(x)f_I, \end{aligned} \tag{2.18}$$

where ψ_I is the enriched RKP approximation shape function

$$\psi_I(x) = \boldsymbol{H}^T(x, x_I, x - x_I)\boldsymbol{M}(x)^{-1}\boldsymbol{R}(x)\Phi(x - x_I, h)\Delta x_I. \tag{2.19}$$

As in the classical RKPM we take $\Delta x_I = 1$. Different quadrature rules exist and they have been tested in [1] without a significant incidence on the reproducing condition accuracy.

3 The Case of a Function with Discontinuous Derivatives

Let Ω be the domain where the problem is defined and Γ_d a point, curve or surface (in 1D, 2D and 3D respectively) where the normal function derivative becomes discontinuous. We assume that this discontinuity curve splits the domain in two subdomains Ω_0 and Ω_1 (see Figure 3.1)

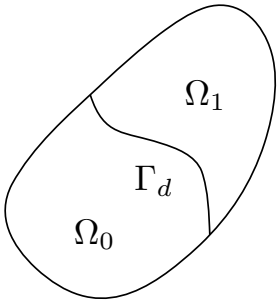


Figure 3.1. Problem domain containing an interface with a discontinuous normal derivative.

$$\begin{gathered}\Omega_0 \cup \Omega_1 \cup \Gamma_d = \Omega, \\ \Omega_0 \cap \Omega_1 = \emptyset,\end{gathered}$$

where Γ_d represents the curve along which the normal derivative of some approximation functions become discontinuous.

The enrichment function $f^e(\boldsymbol{x})$ which will be introduced in the reproduction vector $\boldsymbol{R}(x)$ must satisfy the following conditions

$$\begin{gathered}f^e(\boldsymbol{x}) \in \mathcal{C}^0(\boldsymbol{x} \in \Omega), \\ f^e(\boldsymbol{x}) \in \mathcal{C}^1(\boldsymbol{x} \in \Omega_0), \\ f^e(\boldsymbol{x}) \in \mathcal{C}^1(\boldsymbol{x} \in \Omega_1), \\ Grad\, f^e(\boldsymbol{x}) \cdot \boldsymbol{n} \in \mathcal{C}^{-1}(\boldsymbol{x} \in \Gamma_d),\end{gathered} \tag{3.1}$$

where $\boldsymbol{n}$ denotes the unit outward vector defined on the curve Γ_d.

To locate the discontinuity curve Γ_d we have use of a level set function $\Theta(\boldsymbol{x})$ defined as the signed distance from $\boldsymbol{x}$ to the interface Γ_d. Thus

$$\Theta(\boldsymbol{x}) = \begin{cases} \Theta(\boldsymbol{x}) < 0 \text{ if } \boldsymbol{x} \in \Omega_0 \\ \Theta(\boldsymbol{x}) > 0 \text{ if } \boldsymbol{x} \in \Omega_1 \\ \Theta(\boldsymbol{x}) = 0 \text{ if } \boldsymbol{x} \in \Gamma_d \end{cases}. \tag{3.2}$$

Thus, for satisfying the conditions (3.1) the enrichment function $f^e(\boldsymbol{x})$ could be assumed in the form

$$f^e(\boldsymbol{x}) = H_0(\Theta(\boldsymbol{x}))\Theta(\boldsymbol{x}), \tag{3.3}$$

where $H_0(\boldsymbol{x})$ represents the usual Heaviside function

$$\begin{cases} H_0(\Theta(\boldsymbol{x})) = 1 \; if \; \Theta(\boldsymbol{x}) \geq 0 \\ H_0(\Theta(\boldsymbol{x})) = 0 \; if \; \Theta(\boldsymbol{x}) < 0 \end{cases}. \tag{3.4}$$

4 Properties of the Moment Matrix

If the moment matrix is computed as described in section 2 everywhere, it can become singular, as we will prove later. We can define two boundaries in the discontinuity curve neighborhood Γ_0 and Γ_1 (as illustrated in figure 4.2)

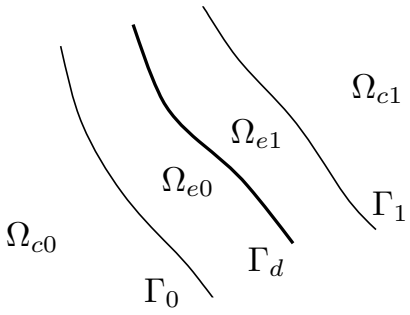


Figure 4.2. Enriched and non-enriched domains.

such that now the domain Ω consists of four subdomains Ω_{c0}, Ω_{e0}, Ω_{c1} and Ω_{e1}, where classical and enriched reproducing kernel approximation will be defined with continuity along Γ_0 and Γ_1 as proved later. Thus, we start from the relations

$$\Omega_0 = \Omega_{c0} \cup \Omega_{e0} \cup \Gamma_0,$$
$$\Omega_1 = \Omega_{c1} \cup \Omega_{e1} \cup \Gamma_1.$$

Obviously, Γ_0 and Γ_1 define the points, curves or surfaces (1D, 2D and 3D respectively) in Ω_0 and Ω_1 respectively, through which the determinant of the moment matrix changes from zero to a non-null value.

We define $\boldsymbol{R}_c^T(x) = [1, x, \ldots, x^n]$, $\boldsymbol{R}^T(x) = \left[\boldsymbol{R}_c^T(x), H_0(\Theta(x))\Theta(x)\right]$, and in a similar manner $\boldsymbol{H}_c^T(x-x_I) = [1, x-x_I, \ldots, (x-x_I)^n]$, and $\boldsymbol{H}^T(x, x_I, x-x_I) = \left[\boldsymbol{H}_c^T(x-x_I), H_0(\Theta(x_I))\Theta(x_I)\right]$.

The following properties of the moment matrix can be deduced:

Property 1: The inverse of the moment matrix exists in the domains Ω_{e0} and Ω_{e1}, i.e. in the subdomains whose boundary involves the discontinuity curve Γ_d. Then, the moment matrix results

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \begin{bmatrix} \boldsymbol{R}_c(x_I)\boldsymbol{H}_c^T(x-x_I) & \boldsymbol{R}_c(x_I)H_0(\Theta(x_I))\Theta(x_I) \\ H_0(\Theta(x_I))\Theta(x_I)\boldsymbol{H}_c^T(x-x_I) & H_0(\Theta(x_I))^2\Theta(x_I)^2 \end{bmatrix} \Phi(x-x_I,h)\Delta x_I.$$

Property 2: In the domain Ω_{c0}, $H_0(\Theta(x_I)) = 0$, $\forall I$. Thus, the moment matrix becomes singular

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \begin{bmatrix} \boldsymbol{R}_c(x_I)\boldsymbol{H}_c^T(x-x_I) & 0 \\ 0 & 0 \end{bmatrix} \Phi(x-x_I,h)\Delta x_I.$$

In this case we propose to replace this moment matrix for the one derived from a standard (non-enriched) approximation

$$\boldsymbol{M}_c(x) = \sum_{I=1}^{NP} \boldsymbol{R}_c(x)\boldsymbol{H}_c^T(x - x_I)\Phi(x - x_I, h)\Delta x_I. \tag{4.1}$$

Property 3: Within the domain Ω_{c1}, $H_0(\Theta(x_I)) = 1$, $\forall I$. Thus, $\boldsymbol{M}(x)$ results

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \begin{bmatrix} \boldsymbol{R}_c(x_I)\boldsymbol{H}_c^T(x - x_I) & \boldsymbol{R}_c(x_I)\Theta(x_I) \\ \Theta(x_I)\boldsymbol{H}_c^T(x - x_I) & \Theta(x_I)^2 \end{bmatrix} \Phi(x - x_I, h)\Delta x_I.$$

The moment matrix is singular in 1D (as will be illustrated in section 5.1) as well as in some 2D and 3D cases. In this case we replace again the enriched moment matrix for the non-enriched one. Effectively, if the enriched moment matrix is not singular everywhere in Ω_{c1}, then Γ_1 does not exist, i.e. $\Omega_{c1} = \emptyset$ et $\Omega_{e1} = \Omega_1$.

Property 4: When we approach the curve Γ_0 the shape functions computed from both the enriched and the non-enriched approximations are connected with continuity, i.e.

$$\lim_{x \in \Omega_{c0} \to \Gamma_0} \psi_I(x) = \lim_{x \in \Omega_{e0} \to \Gamma_0} \psi_I(x), \quad \forall I.$$

And in a similar way when we approach the curve Γ_1 (when it exists)

$$\lim_{x \in \Omega_{c1} \to \Gamma_1} \psi_I(x) = \lim_{x \in \Omega_{e1} \to \Gamma_1} \psi_I(x), \quad \forall I.$$

This property has important consequences because due to the continuity of the shape functions within the whole domain, the numerical integration of variational formulations does not require a specific treatment.

Property 5: Let ζ_I be the boundary of the support related to the node x_I. The following results can be stated:

The curves Γ_0 and Γ_1 belong to the union of the support boundary ζ_I of some nodes. In the 1D case Γ_0 and Γ_1 reduce to two points (as illustrated in section 5). In higher dimensions, both boundaries are composed by a series of arcs of circle (sphere in 3D) (when the support of the kernel function is assumed circular -spherical in 3D-).

Property 6: The boundary ζ_I associated with the node $x_I \in \Omega_1$ (respectively $x_I \in \Omega_0$) closest to the discontinuity interface Γ_d defines the boundary Γ_0 (respectively Γ_1).

5 About the Resulting Shape Function and its Derivatives

In this section we focus in a one-dimensional problem in order to illustrate some of the previous properties. The signed distance is used for defining the level set function

$$\Theta(x) = x - x_d, \tag{5.1}$$

where x_d represents the coordinate of the point where the discontinuity in the function derivative takes place.

5.1 Moment Matrix

For the sake of clarity, from now on, we restrict the consistency requirement to a first order. Thus, the moment matrix results

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \widetilde{\boldsymbol{M}}_I(x)\Phi(x - x_I, h)$$

with

$$\widetilde{\boldsymbol{M}}_I(x) = \begin{bmatrix} 1 & x-x_I & H_0(x_I-x_d)(x_I-x_d) \\ x_I & x_I(x-x_I) & x_I H_0(x_I-x_d)(x_I-x_d) \\ H_0(x_I-x_d)(x_I-x_d) & H_0(x_I-x_d)(x_I-x_d)(x-x_I) & (H_0(x_I-x_d)(x_I-x_d))^2 \end{bmatrix}.$$

For illustrating the third property, we consider $x \in \Omega_{c1}$, which leads to the following expression of the moment matrix

$$\boldsymbol{M}(x) = \sum_{I=1}^{NP} \begin{bmatrix} 1 & x-x_I & (x_I-x_d) \\ x_I & x_I(x-x_I) & x_I(x_I-x_d) \\ (x_I-x_d) & (x_I-x_d)(x-x_I) & (x_I-x_d)^2 \end{bmatrix} \Phi(x-x_I,h).$$

We can notice that the third row is a linear combination of the two first. Thus, the determinant vanishes and the moment matrix becomes singular. This result does not depend on the degree of consistency assumed.

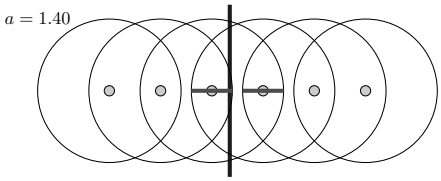
Now we are going to illustrate the position of the boundaries Γ_0 and Γ_1 through which the approximation changes from enriched to non-enriched.

In Figure 5.3 the vertical line indicates the position where the function derivative is expected to be discontinuous, and the horizontal one represents the enrichment domain $\Omega_{e1} \cup \Omega_{e0}$. The distance between two consecutive nodes (assumed regularly spaced) is $h = 1$ and the support radius is taken in the form $R = a \times h$ (where the value of a is indicated in each subfigure).

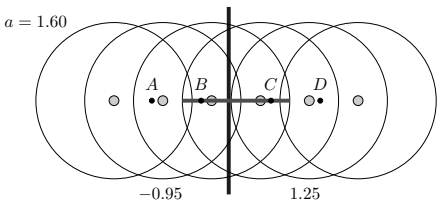
It is well known that for a standard RKP approximation (without any enrichment), the linear consistency implies that each point must be contained by more than one nodal support (in other case the moment matrix becomes singular) [8]. In our case we need to add to this constraint the properties listed in the previous section.

From Figure 5.3(b) we can notice:

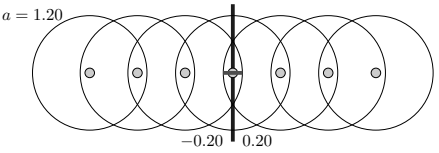
- Point A is in the domain Ω_{c0}, i.e. $A \in \Omega_{c0}$ because any node whose support contains the point A is not in Ω_1 (*Property 2*).
- $D \in \Omega_{c1}$ because any node whose support contains the point D is not in Ω_0 (*Property 3*).



(a) Node not located at the discontinuity: discontinuous $\Omega_{e0} \cup \Omega_{e1}$.



(b) Node not located at the discontinuity: continuous $\Omega_{e0} \cup \Omega_{e1}$.



(c) Node located at the discontinuity.

Figure 5.3. Enriched and non-enriched domains in a one-dimensional test problem.

- $B \in \Omega_{e0}$ and $C \in \Omega_{e1}$ (*Property 1*).
- The position of Γ_1 (respectively Γ_0) is defined by the support of the node within Ω_0 (respectively Ω_1) closest to the discontinuity interface Γ_d (*Properties 5–6*).

If $a < 1.5$ there are points, candidates to belong to $\Omega_{e0} \cup \Omega_{e1}$, such that they belong only to the support of two nodes. In this way the approximation is only defined if $a > 1.5$ (Figure 5.3(a)).

When there is a node located at the discontinuity point this problem is avoided as depicted in Figure 5.3(c). Thus, the approximation is defined only if $a > 1$, like in the standard (non-enriched) RKP approximation.

5.2 Related Shape Functions

Figure 5.4 depicts the shape functions related to each node considered in the interval $\Omega = [-1, 1]$ with $x_d = 0$ and $NP = 10$ ($h = 2/9$). The support radius $a \times h$ is defined by the value of the parameter a, that has been fixed to $a = 2$. Thus, the boundaries defining the transition between the enriched and non-enriched domain are located at $x_0 = -1.5 \times h = -\frac{1}{3}$ and $x_1 = 1.5 \times h = \frac{1}{3}$. Figure 5.5 shows the associated derivatives, whereas Figure 5.6 depicts

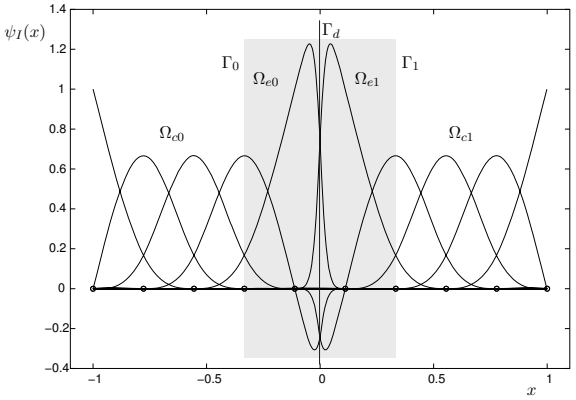


Figure 5.4. Shape functions for a discontinuous derivative located at $x_d = 0$.

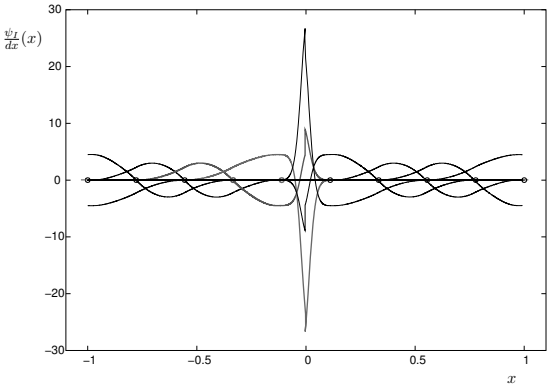


Figure 5.5. Shape function derivatives for a discontinuous derivative located at $x_d = 0$.

similar results when $NP = 9$ and consequently a node is placed just at the discontinuity point.

Remarks:

- We have noticed that the shape functions are perfectly continuous through the interfaces between the enriched and non-enriched domains (*Property 4*). Moreover the shape function derivatives become discontinuous only along the interface $x_d = 0$.
- The shape functions are quite different when a node is or not located at the discontinuity point. In the first case (node located at x_d) the shape functions are very close to those obtained by Krongauz et al. [10] using a PU enrichment.

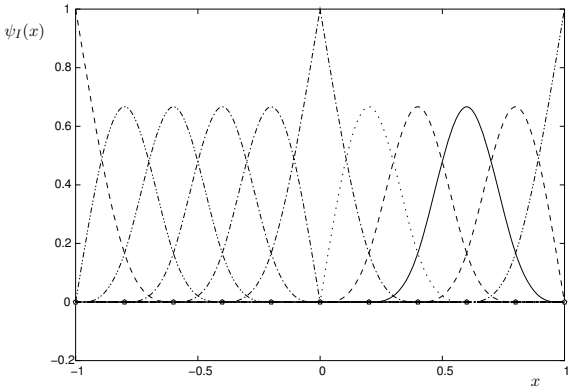


Figure 5.6. Shape functions for a discontinuous derivative located at $x_d = 0$ when a node is placed just at the discontinuity point.

6 Numerical Example

In this section we consider a simple numerical test: the steady heat conduction problem (with a constant source term $g = 2$) defined in the interval $]-1, 1[$ which involves two homogeneous materials with different thermal conductivities

$$\frac{d}{dx}\left(k(x)\frac{dT}{dx}\right) = g, \tag{6.1}$$

where T is the temperature field and k the thermal conductivity defined as follows

$$\begin{array}{c} k(x) = k_0 = 1 \ \text{ if } x \in \Omega_0 =]-1, 0[\,, \\ k(x) = k_1 = 10 \text{ if } \ x \in \Omega_1 =]0, 1[\,. \end{array} \tag{6.2}$$

The boundary conditions are defined by

$$\left\{ \begin{array}{l} T(x = -1) = 0 \\ T(x = 1) = 1 \end{array} \right. . \tag{6.3}$$

The problem is solved using the weak-form of Eq. (6.1), where the integrals were computed using a background mesh. Four integration cells are placed between two consecutive nodes, with 5 gauss integration points in each cell.

Computed temperatures with and without approximation enrichment as well as the exact temperature field are depicted in figure 6.7. In both cases (enriched and non-enriched) a first order consistency has been enforced, been the support radius $R = 2 \times h$. The related derivatives are shown in Figure 6.8. From both figures we can conclude about the better accuracy of the solution computed by using an enriched approximation, mainly in the temperature derivative in the discontinuity neighbourhood.

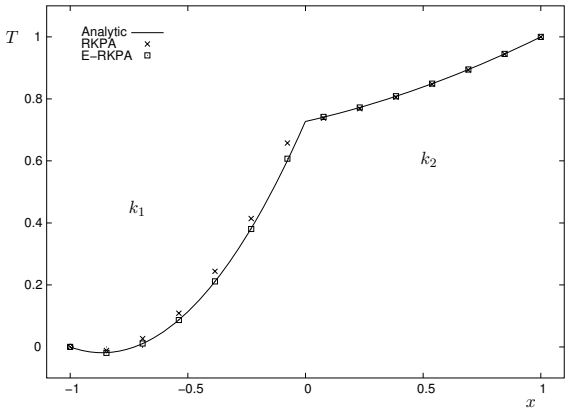


Figure 6.7. Exact and numerical temperature fields computed with and without approximation enrichment.

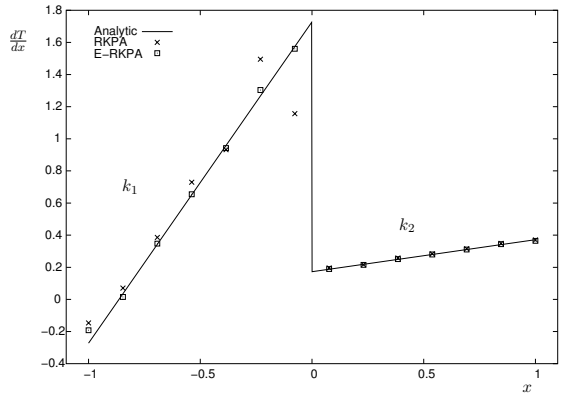


Figure 6.8. Exact and computed temperature derivatives.

6.1 Convergence Analysis

In this section we compare the convergence of both enriched and non-enriched approximations, in the test problem just described. Two norms are considered, the L_2 and the H_1 norms. Moreover, the analysis has been performed for three different conductivity ratios $k_{01} = \frac{k_1}{k_0} = 10, 100, 1000$. Figure 6.9 illustrates these results.

In the case of non-enriched approximations (standard RKPA) we can notice from figure 6.9 that the order of the method is 1 and 0.5 (using the L_2 and H_1 norms respectively). Moreover, the error increases as the conductivity ratio increases.

On the contrary, when an enriched approximation is considered the method results of order 2.4 and 1.5 (using the L_2 and H_1 norms respectively). Moreover, the error is independent of the conductivity ratio.

In conclusion, both the order of convergence and the solution error obtained by using an enriched approximation are significantly better that the ones obtained by using the standard non-enriched formulation.

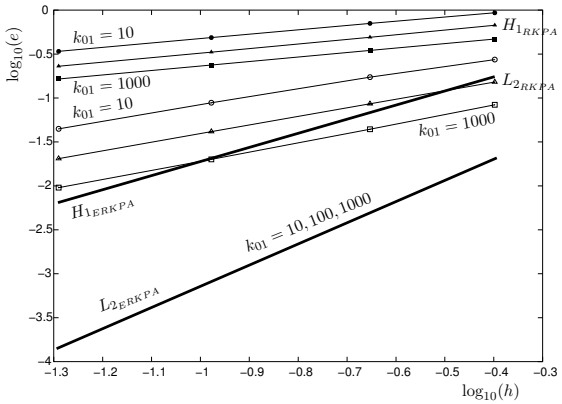


Figure 6.9. Convergence analysis using the L_2 and the H_1 norms.

7 Conclusion

In this work we have proposed a new strategy to enrich reproducing kernel particle approximations in order to reproduce functions involving discontinuous derivatives through some interfaces. The computed results seems to be very accurate (from the point of view of the order of convergence as well as from the associated level of error) and it is very easy to implement.

In some cases the moment matrix becomes singular far from the discontinuity location (in the order of the support size), and consequently the enrichment becomes local in space. In any case, the size of the global discrete system is not affected by the enrichment process, in contrast with the vast majority of enrichment methods making use of the partition of unity.

The case of moving interfaces involving field discontinuities can be easily treated coupling the technique described in this work with a level set description of the interface movement.

The extension to higher dimension problems involving complex discontinuity geometries is a work in progress.

References

1. N. Aluru. A point collocation method based on reproducing kernel approximations. *Int. J. Numer. Meths. Eng.*, 47:1083–1121, 2000.
2. G. T. B. Nayrolles and P. Villon. Generalizing the finite element method: Diffuse approximation and diffuse elements computational mechanics. *Journal of Computational Mechanics*, 10(5):307–318, 1992.
3. I. Babuška and J. Melenk. The partition of unity method. *Int. J. Numer. Meths. Eng.*, 40:727–758, 1997.
4. T. Belytschko, Y. Kronggauz, D. Organ, and M. Fleming. Meshless methods: an overview and recent developments. *Comput. Meths. Appl. Mech. Engrg.*, 139:3–47, 1996.
5. J. Chen, C. Pan, C. Roque, and H.P.Wang. A lagrangian reproducing kernel particle method for metal forming. *Comput. Mech.*, 22:289–307, 1998.
6. M. Fleming, Y. Chu, B. Moran, and T. Belytschko. Enriched element-free galerkin methods for crack tip fields. *Int. J. Numer. Meths. Eng.*, 40, 1997.
7. R. Gingold and J. Monaghan. Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices Royal Astronomical Society*, 181:375–389, 1977.
8. W. Han and X. Meng. Error anlysis of the reproducing kernel particle method. *Comput. Meths. Appl. Mech. Engrg.*, 190:6157–6181, 2001.
9. H. Ji, D. Chopp, and J. Dolbow. A hybrid extended finite/level set method for modeling phase transformtions. *Int. J. Numer. Meths. Eng.*, 54:1209–1233, 2002.
10. Y. Kronggauz and T. Belytschko. Efg approximation with discontinuous derivatives. *Int. J. Numer. Meths. Eng.*, 41:1215–1233, 1998.
11. W. Liu, S. Jun, S. Li, J. Adee, and T. Belytschko. Reproducing kernel particle methods for structural dynamics. *Int. J. Numer. Meths. Eng.*, 38:1655–1679, 1995.
12. J. M. Melenk and I. Babuška. The partition of unity finite element method: Basic theory and applications. *Comput. Meths. Appl. Mech. Engrg.*, 139:289–314, 1996.
13. N. Moës and T. Belytschko. Extended finite element method for cohesive crack growth. *Engineering Fracture Mechanics*, 69:813–833, 2002.
14. J. Sethian. Evolution, implementation and application of level set and fast marching methods for advancing fronts. *J. Comput. Phys.*, 169:503–555, 2001.
15. T. Strouboulis, K. Corps, and I. Babuška. The generalized finite elment method. *Comput. Meths. Appl. Mech. Engrg.*, 190:4081–4193, 2001.
16. N. Sukumar, D. Chopp, N. Moës, and T. Belytschko. Modeling holes and inclusions by level sets in the extended finite-element method. *Comput. Meths. Appl. Mech. Engrg.*, 190:6183–6200, 2001.
17. G. Ventura, J. Xu, and T. Belytschko. A vector level set method and new discontinuity approximations for crack growth by efg. *Int. J. Numer. Meths. Eng.*, 54:923–944, 2002.

Reproducing Kernel Element Interpolation: Globally Conforming $I^m/C^n/P^k$ Hierarchies

Shaofan Li[1⋆], Daniel C. Simkins[1], Hongsheng Lu[2], and Wing Kam Liu[2]

[1] University of California, Berkeley, CA, 94720, USA.
[2] Northwestern University, Evanston, IL 60802, USA.

Abstract In this work, arbitrarily smooth, globally compatible, $I^m/C^n/P^k$ interpolation hierarchies are constructed in the framework of reproducing kernel element method (RKEM) for multi-dimensional domains. This is the first interpolation hierarchical structure that has been ever constructed with both minimal degrees of freedom and higher order continuity and reproducing conditions over multi-dimensional domains. The proposed hierarchical structure possesses the generalized Kronecker property, i.e., $\partial^\alpha \Psi_I^{(\beta)}/\partial x^\alpha(x_J) = \delta_{IJ}\delta_{\alpha\beta}$, $|\alpha|$, $|\beta| \leq m$. The newly constructed globally conforming interpolant is a hybrid of global partition polynomials (C^∞) and a smooth (C^n) compactly supported meshfree partition of unity. Examples of compatible RKEM hierarchical interpolations are illustrated, and they are used in a Galerkin procedure to solve differential equations.

1 Introduction

In this paper, the term, I^m *interpolation field*, denotes the interpolant that can interpolate the derivatives of an unknown function up to the m-th order, whereas the term, C^n *interpolation field*, denotes the interpolant having globally continuous derivatives up to n-th order, and the term, P^k *interpolation field*, denotes for the interpolant that can reproduce complete k-th order polynomials.

Constructing a globally conforming $I^m/C^n/P^k$, $(m, n, k \geq 1)$ interpolation field in multiple dimension was *the challenge* in the early development of finite element methods. It attracted a group of very creative engineers and mathematicians working on the subject. Some of them were intellectual heavy weight of the time, e.g. Clough and Tocher [1965], Bazeley, Cheung, Irons, and Zienkiewicz [1965], Fraeijs de Veubeke [1965], Argyris et al [1968], Irons [1969], Felippa and Clough [1970], Bramble and Zlámal [1970], Birkhoff [1971], Birkhoff and Mansfield [1974], among others.

⋆ li@ce.berkeley.edu

During the past half century, it has been an outstanding problem to construct compatible higher order continuous finite element interpolants in multiple dimensions. Although there were a few compatible C^1 elements constructed in two dimensional (2D) case, e.g. Argyris element [Argyris68] or Bell's element [Bell69], they usually require adding extra degrees of freedom on either nodal points or along the boundary or in the interior of the element to make it work, or eventually do not work well because of their complexity.

As a matter of fact, in many engineering applications, the incompatible element is the only viable choice in numerical computations. During the past fifty years, a main theme of finite element method research is to develop suitable finite element shape functions that can be used in various mixed formulations, or can be used with incompatible modes. Nevertheless, no general solution has been found. Even though engineers invented the so-called patch test to examine applicability of various incompatible elements [Irons72], serious-minded mathematicians view it as a "*variational crime*". It is fair to say that this predicament in FEM has more or less hinged the advancement of finite element technology.

Recently, it has been found that by combining a meshfree interpolant with finite element interpolant one can generate a class of hybrid interpolation functions that may achieve higher order continuous interpolation in multiple dimensions, which is unattainable by traditional finite element technique. This new technology is now termed as *Reproducing Kernel Element Method* (RKEM) ([LHLL, LLHLD]), which is in parallel and in contrast to the meshfree *Reproducing Kernel Particle Method* (RKPM) (Liu et al [LJZ, LJLAB]). In this work, complete $I^m/C^n/P^k$ RKEM interpolation hierarchies are developed.

2 Reproducing Kernel Element Method

2.1 Global partition polynomial

Let $\boldsymbol{Z}^d$ denote the set of all ordered d-tuples of non-negative integers. A multi-index is an ordered collection (d-tuple) of d nonnegative integers, $\alpha = (\alpha_1, \cdots, \alpha_d)$, and its length is defined as

$$|\alpha| = \sum_{i=1}^{d} \alpha_i \tag{2.1}$$

We write $\alpha! = \alpha_1!\alpha_2! \cdots \alpha_d!$ and $\boldsymbol{x}^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_d^{\alpha_d}, \forall \boldsymbol{x} \in \mathbb{R}^d$. For a differentiable function $f(\boldsymbol{x})$ and any α with $|\alpha| \leq k$,

$$D^\alpha f(\boldsymbol{x}) = \frac{\partial^\alpha f(\boldsymbol{x})}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \tag{2.2}$$

is the αth order partial derivative. As usual, $D^0 f(\boldsymbol{x}) = f(\boldsymbol{x})$.

Assume that finite element shape functions, $\phi_{e,i}$, $e \in \Lambda_E := \{1, 2, \cdots, n_{el}\}$, and $i \in \Lambda_e = \{1, \cdots, n_{ne}\}$, are compactly supported, i.e. $supp\{\phi_{e,i}\} = \overline{\Omega_e}$, where n_{el} is the total number of elements, Λ_e is the local index set for all the nodal points in an element, and the total number nodal points in a typical element, e, is n_{ne}. When $\boldsymbol{x} \notin \Omega_e$, $\phi_{e,i}(\boldsymbol{x}) = 0$. For a large class of FEM shape functions, this condition is enforced by multiplying Heaviside function with the so-called global partition polynomial functions, or more precisely,

$$\phi_{e,i}(x) = \psi_{e,i}(x)\chi_e(x) \tag{2.3}$$

where function, $\chi_e(x)$, is the characteristic function of element e, i.e.

$$\chi_e(x) := \begin{cases} 1, & x \in \Omega_e \\ 0, & x \notin \Omega_e \end{cases} \tag{2.4}$$

The characteristic function in (2.4) truncates the analytical polynomial functions so that FEM shape functions are localized in compact supports. Because of the presence of the characteristic function, $\chi_e(x)$, most of FEM shape functions, $\phi_{e,i}(x)$, are $C^0(\Omega)$ functions in multiple dimensions.

We call the set of polynomial functions, $\psi_{e,i}(x)$ as **the global partition polynomials**. The so-called *global partition polynomial* may be viewed as the continuous extension of regular FEM polynomial shape function[3], and it is defined in $\mathbb{R}^d$. Denote such globally continuous extension of a FEM shape functions, $\phi_{e,i}(\boldsymbol{x})$, as $\psi_{e,i}(\boldsymbol{x})$ where $e \in \Lambda_E$, and $i \in \Lambda_e$.

Consider a smooth continuous function that has up to the m-th order continuous derivatives. We approximate the function by using the following general Hermite FEM interpolation,

$$f^h(\boldsymbol{x}) = \sum_{e \in \Lambda_E} \sum_{|\beta| \le m} \sum_{i \in \Lambda_e} \phi_{e,i}^{(\beta)}(\boldsymbol{x}) f_i^{(\beta)}, \tag{2.5}$$

where $\phi_{e,i}^{(\beta)}(\boldsymbol{x})$ is the β-th order interpolant that interpolates the β-th order derivative, and $f_i^{(\beta)} = D^\beta f(\boldsymbol{x}_i)$.

The global partition polynomials have two special properties. First, if a FEM shape function has the following Kronecker delta properties

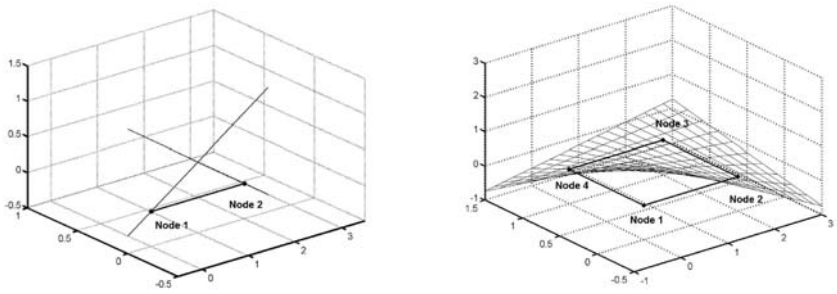
$$D^\alpha \phi_{e,i}^{(\beta)} \Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = \delta_{ij}\delta_{\alpha\beta}, \quad \boldsymbol{x}_i,\ \boldsymbol{x}_j \in \bar{\Omega},\ \ |\alpha|, |\beta| \le m, \tag{2.6}$$

then the corresponding global partition polynomial function has the properties,

$$D^\alpha \psi_{e,i}^{(\beta)} \Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = \delta_{ij}\delta_{\alpha\beta}, \quad \boldsymbol{x}_i,\ \boldsymbol{x}_j \in \bar{\Omega}_e,\ \ |\alpha|, |\beta| \le m\ . \tag{2.7}$$

In fact, in most cases, (2.6) is a consequence or built in property of (2.7).

[3] In this paper, we only consider polynomial type of FEM shape functions



(a) (b)

Figure 2.1. Global partition polynomials in 1-D and 2-D: (a) 1-D linear global partition polynomials; (b) 2-D bilinear global partition polynomials.

As a partition of unity, traditional finite element interpolants satisfy the condition,

$$\sum_{I \in \Lambda_P} \phi_I(\boldsymbol{x}) = 1, \quad \forall\, \boldsymbol{x} \in \bar{\Omega}\ . \tag{2.8}$$

However, since global partition polynomials are not compactly supported, they do not form a partition of unity for the global index set, Λ_P, i.e.

$$\sum_{I \in \Lambda_P} \psi_I(\boldsymbol{x}) \neq 1, \quad \forall\, \boldsymbol{x}\ \in \bar{\Omega}\ . \tag{2.9}$$

Nevertheless, the global partition polynomial has the following amiable properties: *the global partition polynomial forms a partition of unity in every local index set Λ_e, i.e.*

$$\sum_{i \in \Lambda_e} \psi_{e,i}(\boldsymbol{x}) = 1, \quad \forall\, \boldsymbol{x} \in \mathbb{R}^d \ \text{ and } \ \forall\, e \in \Lambda_E \tag{2.10}$$

where $\Lambda_e := \{I \,\big|\, 1, 2, \cdots, nen\}$.

Examples of one dimensional (1D) partition polynomials are the following linear polynomials defined in $\mathbb{R}$,

$$\psi_{e,1}(x) = \frac{x_{e+1} - x}{x_{e+1} - x_e}, \tag{2.11}$$

$$\psi_{e,2}(x) = \frac{x - x_e}{x_{e+1} - x_e}, \tag{2.12}$$

where $-\infty < x < \infty$ and $e \in \Lambda_E$.

A two-dimensional example of global partition shape function set is:

$$\psi_{e,1}(\xi(\boldsymbol{x}),\eta(\boldsymbol{x})) = \tfrac{1}{4}(1-\xi)(1-\eta)\,, \quad \psi_{e,2}(\xi(\boldsymbol{x}),\eta(\boldsymbol{x})) = \tfrac{1}{4}(1+\xi)(1-\eta)\,,$$
$$\psi_{e,3}(\xi(\boldsymbol{x}),\eta(\boldsymbol{x})) = \tfrac{1}{4}(1+\xi)(1+\eta)\,, \quad \psi_{e,4}(\xi(\boldsymbol{x}),\eta(\boldsymbol{x})) = \tfrac{1}{4}(1-\xi)(1+\eta)\,,$$

with $-\infty < \xi, \eta < \infty$ and functions $\xi(\boldsymbol{x})$ and $\eta(\boldsymbol{x})$ can be found by its inverse relationship,

$$\boldsymbol{x} = \sum_{i=1}^{4} \psi_{e,i}(\boldsymbol{\xi})\boldsymbol{x}_{e,i} \tag{2.13}$$

In Fig. 2.1, the global partition polynomials defined in (2.11)–(2.12) and (2.1) are displayed.

2.2 RKEM Interpolant

Now we describe how to construct a valid RKEM interpolant. We define the reproducing kernel element interpolation as

$$\mathcal{I}f(\boldsymbol{x}) = \sum_{e\in\Lambda_E}\left[\int_{\Omega_e} \mathcal{K}_\rho(\boldsymbol{x}-\boldsymbol{y};\boldsymbol{x})d\boldsymbol{y}\left(\sum_{i\in\Lambda_e}\psi_{e,i}(\boldsymbol{x})f(\boldsymbol{x}_{e,i})\right)\right] \tag{2.14}$$

To obtain a valid interpolation scheme, the integral in expression (2.14) is replaced by nodal integration,

$$\mathcal{I}^h f(\boldsymbol{x}) = \underset{e\in\Lambda_E}{\boldsymbol{A}}\left[\left(\sum_{j\in\Lambda_e}\mathcal{K}_\rho(\boldsymbol{x}-\boldsymbol{x}_{e,j};\boldsymbol{x})\Delta V_{e,j}\right)\left(\sum_{i\in\Lambda_e}\psi_{e,i}(\boldsymbol{x})f(\boldsymbol{x}_{e,i})\right)\right] \tag{2.15}$$

where $\Delta V_{e,j}$ is the nodal integration weight, which can be easily assigned for each nodal point by Lobatto quadrature weight. The symbol, $\underset{e\in\Lambda_E}{\boldsymbol{A}}$, denotes the summation over all elements of the mesh.

The reproducing kernel function, $\mathcal{K}^{\varrho}_{e,j}(\boldsymbol{x}) := \mathcal{K}_{\varrho}(\boldsymbol{x}-\boldsymbol{x}_{e,j};\boldsymbol{x})$ is a compact supported function and the radius of its support, $supp\{\mathcal{K}^{\varrho}_{e,j}(\boldsymbol{x})\}$, is $\varrho_{e,j}$. A smooth window function, $w(x) \in C^n(\Omega)$, is chosen to serve as the core of the kernel. Note that the RKEM nodal integration is consistent with the nodal integration employed in previous reproducing kernel particle method (RKPM) (e.g. in [LLB]) that ensures all the reproducing properties in discrete sense and guarantees that discrete summation is a partition of unity.

To ensure the consistency, the RKEM kernel function has to satisfy the following partition of unity condition,

$$\underset{e\in\Lambda_E}{\boldsymbol{A}}\left\{\left(\sum_{j\in\Lambda_e}\frac{1}{\rho^d_{e,j}}w\left(\frac{\boldsymbol{x}-\boldsymbol{x}_{e,j}}{\rho_{e,j}}\right)b(\boldsymbol{x})\Delta V_{e,j}\right)\left(\sum_{i\in\Lambda_e}\psi_{e,i}(\boldsymbol{x})\right)\right\} = 1,\ \forall\,\boldsymbol{x}\in\Omega\,. \tag{2.16}$$

One can then solve (2.16) to obtain the unknown $b(\boldsymbol{x})$,

$$b(\boldsymbol{x}) = \left\{ \underset{e\in\Lambda_E}{\boldsymbol{A}} \left[\left(\sum_{j\in\Lambda_e} \frac{1}{\rho_{e,j}^d} w\left(\frac{\boldsymbol{x}-\boldsymbol{x}_{e,j}}{\rho_{e,j}} \right) \Delta V_{e,j} \right) \left(\sum_{i\in\Lambda_e} \psi_{e,i}(\boldsymbol{x}) \right) \right] \right\}^{-1}. \tag{2.17}$$

An interpolation error estimate for the interpolation defined in (2.14) has been given in [LHLL].

Proposition 2.1. *Consider the reproducing kernel element interpolant (2.15). Assume the discretization mesh having the following surjective connectivity map* $\Lambda_E \times \Lambda_e \to \Lambda_P$,

$$(e_1, i_1), (e_2, i_2), \cdots, (e_k, i_k), \cdots, (e_\ell, i_\ell) \to I. \tag{2.18}$$

Define

$$\Omega_I = \bigcup_{k=1}^{\ell} \Omega_{e_k} \tag{2.19}$$

Let $(e, i) \to I$. *The meshfree kernel at nodal point* (e, i) *may be denoted as,*

$$\mathcal{K}_{e,i}^{\rho}(\boldsymbol{x}) := \mathcal{K}_{\rho}(\boldsymbol{x} - \boldsymbol{x}_{e,i}; \boldsymbol{x}) =: \mathcal{K}_I^{\rho}(\boldsymbol{x}), \quad \forall\ \boldsymbol{x} \in \Omega_e \tag{2.20}$$

Suppose that

1. $$\sum_{i\in\Lambda_e} \psi_{e,i}(\boldsymbol{x}) \boldsymbol{x}_i^{\lambda} = \boldsymbol{x}^{\lambda}, \quad \forall\ \ \boldsymbol{x} \in \Omega, \quad \lambda \leq \ell \tag{2.21}$$

2. *Discretization mesh* $\mathcal{T}_{n_{el}}$ *is quasi-uniform;*
3. *Meshfree interpolant is a signed partition of unity,*

$$\underset{e\in\Lambda_E}{\boldsymbol{A}} \left(\sum_{i\in\Lambda_e} \mathcal{K}_{e,i}^{\rho}(\boldsymbol{x}) \right) = 1, \quad \forall \boldsymbol{x} \in \Omega \tag{2.22}$$

4. *For* $e \in \Lambda_E$ *and* $(e, i) \to I$,

$$supp\{\mathcal{K}_{e,i}^{\rho}(\boldsymbol{x})\} \subset \Omega_I \tag{2.23}$$

The reproducing kernel element interpolation field

$$\mathcal{I}f = \sum_{I\in\Lambda_P} \Psi_I(\boldsymbol{x}) f_I, \quad \Psi_I(\boldsymbol{x}) = \sum_{k=1}^{\ell} \Big(\sum_{j\in\Lambda_{e_k}} \mathcal{K}_{e_k,j}^{\rho}(\boldsymbol{x}) \Delta V_{e_k,j} \Big) \psi_{e_k,i_k}(\boldsymbol{x}), \tag{2.24}$$

has the following properties,

1. $$\mathcal{I}^h \boldsymbol{x}^{\lambda} = \boldsymbol{x}^{\lambda}, \quad \lambda = 0, \cdots, k. \tag{2.25}$$

   In particular $\mathcal{I}^h 1 = 1$.
2. $$\Psi_I(\boldsymbol{x}_J) = \delta_{IJ} \tag{2.26}$$

3 Globally Conforming $I^m/C^n/P^k$ Hierarchy I

3.1 Construction

In this section, we construct the first globally conforming I^m/C^n hierarchy. Consider the nodal point i of the element $e \in \Lambda_E$. Assume that one can find a global partition polynomial basis, $\psi^{(0)}_{e,i}(\boldsymbol{x})$, such that

$$D^\alpha \psi^{(0)}_{e,i} \Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = \delta_{\alpha 0}\delta_{ij}, \quad |\alpha| \le q \ . \tag{3.1}$$

We can then construct a hybrid meshfree/FEM interpolant basis as

$$\Psi^{(0)}_I(\boldsymbol{x}) = \underset{e\in\Lambda_E}{\boldsymbol{A}} \left(\sum_{j\in\Lambda_e} \mathcal{K}^\rho_{e,j}(\boldsymbol{x}) \Delta x_{e,j} \right) \psi^{(0)}_{e.i}(\boldsymbol{x}) \tag{3.2}$$

where connectivity $(e,i) \to I$ is implied. The meshfree kernel functions satisfy the following condition,

$$\underset{e\in\Lambda_E}{\boldsymbol{A}} \left(\sum_{j\in\Lambda_e} \mathcal{K}^\rho_{e,j}(\boldsymbol{x}) \Delta x_{e,j} \right) \sum_{i\in\Lambda_e} \psi^{(0)}_{e.i}(\boldsymbol{x}) = 1 \ . \tag{3.3}$$

Note that it is not necessary to require the global partition polynomials, $\{\psi^{(0)}_{e,i}\}_{i\in\Lambda_e}$, to form a partition of unity.

We construct higher order RKEM bases by multiplying the zero-th order basis with some polynomials, i.e. $I \in \Lambda_P$,

$$\Psi^{(1)}_I(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}_I)\Psi^{(0)}_I(\boldsymbol{x}) \tag{3.4}$$

$$\Psi^{(2)}_I(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_I)^2\Psi^{(0)}_I(\boldsymbol{x}) \tag{3.5}$$

$$\cdots$$

$$\Psi^{(\ell)}_I(\boldsymbol{x}) = \frac{1}{\ell!}(\boldsymbol{x} - \boldsymbol{x}_I)^\ell\Psi^{(0)}_I(\boldsymbol{x}) \ . \tag{3.6}$$

The proposed RKEM I^m/C^n interpolation hierarchy can be written as

$$\mathcal{I}^m f = \sum_{I\in\Lambda_P} \left(\Psi^{(0)}_I(\boldsymbol{x}) f_I + \Psi^{(1)}_I(\boldsymbol{x}) Df_I + \cdots + \Psi^{(m)}_I(\boldsymbol{x}) D^m f_I \right) \tag{3.7}$$

In this construction, $m = k$, which means that the globally conforming interpolants are only capable of reproducing a complete mth order polynomials.

The main result of this interpolation is summarized in the following proposition:

Proposition 3.1. *Assume that* $\exists \psi^{(0)}_{e,i}(\boldsymbol{x}), \quad \forall e \in \Lambda_E, \ i \in \Lambda_e$, *such that*

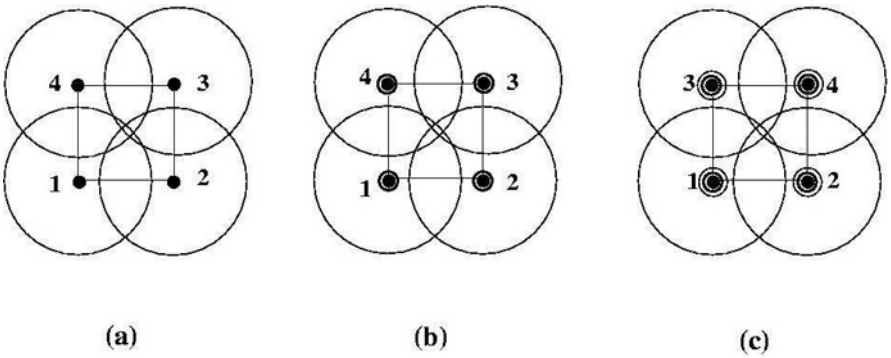


Figure 3.2. Two dimensional quadrilateral RKEM hierarchical interpolation field: (a) I^0 element, (b) I^1 element, and (c) I^2 element.

$$D^{\alpha}\psi_{e,i}^{(0)}\Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = \delta_{\alpha 0}\delta_{ij}, \quad |\alpha| \leq \ell, i, j \in \Lambda_e \tag{3.8}$$

and define,

$$\Psi_I^{(0)}(\boldsymbol{x}) = \underset{e\in\Lambda_E}{\boldsymbol{A}}\left(\sum_{j\in\Lambda_e}\mathcal{K}_{e,j}^{\rho}(\boldsymbol{x})\Delta x_{e,j}\right)\psi_{e.i}^{(0)}(\boldsymbol{x}) \tag{3.9}$$

$$\Psi_I^{(1)} = (\boldsymbol{x}-\boldsymbol{x}_I)\Psi_I^{(0)}(\boldsymbol{x}) \tag{3.10}$$

$$\Psi_I^{(2)} = \frac{1}{2!}(\boldsymbol{x}-\boldsymbol{x}_I)^2\Psi_I^{(0)}(\boldsymbol{x}) \tag{3.11}$$

$$\cdots\cdots$$

$$\Psi_I^{(\ell)} = \frac{1}{\ell!}(\boldsymbol{x}-\boldsymbol{x}_I)^{\ell}\Psi_I^{(0)}(\boldsymbol{x}) \tag{3.12}$$

where $(e,i) \rightarrow I$ *and*

$$\sum_{I\in\Lambda_P}\Psi_I^{(0)}(\boldsymbol{x}) := \underset{e\in\Lambda_E}{\boldsymbol{A}}\left(\sum_{j\in\Lambda_e}\mathcal{K}_{e,j}^{\rho}(\boldsymbol{x})\Delta x_{e,j}\right)\left(\sum_{i\in\Lambda_e}\psi_{e.i}^{(0)}(\boldsymbol{x})\right) = 1\ . \tag{3.13}$$

The interpolation scheme (3.7) has the following properties:

1. $D^{\alpha}\Psi_I^{(\beta)}\Big|_{\boldsymbol{x}=\boldsymbol{x}_J} = \delta_{\alpha\beta}\delta_{IJ}, \quad I,J\in\Lambda_P, \quad |\alpha|,|\beta| \leq m;$
2. $I^m\boldsymbol{x}^{\lambda} = \boldsymbol{x}^{\lambda}, \quad \forall \boldsymbol{x}\in\Omega, \quad |\lambda| \leq m.$

The proof of this proposition can be found in [LLHLD].

3.2 Globally Compatible Quadrilateral Element Q12P1I1 Element

This is a multi-dimensional compatible interpolant. It only has a minimal twelve degrees of freedom over four nodes. It is an I^1 interpolation, i.e. the

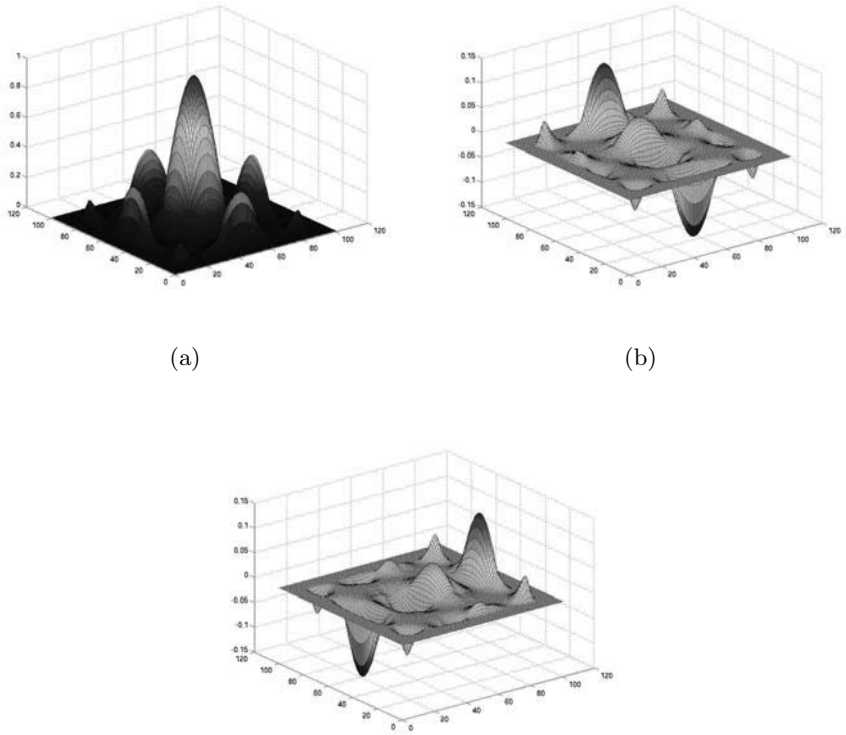


Figure 3.3. Smooth quadrilateral I^1/C^4 RKEM interpolant: (a) the 1st shape function, $\Psi_I^{(00)}(\boldsymbol{x})$, (b) the 2nd shape function $\Psi_I^{(10)}(\boldsymbol{x})$, (c) the 3rd shape function $\Psi_I^{(01)}(\boldsymbol{x})$. See Color Plate 3 on page 294.

primary variable and its first order derivatives are sampled. If the fifth order spline is chosen as the meshfree window function, its smoothness is C^4.

Consider a four-nodes quadrilateral element. Use 1D zero-th order Hermite interpolant to construct global partition polynomials,

$$\psi_{e,1}^{(0)}(\boldsymbol{x}) = H_{c1}^{(0)}(\xi)H_{c1}^{(0)}(\eta)\,, \quad \psi_{e,2}^{(0)}(\boldsymbol{x}) = H_{c2}^{(0)}(\xi)H_{c1}^{(0)}(\eta)\,,$$
$$\psi_{e,3}^{(0)}(\boldsymbol{x}) = H_{c2}^{(0)}(\xi)H_{c2}^{(0)}(\eta)\,, \quad \psi_{e,4}^{(0)}(\boldsymbol{x}) = H_{c1}^{(0)}(\xi)H_{c2}^{(0)}(\eta)\,,$$

where $H_{c1}^{(0)}(\zeta) = 1 - 3\zeta^2 + 2\zeta^3$, $H_{c2}^{(0)}(\zeta) = 3\zeta^2 - 2\zeta^3$, and $\zeta = \xi$ and η.

The coordinate transformation between (x, y) and (ξ, η) is bilinear,

$$x(\xi,\eta) = \alpha_0 + \alpha_1\xi + \alpha_2\eta + \alpha_3\xi\eta \tag{3.14}$$

$$y(\xi,\eta) = \beta_0 + \beta_1\xi + \beta_2\eta + \beta_3\xi\eta \tag{3.15}$$

Or in terms of nodal coordinates, polynomials,

$$x = \sum_{j=1}^{4} N_{e,j}(\xi,\eta)x_{e,j}, \quad y = \sum_{j=1}^{4} N_{e,j}(\xi,\eta)y_{e,j}, \tag{3.16}$$

where $N_{e,1}(\xi,\eta) = 1-\xi-\eta+\xi\eta$, $N_{e,2}(\xi,\eta) = \xi(1-\eta)$, $N_{e,3}(\xi,\eta) = \xi\eta$, and $N_{e,4}(\xi,\eta) = \eta(1-\xi)$.

Three I^1/C^4 quadrilateral RKEM shape functions are plotted in Fig. 3.3. The zero-th order shape functions in an element are

$$\Psi_{e,i}^{(00)}(\boldsymbol{x}) = \Big(\sum_{j=1}^{4} \mathcal{K}_{e,j}^{\varrho}(\boldsymbol{x})\Delta V_{e,j}\Big)\psi_{e,i}^{(0)}(\xi,\eta), \quad i = 1,2,3,4 \tag{3.17}$$

The higher order basis functions are

$$\Psi_{e,i}^{(10)}(\boldsymbol{x}) = (x - x_{e,i})\Psi_{e,i}^{(00)}(\boldsymbol{x}), \quad i = 1,2,3,4 \tag{3.18}$$

$$\Psi_{e,i}^{(01)}(\boldsymbol{x}) = (y - y_{e,i})\Psi_{e,i}^{(00)}(\boldsymbol{x}), \quad i = 1,2,3,4 \tag{3.19}$$

The zero-th order global shape function can be written as

$$\Psi_I^{(00)}(\boldsymbol{x}) = \sum_{k\in\Lambda_I}\Big(\sum_{j\in\Lambda_{e_k}} \mathcal{K}^{\varrho}{}_{e_k,j}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi_{e_k,i_k}^{(0)}(\boldsymbol{x}) = \sum_{k\in\Lambda_I}\Psi_{e_k,i_k}^{(00)}(\boldsymbol{x})$$

where $\Lambda_I = \{1,2,\cdots,\ell\}$. The global higher order basis functions have simple form

$$\Psi_I^{(10)}(\boldsymbol{x}) = (x - x_I)\Psi_I^{(00)}(\boldsymbol{x}) \tag{3.20}$$

$$\Psi_I^{(01)}(\boldsymbol{x}) = (y - y_I)\Psi_I^{(00)}(\boldsymbol{x}) \tag{3.21}$$

One can easily write down 16 degrees of freedom bilinear RKEM quadrilateral interpolant function by adding four additional higher order partition polynomials,

$$\Psi_{e,i}^{(11)}(\boldsymbol{x}) = (x - x_{e,i})(y - y_{e,i})\Psi_{e,i}^{(00)}(\boldsymbol{x}), \quad i = 1,2,3,4 \tag{3.22}$$

or by adding one global shape function,

$$\Psi_I^{(11)}(\boldsymbol{x}) = (x - x_I)(y - y_I)\Psi_I^{(00)}(\boldsymbol{x}). \tag{3.23}$$

It can be shown that the above compatible RKEM Interpolants can reproduce polynomials, $1, x, y$, and xy.

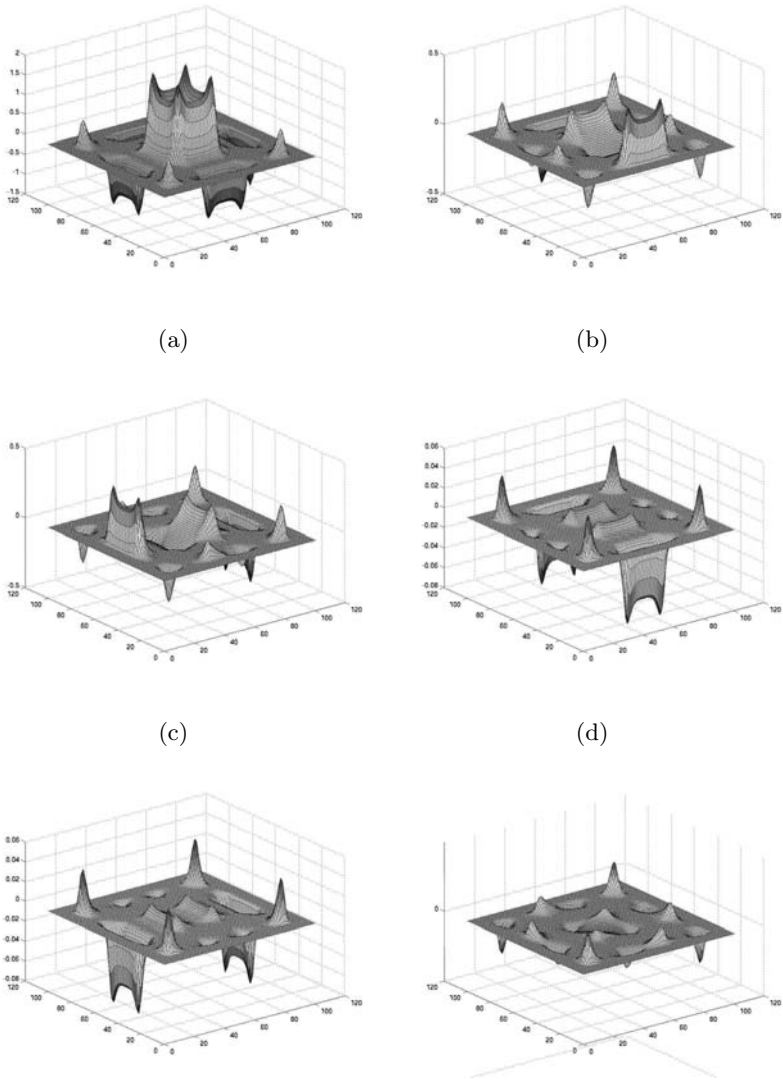


Figure 3.4. Compatible quadrilateral $I^2/C^4/P^2$ RKPM interpolants: (a) the 1st shape function, $\Psi_I^{(00)}(\boldsymbol{x})$, (b) the 2nd shape function $\Psi_I^{(10)}(\boldsymbol{x})$, (c) the 3rd shape function $\Psi_I^{(01)}(\boldsymbol{x})$ (d) the 4th shape function, $\Psi_I^{(20)}(\boldsymbol{x})$, (e) the 5th shape function $\Psi_I^{(02)}(\boldsymbol{x})$, (f) the 6th shape function $\Psi_I^{(11)}(\boldsymbol{x})$. See Color Plate 4 on page 295.

3.3 Globally Compatible Q16P2I2 Quadrilateral Element

Choosing the global partition polynomials as

$$\psi^{(0)}_{e,1}(\boldsymbol{x}) = H^{(0)}_{f1}(\xi)H^{(0)}_{f1}(\eta)\,, \quad \psi^{(0)}_{e,2}(\boldsymbol{x}) = H^{(0)}_{f2}(\xi)H^{(0)}_{f1}(\eta)\,,$$
$$\psi^{(0)}_{e,3}(\boldsymbol{x}) = H^{(0)}_{f2}(\xi)H^{(0)}_{f2}(\eta)\,, \quad \psi^{(0)}_{e,4}(\boldsymbol{x}) = H^{(0)}_{f2}(\xi)H^{(0)}_{f1}(\eta)\,,$$

where $H^{(0)}_{f1}(\zeta) = 1 - 10\zeta^3 + 15\zeta^4 - 6\zeta^5$, and $H^{(0)}_{f2}(\zeta) = 10\zeta^3 - 15\zeta^4 + 6\zeta^5$. The coordinate transformation are bilinear, which is exactly the same as Eq. (3.14)–(3.16).

We can construct a compatible I^2/C^n RKEM interpolant with a minimal 24 degrees of freedom. The first four zero-order basis functions are

$$\Psi^{(00)}_{e,i}(\boldsymbol{x}) = \Big(\sum_{j=1}^{4} \mathcal{K}^{\varrho}{}_{e,j}(\boldsymbol{x})\Big)\psi^{(0)}_{e,i}(\boldsymbol{x}), \quad i = 1,2,3,4 \tag{3.24}$$

and higher order basis functions

$$\Psi^{(10)}_{e,i}(\boldsymbol{x}) = (x - x_{e,i})\Psi^{(00)}_{e,i}(\boldsymbol{x}), i = 1,2,3,4$$
$$\Psi^{(01)}_{e,i}(\boldsymbol{x}) = (y - y_{e,i})\Psi^{(00)}_{e,i}(\boldsymbol{x}), \quad i = 1,2,3,4$$

and

$$\Psi^{(20)}_{e,i}(\boldsymbol{x}) = \tfrac{1}{2}(x - x_{e,i})^2\Psi^{(00)}_{e,i}(\boldsymbol{x}), \quad i = 1,2,3,4$$
$$\Psi^{(11)}_{e,i}(\boldsymbol{x}) = (x - x_{e,i})(y - y_{e,i})\Psi^{(00)}_{e,i}(\boldsymbol{x}), \quad i = 1,2,3,4$$
$$\Psi^{(02)}_{e,i}(\boldsymbol{x}) = \tfrac{1}{2}(y - y_{e,i})^2\Psi^{(00)}_{e,i}(\boldsymbol{x}), \quad i = 1,2,3,4$$

The six I^2/C^4 quadrilateral RKEM shape functions are plotted in Fig. 3.4.

The zero-th order global shape function has the form,

$$\Psi^{(00)}_I(\boldsymbol{x}) = \textstyle\sum_{k=1}^{\ell} \Psi^{(00)}_{e_k,i_k}(\boldsymbol{x})\,, \qquad \Psi^{(10)}_I(\boldsymbol{x}) = (x - x_I)\Psi^{(00)}_I(\boldsymbol{x})\,,$$
$$\Psi^{(01)}_I(\boldsymbol{x}) = (y - y_I)\Psi^{(00)}_I(\boldsymbol{x})\,, \qquad \Psi^{(20)}_I(\boldsymbol{x}) = \tfrac{1}{2!}(x - x_I)^2\Psi^{(00)}_I(\boldsymbol{x})\,,$$
$$\Psi^{(11)}_I(\boldsymbol{x}) = (x - x_I)(y - y_I)\Psi^{(00)}_I(\boldsymbol{x})\,, \qquad \Psi^{(02)}_I(\boldsymbol{y}) = \tfrac{1}{2!}(y - y_I)^2\Psi^{(00)}_I(\boldsymbol{x}).$$

4 Globally Conforming $I^m/C^n/P^k$ Hierarchy II

The globally conforming $I^m/C^n/P^k$ hierarchy constructed in the previous section has its limitations. The globally conforming hierarchy is in fact $I^m/C^n/P^m$, i.e. $k = m$. This means that the reproducing property of the interpolant is limited by the interpolation index order m, or the enrichment order. It can only reproduce complete m-th order polynomials. In order to achieve higher order accuracy, another (the second) globally conforming $I^m/C^n/P^k (k \geq m)$ hierarchy is proposed. This interpolation hierarchy can reproduce a complete kth order polynomials with $k \geq m$.

4.1 Construction

Assume that there exists a set $\{\psi_{e,i}^{(0)}, \psi_{e,i}^{(1)}, \cdots, \psi_{e,i}^{(m)}\}$ of Hermite type global polynomials such that within the element e, they can reproduce λ-th order polynomials, i.e.

$$\sum_{i\in\Lambda_e}\left\{\psi_{e,i}^{(0)}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda}+\lambda\psi_{e,i}^{(1)}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda-1}+\cdots+\frac{\lambda!}{(\lambda-m)!}\psi_{e,i}^{(m)}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda-m}\right\}=\boldsymbol{x}^{\lambda} \tag{4.1}$$

with $|\lambda|\le k$. Assume that the mesh topology has a surjective connectivity map such that

$$\Lambda_E\times\Lambda_e\to\Lambda_P:(e_1,i_1),\cdots,(e_\ell,i_\ell)\to I\ . \tag{4.2}$$

The global RKEM basis functions are constructed as follows,

$$\begin{aligned}\Psi_I^{(0)}(\boldsymbol{x})&=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}_{e_k,j}^{\rho}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi_{e_k,i_k}^{(0)}(\boldsymbol{x})\\ \Psi_I^{(1)}(\boldsymbol{x})&=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}_{e_k,j}^{\rho}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi_{e_k,i_k}^{(1)}(\boldsymbol{x})\\ &\cdots\cdots\end{aligned} \tag{4.3}$$

$$\Psi_I^{(m)}(\boldsymbol{x})=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}_{e_k,j}^{\rho}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi_{e_k,i_k}^{(m)}(\boldsymbol{x}) \tag{4.4}$$

where $I\in\Lambda_P$ and

$$\mathop{\boldsymbol{A}}_{e\in\Lambda_E}\sum_{j\in\Lambda_e}\Big(\mathcal{K}_{e,j}^{\rho}(\boldsymbol{x})\Delta V_{e,j}\Big)=1. \tag{4.5}$$

Again, the proposed RKEM I^m/C^n interpolation scheme can be written as

$$I^m f=\sum_{I\in\Lambda_P}\Big(\Psi_I^{(0)}(\boldsymbol{x})f_I+\Psi_I^{(0)}(\boldsymbol{x})Df\ \Big|_I+\cdots+\Psi_I^{(m)}(\boldsymbol{x})D^m f\ \Big|_I\Big) \tag{4.6}$$

The main properties of this globally conforming I^m/C^n hierarchy are summarized in the following proposition.

Proposition 4.1. *Assume that* $\exists\ \{\psi_{e,i}^{(0)}, \psi_{e,i}^{(1)}, \cdots \psi_{e,i}^{(m)}\}$, $\forall e\in\Lambda_E, i\in\Lambda_e$, *such that*

1.

$$D^{\alpha}\psi_{e,i}^{(\beta)}\ \Big|_{\boldsymbol{x}=\boldsymbol{x}_j}=\delta_{\alpha\beta}\delta_{ij},\quad |\alpha|,|\beta|\le m,\quad\text{and}\quad i,j\in\Lambda_e; \tag{4.7}$$

2.

$$\sum_{i\in\Lambda_e}\left\{\psi^{(0)}_{e,i}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda}+\lambda\psi^{(1)}_{e,i}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda-1}+\cdots+\frac{\lambda!}{(\lambda-m)!}\psi^{(m)}_{e,i}(\boldsymbol{x})\boldsymbol{x}_i^{\lambda-m}\right\}=x^{\lambda}\ . \tag{4.8}$$

Consider the local-global connectivity map for a given mesh,

$$\Lambda_E\times\Lambda_e\to\Lambda_P:(e_1,i_1)\cdots(e_k,i_k)\cdots(e_\ell,i_\ell)\to I\ . \tag{4.9}$$

We construct the following hybrid meshfree/FEM shape functions,

$$\Psi^{(0)}_I(\boldsymbol{x})=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}^{\rho}_{e_k,j}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi^{(0)}_{e_k,i_k}(\boldsymbol{x})$$

$$\Psi^{(1)}_I(\boldsymbol{x})=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}^{\rho}_{e_k,j}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi^{(1)}_{e_k,i_k}(\boldsymbol{x})$$

$$\cdots\cdots \tag{4.10}$$

$$\Psi^{(m)}_I(\boldsymbol{x})=\sum_{k=1}^{\ell}\Big(\sum_{j\in\Lambda_{e_k}}\mathcal{K}^{\rho}_{e_k,j}(\boldsymbol{x})\Delta V_{e_k,j}\Big)\psi^{(m)}_{e_k,i_k}(\boldsymbol{x}) \tag{4.11}$$

where $I\in\Lambda_P$.

Then the interpolation scheme (4.6) has the following properties,

1. $D^{\alpha}\Psi^{(\beta)}_I\Big|_{\boldsymbol{x}=\boldsymbol{x}_J}=\delta_{\alpha\beta}\delta_{IJ},\quad I,J\in\Lambda_P,|\alpha|,|\beta|\le m;$
2. $I^m\boldsymbol{x}^{\lambda}=\boldsymbol{x}^{\lambda},\ \forall\boldsymbol{x}\in\Omega,\ |\lambda|\le k.$

The proof of the proposition can be found in [LLHLD].

4.2 2D Example I: $T12P3I(4/3)$ Triangle Element

We first propose a two-dimensional globally conforming, bilinear, 12 degrees of freedom triangle element. It is illustrated in Fig. 4.5. The notation $I^{4/3}$ means that at each nodal point we interpolate the unknown function, say $f(x,y)$, its two first order derivatives, $\dfrac{\partial f}{\partial x}$ and $\dfrac{\partial f}{\partial y}$ and its mixed derivative, $\dfrac{\partial^2 f}{\partial x\partial y}$, which is one-third of the second derivatives. We denote the one-third of second derivative as a cross in Fig. 4.5. Since we interpolant one third of the second derivatives, the interpolation scheme is neither I^1 nor I^2. We denote it as $I^{(1+1/3)}=I^{(4/3)}$.

The global partition polynomials in an element can form a local interpolation,

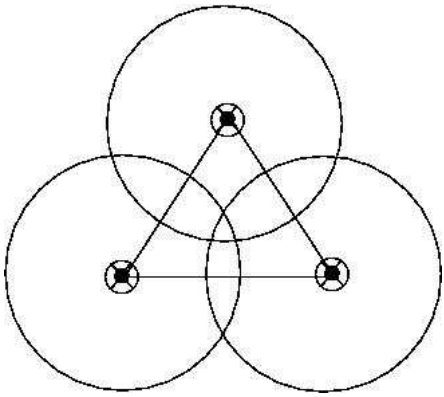


Figure 4.5. An bilinear triangle (T12P3I4/3) element.

$$\begin{aligned}\mathcal{I}_{ele}f &= \sum_{i=1}^{3}\Big(\psi_{e,i}^{(00)}f_{e,i} + \psi_{e,i}^{(10)}\frac{\partial f}{\partial x}\Big|_{(e,i)} + \psi_{e,i}^{(01)}\frac{\partial f}{\partial y}\Big|_{(e,i)} + \psi_{e,i}^{(11)}\frac{\partial^2 f}{\partial x\partial y}\Big|_{(e,i)}\Big)\\ &= c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2 + c_7x^3 + c_8x^2y + c_9xy^2\\ &\quad + c_{10}y^3 + c_{11}x^2(x^2+xy+y^2) + c_{12}y^2(x^2+xy+y^2) \qquad (4.12)\end{aligned}$$

To link the nodal data with constants, $c's$, we define vectors, $\boldsymbol{\psi}$, $\boldsymbol{f}$, $\boldsymbol{p}$, and $\boldsymbol{c}$:

$$\begin{aligned}\boldsymbol{\psi}_e^T(\boldsymbol{x}) &:= \{\psi_{e,1}^{(00)}, \psi_{e,1}^{(10)}, \psi_{e,1}^{(01)}, \psi_{e,1}^{(11)}, \psi_{e,2}^{(00)}, \psi_{e,2}^{(10)}, \psi_{e,2}^{(01)}, \psi_{e,2}^{(11)},\\ &\qquad \psi_{e,3}^{(00)}, \psi_{e,3}^{(10)}, \psi_{e,3}^{(01)}, \psi_{e,3}^{(11)}\}\ ,\\ \boldsymbol{f}_e^T &:= \{f_1, f_{1x}, f_{1y}, f_{1xy}, f_2, f_{2x}, f_{2y}, f_{2xy}, f_3, f_{3x}, f_{3y}, f_{3xy}\}_e\ ,\\ \boldsymbol{p}^T(\boldsymbol{x}) &:= \{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, x^2(x^2+xy+y^2),\\ &\qquad y^2(x^2+xy+y^2)\}\ ,\\ \boldsymbol{c}_e^T &:= \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}\}_e\ .\end{aligned}$$

We can write

$$\mathcal{I}_{loc}f = \boldsymbol{\psi}_e^T\boldsymbol{f}_e = \boldsymbol{p}^T\boldsymbol{c}_e. \qquad (4.13)$$

The nodal values are related with coefficients c's by a set of 12 simultaneous linear algebraic equations,

$$\boldsymbol{f}_e = \boldsymbol{C}_e\boldsymbol{c}_e, \qquad (4.14)$$

where matrix $\boldsymbol{C}_e$ is defined as

$$\begin{aligned}\boldsymbol{C}_e^T = \Big[&\boldsymbol{p}(\boldsymbol{x}_1), \frac{\partial\boldsymbol{p}}{\partial x}(\boldsymbol{x}_1), \frac{\partial\boldsymbol{p}}{\partial y}(\boldsymbol{x}_1), \frac{\partial^2\boldsymbol{p}}{\partial x\partial y}(\boldsymbol{x}_1), \cdots,\\ &\boldsymbol{p}(\boldsymbol{x}_3), \frac{\partial\boldsymbol{p}}{\partial x}(\boldsymbol{x}_3), \frac{\partial\boldsymbol{p}}{\partial y}(\boldsymbol{x}_3), \frac{\partial^2\boldsymbol{p}}{\partial x\partial y}(\boldsymbol{x}_3)\Big]\ . \qquad (4.15)\end{aligned}$$

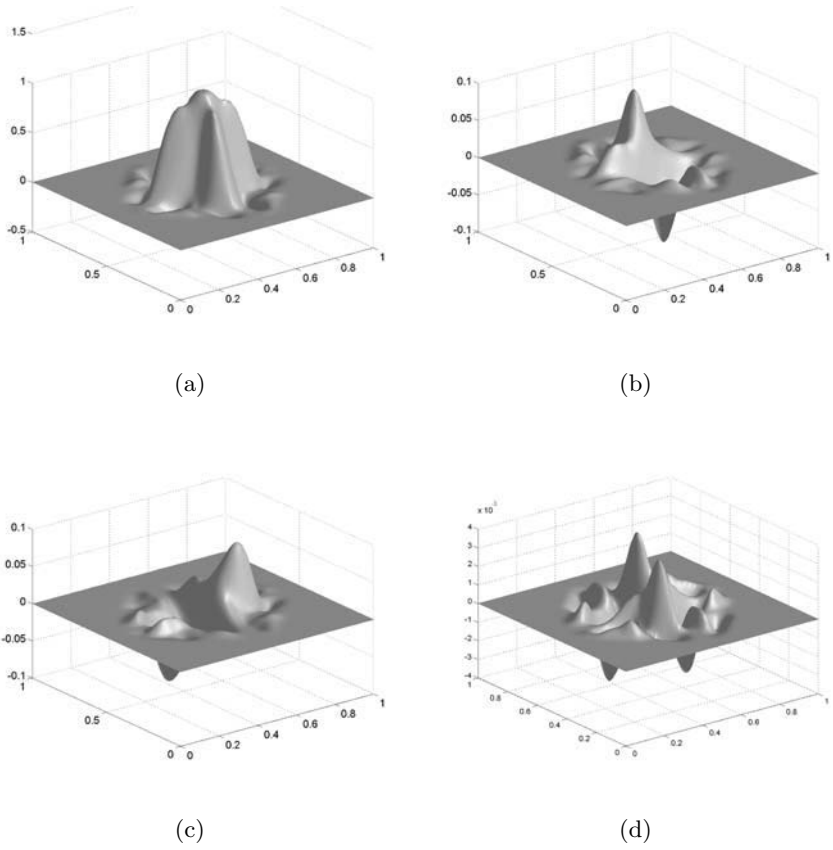


Figure 4.6. The four global shape functions of T12P3I(4/3) element: (a) $\Psi_I^{(00)}(\boldsymbol{x})$; (b) $\Psi_I^{(10)}(\boldsymbol{x})$; (c) $\Psi_I^{(01)}(\boldsymbol{x})$; (d) $\Psi_I^{(11)}(\boldsymbol{x})$.

One can then find vector $\boldsymbol{c}_e$ by solving the linear algebraic equation,

$$\boldsymbol{c}_e = \boldsymbol{C}_e^{-1}\boldsymbol{f} \ . \tag{4.16}$$

Then, the global partition polynomials can be found as

$$\boldsymbol{\psi}_e(\boldsymbol{x}) = \boldsymbol{C}_e^{-T}\boldsymbol{p}(\boldsymbol{x}) \ . \tag{4.17}$$

Assume that the mesh connectivity map is as usual

$$\Lambda_E \times \Lambda_e \to \Lambda_P : (e_1, i_1) \cdots (e_\ell, i_\ell) \to I \ . \tag{4.18}$$

The RKEM global shape functions of the triangle have the same formula,

$$\Psi_I^{(\alpha)}(\boldsymbol{x}) = \sum_{k=1}^{\ell} \sum_{j \in \Lambda_{e_k}} \left(\mathcal{K}^{\rho}_{e_k,j}(\boldsymbol{x}) \Delta V_{e_k,j} \right) \psi^{(\alpha)}_{e_k,ik}(\boldsymbol{x}) \ . \tag{4.19}$$

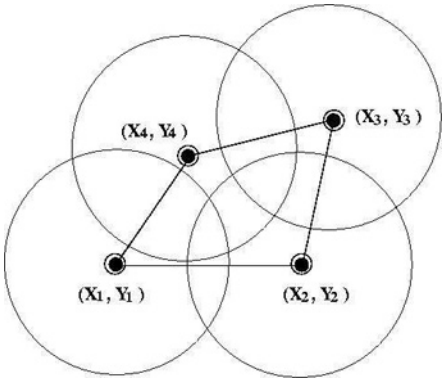


Figure 4.7. Illustration of Q12P3I1 quadrilateral element.

4.3 2D Example III: Q12P3I1 Quadrilateral Element

One can construct quadrilateral elements by using the same technique. To illustrate an example, we outline the procedure to construct a twelve degrees of freedom quadrilateral elemet, $Q12P3I1$. The element interpolation provided by the global partition polynomials in an element is

$$\begin{aligned}\mathcal{I}_{ele}f &= \sum_{i=1}^{4}\Big(\psi_{e,i}^{(0)} f_{e,i} + \psi_{e,i}^{(10)}\frac{\partial f}{\partial x}\Big|_{(e,i)} + \psi_{e,i}^{(01)}\frac{\partial f}{\partial y}\Big|_{(e,i)}\Big) \\ &= c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2 + c_7x^3 + c_8x^2y + c_9xy^2 + c_{10}y^3 \\ &\quad + c_{11}x^3(x+y) + c_{12}y^3(x+y)\ . \end{aligned} \tag{4.20}$$

To link the nodal data with constants,$c's$, vectors, $\boldsymbol{\psi}$, $\boldsymbol{f}$, $\boldsymbol{p}$, and $\boldsymbol{c}$ are defined

$$\begin{aligned}\boldsymbol{\psi}_e^T(\boldsymbol{x}) &:= \{\psi_{e,1}^{(00)}, \psi_{e,1}^{(10)}, \psi_{e,1}^{(01)}, \psi_{e,2}^{(00)}, \psi_{e,2}^{(10)}, \psi_{e,2}^{(01)}, \psi_{e,3}^{(00)}, \psi_{e,3}^{(10)}, \psi_{e,3}^{(01)}, \\ &\qquad \psi_{e,4}^{(00)}, \psi_{e,4}^{(10)}, \psi_{e,4}^{(01)}\} \qquad (4.21) \\ \boldsymbol{f}_e^T &:= \{f_1, f_{1x}, f_{1y}, f_2, f_{2x}, f_{2y}, f_3, f_{3x}, f_{3y}, f_4, f_{4x}, f_{4y}\}_e\ , \qquad (4.22) \\ \boldsymbol{p}^T(\boldsymbol{x}) &:= \{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, x^3(x+y), y^3(x+y)\}\ , \qquad (4.23) \\ \boldsymbol{c}_e^T &:= \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}\}_e\ . \qquad (4.24)\end{aligned}$$

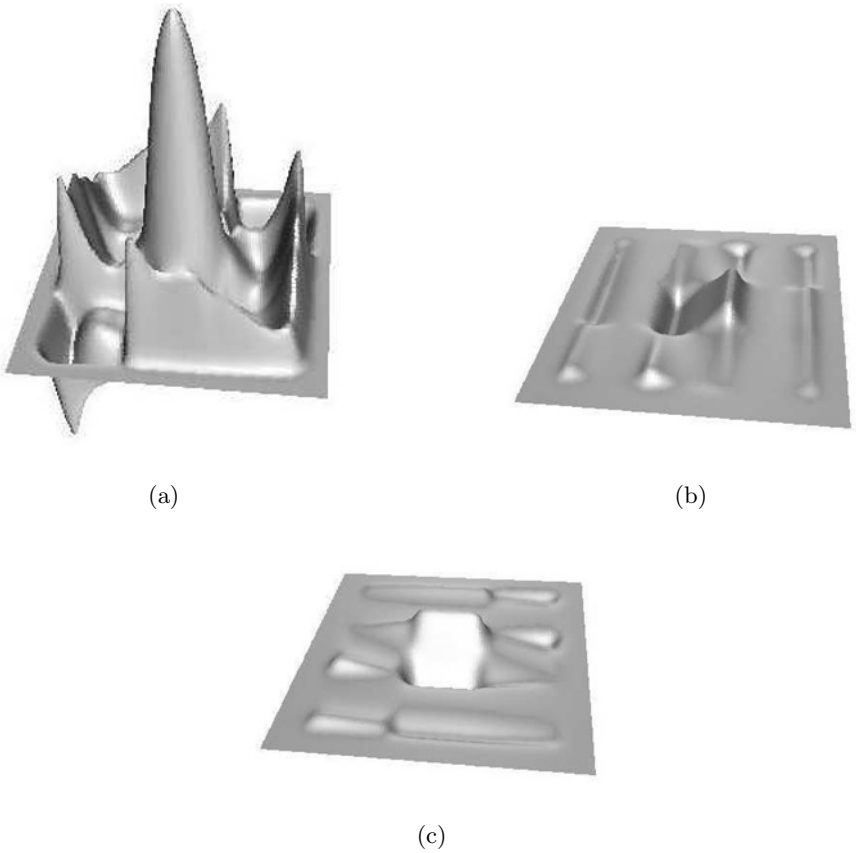
We can write that

$$\mathcal{I}_{loc}f = \boldsymbol{\psi}_e^T\boldsymbol{f}_e = \boldsymbol{p}^T\boldsymbol{c}_e. \tag{4.25}$$

Again, the nodal values are related with coefficients c's by a system of 12 linear algebraic equations,

$$\boldsymbol{f}_e = \boldsymbol{C}_e\boldsymbol{c}_e, \tag{4.26}$$

where matrix $\boldsymbol{C}_e$ is defined as,



(a) (b)

(c)

Figure 4.8. The shape functions of Q12P3I1 element.

$$\boldsymbol{C}_e^T = \left[\boldsymbol{p}(\boldsymbol{x}_1), \frac{\partial \boldsymbol{p}}{\partial x}(\boldsymbol{x}_1), \frac{\partial \boldsymbol{p}}{\partial y}(\boldsymbol{x}_1), \cdots\cdots, \boldsymbol{p}(\boldsymbol{x}_4), \frac{\partial \boldsymbol{p}}{\partial x}(\boldsymbol{x}_4), \frac{\partial \boldsymbol{p}}{\partial y}(\boldsymbol{x}_4)\right] . \quad (4.27)$$

By solving the linear algebraic equations, one can then find vector $\boldsymbol{c}_e$

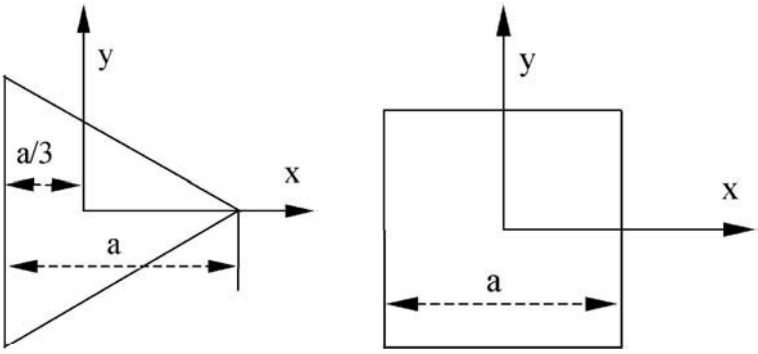
$$\boldsymbol{c}_e = \boldsymbol{C}_e^{-1} \boldsymbol{f}, \quad (4.28)$$

and then the global partition polynomials

$$\boldsymbol{\psi}_e(\boldsymbol{x}) = \boldsymbol{C}_e^{-T} \boldsymbol{p}(\boldsymbol{x}) . \quad (4.29)$$

5 Numerical Examples

To validate the method, the proposed RKEM interpolants are tested in Galerkin procedures to solve various Kirchhoff plate problems, because the



(a) Triangle plate (b) Square plate

Figure 5.9. Problem description for plate.

Galerkin weak formulation of a Kirchhoff plate involves second derivatives and a global $C^1(\Omega)$ interpolation field is the minimum requirement. Moreover, the boundary conditions of Kirchhoff plate problems involve interpolating boundary data of both the first order derivative (slopes) and the second order derivative (curvatures), it provides a severe test to the newly proposed RKEM triangle interpolants. For more information on how to impose boundary conditions for finite element computation of thin plates, readers are referred to Hughes' finite element book [Hughes87] pages 324–327.

We consider three problems: (1) a simply supported triangle plate subjected uniform load, (2) a simply supported square plate subjected uniform load, and (3) a clamped circular plate under uniform load.

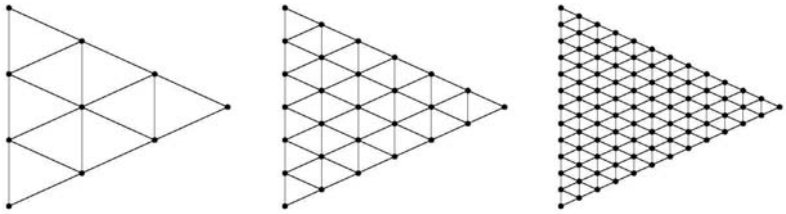
5.1 Equilateral Triangular Plate

To validate the method, an equilateral triangular thin plate under uniform load is solved first by using the proposed method. The coordinate axes are taken as shown in Fig. 5.9(a). In the case of a uniformly loaded plate with simply supported ends, the deflection of surface is given as [TW]:

$$w = \frac{p}{64aD}\left[x^3 - 3y^2x - a(x^2+y^2) + \frac{4}{27}a^3\right]\left(\frac{4}{9}a^2 - x^2 - y^2\right) \qquad (5.1)$$

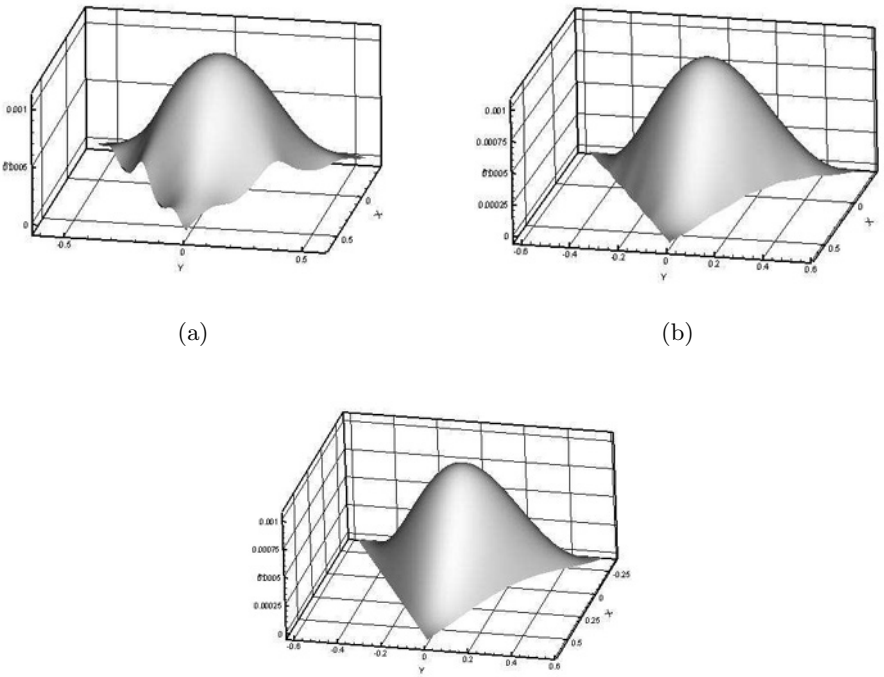
where a=1 is the height of the triangular plate.

The triangle element used in the computation is T12P3I4/3: (T12) 12 degrees of freedoms of a triangle – (P3) reproducing complete cubic (3) polynomials – (I3) interpolating 4/3 order derivatives ($f, f_x, f_x,$ and f_{xy}). Three discretizations with 9, 36, and 144 elements shown in Fig. 5.10 are used for the convergence study. Due to the generalized Kronecker delta property of globally conforming interpolation, it is easy to exactly impose the simply supported boundary conditions at boundary nodes. The maximum



(a) 10 nodes (b) 28 nodes (c) 91 nodes

Figure 5.10. The model discretization for triangle thin plate.

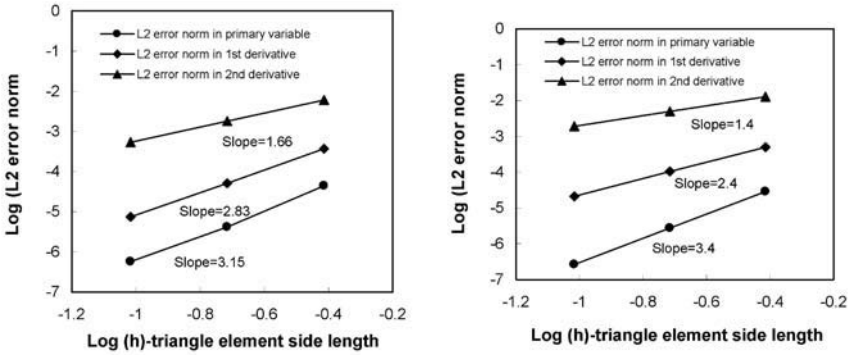


(a) (b)

(c)

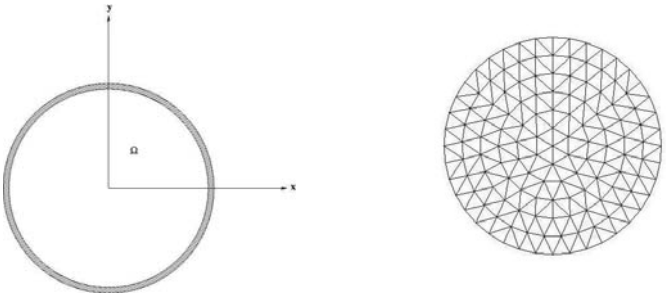
Figure 5.11. Deflection for triangle plate: (a) 9 elements, (b) 36 elements (c) 144 elements.

deflection at the center for the three cases are, $1.08428070 \cdot 10^{-3} pa^4/D$, $1.03376896 \cdot 10^{-3} pa^4/D$, and $1.02974603 \cdot 10^{-3} pa^4/D$, respectively; and the exactsolution is $1.02880658 \cdot 10^{-3} pa^4/D$. The deflection surfaces corresponding to the three models are shown in Fig. 5.11. The L_2 error norms in the primary variable, and its first and second derivatives are shown in Fig. 5.12



(a) (b)

Figure 5.12. Convergence rates of interpolation and Galerkin solutions: (a) L_2 error norms for a Galerkin solution; (b) L_2 error norms for interpolation solution.



(a) (b)

Figure 5.13. Example 3: problem domain and triangle mesh.

for interpolation and Galerkin solutions, respectively. The convergence rates in terms of error norms L_2, H^1 and H^2 are 3.2, 2.8, and 1.7 for the Galerkin solution, respectively. They match well with the convergence rates for the interpolation solution, which are 3.4, 2.4, and 1.4 in terms of error norms L_2, H^1 and H^2, respectively.

5.2 Clamped Circular Plate

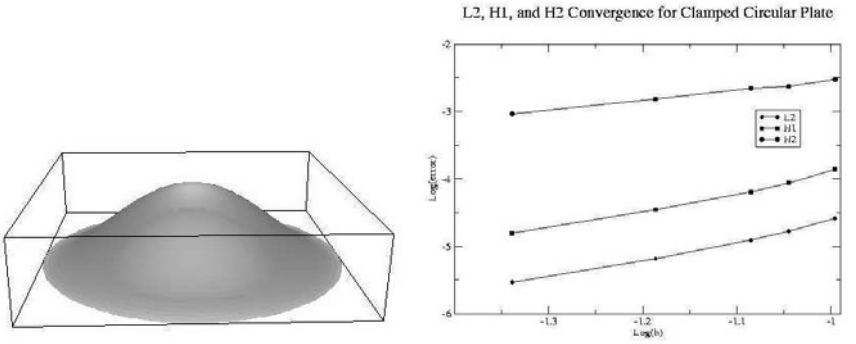
We now solve a clamped unit diameter circular Kirchhoff plate, as depicted in Fig. 5.13. The exact solution is given in [Ugural] for a plate of radius a as

$$w(x,y) = \frac{p}{64D}\left(a^2 - x^2 - y^2\right)^2$$

The deformed shape of the circular plate is juxtaposed with the convergence results depicted in Fig. 5.14. The L_2, H_1 and H_2 errors were computed

Table 5.1. Rates Of Convergence for Clamped Circular Plate.

L_2	H_1	H_2
2.6812	2.6495	1.4606



(a) (b)

Figure 5.14. Computation results for example 3: (a) deflection profile, and (b) convergence results.

and are plotted in Fig. 5.14. The slopes of the lines as determined by regression are presented in Table 5.1.

6 Closure

In conclusion, unlike most of FEM shape functions, RKEM interpolants can be used directly in numerical computations involved with the Galerkin weak formulation that contains high order derivatives of an unknown function; and unlike most meshfree interpolants, RKEM interpolant can satisfy Dirichlet boundary conditions. The basic strategy or philosophy of RKEM method is to build the interpolation field on a special partition of unity, the so-called *global partition polynomials*. In order to recover the Kronecker delta property, we use the RKPM meshfree interpolant to localize the global partition polynomials such that the global RKEM basis function is truly a interpolant that satisfies the general higher order Kronecker delta properties. By doing so, the smoothness of an interpolant only depends on the smoothness of the window function inside the meshfree kernel function, which can be easily controlled by a user in a desired manner.

Acknowledgement. This work is made possible by the support from NSF under grants CMS-0239130 to University of California (Berkeley), DMI-0115079 to Northwestern University, which are greatly appreciated.

References

[Argyris68] Argyris, J. H., Fried, I. and Scharpf, D. W.: The TUBA family of plate elements for the matrix displacement method. *The Aeronautical of the Royal Aeronautical Society*, **72**, 701–709 (1968).

[Barnhill73] Barnhill, R. E., Birkhoff, G., and Gordon, W.J.: Smooth interpolation in triangles. *Journal of Approximation Theory*, **8**, 114-128 (1973)

[Bazeley65] Bazeley, G.P., Cheung, Y. K., Irons, B. M. and Zienkiewicz, O. C.: Triangle elements in bending: conforming and nonconforming solutions. *Proc. 1st Conference on matrix methods in structural mechanics*, Wright-Patterson, AFB, Ohio (1965)

[Bell69] Bell, K. : A refined triangular plate bending finite element. *International Journal for Numerical Methods in Engineering*, **1**, 101-122 (1969)

[BLG] Belytschko, T., Lu, Y. Y. and Gu, L. : Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, **37**, 229–256 (1994)

[Birkhoff65] Birkhoff, G. and de Boor, C.: Piecewise polynomial interpolation and approximation, in *Approximation of functions*, edited by Garabedian, H. L. Elsevier, New York (1965)

[Birkhoff68] Birkhoff, G., Schultz, M. H. and Varga, R. S.: Piecewise Hermite interpolation in one and two variables with applications to partial differential equations, *Numer. Math*, **11**, 232-256 (1968)

[Clough65] Clough, R. W. and Tocher, J. : Finite element stiffness matrices for the analysis of plate bending: *Proc. 1st Conference on matrix methods in structural mechanics*, Wright-Patterson, AFB, Ohio (1965)

[dVFB65] de Veubeke, Fraeijs, B. : Bending and stretching of plates – special models for upper and lower bounds: *Proc. 1st Conference on matrix methods in structural mechanics*, Wright-Patterson, AFB, Ohio (1965)

[dVFB68] de Veubeke, Fraeijs, B. : A conforming finite element for plate bending, *International Journal of Solids and Structures*, **4**, 95-108 (1968)

[Felippa70] Felippa, C. and Clough, R. W. : The finite element in solid mechanics: in *SIAM-AMS Proceedings*, **2**, 210-252, Amer. Math. Soc. Providence, RI (1970)

[Hughes87] Hughes, T. J. R. : *The Finite Element Method*, Prentice Hall Inc., Englewood Cliffs, New Jersey (1987)

[Irons69] Irons, B. M.: A conforming quartic triangular element for plate bending. *International Journal for Numerical Methods in Engineering*, **1**, 29–45 (1969)

[Irons72] Irons, B. M. and Razzaque, A. : Experience with the patch test for convergence of finite element methods. In *Mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations*, Edited by Aziz, A. K. Academic Press. pages 557-587 (1972).

[LL] Li, S. and Liu, W. K. : Meshfree and particle methods and their applications. *Applied Mechanics Review*, **55**, 1-34 (2002).

[LLHLD] Li, S., Lu, H., Han, W., Liu, W. K., and Simkins, Jr. D. C.: Reproducing kernel element, Part II. Globally conforming I^m /C^n hierarchies. *Computer Methods in Applied Mechanics and Engineering*, **193**, 953-987, (2004).

[LZR] Liu, W. K., Zhang, Y. and Ramirez, M. R. : Multiple scale finite element methods: *International Journal for Numerical Methods in Engineering*, **32**, 969-990 (1991)

[LHLL] Liu, W. K., Han, W., Lu, H., Li, S., and Cao, J.: Reproducing kernel element method: Part I. Theoretical formulation. *Computer Methods in Applied Mechanics and Engineering*, **193**, 933-951.

[LJLAB] Liu, W. K., Jun, S., Li, S., Adee, J. and Belytschko, T.: Reproducing kernel particle methods for structural dynamics: *International Journal for Numerical Methods in Engineering*, **38**, 1655–1679 (1995)

[LJZ] Liu, W.K., Jun, S. and Zhang, Y. F. : Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, **20**, 1081–1106 (1995)

[LLB] Liu, W. K., Li, S. and Belytschko, T. : Moving least square reproducing kernel method Part I: Methodology and convergence. *Computer Methods in Applied Mechanics and Engineering*, **143**, 422–453 (1997)

[TW] Timoshenko, S. P. and Woinowsky-Krieger, S.: *Theory of Plates and Shells*, McGRAW-HILL, New York, Secomd Edition, (1959)

[Ugural] Ugural, A. C. : *Stresses In Plates And Shells*, Second Edition, McGraw-Hill, Boston (1999)

Multi-scale Analysis Using Two Influence Radii in EFGM

Sachiko Masuda[1⋆] and Hirohisa Noguchi[2⋆⋆]

[1] School of Science for Open and Environmental Systems Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, 223-8522, Japan.
[2] Department of System Design Engineering Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, 223-8522, Japan.

Abstract In this study, two Element Free Galerkin analyses using two influence radii are presented. One is implemented for one nodal points layout and provides p-like adaptivity, and the other is implemented for different nodal points layouts and multi-scale analysis. The solution of the analyses can be divided into global and local parts and they are caught simultaneously. In this research, analysis using shape function based on large influence radius provides global solutions, while solutions based on small influence radius can be regarded as local solutions or errors when the number of nodes is sufficient. Some numerical examples for the proposed p-like adaptivity show comparable or better convergence of results compared with other methods. Numerical examples for the proposed multi-scale method show that local solutions can catch local fluctuation separated from global solutions.

1 Introduction

With numerical methods such as finite element method, mesh free method, etc., several adaptive methods are proposed in order to achieve the required accuracy or to capture local behavior. One such method is p-adaptivity which does not need more nodal points but changes the order of basis into higher order. With the moving least square approximation which is applied in the element free Galerkin method, even if the order of basis function and nodal points layout are same, different influence radiuses yield different approximation orders. Thus, the order of approximation varies by the size of the influence radius used in the analysis. From this point of view, p-like adaptivity analysis is proposed by using two influence radii for the same nodal points layout. With this method, no new nodal point layout but new additional degree of freedom is required. As increase in the degree of freedom increases the order of basis, this method is termed as p-like adaptivity. In the formulation of this

⋆ sachiko@noguchi.sd.keio.ac.jp
⋆⋆ noguchi@sd.keio.ac.jp

method, the concept of multiple-field RKPM [Liu97] is directly applied. That is, the overlapping of two-scale solutions obtained by different influence radii is eliminated by the projection of local solutions using the shape function of global parts.

Another adaptive method is h-adaptivity which increases nodal points. Finer nodal points layout is used where detailed analysis is required, otherwise coarse nodal points layout is used. In the analysis, different nodal densities provide different scale solutions. With mesh-free methods, it is relatively easy to increase nodal because the method is free from generating mesh. On the other hand, it is difficult to choose the adequate influence radius when the nodal density is changing. With the proposed multi-scale analysis, coarse nodal points layout is overlapped with fine nodal points layout and the adequate influence radius is selected in each domain. In the formulation, the same concept adopted in the proposed p-like adaptivity is applied and it is shown that local solutions can be separated from global solutions. Throughout this study, the element free Galerkin method is applied [Bel94].

1.1 Moving Least Square Approximation

With the Element Free Galerkin Method (EFGM) [Bel94], the component of the displacement vector $u(\mathbf{x})$ is approximated by a polynomial function as follows, where m is the number of terms in the polynomial function. In eq. (1.1), a linear basis vector $\mathbf{p}$ and its coefficient vector $\mathbf{a}$ in a two-dimensional domain is exemplified(m=3).

$$u^h(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x})\, a_j(\mathbf{x}) \equiv \mathbf{p}^T(\mathbf{x})\, \mathbf{a}(\mathbf{x}) \tag{1.1}$$

$$\mathbf{p}^T(\mathbf{x}) = \left(1, x, y\right),\ \mathbf{a}^T(\mathbf{x}) = \left(a_1, a_2, a_3\right) \tag{1.2}$$

The coefficients in $\mathbf{a}(\mathbf{x})$ are determined by minimizing the following weighted functional.

$$J = \sum_{i=1}^{NP} w(r_i) \left\{u(\mathbf{x}_i) - u_i\right\}^2,\ r_i = |\mathbf{x} - \mathbf{x}_i| \tag{1.3}$$

whre u_i is the unknown nodal value of displacement u at node x_i, r_i is the distance between two points $\mathbf{x}$ and $\mathbf{x}_i$ and NP is the number of nodes in the domain of influence whose radius is ρ. In this paper, the following fourth order polynomial is adopted for a weight function, $w(\rho)$, which satisfies $w(\rho)$ =0, $dw/dr(\rho) = 0$ and $d^2w/dr^2(\rho) = 0$.

$$w(r_i) = \begin{cases} 1 - 6\left(\frac{r_i}{\rho}\right)^2 + 8\left(\frac{r_i}{\rho}\right)^3 - 3\left(\frac{r_i}{\rho}\right)^4 & (0 \le r_i \le \rho) \\ 0 & (\rho < r_i) \end{cases} \tag{1.4}$$

Finally, the approximation displacement $u^h(\mathbf{x})$ can be represented by the nodal value u_i, as

$$u^h(\mathbf{x}) = \sum_{i=1}^{NP} \mathbf{p}^T(\mathbf{x}) \left(\mathbf{A}^T(\mathbf{x})\mathbf{B}(\mathbf{x})\right)_i u_i \equiv \sum_{i=1}^{NP} \phi_i(\mathbf{x}) u_i \tag{1.5}$$

$$\mathbf{A}(\mathbf{x}) = \sum_{i=1}^{NP} w(r_i)\mathbf{p}(\mathbf{x}_i)\mathbf{p}^T(\mathbf{x}_i) \tag{1.6}$$

$$\mathbf{B}(\mathbf{x}) = [w(r_1)\mathbf{p}(\mathbf{x}_1), w(r_2)\mathbf{p}(\mathbf{x}_2), \cdots, w(r_{NP})\mathbf{p}(\mathbf{x}_{NP})] \tag{1.7}$$

where $\phi_i(\mathbf{x})$ corresponds to the shape function used in the finite element method.

1.2 MLSA with Discontinuous Derivatives Basis Function

With comventional moving least squars approximation(MLSA), not only the displacement field but its partial derivatives are approximated continuously. For this reason, some special treatment is needed where the derivative of displacement is discontinuous, such as strain jumps at material interfaces. In this study, for modeling of discontinuous derivative, discontinuous basis function approach [Nog03] is introduced in MLSA. When a material interface is located at $x = x_d$ in 1D problem, a bi-limear basis function is simply adopted for the discontinuous basis function insterd of the comventional linear basis function as follows;

$$\begin{cases} u^h(x) = a + b(x - x_d) & (x \le x_d) \\ u^h(x) = a + c(x - x_d) & (x > x_d) \end{cases} \tag{1.8}$$

This basis function is continuous at the interface but the slope is different at both sides of the interface. In this case, the functional to be minimized is rewritten as follows;

$$J = \sum_{i=1}^{NP_{d1}} W(r_i)\left(a + b(x_i - x_d) - u_i\right)^2 + \sum_{i=NP_{d1}+1}^{NP_{d1}+NP_{d2}} w(r_i)\left(a + c(x_i - x_d) - u_i\right)^2.$$

Where NP_{d1} is the mumber of poins in the left side of the interface ($x \le x_d$), and NP_{d2} that in the right ($x > x_d$). Therefore, the total number of sampling points in the domain of influence is $NP_{d1} + NP_{d2}$. This discontinuous basis function can completely represent bi-liner function and maintain linear reproducibility for obtained approximate function.

2 P-like Adaptivity Analysis

2.1 Displacement Approximation and Shape Functions

From eq. (1.5), approximated displacement fields calculated by large and small influence radii can be written as;

$$u^{Lh}(\mathbf{x}) = \sum_{i=1}^{NP} \phi_i^L(\mathbf{x}) u_i \tag{2.1}$$

$$u^{Sh}(\mathbf{x}) = \sum_{i=1}^{NP} \phi_i^S(\mathbf{x}) w_i \tag{2.2}$$

where NP is the total number of nodes in the domain, u_i and w_i are corresponding nodal values respectively, ϕ_i is the shape function of point i, L denotes the value related to the large influence domain, and S is the small influence domain. To evaluate displacement correctly by both u^{Lh} and u^{Sh}, overlap between the fields must be eliminated from the sum of u^{Lh} and u^{Sh}. Following [Liu97], the overlap can be calculated by multiplying one field by the shape function of another field as;

$$u^h(\mathbf{x}) = \sum_{i=1}^{NP} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NP} \phi_i^S(\mathbf{x}) w_i - \sum_{i=1}^{NP} \phi_i^L(\mathbf{x}) \left\{ \sum_{j=1}^{NP} \phi_j^S(\mathbf{x}_i) w_j \right\} \tag{2.3}$$

The last term in eq. (2.3) shows the subtracted overlap displacement. Finally displacement can be divided into two scales defined by the different influence radii.

$$\begin{aligned} u^h(\mathbf{x}) &= \sum_{i=1}^{NP} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NP} \left\{ \phi_i^S(\mathbf{x}) - \sum_{j=1}^{NP} \phi_j^L(\mathbf{x}) \phi_i^S(\mathbf{x}_j) \right\} w_i \\ &= \sum_{i=1}^{NP} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NP} \psi_i(\mathbf{x}) w_i \end{aligned} \tag{2.4}$$

where $\phi_i^L(\mathbf{x})$ is the shape function for u_i which satisfies the partition of unity condition $\psi_i(\mathbf{x})$ for w_i which satisfies the partition of nullity condition. Figure 2.1 shows shape functions of node 1, 2, 11, 20,21 and nodal configuration.

2.2 Numerical Example

As a numerical example, the Helmholtz equation under Dirichlet conditions at both sides is analyzed by the proposed method and the conventional EFGM. The following is the considered problem.

$$\frac{\partial^2 u}{\partial x^2} + k^2 u = 0, u(0) = \bar{u}, u(L) = 0 \tag{2.5}$$

The theoretical solution of eq. (2.5) is;

$$u(x) = \bar{u} \frac{\sin(k(L-x))}{\sin kL}. \tag{2.6}$$

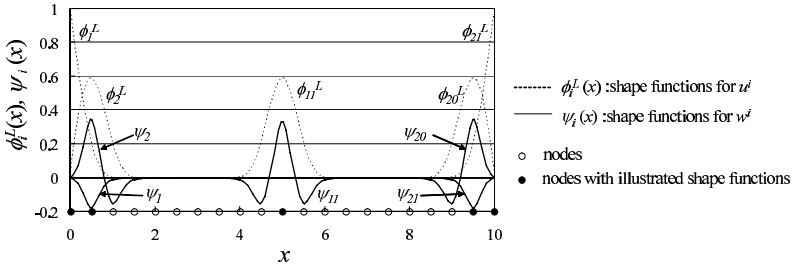


Figure 2.1. Shape functions of node 1, 2, 22, 20, 21 for p-like aduptivity analysis

The larger k is, the larger is the frequency. 2.2 shows the numerical results when $k = 32$. In all examples, the total number of nodal points are 11. The figures in (a) and (b) are analyzed by the conventional EFGM using the influence radius $\rho = 4.4c$ and $\rho = 2.2c$, respectively where c is the minimum distance between nodal points. Accuracy of the results is poor for both analyses. The proposed method shown in (c) provides reasonabe accurate.

Figure 2.3 is a convergence plot when $k = 8$ for the proposed method compared with the other methods. Linear basis are adopted in all of these methods. The reproducing kernel hierarchical partition of unity method [Li99] [Li99-2] is also p-like adoptivity method. It utilizes two shape functions, one which satisfies the partition of unity condition and the other satisfies the partition of nullity condition. Compared with other methods, a slight increase in convergence rate is observed for the proposed method. These numerical results show that the proposed method can improve the accuracy of the numerical solution of Helmholtz equations and that p-like adaptivity is achieved.

3 Multi-Scale Analysis Using Two Nodal Points Layouts

3.1 Displacement Approximation and Shape Functions

Similar to the previous section, the approximated displacement fields calculated by coarse nodal and fine nodal points layouts can be written as;

$$u^{Lh}(\mathbf{x}) = \sum_{i=1}^{NPL} \phi_i^L(\mathbf{x}) u_i \tag{3.1}$$

$$u^{Sh}(\mathbf{x}) = \sum_{i=1}^{NPS} \phi_i^S(\mathbf{x}) w_i \tag{3.2}$$

where NPL and NPS are the total numbers of nodes in the domain of coarse and fine nodal points layout, u_i and w_i are corresponding nodal values respectively. The large influence radius is adopted for the coarse nodal points layout

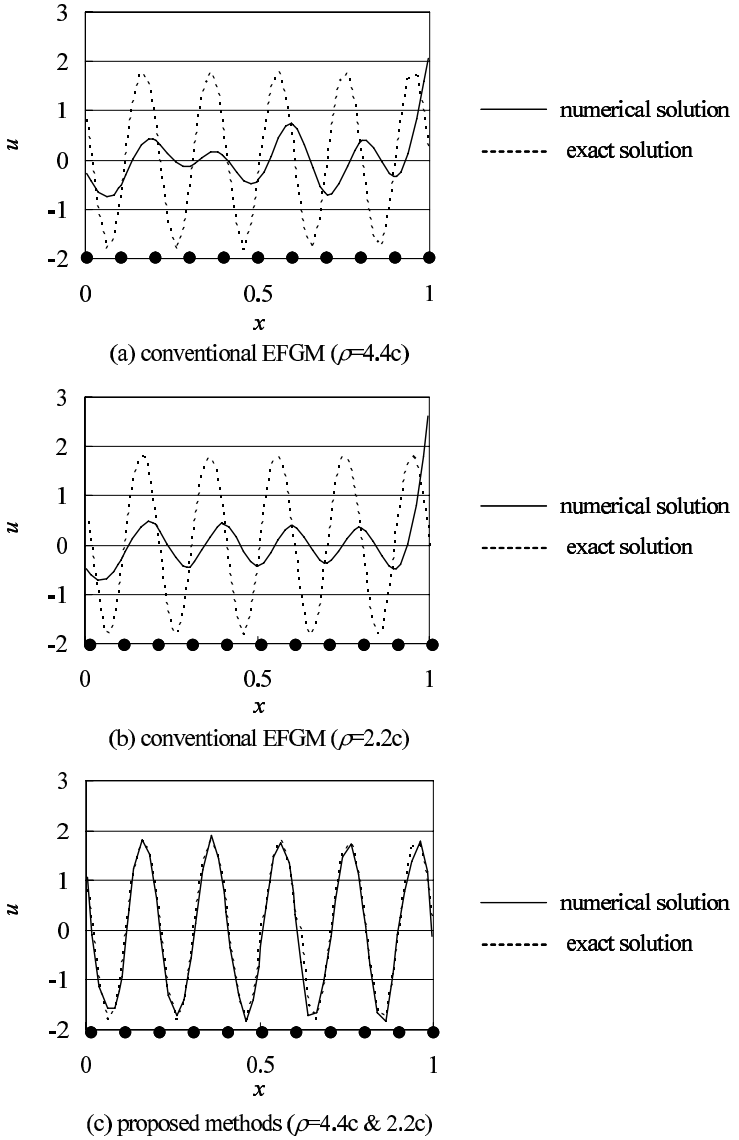


Figure 2.2. Analysis of Helmholtz equation by conventional EFGM and proposed method (c: minimum distance between nodal points)

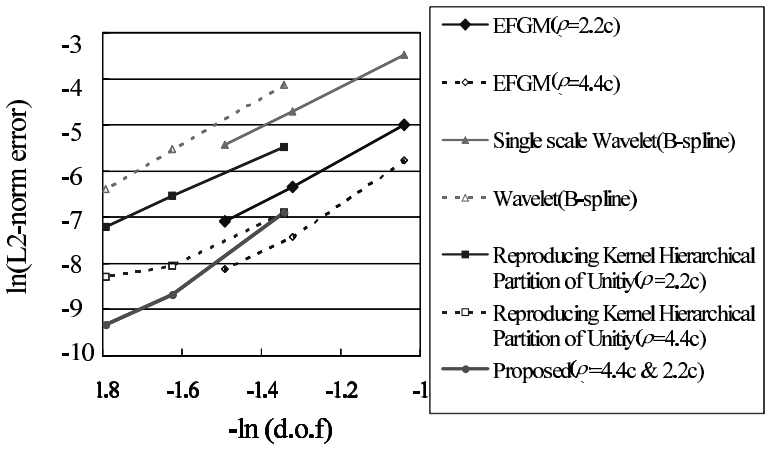


Figure 2.3. Convergence plot for proposed method ($k = 8$)

while the small influence radius is adopted for the fine nodal points layout. The overlap between the fields is calculated in more or less the same manner as described in section 2.

$$u^h(\mathbf{x}) = \sum_{i=1}^{NPL} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NPS} \phi_i^S(\mathbf{x}) w_i - \sum_{i=1}^{NPL'} \phi_i^{L'}(\mathbf{x}) \left\{ \sum_{j=1}^{NPS} \phi_j^S(\mathbf{x}_i) w_j \right\} \quad (3.3)$$

It must be noted that the multiplying shape function $\phi_i^{L'}(\mathbf{x})$ is not same as calculated in eq.(3.1), $\phi_i^L(\mathbf{x})$. For this shape function, the selected nodal points in the coarse layout, where the multi-scale analysis is conducted, are considered so that the corresponding shape function satisfies the partition of nullity cindition (Fig. 3.5). Equation (3.3) can be divided into two terms.

$$\begin{aligned} u^h(\mathbf{x}) &= \sum_{i=1}^{NPL} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NPS} \left\{ \phi_i^S(\mathbf{x}) - \sum_{j=1}^{NPL'} \phi_j^{L'}(\mathbf{x}) \phi_i^S(\mathbf{x}_j) \right\} w_i \\ &= \sum_{i=1}^{NPL} \phi_i^L(\mathbf{x}) u_i + \sum_{i=1}^{NPS} \psi_i(\mathbf{x}) w_i \end{aligned} \quad (3.4)$$

where $\phi_i^L(\mathbf{x})$ is the shape function for u_i which satisfies the partition of unity condition and $\psi_i(\mathbf{x})$ is for w_i which satisfies the partition of nullity condition. The first term of eq. (3.4) on the right hand can be regarded as global solution, and the last term as local solution. Figure 3.5 shows the shape functions for multi-scale analysis.

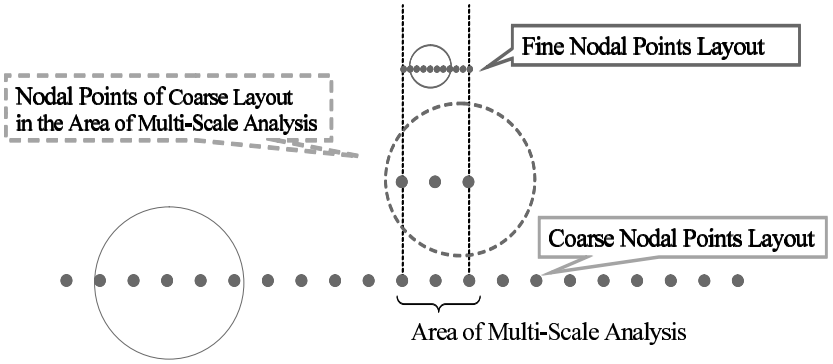


Figure 3.4. Nodal points layout for calculating shape functions

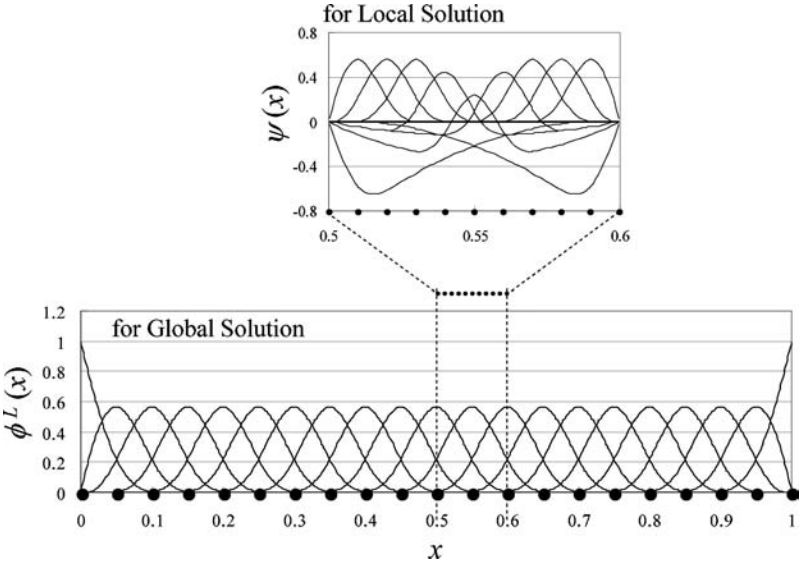


Figure 3.5. Shape functions for multi-scale analysis

3.2 Numerical Example

Poisson's equation. Poisson's equation in eq. (3.5) under the Dirichlet conditions at both sides of the domain is analyzed by the proposed method for dual-scale analysis and the conventional EFGM for single scale analysis.

$$\frac{d}{dx}\left(\lambda\frac{du}{dx}\right) = -q\,(x) \tag{3.5}$$

The following two cases are analyzed. In the first case, $q(x)$ is the following constant;

$$q(x) = 1 \tag{3.6}$$

In the second case, in order to check if the present method can capture the local fluctuation, the following q is adopted, where sinusoidal function is locally inserted incidentally.

$$q(x) = \begin{cases} 1 & (x \leq 0.5, 0.6 \leq x) \\ 1 + 40\pi^2 \cos\left[20\pi\left(x - 0.55\right)\right] + \frac{x}{10} & (0.5 \leq x \leq 0.6) \end{cases} \tag{3.7}$$

In both cases, material property and boundary conditions are followings;

$$\begin{aligned} &\lambda = 1 \\ &u(0) = u(1) = 0 \end{aligned} \tag{3.8}$$

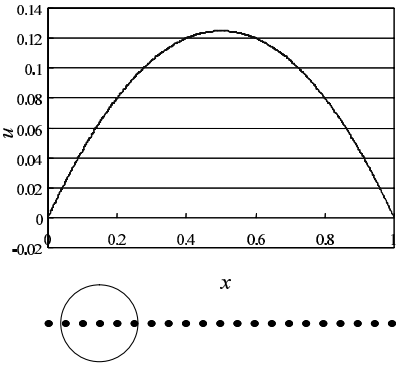
In the analysis by the proposed method, 21 nodal points are distributed for the global solution and 11 nodal points are distributed between $[0.5, 0.6]$ for the local solution. The adopted influence radii are $2.2c^S$ and $2.2c^L$ respectively, where c is minimum distance between nodal points.

Figure 3.6 shows the results in which the source term is constant. In dual scale analysis, the sum of global and local solutions become a solution of the governing equation. The results obtained from both single scale and dual-scale analysis are accurate. In dual-scale analysis, only the global solution is obtained and the local solution is 0.

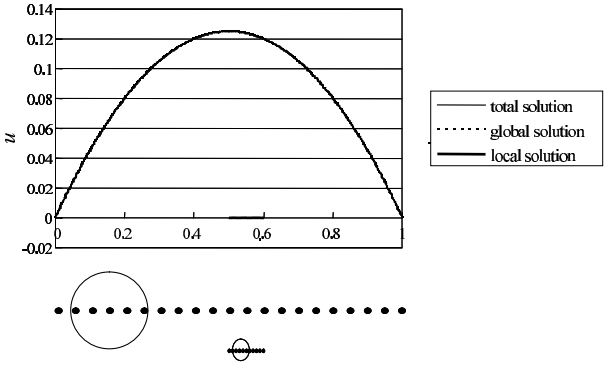
Figure 3.7 shows the results of analyses for local fluctuation. In the single scale analysis shown in (a) using different nodal points density according to the existence of local fluctuation and one uniform influence radius in the whole area, oscillation is obtained around the area with local fluctuation. In the other single scale analysis shown in (b) using fine uniform nodal points layout and uniform size of influence radius, accurate results are obtained In the proposed method shown in (c), accurate results are obtained using relatively few points compared with (b). The figure (c) shows that local solution is completely separated from global solution.

Poisson's equation with derivative discontinuity . In some cases, more detail analysis is required around the surface of discontinuity in such analysis of composite structures. The proposed multi-scale method is applied for evaluation around the surface of derivative discontinuity. As noted before, special treatment is necessary in EFGM for discontinuous displacement derivatives. In the proposed method, DBF described in section 1.2 is adopted. As a numerical example, Poisson's equation under Dirichlet boundary conditions is analyzed by the proposed method and the conventional EFGM. Material property changes at $x = 0.5$. In this study, two cases were analyzed. One has no source term and the other has source term only around material interface similar to the previous example;

$$q(x) = \begin{cases} 40\pi^2 \cos\left[20\pi\left(x - 0.45\right)\right] + \frac{x}{10} & (0.4 \leq x \leq 0.5) \\ 40\pi^2 \cos\left[20\pi\left(x - 0.55\right)\right] + \frac{x}{10} & (0.5 \leq x \leq 0.6) \\ 0 & (x \leq 0.4, 0.6 \leq x) \end{cases} \tag{3.9}$$



(a) Single Scale Analysis (Conventional EFGM)



(b) Dual Scale Analysis (Proposed Method)

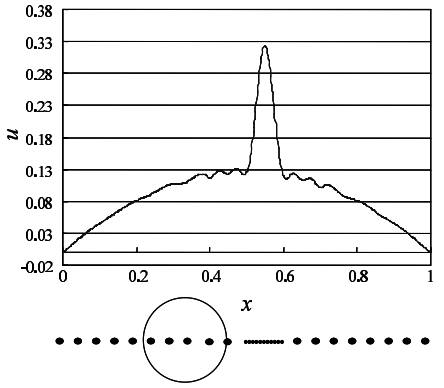
Figure 3.6. Analysis of Poisson's equation with constant source term

In both cases, material properties and boundary conditions are as follows.

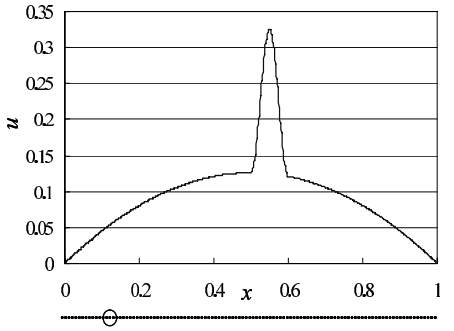
$$\lambda = \begin{cases} 1 \ (0 \leq x \leq 0.5) \\ 2 \ (0.5 \leq x \leq 1) \end{cases} \\ u(0) = 0, u(1) = 0.75 \tag{3.10}$$

In analysis without local fluctuation, bi-liner solution is obtained and the derivative has discontinuity at material interface $x = 0.5$. In analysis with local fluctuation, there is derivative discontinuity and local fluctuation around $x = 0.5$.

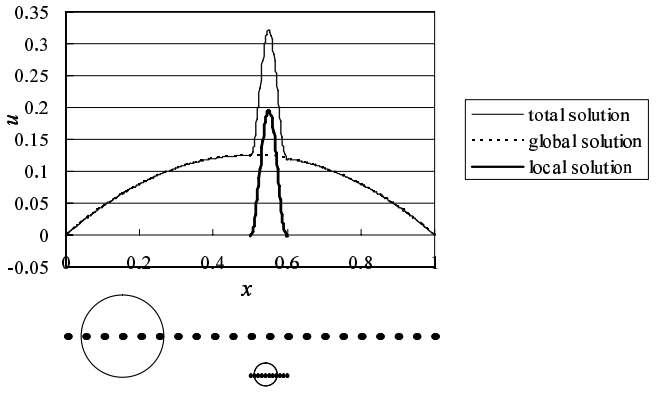
Figure 3.8 shows the results of analysis without local fluctuation using discontinuous basis function as special treatment for discontinuous derivative. In the both analyses by the conventional EFGM and the proposed method, perfect solutions are obtained and error does not occur. In the analysis by the proposed method (c), only global solution is obtained and local solution is 0. If the special treatment for derivative discontinuity is not utilized, oscillations



(a) Single Scale Analysis (29 d.o.f)
(Conventional EFGM)



(b) Single Scale Analysis (101 d.o.f)
(Conventional EFGM)



(c) Dual Scale Analysis (21+11 d.o.f.)
(Proposed Method)

Figure 3.7. Analysis of Poisson's equation with local fluctuation and constant source term

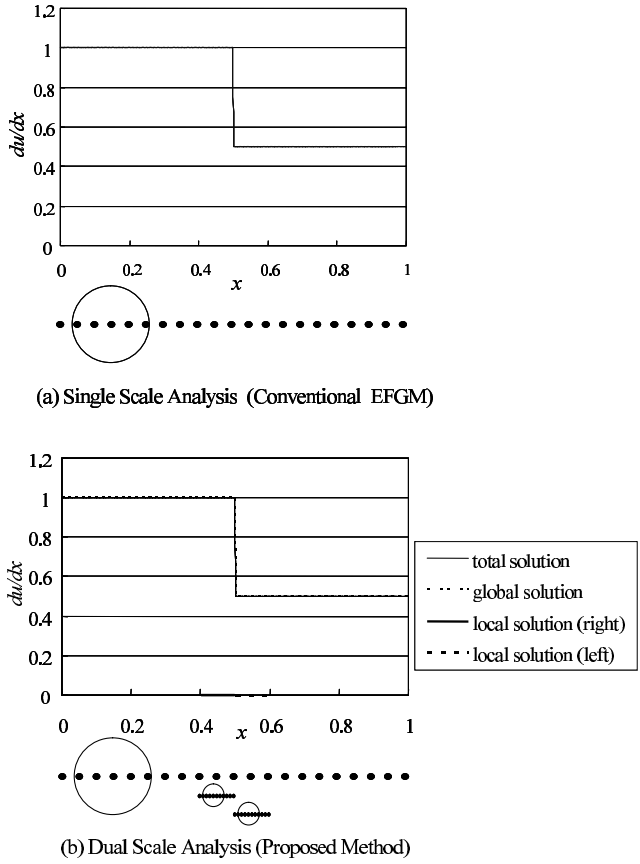
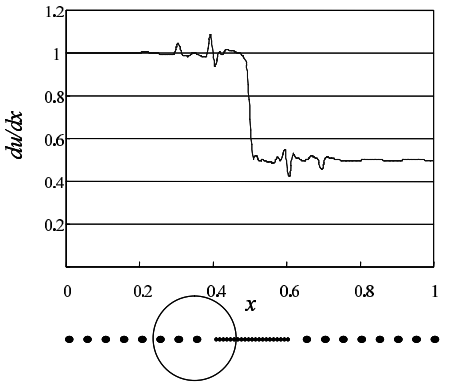


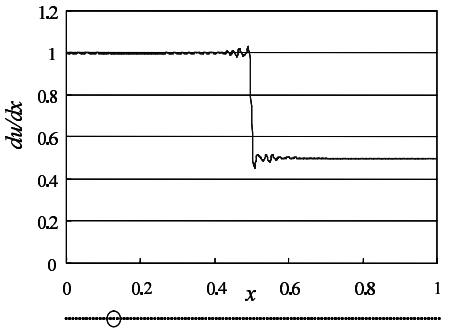
Figure 3.8. Analysis of Poisson's equation with derivative discontinuity (discontinuous basis function is adopted)

are observed in the conventional EFGM analysis as shown in Fig.3.9. By using proposed method (c), local solution functions to catch the discontinuous derivative and the solution is improved but not perfect.

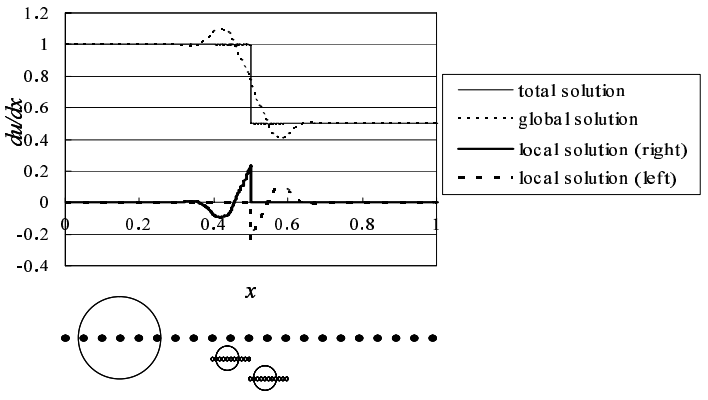
Figure 3.10 shows results of analyses with local fluctuation around material interface using DBF. In all of these analyses, discontinuous derivative is obtained by using DBF. In conventional EFGM, oscillations occur due to local fluctuation when coarse nodal points layout is used (a). On the other hand, accurate results can be obtained with relatively few points (c) by the proposed method, although finer nodal points are necessary to obtain the same accuracy in conventional EFGM (b).



(a) Single Scale Analysis (37 d.o.f)
(Conventional EFGM)



(b) Single Scale Analysis (101 d.o.f)
(Conventional EFGM)



(c) Dual Scale Analysis (21+11+11 d.o.f.)
(Proposed Method)

Figure 3.9. Analysis of Poisson's equation with derivative discontinuity (no special treatment for discontinuous derivative is adopted)

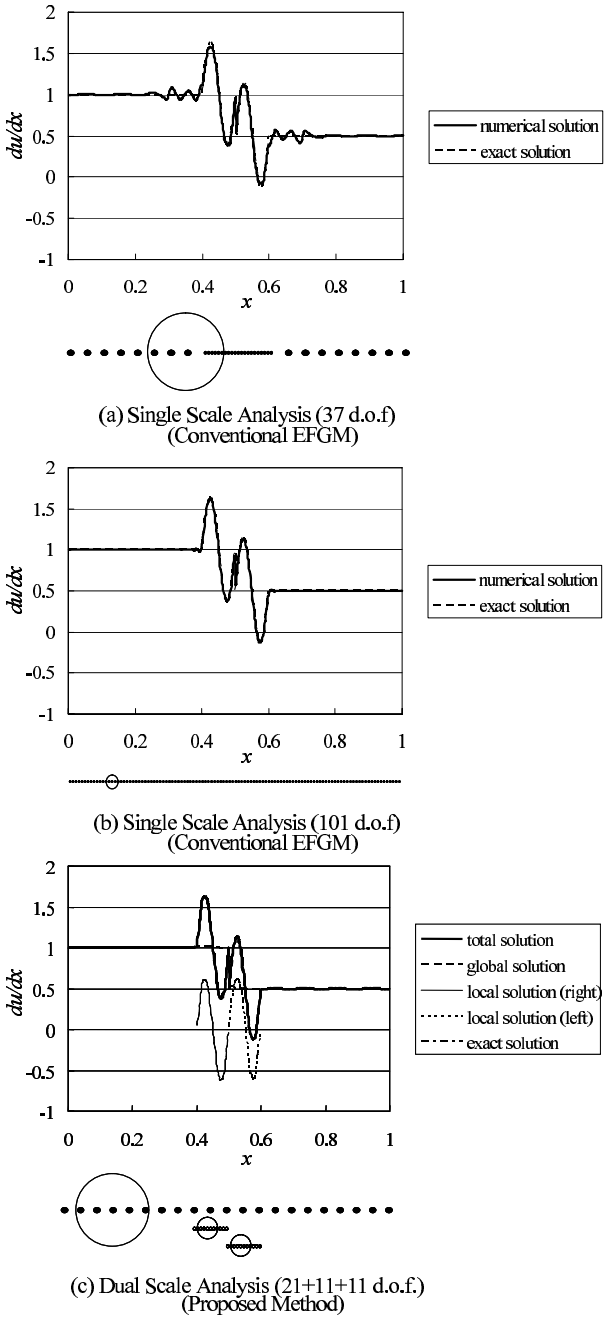


Figure 3.10. Analysis of Poisson's equation with derivative discontinuity and local fluctuation (discontinuous basis function is adopted)

4 Conclusion

By using two influence radii in EFGM, p-like adaptivity is achieved for same nodal points layout, and dual-scale analysis is achieved for different nodal points layouts with adequate sizes of influence radii. This method simplifies the choice of size of influence radius. Even for the discontinuous derivative field, the proposed method always provides accurate results together with the discontinuouos basis function.

Acknowledgement. The supported of this work by Grant in Aid for the 21st century center of Excellence for "System Design : Paradigm Shift from Intelligence to Life" from Ministry of Education, Culture, Sport, and Technology in Japan is greatly acknowledged.

References

[Bel94] Belytschko, T., Lu, Y.Y., Gu, L.: Element-free Galerkin methods. Int. J. numer. methods eng., **37**, 229–256 (1994)

[Liu97] Liu, W.K., Uras, R.A., Chen, Y.: Enrichment of the finite element method with the reproducing kernel particle method. Trans. ASME J. App. Mech., **64**, 861–870 (1997)

[Li99] Li, S., Liu, W.K.: Reproducing kernel hierarchical partition of unity, Part1. Int. J. numer. methods eng., **45**, 251–288 (1999)

[Li99-2] Li, S., Liu, W.K.: Reproducing kernel hierarchical partition of unity, Part2. Int. J. numer. methods eng., **45**, 289–317 (1999)

[Nog03] Noguchi, H., Sato, Y., Masuda, S.: Analysis of Structure with Material Interface by Meshfree Method. Computers and Structures, submitting

Solution of a Dynamic Main Crack Interaction with a System of Micro-Cracks by the Element Free Galerkin Method

Boris Muravin[1,2⋆], Eli Turkel[1], and Gregory Muravin[2]

[1] School of Mathematical Sciences, Department of Applied Mathematics, Tel-Aviv University, Tel-Aviv 69978, Israel.
[2] Margan Physical Diagnostics Ltd., P.O.B. 8155, Ha-Omanut 12, Netanya 42160, Israel.

Abstract Damage and failures in high-pressure equipment and in high-energy piping have increased significantly during the second part of the twentieth century, in spite of improved construction procedures and the high quality of materials used. The result has been a grave expansion in the number of fatal disasters and ecological catastrophes, and their harmful social and economic consequences. This trend is apparent from a brief analysis of extensively developing industrial activities in different countries of the world, such as chemical, refinery and gas-treatment enterprises, power and the nuclear power industry. The statistics of failures in these industries show that most of the damage was caused by systems of interacting flaws.

To numerically tackle these problems a previously developed code by the authors, based on the Element Free Galerkin (EFG) solution of systems of strongly interacting static cracks, was modified and adapted for dynamic problems in fracture mechanics. Several numerical examples of single crack propagation under impulse loading are solved. Accuracy of the results is verified comparing several analytical and numerical methods. The developed method is then applied to the physical model of dynamic crack propagation in the field of interacting flaws.

1 Introduction

Dynamic fracture and crack arrest criteria have only begun to appear in the literature. The models of the interaction of flaws under dynamic load, the interaction of a main crack with a system of flaws, and the dynamic growth of a crack in a field of interacting cracks are not well developed. There are no general, reliable, efficient and accurate methods for the numerical solution of these problems.

As result of the lack of reliable experimental methods and procedures for evaluation of stress and strain fields, stress intensity factors and energy release

⋆ bm@margan.com

rates , unexpected and rarely predictable failures caused by individual flaws and systems of interacting flaws continue to occur in high-pressure equipment operating under dynamic loading, especially in enclosed spaces (nuclear reactors, submarines, shuttles). These failures jeopardize operational safety, and can lead to fatal disasters, catastrophes, and social and economical losses.

The problem of single crack behavior under dynamic pulse load was investigated numerically by different authors. Parton and Boriskovsky (1985, [1]), Nakamura, Shih and Freund (1985, [2]) were the first who solved it by the Finite Element method. Later, the problem was analyzed by other numerical methods. Lu, Belytschko and Tabbara in 1995 [3] and Organ in 1996 [4] have applied the Element Free Galerkin method and Chen, Gerlach, and Belytschko (2001, [5]) used the Extended Finite Elements method for the solution of the problem. Techniques of these works and the results of our investigations devoted to the solution of static interacting cracks problems are used here with certain adaptation for the solution of dynamic multi-crack problems. These will be considered below in detail.

2 The EFG Method for Dynamic Linear Elastic Fracture Mechanics

For numerical solution of dynamic linear elastic fracture mechanics problems by the EFG method, the discrete equation is derived from the variational form of the governing equation of motion.

Consider the analytical equation of motion

$$\sigma_{ij,j} + b = \rho \ddot{u}_i, \qquad \text{in domain } \Omega \text{ bounded by } \Gamma, \tag{2.1}$$

where σ_{ij} is the stress tensor, b is the body force and ρ is the material density.

Equation (2.1) has the following boundary conditions (b.c.) and initial conditions (i.c.):

$$\text{Essential b.c.: } u_i = \bar{u}_i \text{ on } \Gamma_u, \tag{2.2}$$

$$\text{Natural b.c.: } \sigma_{ij} n_j = \bar{t}_i \text{ on } \Gamma_t, \tag{2.3}$$

$$\text{Displacement i.c.: } u(x,0) = u_0(x), \tag{2.4}$$

$$\text{Velocity i.c.: } \dot{u}(x,0) = v_0(x), \tag{2.5}$$

where $\bar{u}, \bar{t}$ are prescribed boundary displacements and tractions respectively, and n is the unit normal vector to Γ.

The variational form of the equation (1) is

$$\int_{\Omega} \rho \delta u \cdot \ddot{u} d\Omega + \int_{\Omega} \sigma^T \delta \varepsilon d\Omega = \int_{\Omega} \delta u \cdot b d\Omega - \int_{\Gamma_t} \delta u \cdot \bar{t} d\Gamma, \tag{2.6}$$

where δ is the variation operator.

The discrete form of equation (2.6) is

$$M\ddot{u} + f^{int} = f^{ext}, \tag{2.7}$$

M is the mass matrix diagonalized by the row-sum method, f^{ext} and f^{int} are the vectors of external and internal forces respectively.

$$M_{II} = \sum_J \int_\Omega \rho \phi_I \phi_J d\Omega, \tag{2.8}$$

$$f_I^{ext} = \int_{\Gamma_t} \phi_I \bar{t} d\Gamma + \int_\Omega \phi_I b d\Omega, \tag{2.9}$$

$$f_I^{int} = \int_\Omega B_I^T \sigma d\Omega, \tag{2.10}$$

where ϕ_I are the EFG shape functions, B_I is a matrix of shape function derivatives:

$$B_I = \begin{bmatrix} \phi_{I,x} & 0 \\ 0 & \phi_{I,y} \\ \phi_{I,y} & \phi_{I,x} \end{bmatrix}. \tag{2.11}$$

For the time evolution approximation we use the following three schemes: a regular second-order accurate central difference scheme, a fourth-order accurate difference scheme and the Newmark method.

The time evolution approximation by a regular central difference scheme is given by:

$$u_{t+\Delta t} = 2u_t - u_{t-\Delta t} + \Delta t^2 M_t^{-1} \left(f_t^{ext} - f_t^{int} \right). \tag{2.12}$$

We derive the time evolution approximation by a fourth-order accurate scheme from the Taylor series:

$$u_{t+\Delta t} = u_t + \Delta t \dot{u}_t + \frac{\Delta t^2}{2} \ddot{u}_t + \frac{\Delta t^3}{6} \dddot{u}_t + \frac{\Delta t^4}{24} \ddddot{u}_t + ... \tag{2.13}$$

$$u_{t-\Delta t} = u_t - \Delta t \dot{u}_t + \frac{\Delta t^2}{2} \ddot{u}_t - \frac{\Delta t^3}{6} \dddot{u}_t + \frac{\Delta t^4}{24} \ddddot{u}_t + ... \tag{2.14}$$

Summarizing (2.13) and (2.14) and truncating the series we obtain:

$$u_{t+\Delta t} + u_{t-\Delta t} = 2u_t + \Delta t^2 \ddot{u}_t + \frac{\Delta t^4}{12} \ddddot{u}_t + O\left(\Delta t^6\right). \tag{2.15}$$

Then

$$\ddot{u}_t = \frac{u_{t+\Delta t} - 2u_t + u_{t-\Delta t}}{\Delta t^2} - \frac{\Delta t^2}{12} \ddddot{u}_t + O\left(\Delta t^4\right). \tag{2.16}$$

The second-order accurate backward difference is used to approximate the fourth-order material derivative u_t, which results in fourth-order convergence for the difference equation.

$$\overset{\cdots\cdot}{u_t} = \frac{2\ddot{u}_t - 5\ddot{u}_{t-\Delta t} + 4\ddot{u}_{t-2\Delta t} - \ddot{u}_{t-3\Delta t}}{\Delta t^2} + O\left(\Delta t^2\right). \tag{2.17}$$

Finally we get the following time evolution rule:

$$u_{t+\Delta t} = 2u_t - u_{t-\Delta t} + \Delta t^2 A_t + \frac{\Delta t^2}{12}\left(2A_t - 5A_{t-\Delta t} + 4A_{t-2\Delta t} - A_{t-3\Delta t}\right), \tag{2.18}$$

where

$$A_t = M_t^{-1}\left(f_t^{ext} - f_t^{int}\right). \tag{2.19}$$

Another technique that can be used for numerical integration is the two parameter Newmark family of methods. These methods truncate the Taylor series:

$$u_t = u_{t-\Delta t} + \Delta t\dot{u}_{t-\Delta t} + \frac{\Delta t^2}{2}\ddot{u}_{t-\Delta t} + \frac{\Delta t^3}{6}\overset{\cdots}{u}_{t-\Delta t} + \ldots, \tag{2.20}$$

$$\dot{u}_t = \dot{u}_{t-\Delta t} + \Delta t\ddot{u}_{t-\Delta t} + \frac{\Delta t^2}{2}\overset{\cdots}{u}_{t-\Delta t} + \ldots. \tag{2.21}$$

The parameters β and γ of the Newmark method are defined in following expressions using (2.20) and (2.21):

$$u_t = u_{t-\Delta t} + \Delta t\dot{u}_{t-\Delta t} + \frac{\Delta t^2}{2}\ddot{u}_{t-\Delta t} + \beta\Delta t^3\overset{\cdots}{u}_{t-\Delta t}, \tag{2.22}$$

$$\dot{u}_t = \dot{u}_{t-\Delta t} + \Delta t\ddot{u}_{t-\Delta t} + \gamma\Delta t^2\overset{\cdots}{u}_{t-\Delta t}. \tag{2.23}$$

If the acceleration is assumed to be linear then:

$$\overset{\cdots}{u_t} = \frac{\ddot{u}_t - \ddot{u}_{t-\Delta t}}{\Delta t}, \tag{2.24}$$

$$u_t = u_{t-\Delta t} + \Delta t\dot{u}_{t-\Delta t} + \left(\frac{1}{2} - \beta\right)\Delta t^2\ddot{u}_{t-\Delta t} + \beta\Delta t^2\ddot{u}_t, \tag{2.25}$$

$$\dot{u}_t = \dot{u}_{t-\Delta t} + (1-\gamma)\,\Delta t\ddot{u}_{t-\Delta t} + \gamma\Delta t\ddot{u}_t. \tag{2.26}$$

Finally,

$$\left(b_1 M + K\right)u_t = f_t^{ext} + M\left(b_1 u_{t-\Delta t} - b_2\dot{u}_{t-\Delta t} - b_3\ddot{u}_{t-\Delta t}\right), \tag{2.27}$$

$$\ddot{u}_t = b_1\left(u_t - u_{t-\Delta t}\right) + b_2\dot{u}_{t-\Delta t} + b_3\ddot{u}_{t-\Delta t}, \tag{2.28}$$

$$\dot{u}_t = b_4 \left(u_t - u_{t-\Delta t}\right) + b_5 \dot{u}_{t-\Delta t} + b_6 \ddot{u}_{t-\Delta t}, \tag{2.29}$$

where

$$b_1 = \frac{1}{\beta \Delta t^2}, b_2 = -\frac{1}{\beta \Delta t}, b_3 = -\left(\frac{1}{2\beta} - 1\right),$$
$$b_4 = \gamma \Delta t b_1, b_5 = 1 + \gamma \Delta t b_2, b_6 = \Delta t \left(1 + \gamma b_3 - \gamma\right). \tag{2.30}$$

The method is unconditionally stable if

$$2\beta \geq \gamma \geq \frac{1}{2}, \tag{2.31}$$

and there is energy damping if

$$\gamma > \frac{1}{2}. \tag{2.32}$$

It should be noted that in the dynamic problems the weight functions and their domains of influence are time dependent. That leads to the time dependence of the shape functions. Since the time dependence of the shape functions and their derivatives can not be analytically expressed for the case of arbitrary dynamic fracture, these functions should be updated with every time step, based on the new crack geometry and the updated weight function. Next we will use previously described methods for the numerical solution of the dynamic problems of our interest.

3 Solution of Multi-Crack Problems by the EFG Method

Currently, the solution of problems with many cracks by a finite element method requires enormous mesh refinement near each crack tip, including the embedding of many singular elements. When using meshless methods and the EFG method in particular, many additional nodes should be added around and between the cracks and the nodal domain of influence should be lowered significantly. As a result a huge computational effort is needed for the solution.

To overcome the problem, we developed an algorithm for the construction of weight functions to handle strongly interacting multi-crack, where the distance between cracks can be smaller then the domain of influence of the nodes [6]. This algorithm modifies the diffraction method [7] so that it can characterize simultaneously all crack tips located in the nodal domain of influence. The same algorithm can be used to modify other methods for construction of weight functions. Among these are methods based on the modification of the weight function shape near crack tips, such as transparency [7] and visibility methods [8]. The algorithm was presented for example of the diffraction method in [6]. We call this method the *multiple crack weight* (MCW) method.

This section presents the reliability and accuracy of the EFG method together with the MCW method for the solution of multiple cracks problems.

Numerical examples of interacting and intersecting cracks are calculated in terms of the stress intensity factors and compared to available reference solutions provided by other numerical methods. Problems about star-shaped cracks under bi-axial static load are chosen to illustrate the main aspects of solution of multiple crack problems. The convergence of the stress intensity factors as a function of number of nodes is analyzed and discussed.

In all the calculations the nodal distribution was regular with additional nodes along the cracks surfaces, star-shaped array of nodes around of the free tips of the cracks and an additional node at the free tips of the cracks. The radius of the outer ring of star-shaped additional nodes was equal to 0.75 of the distance between regular nodes. The regular nodal distribution was used as a mesh for the numerical integration by the Gauss quadrature rule with 12×12 Gauss quadrature points in each cell. The fully enriched basis [9] was coupled with a linear basis. The outer radius of the enrichment was smaller than the outer radius of the star-shaped additional nodes. A plane strain condition was assumed.

Six intersecting cracks were used to model a star-shaped crack (Fig. 1a). The normalized stress intensity factors $F_I^A = K_I^A/\sigma\sqrt{\pi a}$, $F_I^B = K_I^B/\sigma\sqrt{\pi a}$ and $F_{II}^B = K_{II}^B/\sigma\sqrt{\pi a}$ were calculated using the domain form of the interaction integral for a/W=0.5 and several ratios of a/h, where W is the half width of the specimen, a is the crack length and h is the average nodal spacing. The results presented in Fig. 2 show the convergence of the solution as the mesh is refined. The stress intensity factors are oscillating while they converge to their limiting values. The amplitude of the oscillations vanishes as the ratio a/h increases. The relative differences between the stress intensity factors, F_I^A, F_I^B, F_{II}^B calculated with a/h=5 mesh and those calculated with a/h=10 mesh were 0.23%, 0.2%, 0.75% respectively.

Comparing these results with the reference solutions [10,11], one sees that the accuracy of the solution is acceptable even for the relatively small ratio of a/h= 5 (mesh with 21×21 regular nodes) and that the results agree satisfactorily with the references in [10,11] (Table 1, case a/W=0.5). This is despite the fact that the small regular nodal distribution can not match inclined crack lines properly, several cells of the integration mesh are crossed by cracks and two crack tips are in the nodal domain of influence of many nodes. This demonstrates that the modified EFG method combined with MCW method is able to solve accurately multiple crack problems with relatively small nodal distributions.

The same mesh distributions that were used for the cross-shaped crack problem were used to solve the star-shaped crack problem for different a/W ratios. The calculated normalized stress intensity factors (Table 1) were compared with those calculated by X-FEM in [10] for two different meshes distributions (Table 1 ref 1 and ref 2) and by Cheung et al in [11] (Table 1 ref 3). Our results show good agreement with the reference results and are closer to those provided by X-FEM than to results of Cheung et al.

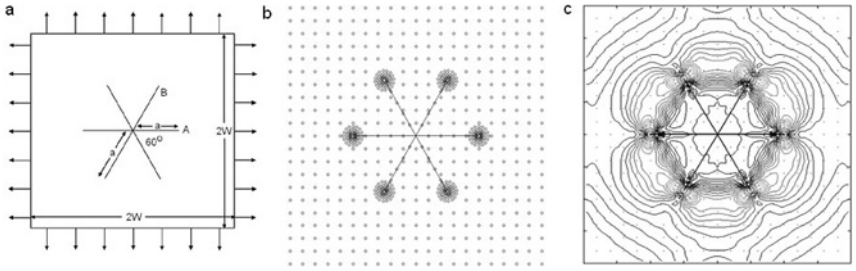


Figure 3.1. Finite-size plate with star-shaped crack under bi-axial loading (a). Mesh distribution for a/W=0.5 (b). Von Mises stress distribution for a/W=0.5 (c).

Table 3.1. Normalized stress intensity factors for the problem in Fig. 1.

		F_I	F_I ref 1	F_I ref 2	F_I ref 3	E_1, %	E_2, %	E_3, %
$a/W = 0.2$	F_I^B	0.7690	0.7683	-	0.7578	0.09	-	1.48
$a/W = 0.2$	F_{II}^B	0.0007	0.0005	-	0.0004	*	-	*
$a/W = 0.2$	F_I^A	0.7691	0.7670	0.7746	0.7570	0.27	0.71	1.60
$a/W = 0.3$	F_I^B	0.7994	0.7983	0.7973	0.7884	0.14	0.26	1.40
$a/W = 0.3$	F_{II}^B	0.0020	0.0021	0.0021	0.0022	*	*	*
$a/W = 0.3$	F_I^A	0.7970	0.7931	0.7942	0.7846	0.49	0.35	1.58
$a/W = 0.4$	F_I^B	0.8527	0.8466	0.8466	0.8365	0.72	0.72	1.94
$a/W = 0.4$	F_{II}^B	0.0077	0.0080	0.0064	0.0070	*	*	*
$a/W = 0.4$	F_I^A	0.8352	0.8287	0.8332	0.8255	0.78	0.24	1.18
$a/W = 0.5$	F_I^B	0.9232	0.9255	0.9208	0.9087	0.25	0.26	1.60
$a/W = 0.5$	F_{II}^B	0.0201	0.0184	0.0168	0.0168	*	*	*
$a/W = 0.5$	F_I^A	0.8921	0.8864	0.8928	0.8815	0.64	0.08	1.20
$a/W = 0.6$	F_I^B	1.0405	1.0445	1.0401	1.0182	0.38	0.04	2.19
$a/W = 0.6$	F_{II}^B	0.0451	0.0364	0.0350	0.0388	*	*	*
$a/W = 0.6$	F_I^A	0.9749	0.9673	0.9760	0.9758	0.79	0.11	0.09
$a/W = 0.7$	F_I^B	1.2384	1.2367	1.2369	1.1936	0.14	0.12	3.75
$a/W = 0.7$	F_{II}^B	0.0622	0.0593	0.0614	0.0529	*	*	*
$a/W = 0.7$	F_I^A	1.1022	1.0971	1.1120	1.1142	0.46	0.88	1.08
$a/W = 0.8$	F_I^B	1.5577	1.5624	1.5593	-	0.30	0.10	-
$a/W = 0.8$	F_{II}^B	0.0804	0.0864	0.0826	-	*	*	-
$a/W = 0.8$	F_I^A	1.3454	1.3423	1.3581	-	0.23	0.94	-
$a/W = 0.9$	F_I^B	2.1605	2.1927	2.1659	-	1.47	0.25	-
$a/W = 0.9$	F_{II}^B	0.0906	0.0868	0.088	-	*	*	-
$a/W = 0.9$	F_I^A	1.9146	1.9037	1.9578	-	0.57	2.21	-

$E_{1,2,3}$ represent the percent difference of normalized stress intensity factors F_I^A, F_I^B with reference solutions ref 1,2,3.

* The percent difference is not representative in this case since the calculated and reference values of F_{II}^B are small or close to zero. We note that in this case there is even a significant percent difference between the three reference solutions.

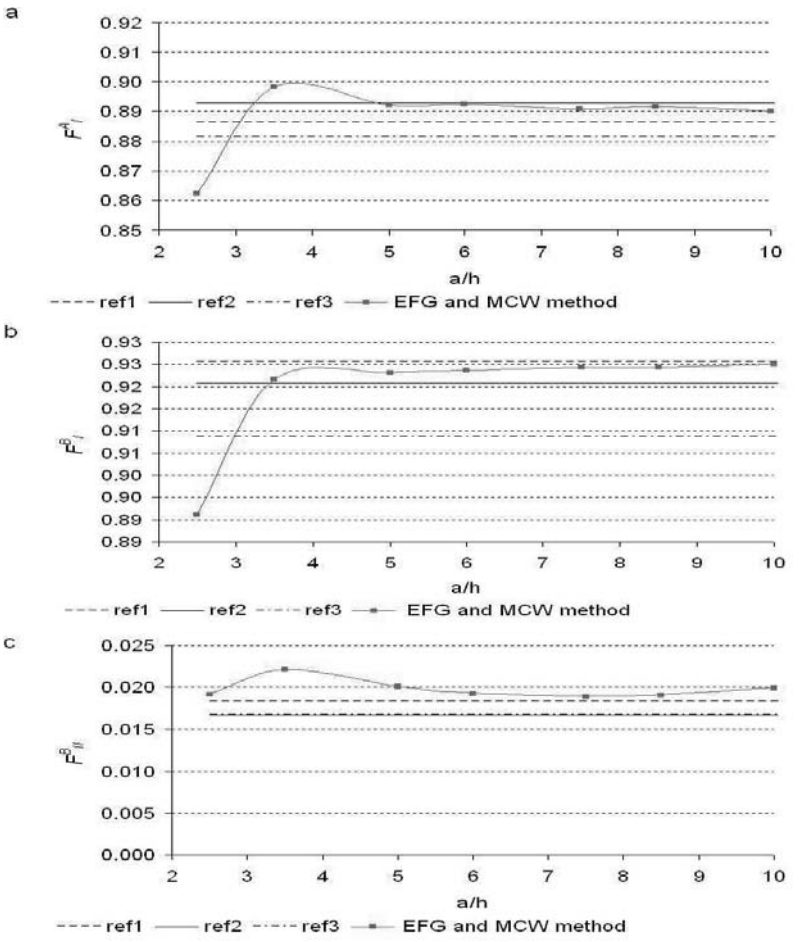


Figure 3.2. Convergence of the normalized stress intensity factors (a) F_I^A, (b) F_I^B, (c) F_{II}^B as a function of the average nodal spacing. The results of ref 2 and ref 3 are identical in (c).

The results show that the MCW method allows the solution of strongly interacting cracks without enormous mesh refinement and reduces significantly the computational efforts. Thus in different numerical examples the computational time was two to four time shorter when using the EFG method with MCW method because smaller nodal distributions were required compared with the standard EFG method for obtaining accurate results.

4 Individual Crack Subjected to Pulse Loading

4.1 Introduction

In this section we consider the physical model of Cherepanov [12] which describes crack behavior under dynamic pulse load and discuss the accuracy of the numerical solution by the EFG method. This model considers semi-infinite crack development in an infinite body subjected to dynamic impulse and includes several stages:

- The concentration of the stress around the crack tip and the increasing of stress intensity factor.
- Initiation of a crack's propagation and subsequent crack elongation with increasing speed.
- Decelerating and arresting of a crack after removal of applied forces.

Cherepanov solved analytically the stress intensity factor for the stationary crack under the pulse of a rectangular pulse of finite duration $\sigma_y(t)$ applied normally to the crack plane:

$$\sigma_y(t) = \sigma_0 \left[H(t) - H(t-T) \right], \tag{4.1}$$

where σ_0 is the pulse amplitude; T is the pulse duration; H is the Heaviside function.

According to his solution, the stress intensity factor, K_{st} of static crack subjected to impulse loading is:

$$K_{st}(t) = \frac{4\sigma_0 c_2}{c_1\sqrt{\pi c_1}} \sqrt{c_1 - c_2} \left[\sqrt{t} H(t) - \sqrt{t-T} H(t-T) \right], \tag{4.2}$$

where c_1 and c_2 are dilatation and shear wave speeds, respectively.

The corresponding to K_{st} energy release rate G_{st} in condition of opening mode of fracture is:

$$G_{st} = \frac{K_{st}^2}{E'}, \tag{4.3}$$

where $E' = E$ for plane stress and $E' = E/(1-\nu)^2$ for plane strain problems, where E is the elastic modulus and ν is the Poisson ratio.

The energy release rate G of propagating crack with velocity $\dot{l}$, can be obtain from the expression provided by Freund [13]:

$$G\left(t, \dot{l}\right) = g\left(\dot{l}\right) G_{st}(t), \tag{4.4}$$

where

$$g\left(\dot{l}\right) = 1 - \dot{l}/c_R, \tag{4.5}$$

and c_R is the Rayleigh wave speed. The velocity of the crack propagation is obtained from:

$$\dot{l} = c_R \left(1 - G_{critical}/G_{st}\left(t\right)\right). \tag{4.6}$$

For numerical solution of the energy release rate, we use a domain form of path-independent integral proposed by Moran and Shih [14]:

$$G = -\int\limits_A \left(\left(W + L\right)\delta_{1j} - \sigma_{ij}u_{i,1}m_j\right)q_{,j}dA - \int\limits_A \left(\rho\dot{u}_i\dot{u}_{i,1} - \rho\ddot{u}_i u_{i,1}\right)qdA, \tag{4.7}$$

where q is a weight function that equals one on the inner boundary of the integral domain, zero on the outer boundary and is arbitrary elsewhere.

We assume that Cherepanov's model satisfies the real conditions of fracture in high energy equipment. Nevertheless, the further research of the phenomenology of dynamic fracture under pulse loading especially for the case of interacting cracks requires the analysis of the cracks characteristics such energy release rates, calculation of the direction of the crack propagation and the dynamic stress and strain fields.

4.2 Analysis of Accuracy

In this section we discuss the accuracy of the EFG method for the solution of static and propagating individual cracks under impulse load. For this we solve the numerical example of a finite plate with width 10m and height 4m and an edge crack with length of 5m subjected to impulse load as illustrated in Fig. 3. The time of the problem is limited to the time that it takes for the stress wave to pass down to the bottom boundary, reflect from its surface and then arrive back to the crack line, t=$3H/c_1$, where H is the half of specimen's height. For longer times the comparison of the numerical solution with the existing analytical solutions [12,13] is not valid.

The dynamic loading was created by an incident tensile pulse with amplitude σ_0=1000 N/m^2 suddenly applied at the top boundary of the stress free specimen with the following material properties: $E = 211 \cdot 10^9$ N/m^2, ρ=7800 kg/m^3, ν=0.3. A plane strain condition was assumed.

The selected EFG scheme includes a regular nodal distribution, a 4×4 Gauss quadrature rule, and linear basis functions. The time step used for numerical integration was calculated according to the Courant condition, where the critical time step is $dt_c = h/c_1$, and h is the minimum nodal spacing. The time evolution approximation was done using the central difference scheme.

For the analysis of the numerical scheme we started with two examples of the crack behavior under impulse loading:

- Crack remains stationary for the entire calculation.
- Crack remains stationary until $t_1 = 1.5H/c_1$ and then starts to propagate with constant velocity $V = 0.4c_2$. We note that the analytical value of the energy release rate at $t_1 = 1.5H/c_1$ is equal to $0.475G^* = 4.5 \cdot 10^{-6} Pa \cdot m$, where $G^* = H\sigma_0^2/E$. We will assume later in numerical examples that the critical energy release rate G_{1c} is equal $0.475G^*$.

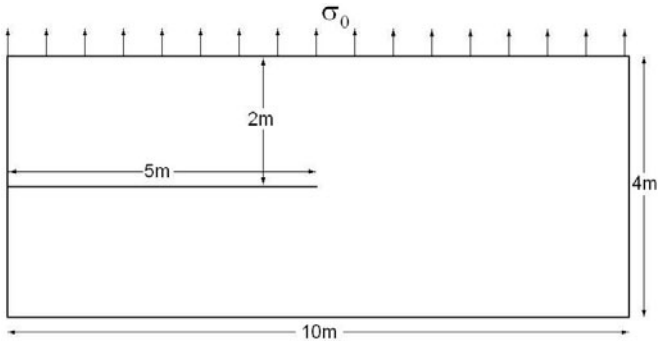


Figure 4.3. Edge crack in the specimen modeling infinite plate subjected to pulse loading.

In both cases the energy release rates were calculated and compared to the analytical values which were obtained from the solution of Cherepanov [12] and Freund [13]. It should be noted that the energy release rate is a linear function of time in this example. Later this characteristic will be used in the accuracy analysis.

Several nodal distribution were used (120×48, 180×72, 240×96, 300×120 for the first case of stationary crack and 120×48, 240×96 for the second case of a stationary and then propagating crack) to show the convergence of the solution as a function of nodal density. To satisfy the Courant condition for all nodal densities we selected the time step equal to $0.5 \cdot 10^{-6}$ sec.

The analysis of the results concerning the first case showed that the energy release rate is growing smoothly but oscillating along the analytical line (Fig. 4a, b, c, d) from the moment when the stress wave arrives at the crack line ($t = H/c_1$). For the higher nodal densities the amplitude of the oscillation is decreases. To assess the accuracy of the solution quantitatively, we propose to use several statistical tools. Thus, we calculate the average, E and the standard deviation, S of the relative errors in the energy release rates calculated for every time step, and the coefficient of correlation, R between the energy release rates and time.

Since the analytical energy release rate is a linear function of time, the coefficient of correlation was used as a tool to assess the linear dependence of the numerically calculated energy release rate on time. The standard deviation is calculated to characterize the dispersion of relative errors between the calculated energy release rates and the analytical values. Thus, one can see in Table 2 that the coefficient of correlation, R converges to 1 as the nodal density increased and the linear character of the energy release rate in time improves. At the same time, the standard deviation and the average error of the numerical results decrease.

In the second case, after the time $t = 1.5H/c_1$ when the crack starts to propagate at a constant velocity one can see that the solution oscillates

with higher amplitudes compared to the case of a static crack, see Fig. 4e, f. The drop in the energy release rate predicted by the analytical solution of Freund [13] at $t = 1.5H/c_1$ is clearly seen in our results. For the higher nodal discretization this drop was sharper.

Table 4.2. Average, E, standard deviation, S, and coefficient of correlation, R, of the relative error in the energy release rate.

Nodal distribution	E	S	R
120×48	0.07046	0.14017	0.99899
180×72	0.05530	0.12307	0.99947
240×96	0.04768	0.11322	0.99963
300×120	0.04008	0.04008	0.99977

To analyze the solution's oscillations we conducted several investigations. It was established, that the solution with a higher order Gauss quadrature for numerical integrations of the mass matrix and f^{ext}, f^{int}, and energy release rate did not yield significant improvement, leaving the same character of oscillations. Different sizes of energy release rate domain also did not change the results, proving that the dynamic energy release rate calculated by (4.7) is independent of the domain size. Decreasing the time step, application of enrichment techniques and a fourth-order accurate scheme did not show any obvious differences in the results. We therefore concluded that the accuracy is more sensitive to the nodal density than to other parameters of the appropriate numerical scheme selected before. In our opinion one of the possible ways to reduce the oscillations for dynamic crack is to add the moving star-shaped arrays of nodes around the tip of the developing crack.

In addition, the Newmark family of methods was used for numerical integration. Solving the energy release rate for static cracks with different values of β and γ parameters, it was established that for $\beta = 1/4$ and $\gamma = 3/4$ the solution is more accurate and smoother, Fig. 5a,b. Thus, for 120×48 nodal distribution, E equals 0.06682, $S = 0.13845$ and $R = 0.99956$, and for a 180×72 nodal distribution, $E = 0.04838$, $S = 0.11549$ and $R = 0.99978$.

This Newmark scheme induces small energy damping, which improves the solution of this particular problem. However, it should be used with care in other problems. So far there is no well established proof that the procedure can provide an accurate solution for multiple crack problems, where small solution oscillations could be the result of crack interactions and not due to numerical error.

We note that the level of the oscillations of the energy release rate calculated by (4.7) in Fig. 4e,f are lower than found in other works [15], [16]. In [15] it was noted that "the oscillations for the moving cracks coincide with the number of nodes passed during propagation; the peaks occur just before the crack tip passes a node". Therefore it is important to take into consideration

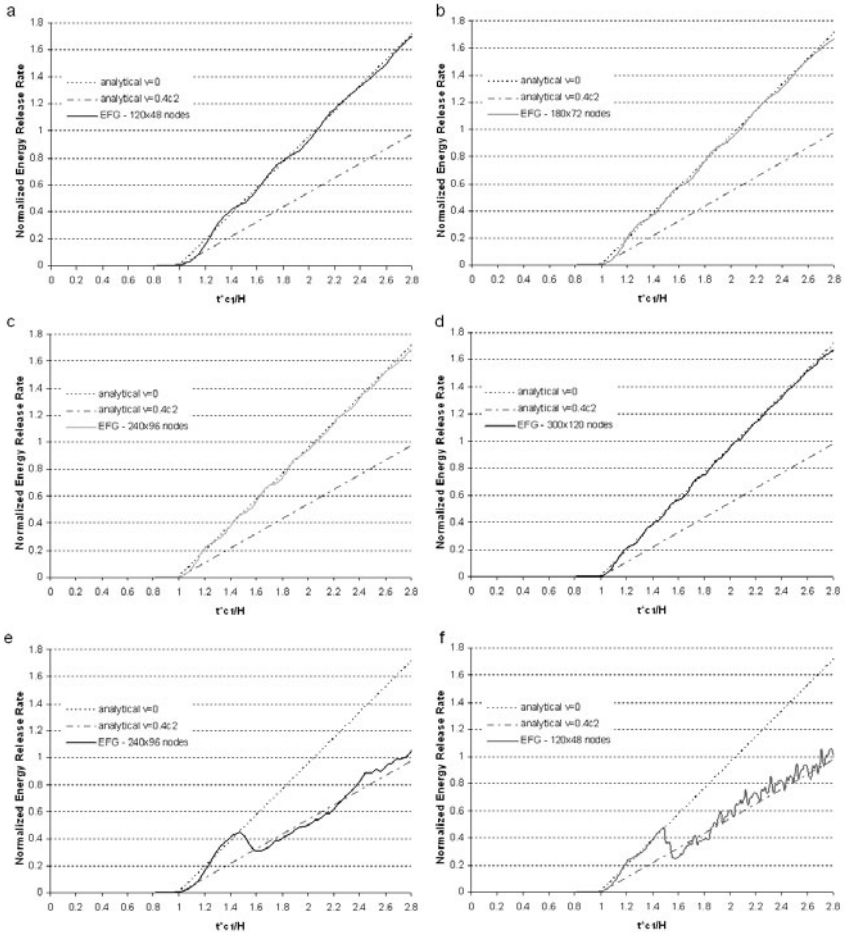


Figure 4.4. Energy release rates for the static crack and static and then dynamic crack propagating at constant velocity, normalized by $H\sigma_0^2/E$.

the character of the oscillations when analyzing different physical phenomena of propagating cracks to prevent misleading results. Next we shall apply the considered EFG scheme with the central difference scheme to the numerical calculation of Cherepanov's model.

5 Main Crack Propagation in a Field of Interacting Flaws under Dynamic Pulse Loading

Below we shall consider the typical case of failure which occurs in high-pressure equipment - main crack propagation in a field of interacting flaws under in-

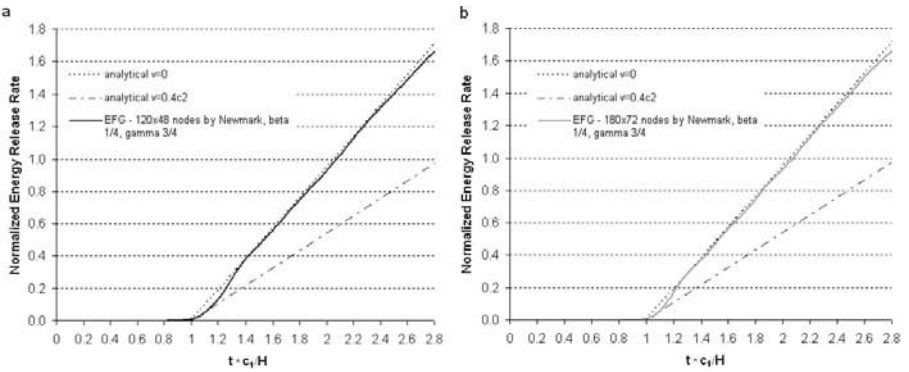


Figure 4.5. Energy release rates normalized by $H\sigma_0^2/E$ for a static crack subject to pulse loading calculated by Newmark method with $\beta = 1/4$ and $\gamma = 3/4$.

fluence of pulse load of finite duration. We use the same specimen for the modeling of the problem as in the previous section. The applied impulse has a finite duration equal H/c_1. The geometry of the main crack and the system of micro-cracks is shown in Fig. 6. In addition we solve the problem of the single main crack propagation without micro-cracks for the same specimen and pulse load. Three stages of Cherepanov's model were analyzed and compared to the case of a single crack problem to establish the effect of a system of micro-cracks on the energetic characteristics of the main crack.

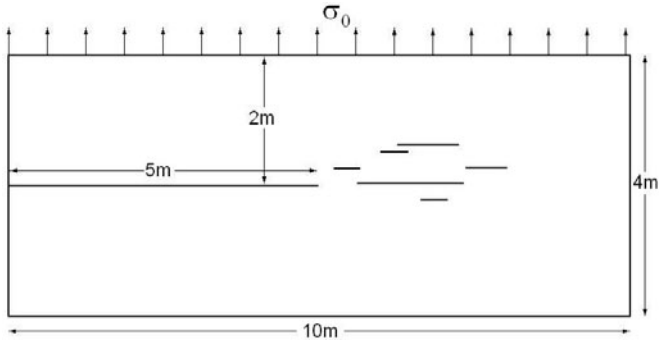


Figure 5.6. Main crack interacting with system of micro-cracks in the plate modeling infinite plate subjected to tensile pulse loading.

The calculation established that initially a pulse load creates a uniformly distributed stress field that moves toward the main crack (Fig. 7a). Later, when the stress wave arrives at time H/c_1 to the main crack line, the tip of the main crack starts to accumulate and concentrate energy (Fig. 7b). As a result the energy release rate grows linearly (Fig. 7f). There is no clear

indication of interaction between the main crack and the system of micro-cracks. The energy release rate of the main crack is not different from the case of the single crack. However, as the stress intensity continues to rise around the tip of the main crack and system of micro-cracks, the interaction effect begins at approximately $t = 1.3H/c_1$. The energy release rate increases relative to the case of the single main crack, indicating that an amplification effect is taking place.

In the zone where the system of cracks is located there are clear signs of a non-uniform stress concentration (Fig. 7b). Some of the cracks concentrate the energy intensively, while others, shielded by external cracks from the pulse stress wave, which is moving in their direction, and do not accumulate stresses. The combined stress field around the system of the cracks is extremely anisotropic.

The main crack starts to propagate when the energy release rate of the main crack exceeds its critical value (Fig. 7f). The initiation of crack growth occurs earlier than in the case of a single crack due to the stress amplification effect. From the beginning, the crack propagated with higher velocity and elongates faster relative to the case of a single crack without the system of micro-cracks (Fig. 7g, h). As the distance between the main crack and the system of micro-cracks decrease, the amplifying effect increases and the redistribution of the stress field becomes obvious (Fig. 7c, d). The maximum velocity that the main crack interacting with the system of micro-cracks reaches is 1050m/s, while for the single crack it is 700m/s. The stress field around the tip of the main crack starts to interact with the stress field of system of micro-cracks and the redistribution of the stress field becomes obvious (Fig. 7c, d), indicating dynamic interaction.

The main crack decelerates as the impulse interrupts at time $t = 2H/c_1$. Its velocity and the rate of its elongation decrease (Fig. 7g, h). However, the energy accumulated in the specimen supports fracture development and the main crack continues its motion toward the micro-crack system. It increases the interaction effect and as a result an interesting phenomenon occurs: the main crack continues to propagate for a longer distance compared with the case of a single crack despite the absence of load on the specimen's boundaries (Fig. 7h). The deceleration of a main crack (Fig. 7g) continues until approximately t=3H/c1 relative to t=2.14H/c1 in the case of a single crack.

One can observe oscillations in the numerical results of the velocity in Fig. 7g after t=2H/c1. As in the case of a single crack propagating with constant velocity considered previously, the peaks occur just before a crack tip passes a node. However, the amplitude of the velocity oscillations attenuates indicating further crack arrest. It should be noted, that the character of the oscillation does not adversely affect the main physical phenomena observed in this example - the main crack elongates for larger lengths if interacting with other cracks. We wish to stress that the same level of oscillations appears in the velocity of a moving single crack in [15].

After all these, the results of the analysis showed us that:

- The energy release rate varies and depends strongly on the position of the applied dynamic load, the direction of the stress wave propagation and on the mutual position of the tip of the main crack and the system of cracks and their sizes.
- The crack may start to propagate earlier if interacting with other flaws.
- The interaction of the main crack with the system of micro-cracks changes the velocity of the main crack propagation.
- The process of crack development under pulse loading and crack deceleration can take longer due to the stress amplification effect.
- The main crack may elongate for larger lengths if interacting with other cracks. This may lower the component's life time subjected to impulse loading.

In the above, we forced the main crack to propagate linearly and the system of micro-cracks was assumed to be static. This was done in order to:

- Compare the behavior of a single main crack undergoing pulse load and when a main crack interacting with a system of micro-cracks.
- Investigate how a system of micro-cracks located at different distances from the tip of the main crack changes the energy release rate, velocity, energy flow density distribution around the main crack.

We will consider below the problem where the main crack and micro-cracks inside of the system of flaws start to propagate at a constant velocity equal to $0.4c_2$when their energy release rates exceed the critical value G_{1c}, under the influence of a pulse load. We postulate that cracks propagate according to the maximum principal stress criterion.
The results of the calculations show that:

- When the accumulated energy at the cracks tips becomes critical and cracks start to propagate, the interaction effect is taking place. This result in the local redistribution of stresses at the tips of the cracks. This changes the direction of propagation of the main crack and of individual cracks of the system of micro-cracks.
- Cracks initially located perpendicular to the applied forces do not propagate linearly. The trajectory of the growing cracks is presented in Fig. 8.
- Some micro- cracks that accumulate more energy start to propagate early and pass through longer distances than others do.
- Micro-cracks that are shielded by other cracks from the pulse stress wave do not propagate.
- Cracks that are located parallel to each other tend to increase the distance between themselves to reduce the shielding effect and then propagate parallel to each other.
- Propagating cracks which are collinear or shifted relative to each other, including the main crack tend to coalescence similar to what was observed in of metallurgical investigations.

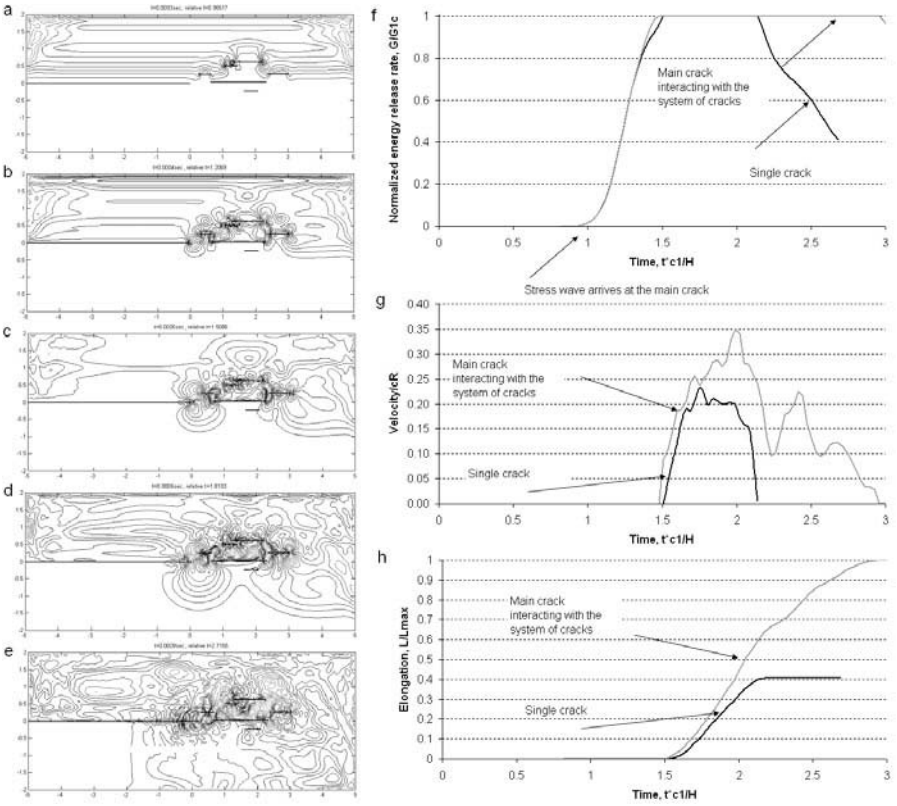


Figure 5.7. (a)-(e) Von-Mises stress distribution for the main crack interacting with the system of micro-crack and (f) energy release rate, (g) velocity and (h) elongation of the propagating main crack interacting with a system of micro-cracks compared with a single main crack. See Color Plate 5 on page 296.

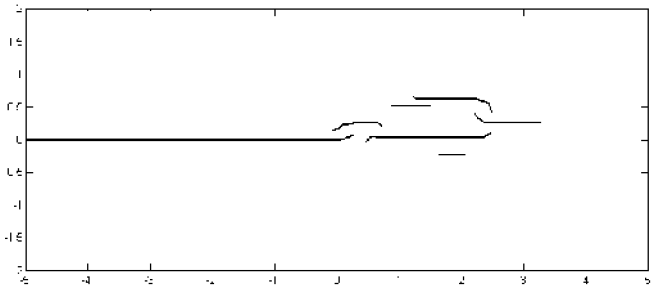


Figure 5.8. Main crack and micro-cracks interacting and propagating with changing direction.

- As a result of the main crack's propagation and interaction with the system of developing micro-cracks, they tend to coalescence with each other to create one macro-flaw.

6 Conclusions

The physical models and numerical methods for investigating individual and interacting randomly dispersed cracks were elaborated and developed. We evaluated their danger level, and the velocity and the direction of a main propagating in a system of interacting cracks. Phenomena of dynamic crack interactions were investigated. These in particular included:

A special, the multiple crack weight (MCW), method for constructing weight functions for multi-crack problems was created. This defines the weight functions around cracks in such a way that they can simultaneously characterize all the crack tips present in the nodal domain of influence. Thus it allows the solution of strongly interacting static and dynamic cracks without enormous mesh refinement and reduces significantly the computational efforts.

The results of the calculated numerical examples show that the new method enables an accurate and reliable solution for static and dynamic strongly interacting cracks problems with relatively small nodal densities. There was less than 0.5% difference between the calculated and at least one of the three corresponding reference values of normalized stress intensity factors F_I^A and F_I^B in all cases, except one for F_I^A and one for F_I^B.

A number of numerical examples of individual interacting cracks undergoing pulse load were solved by the EFG method using different time evolution approximation techniques. The results of the calculations were analyzed. Satisfying accuracy of the solution of numerical examples was achieved compared with analytical solution and the results of other numerical methods.

The problem of main crack propagation in a field of interacting micro-cracks was solved. The analysis has shown that the character of the interaction depends strongly on the position of the applied pulse load, the direction of the stress wave propagation and on the position of the cracks, and their orientation and distance from each other within the system. The interaction of cracks can result in an increase of energy release rate for some of them and lead to their future propagation. Main crack may start to propagate earlier if interacting with other flaws. The interaction of main crack with a system of micro-cracks changes the velocity of its propagation. The process of crack development under pulse loading and crack deceleration can take longer due to the stress amplification effect. Main crack may elongate for longer lengths, when it interacts with other cracks.

References

1. Parton, V.Z. and Boriskovsky, V.G., "Dynamic Fracture Mechanics", Moscow, Mashinostroenie, 1985.
2. Nakamura, T., Shih, C.R. and Freund, L.B., "Computational Methods Based on an Energy Integral in Dynamic Fracture", *International Journal of Fracture,* Vol. 27, 1985, pp. 229–243.
3. Lu, Y.Y., Belytschko, T. and Tabbara, M., "Element-Free Galerkin Methods for Wave Propagation and Dynamic Fracture", *Computer Methods in Applied Mechanics and Engineering,* Vol. 126, 1995, pp. 131–153.
4. Organ, D.J., "Numerical Solutions to Dynamic Fracture Problems Using the Element-Free Galerkin Method" *Ph.D. Thesis,* Northwestern University, 1996.
5. Chen, H., Gerlach, C. and Belytschko, T., "Dynamic Crack Growth with X-FEM" *Presentation,* Dear Born, MI, August 3, 2001.
6. Muravin, B., Turkel, E., "Advance Diffraction Method as a Tool for Solution of Complex Non-Convex Boundary Problems. Implementation and Practical Applications". *Lecture Notes in Computational Science and Engeering: Meshfree Methods for Partial Differential Equations*, Vol. 26, Springer, 2002, pp. 307–317.
7. Organ, D.J., Fleming M.A., and Belytschko T., "Continuous Meshless Approximations for Nonconvex Bodies by Diffraction and Transparency." *Computational Mechanics,* Vol. 18, 1996, pp. 225–235.
8. Belytschko, T., Lu Y.Y., and Gu L., "Element-Free Galerkin Methods." *International Journal for Numerical Methods in Engineering,* Vol. 37, 1994, pp. 229–256.
9. Fleming, M., Chu Y., Moran B., and Belytschko T., "Enriched element free Galerkin methods for crack tip fields". *International Journal for Numerical Methods in Engineering*, Vol. 40, 1997, pp. 1483–1504.
10. Daux, C., Moes, N., Dolbow, J., Sukumar, N. and Belytschko, T., "Arbitrary branched and intersecting cracks with the extended finite element method." *International Journal for Numerical Methods in Engineering*, Vol. 48, 2000, pp. 1741–1760.
11. Cheung, Y., Wang, Y. and Woo, C., "A general method for multiple crack problems in a finite plate." *Computational Mechanics*, Vol. 10, 1992, pp.335–343.
12. Cherepanov, G.P., "Mechanics of Brittle Fracture" /in Russian/, Moscow, Nauka, 1974, 640p.
13. Freund, L.B., "Dynamic Fracture Mechanics", Cambridge University Press, Cambridge, 1990, 581p.
14. Moran, B. and Shih, C.F., "Crack Tip and Associated Domain Integrals from Momentum and Energy Balance", *Engineering Fracture Mechanics,* Vol. 27, No. 6, 1987, pp. 615–641.
15. Organ, D.J., "Numerical Solutions to Dynamic Fracture Problems Using the Element-Free Galerkin Method" *Ph.D. Thesis,* Northwestern University, 1996.
16. Chen, H., Gerlach, C. and Belytschko, T., "Dynamic Crack Growth with X-FEM" *Presentation,* Dear Born, MI, August 3, 2001.

Finite Cover Method for Physically and Geometrically Nonlinear Problems

Kenjiro Terada[1⋆] and Mitsuteru Asai[2⋆⋆]

[1] Department of Civil Engineering, Tohoku University, Aramaki 06, Aoba-ku, Sendai 980-8579, Japan.
[2] Department of Mechanical Engineering, The Ohio State University, 206 West 18th Avenue, Columbus, Ohio 43210-1107, USA.

1 Introduction

The generalization or extention of the standard finite element methods (FEM) has been receiving increasing attention in the field of computational mechanics. The start of this trend is probably made by Babuška and Melenk [MB96, BM97] (for PU-FEM), and Durate and Oden [DO96a, DO96b] (for hp cloud method), in which the implication of the partition of unity (PU) is manifest in the framework of meshless methods. It is then recognized that the PU is the most fundamental theoretical basis for generalizing the FEM. In this context, the overwhelming success has been achieved in the extended FEM (X-FEM) by Belytschko and his co-workers [MDB99, DMB00], which enables us to deal with arbitrary discontinuities in FE analyses by the introduction of discontinuous enrichment functions associated with nodes. Another important development is the generalized FEM (GFEM) introduced by Babuška and his co-workers [SBC00, SCB01], which provides a solid mathematical basis in generalizing the p-adaptive FEM.

With all similar developments, the earliest work that realizes the generalization of the FEM based on the PU property was done by G.H. Shi to propose the manifold method (MM) [Shi91]. The MM, which combines the PU property with differentiable manifolds, was originally developed as an analysis method that integrates the method of discontinuous deformation analysis (DDA) [SG89] into the standard FE technologies, and has recently been introduced in the area of computational mechanics; see, e.g., Chen et al. [COIB98]. To be more adequate, the name change of the MM was made by Ohtsubo et al. [OSTN97] and Terada et al. [TAY03] as the finite cover method (FCM) and by Lin [Lin03] as the mesh-based partition of unity method. It is noted

⋆ tei@civil.tohoku.ac.jp
⋆⋆ asai.7@osu.edu

that the meshfree nature is rather emphasized in these developments than the distinction with the DDA in the MM.

In this study, we introduce the FCM as one of generalized analysis methods or a sort of mesh-free methods, and try to enhance its functional capability to analyze physically and geometrically nonlinear problems. Since the approximation concept and techniques of the FCM are almost the same as those of the FEM, the material nonlinearities and the large deformation kinematics are easily incorporated into the FCM. In this study, we focus our attention to the problem of evolving discontinuities with small strains for the former nonlinearity. On the other hand, for the latter, the standard weak formulation is simply provided in the Lagrangian frame for introducing finite deformation kinematics to the finite cover approximation. Here, a representative numerical example for a hyperelastic body demonstrates the limitation by the Lagrangian approach. Thus, we finally propose a new computational method that enables us to analyze the quasi-static equilibrium problem of a finitely deforming body with a spatially fixed mathematical mesh — we have coined the pseudo-Eulerian FCM for it. It is to be noted that no studies have ever tried so far to extend the generalized versions of the FEM for finite deformation problems.

2 Finite Cover Method

The formulation of the finite cover method (FCM) is presented with reference to the original development of the manifold method (MM) by Shi[Shi91] and ours[TAY03]. After introducing the definitions of the components of the FCM, we explain the device to approximate the solution of boundary value problems (BVP) in comparison with the standard FEM.

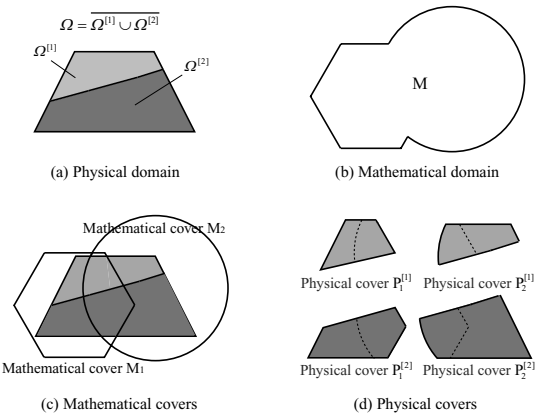


Figure 2.1. Definition of mathematical and physical covers.

2.1 FCM Components

Let us suppose that the physical domain Ω composed of subdmains $\Omega^{[1]}$ and $\Omega^{[2]}$ as shown in Fig. 2.1(a), each of which has its own physical quantities. On the other hand, mathematical properties such as differentiability are introduced to the mathematical domain denoted by M, which is independent of the physical domain. We assume that this mathematical domain can be constructed as a union of a finite number of patches M_I, which can overlap either partially or totally, as

$$\mathrm{M} = \bigcup_{I=1}^{N^{\mathrm{M}}} \mathrm{M}_I, \tag{2.1}$$

where N^{M} is the number of patches. We call these patches *mathematical covers* and assume they have their own mathematical properties. For instance, in Fig. 2.1(b), the mathematical domain composed of two mathematical covers, M_1 and M_2, covers the physical domain Ω. As can be illustrated here, the mathematical domain M, or equivalently the union of mathematical covers, needs not coincide with the physical domain even in the description of the BVP, provided that the former domain completely covers the latter; see Fig. 2.1(c). Also, the common region of mathematical cover M_I and physical domain $\Omega^{[\alpha]}$ is called *physical cover*, which is denoted by $\mathrm{P}_I^{[\alpha]}$ as in Fig. 2.1(d), and accommodates the functions representing physical quantities.

Although these definitions of finite covers may be enough to explain the *differentiable manifolds* in the framework of differential geometry, appropriate mathematical properties must be introduced to approximate the solution of the BVP. If we consider the approximation of physical quantities such as displacement fields in the context of structural or continuum mechanics, certain conditions have to be enforced upon the covers. In this respect, we introduce a weight function to each mathematical cover and spatially arrange all of them so that the union of the weight functions provides a partition of unity (PU) for approximation. The weight function is generally defined as follow:

$$\begin{cases} w_I(\boldsymbol{x}) \geq 0 & \text{for } \boldsymbol{x} \in \mathrm{M}_I \\ w_I(\boldsymbol{x}) = 0 & \text{for } \boldsymbol{x} \notin \mathrm{M}_I, \end{cases} \tag{2.2}$$

along with the PU

$$\sum_{I=1}^{N^{\mathrm{M}}} w_I(\boldsymbol{x}) = 1 \quad \text{on M}, \tag{2.3}$$

where N^{M} is the number of mathematical covers used. Then, we call the common part of mathematical covers the *mathematical element*. Besides, a union of all the mathematical elements is called *mathematical mesh*. Figure 2.2 would help one to understand the covering concept of approximation in

the two-dimensional setting, in which the mathematical element is expressed as

$$\Omega_e = \bigcup_{\alpha=I,J,K,L} \mathrm{M}_\alpha, \tag{2.4}$$

and the *physical element* is defined as the common part of physical covers. In this particular example, the set of weight functions can be bilinear and is illustrated in Fig. 2.3.

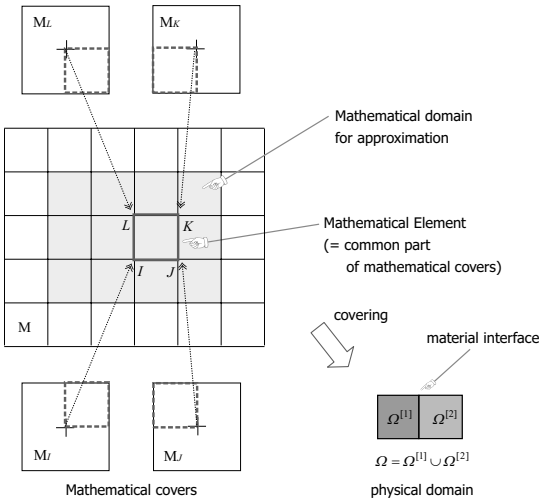


Figure 2.2. Basic components in 2D FCM.

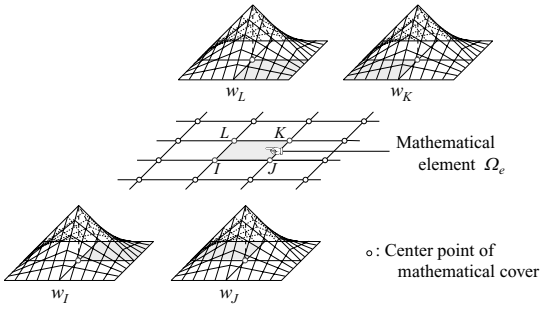


Figure 2.3. Example of bilinear weight functions.

Given the physical problem and the mathematical mesh in Fig. 2.2, we can take any way of covering. Let us consider two typical covering patterns

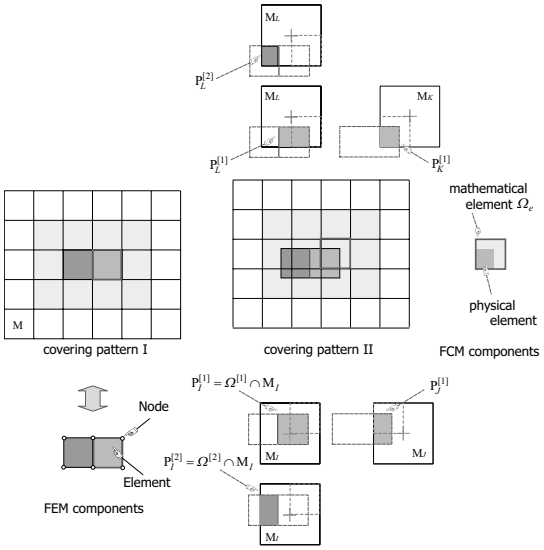


Figure 2.4. Covering patterns and generalized elements.

illustrated in Fig. 2.4; Pattern I and II. Covering Pattern I is performed so that (a part of) the mathematical mesh coincides with the physical domain, or equivalently, so that the mathematical elements and the physical elements are identical. This leads to the discretization of the standard FEM. On the other hand, in Pattern II, the mesh covers the physical domain, but the mathematical elements need not be compatible with the physical domain. In fact, the second type of covering, i.e., Pattern II, characterizes the FCM as a generalized version of the FEM or a kind of meshfree methods.

Keeping in mind the definition of physical covers, we recognize the mathematical element that covers a physical domain in part and is highlighted in Fig. 2.4 for Pattern II. We call this mathematical element the *generalized element* in this paper. It is noteworthy that a physical element is defined as an overlapped region of physical covers, each of which is the common part of a mathematical cover and a physical domain. Therefore, a physical element inherits the mathematical properties from the FCM mesh, so does a generalized element in the light of the fictitious domain concept [GPP94] or the idea of immersed interface[Li98].

The generalized elements are important components of the FCM to represent arbitrary discontinuities in numerical analyses. When we are in the meshfree framework in one hand, a spatially fixed mesh is used for arbitrary physical boundaries including external boundaries and material interfaces. On the other hand, when concerned with the localization phenomena due to material softening or cracking, we are able to represent weak or strong discontinuities within elements. The generalized elements encountered in such situations can

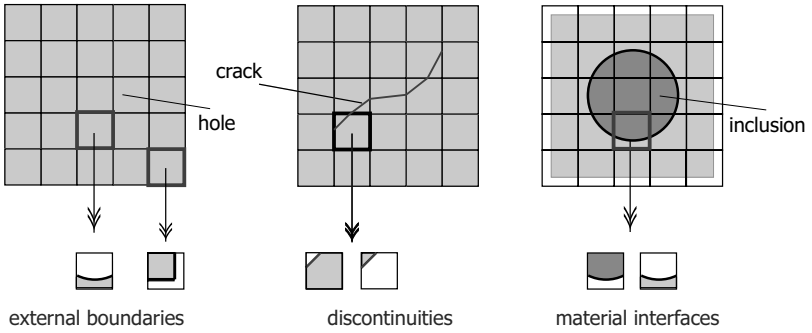


Figure 2.5. Variety of generalized elements in FCM meshes.

be schematized as in Fig. 2.5. The performance of these non-standard elements has recently been examined in Terada and Kurumatani[TK03].

2.2 Finite Cover Approximation

In the original formulation of the MM by Shi [Shi91], the displacement function is approximated by cover functions, which are defined on physical covers, multiplied by weight functions defined on mathematical covers. That is, we employ the following approximation:

$$\boldsymbol{u}(\boldsymbol{x}) \approx \boldsymbol{u}^h(\boldsymbol{x}) = \sum_{I=1}^{N^{\mathrm{M}}} w_I(\boldsymbol{x}) \boldsymbol{f}_I^{[\alpha]}(\boldsymbol{x}) \quad \text{on} \quad \Omega^{[\alpha]}, \tag{2.5}$$

where N^{M} is the number of mathematical covers, $w_I(\boldsymbol{x})$ is the weight function on the mathematical cover I, and $\boldsymbol{f}_I^{[\alpha]}(\boldsymbol{x})$ is the cover function defined only on physical cover $\mathrm{P}_I^{[\alpha]}$. Here, each cover function can be chosen arbitrarily and is formally given as

$$\boldsymbol{f}_I^{[\alpha]}(\boldsymbol{x}) = \sum_{i=0}^{m} p_i(\boldsymbol{x}) \boldsymbol{a}_{iI}^{[\alpha]} \quad \text{on} \quad \mathrm{P}_I^{[\alpha]}. \tag{2.6}$$

Here, $p_i(\boldsymbol{x})$ are monomials to form a polynomial basis with $p_0 = 1$, and $\boldsymbol{a}_{iI}^{[\alpha]}$ are constant coefficient vectors. The complete set of polynomials is a possible choice of $p_i(\boldsymbol{x})$, but is not eligible unless the monomials that are dependent on the weight functions are excluded; see, e.g., Durate and Oden[DO96b]. Substituting (2.6) into (2.5) along with $\boldsymbol{d}_I^{[\alpha]} = \boldsymbol{a}_{0I}^{[\alpha]}$, we arrive at the following expression:

$$\boldsymbol{u}(\boldsymbol{x}) \approx \boldsymbol{u}^h(\boldsymbol{x}) = \sum_{I=1}^{N^{\mathrm{M}}} w_I(\boldsymbol{x}) \left(\boldsymbol{d}_I^{[\alpha]} + \sum_{i=1}^{m} p_i(\boldsymbol{x}) \boldsymbol{a}_{iI}^{[\alpha]} \right) \quad \text{on} \quad \Omega^{[\alpha]}, \tag{2.7}$$

in which, of course, cover functions $p_i(\boldsymbol{x})$ can be chosen as analytical or handbook functions as in the GFEM and as jump or singular functions as in the X-FEM. In this study, we employ the lowest order of weight functions of C^0 partition of unity for the sake of simplicity; that is, we neglect the higher-order cover functions $p_i(\boldsymbol{x})$ $(i \geq 1)$.

What has to be noticed is the way to represent the discontinuities in the FCM, which is somewhat different from the other generalized versions of the standard FEM. In fact, as was seen in the definitions of the FCM components, we distinguish separate physical domains by defining separate physical covers in approximating functions. In other words, a mathematical cover can accommodate multiple physical covers, over which distinct cover functions are defined. That is why we have specified the physical domain $\Omega^{[\alpha]}$ in the above expression for approximation. The representation of discontinuities by means of multiple covers affords the scope of this paper and can also be found in Terada et al. [TAY03].

3 Application to Evolution Problems of Failure Surfaces

The FCM enables us to locate any type of discontinuities in arbitrarily generated mathematical mesh. To demonstrate such a capability of the FCM, we here perform the analysis of an evolving failure surface employing the classical failure condition. Since the use of local values of stress will cause the so-called mesh-dependence, we employ the non-local strain/strain to judge the failure.

3.1 Governing Equations

Let us consider the quasi-static equilibrium problem of the physical domain Ω. which might be composed of two separate physical subdomains, $\Omega^{[1]}$ and $\Omega^{[2]}$. The variational formulation of the displacement method for the problem provides the following weak form of the governing equations to be solved for the displacement field $\boldsymbol{u} \in \mathcal{U}$:

$$\int_{\Omega} (\nabla \delta \boldsymbol{u}) : \boldsymbol{\sigma} \, dv = \int_{\Omega} \delta \boldsymbol{u} \cdot \bar{\boldsymbol{b}} \, dv + \int_{\Gamma_\sigma} \delta \boldsymbol{u} \cdot \bar{\boldsymbol{t}} \, ds \quad \forall \delta \boldsymbol{u} \in \mathcal{V}, \tag{3.1}$$

along with an appropriate constitutive equation that relates the Cauchy stress $\boldsymbol{\sigma}$ and the infinitesimal strain $\boldsymbol{\epsilon}$, and the strain-displacement relationship, $\boldsymbol{\epsilon} = \nabla^{(\mathrm{S})} \boldsymbol{u}$. Here, ∇ is the gradient operator, $\nabla^{(\mathrm{S})}$ is that makes its symmetric part, $\bar{\boldsymbol{t}}$ is the prescribed traction vector on Γ_σ and $\bar{\boldsymbol{b}}$ is the body force. Also, the spaces of trial and test functions are respectively given as

$$\mathcal{U} = \left\{ \boldsymbol{u} \;\middle|\; u_i \in H^1(\Omega),\ \boldsymbol{u} = \bar{\boldsymbol{u}} \text{ on } \Gamma_u \right\}, \tag{3.2}$$

$$\mathcal{V} = \left\{ \delta \boldsymbol{u} \;\middle|\; \delta u_i \in H^1(\Omega),\ \delta \boldsymbol{u} = \boldsymbol{0} \text{ on } \Gamma_u \right\}, \tag{3.3}$$

where $H^1(\Omega)$ is the Sobolev space of first order. Note here that weak form (3.1) implicitly involves the interface compatibility condition $\boldsymbol{u}^{[1]} = \boldsymbol{u}^{[2]}$ on $\Gamma^{[1-2]}$, where $\boldsymbol{u}^{[\alpha]}$ is the displacement, and superscript $[\alpha]$ indicates the association with physical domain $\Omega^{[\alpha]}$.

As was seen in the previous section, the constraints on physical boundaries are not necessarily imposed on the "nodal" degrees of freedom (DOF) in the FCM. Therefore, the compatibility of the displacement on the essential boundary Γ_u and internal boundaries $\Gamma^{[1-2]}$ must be satisfied in a weak sense by means of the method of Lagrange multipliers. In this context, the hybrid variational forms for the set of solutions $(\boldsymbol{u}, \boldsymbol{\lambda}) \in \mathcal{H} \times \mathcal{L}$ are given as

$$\int_{\Omega} \nabla \delta \boldsymbol{u} : \boldsymbol{\sigma} \, dv + \int_{\Gamma^{[1-2]}} \delta \boldsymbol{u}^{[1]} \cdot \boldsymbol{\lambda} \, ds - \int_{\Gamma^{[1-2]}} \delta \boldsymbol{u}^{[2]} \cdot \boldsymbol{\lambda} \, ds + \int_{\Gamma_u} \delta \boldsymbol{u} \cdot \boldsymbol{\lambda} \, ds$$

$$= \int_{\Omega} \delta \boldsymbol{u} \cdot \bar{\boldsymbol{b}} \, dv + \int_{\Gamma_\sigma} \delta \boldsymbol{u} \cdot \bar{\boldsymbol{t}} \, ds \quad \forall \delta \boldsymbol{u} \in \mathcal{H}, \tag{3.4}$$

$$\int_{\Gamma^{[1-2]}} \delta \boldsymbol{\lambda} \cdot \left(\boldsymbol{u}^{[1]} - \boldsymbol{u}^{[2]} \right) \, ds + \int_{\Gamma_u} \delta \boldsymbol{\lambda} \cdot (\boldsymbol{u} - \bar{\boldsymbol{u}}) \, ds = 0 \quad \forall \delta \boldsymbol{\lambda} \in \mathcal{L}, \tag{3.5}$$

where we assume $\mathcal{H} = \{H^1(\Omega)\}^{n_{\mathrm{dim}}}$ and $\mathcal{L} = \{L_2(\Gamma_u \text{ or } \Gamma^{[1-2]})\}^{n_{\mathrm{dim}}-1}$.

The formulation is completed by the introduction of the specific constitutive laws for the material. In this section, we consider only a linearly elastic material governed by the constitutive law $\boldsymbol{\sigma} = \boldsymbol{C} : \boldsymbol{\epsilon}$ where $\boldsymbol{C}$ is the symmetric elasticity moduli tensor. The material is assumed to be isotropic, but might be subject to failure in this particular modeling.

The formulation in the above is the most general. However, we do not consider the problems that necessitate the Lagrange multipliers in the numerical examples. The illustrative examples that extensively utilize the multipliers to impose the essential boundary and material compatibility conditions are found in Terada et al.[TAY03].

3.2 Failure Conditions with Nonlocal Strain/Stress

Failure Condition. Various types of failure conditions have been proposed based on different hypothesis depending on the kinds of materials. For instance, the values of path-independent integrals such as J-integral are used to identify the crack initiation in the framework of classical fracture mechanics. This approach seem to be very effective for determining the state of initiation and the evolving length and direction for existing cracks. In contrast, most of the conventional ways to deal with the successive process of crack generation and evolution seem to place their theoretical basis on the theories of material instabilities [Ric76, deB01].

Since, in this study, our intention is to demonstrate the capability the FCM, we simply employ the latter condition; in particular, we utilize the

Rankine type failure condition based on the values of positive principal stress. Then, we set the following condition for the generation of discontinuities by

$$F = \sigma_{\max} - \sigma_{\mathrm{cr}} = 0, \tag{3.6}$$

where $\sigma_{\max}$ is the positive maximum principle stress that can be evaluated from the Cauchy stress $\boldsymbol{\sigma}$ and σ_{cr} is the critical value of stress for generating a discontinuity. Also, the orientation of the discontinuity, θ_{cr}, is determined by the direction perpendicular to the major principal direction $\theta_{\max}$ as $\theta_{\mathrm{cr}} = \theta_{\max} + \pi/2$.

Introduction of Nonlocal Strain/Stress. As was mentioned in the previous section, the approximation properties of the FCM are almost the same as those of the FEM. Inevitably therefore, the use of the failure criterion based on the local values of stress causes the mesh-dependence problem because of its non-smooth distribution. Within the framework of finite element methods along with softening constitutive models, the nonlocal strain/stress is often employed as a localization limiter; see, e.g., [BP88, BL89]. In this study, the effect of nonlocality is expected only on the judgment of failure, but not really on the material softening, since the discontinuities are explicitly represented in the FCM.

In this study, integral averaging is adopted to achieve the nonlocal effect. The theory states that the material behavior can be evaluated by the nonlocal physical quantities that are obtained by performing weighted averaging of the corresponding local ones distributed around the evaluation point. For example, the nonlocal stress $\overline{\boldsymbol{\sigma}}(\boldsymbol{x})$ can be evaluated by means of the local one $\boldsymbol{\sigma}(\boldsymbol{x})$ as

$$\overline{\boldsymbol{\sigma}}(\boldsymbol{x}) := \int_{\Omega_R} \alpha\left(\boldsymbol{x}, \boldsymbol{\xi}\right) \boldsymbol{\sigma}\left(\boldsymbol{\xi}\right) d\boldsymbol{\xi}, \tag{3.7}$$

where Ω_R is the nonlocal domain of integration and $\alpha\left(\boldsymbol{x}, \boldsymbol{\xi}\right)$ is the nonlocal weighting function given by

$$\alpha\left(\boldsymbol{x}, \boldsymbol{\xi}\right) = \frac{\alpha_0\left(|\boldsymbol{x} - \boldsymbol{\xi}|\right)}{\int_{\Omega_R} \alpha_0\left(|\boldsymbol{x} - \boldsymbol{\zeta}|\right) d\zeta}. \tag{3.8}$$

In this study, we assume the following bell-shaped function for $\alpha_0(r)$ with the r-coordinate defined by $r = |\boldsymbol{x} - \boldsymbol{\xi}|$:

$$\alpha_0(r) = \begin{cases} \left(1 - \dfrac{r^2}{R^2}\right)^2 & \text{if} \quad 0 \le r \le R, \\ 0 & \text{if} \quad R \le r, \end{cases} \tag{3.9}$$

where R is the radius of nonlocal domain Ω_R for weighted integration.

To incorporate the nonlocal variable with the numerical analyses by the FCM, we simply replace the local Cauchy stress in (3.6) by the nonlocal one as

$$F = \overline{\sigma}_{\max} - \sigma_{\rm cr} = 0, \tag{3.10}$$

where $\overline{\sigma}_{\max}$ is the positive maximum principle stress that can be computed from the nonlocal stress $\overline{\boldsymbol{\sigma}}$. It is to be noted that the weighted averaging plays only a role of regularization of the non-smooth distribution of stress. In other words, the nonlocal stress is used only in the failure criterion to judge the generation and evolution of discontinuities, but not for the equation of equilibrium.

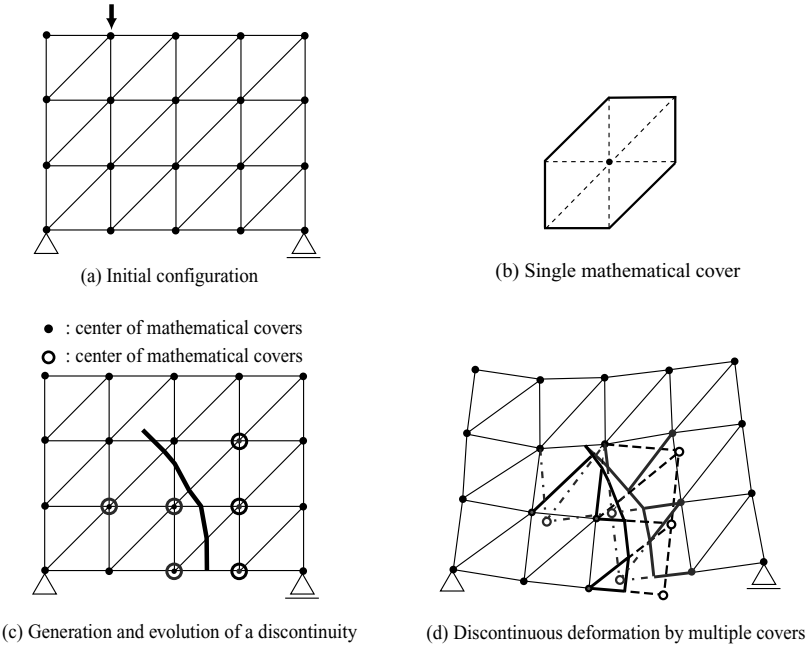


Figure 3.6. Schematic of representing a discontinuity in the FCM.

3.3 Numerical Algorithm for the Evolution of Discontinuities

When an evolving discontinuity intersects a mathematical cover, the associated physical cover that is endowed with cover functions is divided into two covers. This implies that the number of degrees of freedom, which is determined by the cover functions, is increased when the discontinuity evolves. Below, we explain the rule to add such new degrees of freedom by using the example schematized in Fig. 3.6, in which we use the mathematical cover shown in Fig. 3.6(b). Here, the external boundary of the mathematical mesh coincides with the external physical boundary in this particular example, though it is not mandatory in general.

Suppose that the state of a discontinuity is obtained as shown in Fig. 3.6(c) at a certain loading step. As can be seen, some of the mathematical covers cover two separate physical domains. Here, the central point of each mathematical cover, which is marked by ∘ in this figure, is often identified with *node* when constant cover functions are used for approximation. Then, we simply add new cover functions or, equivalently, physical covers according to the definition of finite covers. Also, Fig. 3.6(d) schematizes the representation of the discontinuous motion or deformation with multiple covers. In the actual computer implementation, two sets of *cover layers* are prepared beforehand, each of which is expected to be associated with the separate physical domain. Only the single cover layer is used to analyze the linearly elastic deformation. Then, once the aforementioned failure condition is satisfied, we make the other active to represent the discontinuity.

3.4 Representative Numerical Example

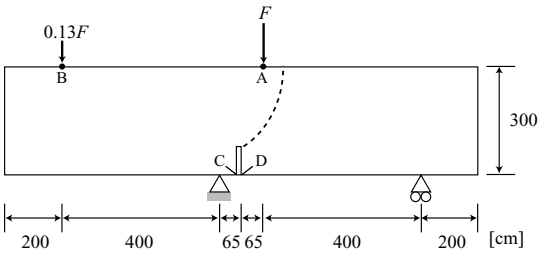


Figure 3.7. Analysis model of four point shear tests and crack path reported in experiments.

We perform the numerical analyses for the evolution of a discontinuity in the four point shear problem by the FCM with the nonlocal failure criterion. The structure under consideration is a beam-like plane solid with a notch at the center of its bottom surface; see Fig. 3.7. This model problem is often utilized to validate specific numerical schemes for this kind of discontinuous deformation [JZ98].

The loading is applied by controlling the vertical displacements at Points A and B in Fig. 3.7 so that the resultant forces at these points keep the same proportion. The experimental facts for concrete beams may be found in Arrera et al. [Arr81], in which the crack initiate at the tip of the notch, and is bended up to the right portion of Point A, which is depicted by a dotted line in Fig. 3.7; see also [JZ98] for the results of numerical analyses.

We prepared three models, Mesh-a, Mesh-b and Mesh-c, of different numbers of mathematical or physical elements, which are actually constant strain triangle finite elements of linear weight functions. The representative element

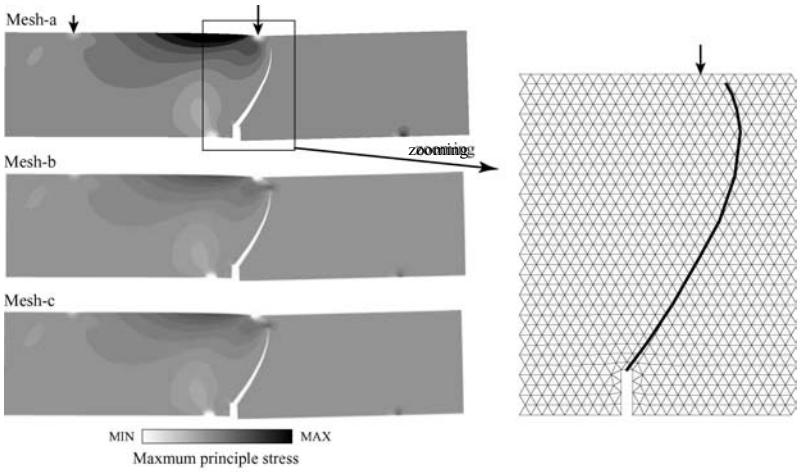
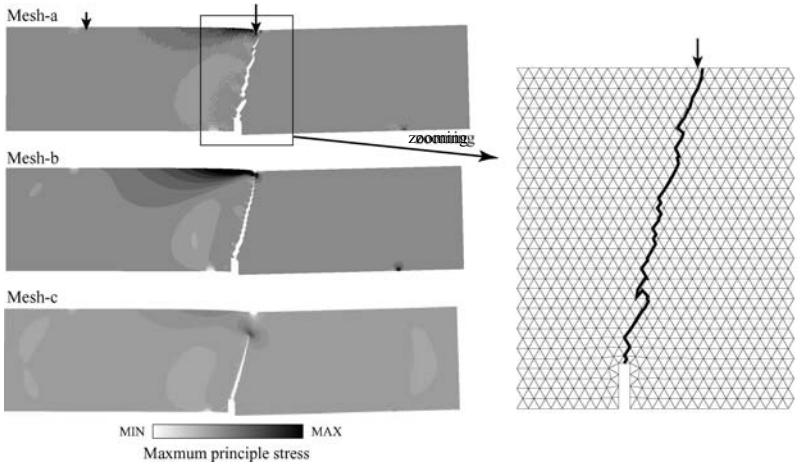


Figure 3.8. Deformed configurations in the nonlocal case in the ultimate state.



(a) Deformed configurations with stress distribution (b) Enlarged illustration of the discontinuity

Figure 3.9. Deformed configurations in the local case in the ultimate state.

sizes of the three models are 10, 5, 2.5 cm and the numbers of elements are 9046, 36580, 146738, respectively. Also, we consider the failure condition with local stress in comparison — we call it the local case for convenience, while the case using the failure condition with nonlocal stress is simply called the nonlocal case. The material parameters used in the numerical analyses are given as follows: Young's modulus $E = 20$(GPa), Poisson's ratio $\nu = 0.3$, the

critical value of the failure condition $\sigma_{cr} = 2(\text{MPa})$. Also, the radius R in (3.9) of nonlocal domain is set by 12.0 cm for all the models, which is comparable to the representative element size of Mesh-a. In general, the size of the radius is chosen so that the local effect in elements is most effectively eliminated, and thus is relevant to neither actual physics nor computational robustness.

Figure 3.8 shows the deformed configurations obtained in the nonlocal cases in the ultimate state. This figure also presents the enlarged illustration of the discontinuity with Mesh-a. As can be seen from the figures, the evolving discontinuities are very similar to that reported in experiments. Also, all the models exhibit almost the same configuration – this means that the numerical solutions are independent of the mesh size. On the other hand, the results obtained in the local case are presented in Fig. 3.9, which shows very different deformed configurations from the experimental one and have great dependence on the mesh size. Thus, the evolution of discontinuities is successfully simulated by the FCM with the simple failure condition by means of nonlocal stress, but not by local stress.

The same observation can be made by depicting the load versus displacement curve, which is shown in Fig. 3.10. Here, the level of loading is the vertical component of the resultant force at Point A and the displacement is a relative one between those of Points C and D in Fig. 3.7, which is often called the crack mouth sliding displacement (CMSD) in the literature.

In conclusion, the FCM enables us to simulate arbitrary discontinuities which evolve independently of the mathematical mesh. It is, however, noted that the incorporation of their mechanical models into boundary value problems is not relevant to their geometrical representation associated with specific analysis methods. In fact, the X-FEM[MDB99] employs almost the same way of geometrical representation for the discontinuities as that of the FCM, but utilizes the analytical functions that represent the crack tip displacements in the mathematical modeling.

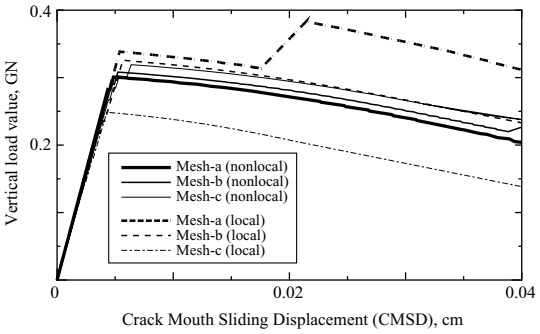


Figure 3.10. Load versus displacement curves obtained by the FCM analyses.

4 Application to Finite Deformation Problems

The FCM is applied to finite deformation problems with the Lagrangian mathematical mesh and extended to that for the Eulerian mesh. First, the limitation of the Lagrangian approach is demonstrated by a representative numerical example, and then propose a new computational method that enables us to analyze the quasi-static equilibrium problem of a finitely deforming body with a spatially fixed, or Eulerian, mathematical mesh.

4.1 Boundary Value Problem and its Linearization

Let $\mathcal{B}_0 \subset \mathcal{R}^{n_{\rm dim}}$ ($n_{\rm dim}$ =1,2, or 3) be the reference configuration of a continuum body with smooth boundary $\partial\mathcal{B}_0$ and closure $\bar{\mathcal{B}}_0 := \mathcal{B}_0 \cup \partial\mathcal{B}_0$. We identify a particle in the body labeled by its position vector $\boldsymbol{X} \in \bar{\mathcal{B}}_0$ relative to the standard basis in $\mathcal{R}^{n_{\rm dim}}$. On the other hand, position $\boldsymbol{x} \in \mathcal{R}^{n_{\rm dim}}$ in the current configuration $\mathcal{B}$ of the body is obtained through the mapping $\boldsymbol{\varphi} : \bar{\mathcal{B}}_0 \to \mathcal{B} \subset \mathcal{R}^{n_{\rm dim}}$, defined as $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X})$ for all $\boldsymbol{X} \in \mathcal{B}_0$, with $J(\boldsymbol{X}) := \det(\boldsymbol{F}(\boldsymbol{X}))$ where we have defined the deformation gradient as

$$\boldsymbol{F}(\boldsymbol{X}) := \nabla_X \boldsymbol{\varphi}(\boldsymbol{X}) = \frac{\partial \boldsymbol{\varphi}(\boldsymbol{X})}{\partial \boldsymbol{X}}. \tag{4.1}$$

It is assumed that the elastic body is subjected to dead loading with body force per unit mass defined by $\boldsymbol{B} : \mathcal{B}_0 \mapsto \mathcal{R}^{n_{\rm dim}}$. We also assume that the deformation is prescribed as $\boldsymbol{u} = \bar{\boldsymbol{u}}$ on $\Gamma_u \subset \partial\mathcal{B}_0$, and that the nominal traction vector is also prescribed as $\bar{\boldsymbol{T}} : \partial\mathcal{B}_0 \mapsto \mathcal{R}^{n_{\rm dim}}$ on Γ_σ. The potential energy function of the external loading takes the form

$$G_{\rm ext}(\boldsymbol{u}) := \int_{\mathcal{B}_0} \rho_0 \boldsymbol{B}.\boldsymbol{u}\, dV + \int_{\Gamma_\sigma} \bar{\boldsymbol{T}}.\boldsymbol{u}\, d\Gamma, \tag{4.2}$$

where ρ_0 is the density in the reference configuration.

In order to have the weak form of the equilibrium equation, we introduce the function space $\mathcal{U}_{\mathcal{B}_0}$ of the solution $\boldsymbol{u}$, and $\mathcal{V}_{\mathcal{B}_0}$ for its variations $\boldsymbol{\eta}$. They are respectively defined as

$$\mathcal{U}_{\mathcal{B}_0} := \left\{ \boldsymbol{u} : \mathcal{B}_0 \mapsto \mathcal{R}^{n_{\rm dim}} \;\middle|\; u_i \in W^{1,p}(\mathcal{B}_0),\ \boldsymbol{u} = \bar{\boldsymbol{u}} \ \text{ on } \ \Gamma_u \right\}, \tag{4.3}$$

$$\mathcal{V}_{\mathcal{B}_0} := \left\{ \boldsymbol{\eta} : \mathcal{B}_0 \mapsto \mathcal{R}^{n_{\rm dim}} \;\middle|\; \eta_i \in W^{1,p}(\mathcal{B}_0),\ \boldsymbol{\eta} = \boldsymbol{0} \text{ on } \Gamma_u \right\}, \tag{4.4}$$

where $W^{1,p}(\mathcal{B}_0)$ is the Sobolev space of functions whose first derivatives belong to the space of p-th Lebesgue integrable functions, $L_p(\mathcal{B}_0)$ ($p \geq 2$). Then, the equilibrium equation in the weak form for solution $\boldsymbol{u} \in \mathcal{U}_{\mathcal{B}_0}$ is given as follows:

$$G_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta}) := \int_{\mathcal{B}_0} \boldsymbol{P} : \nabla_X \boldsymbol{\eta}\, dV - G_{\rm ext}(\boldsymbol{\eta}) = 0, \quad \forall \boldsymbol{\eta} \in \mathcal{V}_{\mathcal{B}_0}, \tag{4.5}$$

along with the local equations in $\mathcal{B}_0$ as

$$\boldsymbol{F}(\boldsymbol{X}) = \nabla_X \boldsymbol{u}(\boldsymbol{X}) + \mathbf{1}, \quad \boldsymbol{P}(\boldsymbol{X}) = \boldsymbol{F}.\boldsymbol{S} = \frac{\partial \mathcal{W}\left(\boldsymbol{X}, \boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\right)}{\partial \boldsymbol{F}}. \tag{4.6}$$

Here, we assume the hyperelastic material behavior and $\mathcal{W}$ is the associated energy function. We also define the second Piola-Kirchhoff stress as $\boldsymbol{S} := 2\partial\mathcal{W}/\partial\boldsymbol{C}$ where $\boldsymbol{C} = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}$ is the right Cauchy-Green strain tensor.

The above nonlinear governing equation is usually solved by the Newton-Raphson method for the incremental displacement $\Delta\boldsymbol{u}$, or the *corrector* more strictly speaking. In this context, the equilibrium equation is linearized to the direction of $\Delta\boldsymbol{u}$ as

$$G_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta}, \Delta\boldsymbol{u}) := DG_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta})[\Delta\boldsymbol{u}] + G_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta}) = 0, \quad \forall\boldsymbol{\eta} \in \mathcal{V}_{\mathcal{B}_0}. \tag{4.7}$$

Here, $DG_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta})[\Delta\boldsymbol{u}]$ denotes the directional derivative of $G_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta})$ in the direction of $\Delta\boldsymbol{u}$ and is defined by

$$DG_{\mathcal{B}_0}(\boldsymbol{P}; \boldsymbol{\eta})[\Delta\boldsymbol{u}] = \int_{\mathcal{B}_0} \nabla_X \boldsymbol{\eta} : \frac{\partial \mathcal{W}}{\partial \boldsymbol{F}\partial \boldsymbol{F}} : \nabla_X(\Delta\boldsymbol{u})\, dV, \tag{4.8}$$

which yields the tangent stiffness matrix multiplied by the displacement corrector vector. Thus the Newton-Raphson iterative procedure can be constructed.

In the numerical examples presented in the next subsections, we consider the regularized logarithmic law for the compressible hyperelastic material behavior, which is derived by the following stored energy function of Hencky's type[Sim98]:

$$\mathcal{W}(\lambda_1, \lambda_2) = \frac{1}{2}\lambda(\ln\lambda_1 + \ln\lambda_2) + \mu\left[(\ln\lambda_1)^2 + (\ln\lambda_2)^2\right], \tag{4.9}$$

where λ and μ are the Lam*é*'s constants. Here, λ_1 and λ_2 denote the principal stretches, namely, the square roots of the eigenvalues of $\boldsymbol{C}$.

4.2 Lagrangian Approach

First, we present the standard Lagrangian approach for solving the boundary value problem given in the above to motivate ourselves to incorporate the Eulerian scheme with the FCM, whose basic ideas will be provided in the next subsection. Nonetheless, to the best of our knowledge, the generalized versions of the FEM, such as GFEM, X-FEM and FCM, have not been applied to the finite deformation problems even in the Lagrangian framework.

Let us consider the plane structure shown in Fig. 4.11, which may be subject to the so-called snap-through phenomenon, and cover it with the mathematical mesh of square covers of the same size. Since the finite cover approximation can be set in almost the same way as the standard FEM in the Lagrangian framework, we can easily perform the numerical analyses to obtain the load-displacement curve in Fig. 4.12. Here, we have used the following material parameters: Young's modulus E=10 GPa, Poisson's ratio ν=0.3

($\lambda = 3/0.52$ GPa, $\mu = 10/2.6$ GPa as Lamé's constants in (4.9)). For comparison, the results by the standard FEM is provided in the same figure. As can be seen, the numerical analysis by the FCM successfully conducted, but cannot be completed, though the standard FEM provides the relevant solution. The reason for this is provided below.

Figures 4.13(a) and (b) respectively show the deformed configuration along with the von-Mises stress distribution and the deformed mathematical mesh at the loading level marked by the arrow in Fig. 4.12. In particular, the mathematical element indicated by the arrow in Fig. 4.13(b) has a collapse, so the approximation by the cover functions multiplied by the weight functions cannot be achieved. This situation is also found in the FE analyses especially for finite deformation problems. However, the mathematical elements that accommodate a very small physical domain sustain severer deformation than the regular elements in the FEM. This brings element collapse and makes the FCM less robust than the FEM.

Therefore, the FCM must evolve so that the mathematical mesh does not deform even when a body moves. This motivate us to develop a new computational strategy that realizes the Eulerian mesh in the next subsection.

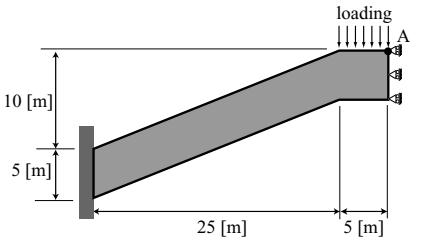


Figure 4.11. Plane structure subject to snap-through deformation.

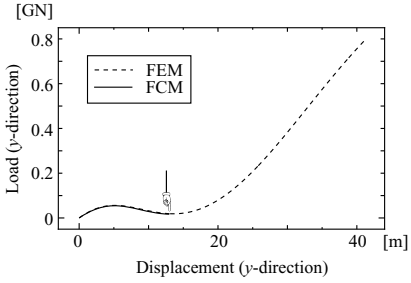
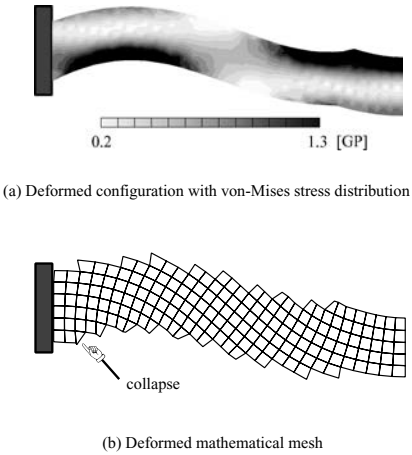


Figure 4.12. Load-displacement curve at Point A.

4.3 Pseudo Eulerian Approach

Introductory Remarks. We shall propose a new analysis method that enables us to follow large deformation with a spatially fixed mathematical mesh. As is well-known, a class of numerical schemes with a spatially fixed mesh is commonly called the Eulerian approach and is usually utilized in the numerical analyses for the motion of fluids. Thus, we here intend to deal with the motion or deformation of solids like that of fluids. This is the first trial in the literature within the framework of generalized finite element methods.

It is noteworthy that there have been several numerical methods, in which a solid body is immersed in a spatially fixed mathematical mesh as if it would



(a) Deformed configuration with von-Mises stress distribution

(b) Deformed mathematical mesh

Figure 4.13. Numerical results by the Lagrangian FCM.

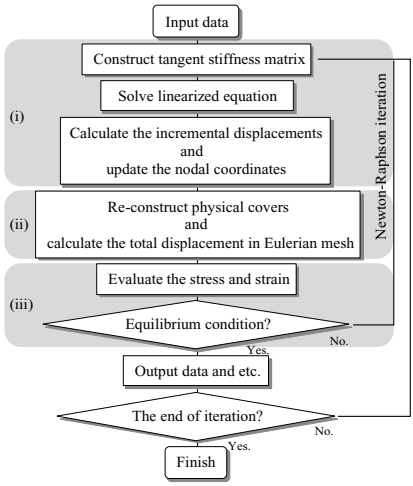


Figure 4.14. Flow chart for the PE-FCM.

be a fluid whose motion is measured by a fixed frame; see, for example, Glowinski et al. [GPP94] and Li [Li98]. On the other hand, the Eulerian FEM by the Benson[Ben92] can deal with large deformation with a spatially fixed mesh, but is classified into the hydrocodes, which do not has the mathematical structure of the generalized finite element techniques.

The messy task to obtain the quasi-static equilibrium of a finitely deforming body with a spatially fixed mathematical mesh is to trace the motion of each material point by the fixed approximation space. This task intrinsically involves two difficulties. One is the treatment of the convected physical quantities at each material point of a solid body when the Eulerian description is employed, and the other is the representation of moving boundaries of the body within a fixed mesh. To avoid the former difficulty, we simply employ the Lagrangian description for the mathematical modeling for a continuum body, but deal with the convective effects with the mathematical mesh that is first attached to the current configuration and then pulled back to the fixed or initial configuration. The latter difficulty is easily overcome in the present context, since the FCM allows the mesh to be located arbitrarily or independently of the actually moving physical domain. In other words, the FCM enables us to immerse a body in a spatially fixed mathematical mesh once the generalized elements that partially have physical domains (especially near the moving physical boundaries) is generated automatically during the analysis.

The methodology with these features is not a pure Eulerian approach, but can be a "pseudo" Eulerian one in the sense that the numerical algorithm presented in Fig. 4.14 realizes the algorithmic treatment of convection and moving boundaries. In this respect, we name this technique the pseudo-Eulerian finite cover method (PE-FCM).

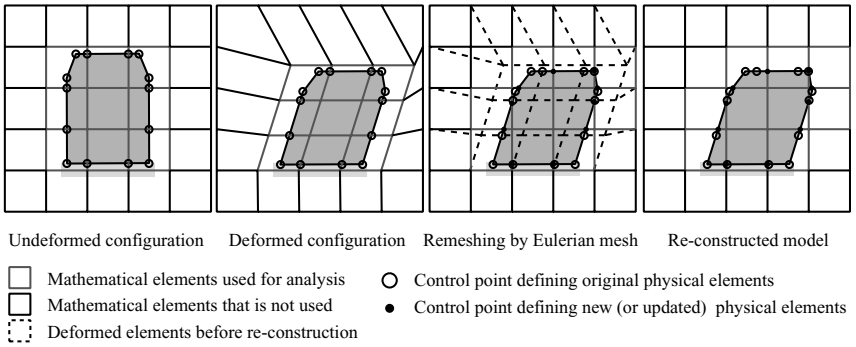


Figure 4.15. Schematic of the updating process for mathematical and physical elements.

Numerical Algorithm. In the numerical algorithm given in Fig. 4.14 for the PE-FCM, Process (i) is exactly the same as the that of the Lagrangian FCM — we solve the linearized equation to obtain the incremental displacement and then update the coordinates of the current positions of *nodes*, which are the central points of spatially fixed mathematical covers. In process (ii), the new physical elements are generated by means of the spatially fixed mathematical mesh and the updated geometry. Then, we evaluate the total displacements at the central points of the fixed mesh by the interpolation in terms of the values of the total displacements obtained in (i). Process (iii) requires us to evaluate the stress and strain, and check the equilibrium of the body. Here, the stress and strain are computed in terms of the total displacements that are interpolated in process (ii), since the integration of the weak form to have the residual force is performed on the fixed mesh.

The numerical algorithm just explained is almost the same as the standard implicit solution scheme for the quasi-static equilibrium of a solid body subject to large deformation. Nonetheless, the interpolation of the total displacements on the fixed Eulerian mesh by those on the deformed Lagrangian mesh corresponds to the physical process of convection. Thus, the PE-FCM can realize the Eulerian approach for finite deformation problems in the algorithmic sense.

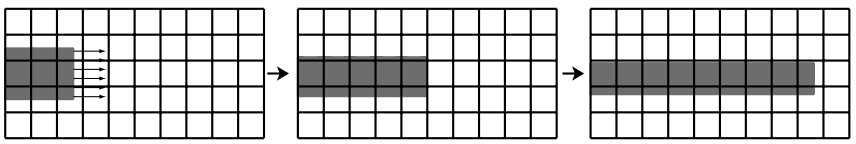


Figure 4.16. Deformation process obtained by the PE-FCM applied to the uniform tension problem.

Representative Numerical Examples.

(1) Uniform tension of a bar-like solid structure

The first example to demonstrate the idea of the PE-FCM is the problem of uniform tension of a bar-like solid structure. The material parameters used here are the same as in the previous subsection. The numerical simulation exactly reproduces the analytical solution that satisfies the following relationship between two stretches λ_1 and λ_2:

$$\frac{\lambda}{\lambda_2}(\ln\lambda_1 + \ln\lambda_2) + 2\frac{\mu}{\lambda_2}\ln\lambda_2 = 0 \tag{4.10}$$

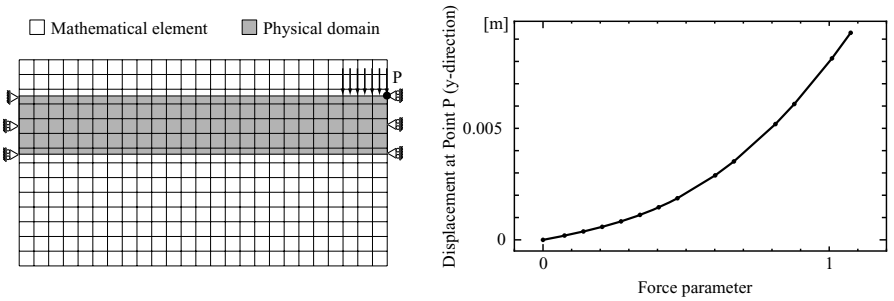
The deformation process is presented in Fig. 4.16, which well illustrates the motion of the body whose boundaries intersects the fixed grid of the mathematical mesh. In this case, the field variables such as stress and strain are constant everywhere in the body, but the displacement field is linear with respect to x_1-direction. Note here that the convection of all the field variables is evaluated based on the total displacement field on the fixed mesh, which is interpolated in terms of those on the deformed mesh. This fact confirms that the algorithm works well.

(2) Bending of a beam-like solid structure

Another numerical example is the problem of bending of a beam-like solid structure as shown in Fig. 4.17(a). The figure also depicts the spatially fixed mathematical mesh and support/loading conditions. The material parameters are the same as before. The numerical analysis by the PE-FCM provides the load-displacement curve shown in Fig. 4.17(b) and the deformation process with the von-Mises stress distribution as shown in Fig. 4.18. Although the transient deformation process is well demonstrated, the verification of the numerical solution is not conducted enough. Actually, we have confirmed in the context of the linear FCM that the numerical solution by the coarse mesh in the FCM is less accurate than that of the FEM, but in some cases the FCM will provides more fast convergence than the FEM[TK03]. Nonetheless, the contribution of this study is manifest in the idea of incorporating the FCM with the pseudo Eulerian approach presented here.

5 Conclusion

The FCM has been introduced as one of generalized analysis methods and extended for physically and geometrically nonlinear problems. For the physical nonlinearity, we were concerned with the evolution of discontinuities in an arbitrarily generated mathematical mesh. We also have tried to extend the FCM to deal with finite deformation problems. After recognizing that the



(a) Beam-like structure with mathematical mesh (b) Load-displacement curve

Figure 4.17. Analysis model and load-displacement curve obtained by the PE-FCM.

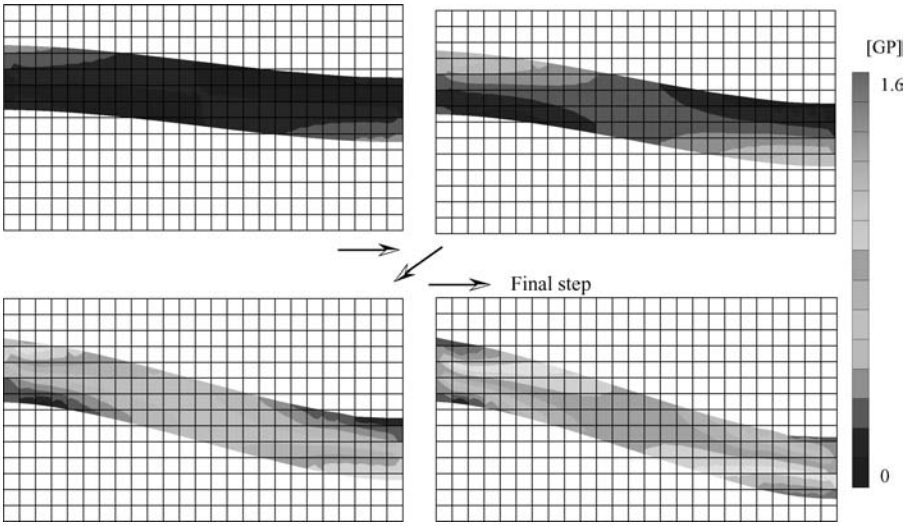


Figure 4.18. Deformed configuration and stress distribution along with the spatially fixed mathematical mesh. See Color Plate 6 on page 297.

Lagrangian approach in the FCM is not appropriate for large deformation problems, we have proposed the new computational method, which we call the pseudo-Eulerian FCM (PE-FCM).

The FCM has almost the same mathematical basis as other generalized versions of the FEM such as the GFEM and the X-FEM in the sense that they are based on the PU with FE mesh. However, the FCM was originally developed as a unified analysis method that comprehends the DDA and its functional capability has been enhanced by the present authors independently of the other PU-based generalizations. In particular, the FCM enjoys the feature that the

physical domain is placed arbitrarily in spatially fixed mathematical mesh. This has been first utilized in this study to combine the Eulerian approach to accommodate finite deformation kinematics in the generalized versions of the FEM.

References

[MB96] Melenk, J.M., Babuška, I.: The partition of unity finite element method: basic theory and applications. Comput. Meth. Appl. Mech. Engrg., **139**, 289–314 (1996)

[BM97] Babuška, I., Melenk, J.M.: The partition of unity method. Int. J. Numer. Meth. Engrg., **40**, 727–758 (1997)

[DO96a] Durate, C.A.M. Oden, J.T: An h-p adaptive method using clouds. Comput. Meth. Appl. Mech. Engrg., **139**, 237–262 (1996)

[DO96b] Durate, C.A.M., J.T. Oden, J.T.: Hp clouds — an hp meshless method. Numer. Meth. Partial Diff. Eqs., **12**, 673–705 (1996)

[MDB99] Moës, N, Dolbow, J, Belytschko, T.: A finite element method for crack growth without remeshing. Int. J. Numer. Meth. Engrg., **46**, 131–150 (1999)

[DMB00] Dolbow, J., Moës, N., Belytschko, T. Discontinuous enrichment in finite elements with a partition of unity method. Finite Elements in Analysis and Design, **36**, 235–260 (2000)

[SBC00] Strouboulis, T., Babuška, I., Copps, K.: The design and analysis of the generalized finite element method. Comput. Meth. Appl. Mech. Engrg., **181**, 43–69 (2000)

[SCB01] Strouboulis, T., Copps, K., Babuška, I.: The generalized finite element method. Comput. Meth. Appl. Mech. Engrg., **190**, 4081–4193 (2001)

[Shi91] Shi, G.H.: Manifold method of material analysis. In: Transactions of the 9^{th} Army Conference On Applied Mathematics and Computing, Report No.92-1, U.S. Army Research Office (1991)

[SG89] Shi, G.H., Goodman, R.E.: Generalization of two-dimensional discontinuous deformation analysis for forward modeling. Int. J. Numer. Anal. Meth. Geomech., **13**, 359–380 (1989)

[COIB98] Chen, G., Ohnishi, Y., Ito, T.: Development of higher-order manifold method. Int. J. Numer. Meth. Engrg., **43**, 685–712 (1998)

[OSTN97] Ohtsubo, H., Suzuki, K., Terada, K., Nakanishi, K.: Utilization of finite covers in the manifold method for accuracy control. In: Ohnishi, Y. (ed) Proc. 2nd Int. Conf. on Analysis of Discontinuous Deformation. Kyoto, 317–322 (1997)

[TAY03] Terada, K., Asai, M., Yamagishi, M.: Finite cover method for linear and nonlinear analyses of heterogeneous solids. Int. J. Numer. Meth. Engrg., **58**, 1321–1346 (2003)

[Lin03] Lin, J-S.: A mesh-based partition of unity method for discontinuity modeling. Comput. Meth. Appl. Mech Engrg., **192**, 1515–1532 (2003)

[GPP94] Glowinski, R., Pan, T., Periaux, J.: A fictitious domain method for Dirichlet problem and applications. Comput. Methods Appl. Mech. Engng., **111**, 283–303 (1994)

[Li98] Li, Z.: The immersed interface method using a finite element formulation. Appl. Numer. Math., **27**, 253–267 (1998)

[TK03] Terada, K., Kurumatani, M.: Performance assessment of generalized elements in the finite cover method. Finite Elements in Analysis and Design, in press.

[Ric76] Rice, J.R.: The localization of plastic deformation. In: Theoretical and Applied Mechanics. North-Holland, Amsterdam, 207–220 (1976)

[deB01] de Borst, R.: Some recent issues in computational failure mechanics. Int. J. Numer. Meth. Engrg., **52**, 63–95 (2001)

[BP88] Bažant, Z.P., Pijaudier-Cabot, G.: Nonlocal continuum damage, localization instability and convergence. Journal of Applied Mechanics, ASME, **55**, 287–293 (1988)

[BL89] Belytschko, T., Lasry, D.: Localization limiters and numerical strategies for strain softening materials. In: Mazars J., Bazent, Z.P. (ed) Cracking and Damage – Strain Localization and Size Effect. Elsevier, 264–294 (1989)

[JZ98] Jirásek, M., Zimmermann, T.: Rotating crack model with transition to scalar damage. Journal of Engineering Mechanics, ASCE, **124**, 277–284 (1998)

[Arr81] Arrea, M., Ingraffea, A.R.: Mixed-mode crack propagation in mortar and concrete. Dept. of Struct. Engrg. Rep. 81-13, Cornell University, Ithaca, New York (1981)

[Sim98] Simo, J.C.: Numerical analysis and simulation of plasticity. In: Handbook of Numerical Analysis, Vol. VI. Elsevier Science (1998)

[Ben92] Benson, D.J.: Computational methods in Lagrangian and Eulerian hydrocodes. Comput. Methods Appl. Mech. Engrg., **99**, 235–394 (1992)

A Numerical Scheme for Solving Incompressible and Low Mach Number Flows by the Finite Pointset Method

Sudarshan Tiwari* and Jörg Kuhnert**

Fraunhofer Institut für Techno- und Wirtschaftmathematik, Gottlieb Daimler Strasse, Geb. 49, D–67663 Kaiserslautern, Germany.

Abstract A meshfree projection method for compressible as well as incompressible flows and the coupling of two phase flows with high density and viscosity ratios is presented. The Navier-Stokes equations are considered as the basic mathematical model and are solved by the implicit projection method. The implicit projection method yields the linear second order partial differential equations. These equations are solved by the weighted least squares method and are compared with the exact solutions. A one dimensional shock tube problem is exhibited for compressible flows. Finally, two phase incompressible and quasi compressible flows are used to simulate a two phase cavity filling problem.

1 Introduction

The Finite Pointset Method (FPM) is a meshfree method to solve partial differential equations. The computational domain is represented by a finite number of particles (pointset), also referred to as numerical points. These points can be arbitrarily distributed, however they have to provide a neighborhood relationship governed by the smoothing length, i.e. each point needs to find sufficiently many neighbor points within a ball of certain radius. Considering the equation of fluid dynamics, the numerical points move with fluid velocity and carry all information which completely describes the flow problem concerned. Of course, this is a fully Lagrangian method being appropriate for flow simulations with complicated as well as rapidly changing geometry [KTU00], involving free surfaces [TK202, TK03] or phase boundaries [HJKT03].

The classical meshfree Lagrangian method to handle problems in fluid dynamics is the Smoothed Particle Hydrodynamics (SPH). SPH was initially developed to study phenomena in astrophysics [GM97, 15]. Later, it was extended to flow cases even on earth [CR99, 16, Mor00, MFZ97]. Unfortunately,

* tiwari@itwm.fhg.de
** kuhnert@itwm.fhg.de

SPH has poor approximation properties, especially of the second order derivatives, required to model the Navier-Stokes equations. Moreover, it is difficult to incorporate boundary conditions of certain types. In SPH, incompressible flows are approximated by using the compressible approach together with a very stiff equation of state.

The FPM is based on least squares approximations, where the higher order derivatives can be approximated very accurately and the boundary conditions can be treated in a classical sense [Ku99]. Several computations of flow problems using the method of least squares or moving least squares are reported by different authors, see [Dil96, Ku99, Ku02, TK01, TK102, TK202, TK03, TM03, Tiw00] and other references therein.

The numerical scheme for incompressible and slightly compressible flow phenomena, presented in this article, is based on the classical projection idea of Chorin [Cho68, TK102]. Due to that, the solutions of Poisson as well as Helmholtz differential equations, in particular, form a central task of FPM. These equations can be solved directly in the given meshfree structure with Dirichlet, Neumann or Cauchy boundary conditions in a very accurate way [TK01]. Also, see section 3.1 and 3.2. Moreover, free surfaces can be incorporated very efficiently [TK202].

For some industrial applications, such as simulations of car tank refueling, several fluid phases like fuel, air and foam might be involved. Not all phases can be assumed to be incompressible, as for instance the air inside of the tank might be compressed during the filling process. This is a rather slow compression, with the Mach number tending to zero. However, the compression plays a big role as it partially governs the filling process. Thus, we would like to incorporate compressibility effects into the classical re-projection idea and finally come up with an implicit scheme for compressible as well as incompressible flows.

Therefore, we are going to present an idea to simulate low Mach number and incompressible flows with exactly the same procedure, i.e. the incompressible case turns out to be a special case of the compressible regime. We consider the Navier-Stokes equations as the mathematical model. We solve these equations by the projection method implicitly. The implicit scheme results in linear second order partial differential equations (Poisson, Helmholtz). We solve them using the constraint least squares method suggested in [TK01], see also section 4.

Most of the methods for solving multi phase flows are based on meshgrid techniques [BKZ92, GW01, HW65, HN81, KP97], where additional computational effort has to be put in order to model the dynamics of interphase boundaries. The advantage of using the particle method is that phases can be distinguished by simply assigning flags to the fluid particles which identify their proper phase. The phase-flags are carried in the same fashion as all other physical data.

Since the particles move with fluid velocity, they may scatter or accumulate together. If they scatter and create holes in the computational domain,

singularities may arise. Hence, holes have to be detected and new particles have to be added. Similarly, any two particles being too close to each other, have to be replaced by a single one.

In this paper we have excluded surface tension effects. The CSF model [BKZ92] can easily be extended by using the approach proposed in [Mor00]. The work is in progress.

We have obtained results from convergence studies for general second order linear partial differential equations. If the coefficients are constant, the scheme has second order convergence. If the coefficients are discontinuous, which occur for solving multi phase flows, the proposed scheme is of first order convergence. The implicit projection method is tested for compressible flows by solving a 1D shock tube problem and the results are compared with the exact solutions. Finally, we present a two phase flow case for cavity filling, where the air is considered to be compressible.

The paper is organized as follows. In section 2, we introduce the mathematical model and the numerical scheme. In section 3, we present the FPM for solving general elliptic partial differential equations. The numerical results are presented in section 4.

2 Governing Equations

We consider two immiscible fluids, for example, liquid and gas. We distinguish the liquid and gas particles by assigning appropriate flags on them. We assume that the viscosity μ and the density ρ jump are discontinuous in the phase boundary. These discontinuities can cause numerical instabilities around the interfaces. To avoid them, in every time step we consider the smoothed densities and viscosities on and around the interface. This means for the discretization of the momentum equations, we consider the smooth density and viscosity and then reassign these values to the original (non smoothed) ones. The interface region can be detected by checking the flags of particles in the neighborhood. We update the smoothed density $\tilde{\rho}$ and the smoothed viscosity $\tilde{\mu}$ in each time step at each particle position $\boldsymbol{x}$ near the interface by using the Shepard interpolation

$$\tilde{\rho}(\boldsymbol{x}) = \frac{\sum_{i=1}^{m} w_i \rho_i}{\sum_{i=1}^{m} w_i}, \qquad \tilde{\mu}(\boldsymbol{x}) = \frac{\sum_{i=1}^{m} w_i \mu_i}{\sum_{i=1}^{m} w_i}, \tag{2.1}$$

where m is the total number of neighbor particles related to $\boldsymbol{x}$ (i.e. all numerical points being within a circle of radius h around $\boldsymbol{x}$, h is called smoothing length). The neighbor particles in the interface region are taken from the liquid as well as from the gas phases. Far from the interface we have $\tilde{\rho} = \rho$ and $\tilde{\mu} = \mu$. We consider a truncated Gaussian as weight function, in general this can be any compactly supported smooth function.

2.1 Navier–Stokes Equations

Let Ω be an open bounded domain in $\mathbb{R}^d$ $(d = 1, 2, 3)$ with boundary Γ. Let $\boldsymbol{v}$, T and p be the velocity, temperature and pressure fields representing the state variables. The compressible Navier-Stokes equations in the Lagrangian form can be written as

$$\frac{D\rho}{Dt} = -\rho\nabla\cdot\boldsymbol{v} \tag{2.2}$$

$$\rho\frac{D\boldsymbol{v}}{Dt} = -\nabla p + \nabla\cdot\sigma(\boldsymbol{v}) + \rho\boldsymbol{g} \tag{2.3}$$

$$\rho c_v\frac{DT}{Dt} = -p\nabla\cdot\boldsymbol{v} + (\sigma\cdot\nabla)\cdot\boldsymbol{v} + \nabla\cdot(\kappa\nabla T). \tag{2.4}$$

Here, κ denotes the heat conduction coefficient, $\boldsymbol{g}$ the body force, c_v the specific heat capacity. By D/Dt we denote the Lagrangian derivative. The stress tensor is

$$\sigma_{ij}(\boldsymbol{v}) = \mu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} - \frac{2}{3}\nabla\cdot\boldsymbol{v}\delta_{ij}\right),$$

where $\delta_{ij} = 0$ is the Kronecker delta. We close the system (2.2-2.4) by the equation of state

$$\rho = \rho(p, T). \tag{2.5}$$

Then, from the continuity equation (2.2) we obtain

$$\nabla\cdot\boldsymbol{v} = -\frac{1}{\rho}\frac{D\rho}{Dt} = -\frac{1}{\rho}(\frac{\partial\rho}{\partial p}\frac{Dp}{Dt} + \frac{\partial\rho}{\partial T}\frac{DT}{Dt}). \tag{2.6}$$

Equation (2.6) is a very important relation which later allows us to derive a projection idea for compressible flow phenomena.

Since the density and the viscosity are smoothed according to (2.1) near the interface, we can rewrite the momentum equations (2.3) whose spatial components are given by

$$\frac{D\boldsymbol{v}}{Dt} = \boldsymbol{g} - \frac{1}{\tilde{\rho}}\left[\nabla p + (\nabla\tilde{\mu}\cdot\nabla)\boldsymbol{v} + \tilde{\mu}\Delta\boldsymbol{v} + \nabla\boldsymbol{v}\cdot\nabla\tilde{\mu} - \frac{2}{3}\nabla\tilde{\mu}(\nabla\cdot\boldsymbol{v}) + \frac{1}{3}\tilde{\mu}\nabla(\nabla\cdot\boldsymbol{v})\right]$$

The above presented equations are to be solved with appropriate initial and boundary conditions which are specified in the section where numerical tests are performed.

3 Numerical Scheme

We consider Chorin's projection method [Cho68] implicitly for both compressible as well as incompressible flows. It consists of two fractional steps. We first compute the new particle positions at time level t^{n+1} by

$$\boldsymbol{x}^{n+1} = \boldsymbol{x}^n + \Delta t \boldsymbol{v}^n. \tag{3.1}$$

Then, for each particle, we compute the smoothed density $\tilde{\rho}^{n+1}$ and viscosity $\tilde{\mu}^{n+1}$ according to (2.1) and then compute the intermediate velocity $\boldsymbol{v}^*$ by

$$\begin{aligned} \boldsymbol{v}^* + \frac{\Delta t}{\tilde{\rho}} \left[(\nabla \tilde{\mu} \cdot \nabla) \boldsymbol{v}^* + \tilde{\mu} \Delta \boldsymbol{v}^* \right] = \\ \boldsymbol{v}^n - \frac{\Delta t}{\tilde{\rho}} \left[\nabla \boldsymbol{v}^n \cdot \nabla \tilde{\mu} - \frac{2}{3} \nabla \tilde{\mu} (\nabla \cdot \boldsymbol{v}^n) + \frac{1}{3} \tilde{\mu} \nabla (\nabla \cdot \boldsymbol{v}^n) \right] + \Delta t \boldsymbol{g}. \end{aligned} \tag{3.2}$$

Here, Δt represents the time step, $\Delta \boldsymbol{v}$ is the Laplacian of $\boldsymbol{v}$, $\tilde{\rho} = \tilde{\rho}^{n+1}$ and $\tilde{\mu} = \tilde{\mu}^{n+1}$.

The second step consists in establishing the new velocity $\boldsymbol{v}^{n+1}$ by correcting the intermediate velocity $\boldsymbol{v}^*$. For this, we need to solve the equation

$$\boldsymbol{v}^{n+1} = \boldsymbol{v}^* - \frac{\Delta t}{\tilde{\rho}} \nabla p^{n+1} \tag{3.3}$$

with the constraints

$$\nabla \cdot \boldsymbol{v}^{n+1} = -\frac{1}{\tilde{\rho}} \frac{D\tilde{\rho}}{Dt} \tag{3.4}$$

with respect to p^{n+1}. By applying the divergence operator to equation (3.3), and using the relation (2.6), we obtain

$$-\frac{1}{\tilde{\rho}} \left(\frac{\partial \tilde{\rho}}{\partial p} \frac{Dp}{Dt} + \frac{\partial \tilde{\rho}}{\partial T} \frac{DT}{Dt} \right) = \nabla \cdot \boldsymbol{v}^* - \Delta t \nabla \cdot \left(\frac{\nabla p^{n+1}}{\tilde{\rho}} \right). \tag{3.5}$$

Now, using $\frac{Dp}{Dt} = \frac{p^{n+1} - p^n}{\Delta t}$, equation (3.5) can be expressed in the form

$$\frac{-1}{\tilde{\rho} \Delta t^2} \frac{\partial \tilde{\rho}}{\partial p} p^{n+1} + \nabla \cdot \left(\frac{\nabla p^{n+1}}{\tilde{\rho}} \right) = \frac{1}{\Delta t} \left(\frac{-1}{\tilde{\rho}} \frac{\partial \tilde{\rho}}{\partial p} p^n + \frac{\partial \tilde{\rho}^n}{\partial T^n} \frac{DT^n}{Dt} + \nabla \cdot \boldsymbol{v}^* \right). \tag{3.6}$$

Using the quotient rule for the second term on the left hand side of (3.6), we obtain

$$\frac{-1}{\Delta t^2} \frac{\partial \tilde{\rho}}{\partial p} p^{n+1} - \frac{1}{\tilde{\rho}} \nabla \tilde{\rho} \cdot \nabla p^{n+1} + \Delta p^{n+1} = \frac{\tilde{\rho}}{\Delta t} \left(\frac{-1}{\tilde{\rho}} \frac{\partial \tilde{\rho}}{\partial p} p^n + \frac{\partial \tilde{\rho}^n}{\partial T^n} \frac{DT}{Dt} + \nabla \cdot \boldsymbol{v}^* \right). \tag{3.7}$$

On the right hand side of (3.7), it is obvious to replace $\frac{DT}{Dt}$ by equation (2.4).

The boundary condition for p^{n+1} is obtained by projecting the equation (3.3) on the unit normal vector $\boldsymbol{n}$ to the boundary Γ. Thus, we obtain the Neumann boundary condition

$$\frac{\partial p^{n+1}}{\partial \boldsymbol{n}} = -\frac{1}{\Delta t} (\boldsymbol{v}_\Gamma^{n+1} - \boldsymbol{v}_\Gamma^*) \cdot \boldsymbol{n}, \tag{3.8}$$

where $\boldsymbol{v}_\Gamma$ is the value of $\boldsymbol{v}$ on Γ. Assuming that $\boldsymbol{v} \cdot \boldsymbol{n} = 0$ on Γ, we obtain

$$\frac{\partial p^{n+1}}{\partial \boldsymbol{n}} = 0 \tag{3.9}$$

on Γ. If no boundary velocity $\boldsymbol{v}_\Gamma$ is known a priori (i.e. if the velocity at the boundary is a result of the computations itself, such as for free surfaces or outflow boundaries), the Dirichlet boundary conditions are appropriate for p^{n+1}.

In the case of incompressible flow, the first term on the left hand side and the first and second terms of the right hand side of equations (3.7) vanish, which results in the classical projection idea of Chorin.

Furthermore, for the temperature we have to solve the following equation

$$T^{n+1} - \frac{\Delta t}{c_v \rho^n} \nabla \cdot \left(\kappa \nabla T^{n+1} \right) = T^n + \frac{\Delta t}{c_v \rho^n} \left[-p^n \nabla \cdot \boldsymbol{v}^n + \left(\sigma(\boldsymbol{v}^n) \cdot \nabla \right) \boldsymbol{v}^n \right]. \tag{3.10}$$

Finally, we update the density for the compressible flow by

$$\rho^{n+1} = \rho(p^{n+1}, T^{n+1}). \tag{3.11}$$

The remaining task is the discretization and solution of the equations (3.2, 3.7, 3.10) on the given (meshfree) point cloud. For this, we establish big linear systems of equations, where the matrix represents the discrete approximation of the differential operators involved, and the right hand side reflects the source terms. In order to establish the mentioned discrete operators with respect to the point cloud, we employ the weighted least squares method, presented in section 3.1.

3.1 Least Squares Method for Approximation of Derivatives

Let $\psi : \Omega \longrightarrow \mathbb{R}$ be a scalar function and ψ_i its discrete values at the particle positions $\boldsymbol{x}_i$ for $i = 1, 2, \ldots, N$. Consider the problem to approximate spatial derivatives of that particular function $\psi(\boldsymbol{x})$ at some particle position $\boldsymbol{x}$ based on the discrete function values of its neighbor points. In order to restrict the number of points we introduce a weight function $w = w(\boldsymbol{x}_i - \boldsymbol{x}; h)$ with small compact support, where h determines the size of the support.

The weight function can be quite arbitrary, however it makes sense to choose a Gaussian weight function of the form

$$w(\boldsymbol{x}_i - \boldsymbol{x}; h) = \begin{cases} exp(-\alpha \frac{\|\boldsymbol{x}_i - \boldsymbol{x}\|^2}{h^2}), & \text{if } \frac{\|\boldsymbol{x}_i - \boldsymbol{x}\|}{h} \leq 1 \\ 0, & \text{else,} \end{cases}$$

where α is a positive constant and is considered to be in the range of 6. So far, in our implementation, we allow user given h as a function in space and time. However, no adaptive choice of h is realized yet. Working with user given h implies that new particles will have to be brought into play as the particle distribution becomes too sparse or, logically, particles will have to be removed from the computation as they become too dense.

Let $P(\boldsymbol{x}, h) = \{\boldsymbol{x}_i : i = 1, 2, \ldots, m\}$ be the set of m neighbor points of $\boldsymbol{x} = (x, y, z)$ in a ball of radius h. We note that the central particle $\boldsymbol{x}$ is one element of the neighbor set $P(\boldsymbol{x}, h)$. For consistency reasons, some obvious restrictions are required, for example, in $3D$ there should be at least 9 particles in addition to the central point and they should neither be on the same line nor on the same circle. In the following, we derive the Least Squares Method for three dimensional problems.

We determine the derivatives of a function by using the Taylor series expansion and the least squares approximation. Hence, consider m Taylor expansions of $\psi(\boldsymbol{x_i})$ about $\boldsymbol{x}$

$$\psi(\boldsymbol{x}_i) = \psi(\boldsymbol{x}) + \sum_{j=1}^{m} \frac{\partial \psi^{|j|}}{\partial x^{j_1} \partial y^{j_2} \partial z^{j_3}} \frac{1}{j!} (x_i - x)^{j_1} (y_i - y)^{j_2} (z_i - z)^{j_3} + e_i, \quad (3.12)$$

for $i = 1, \ldots, m$, where e_i is the error in Taylor's expansion at the point $\boldsymbol{x}_i$.

Denote the coefficients

$$a_1 = \tfrac{\partial \psi}{\partial x},\ a_2 = \tfrac{\partial \psi}{\partial y},\ a_3 = \tfrac{\partial \psi}{\partial z},\ a_4 = \tfrac{\partial^2 \psi}{\partial x^2},\ a_5 = \tfrac{\partial^2 \psi}{\partial x \partial y},$$
$$a_6 = \tfrac{\partial^2 \psi}{\partial x \partial z},\ a_7 = \tfrac{\partial^2 \psi}{\partial y^2},\ a_8 = \tfrac{\partial^2 \psi}{\partial y \partial z},\ a_9 = \tfrac{\partial^2 \psi}{\partial z^2}.$$

Let us assume that $\psi(\boldsymbol{x}) = \psi$ is the known discrete function value at the particle position $\boldsymbol{x}$. For $m > 9$, this system is overdetermined with respect to the unknowns a_i and can be rewritten as

$$\boldsymbol{e} = M\boldsymbol{a} - \boldsymbol{b}, \quad (3.13)$$

where

$$M = \begin{pmatrix} dx_1 & dy_1 & dz_1 & \frac{1}{2}dx_1^2 & dx_1 dy_1 & dx_1 dz_1 & \frac{1}{2}dy_1^2 & dy_1 dz_1 & \frac{1}{2}dz_1^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ dx_m & dy_m & dz_m & \frac{1}{2}dx_m^2 & dx_m dy_m & dx_m dz_m & \frac{1}{2}dy_m^2 & dy_m dz_m & \frac{1}{2}dz_m^2 \end{pmatrix},$$

$\boldsymbol{a} = (a_1, a_2, \ldots a_9)^T$, $\boldsymbol{b} = (\psi_1 - \psi, \ldots, \psi_m - \psi)^T$, $\boldsymbol{e} = (e_1, \ldots, e_m)^T$ and $dx_i = x_i - x,\ dy_i = y_i - y, dz_i = z_i - z$.

The unknowns a_i are computed by minimizing a weighted error over the neighboring points. Thus, we have to minimize the following quadratic form

$$J = \sum_{i=1}^{m} w_i e_i^2 = (M\boldsymbol{a} - \boldsymbol{b})^T W (M\boldsymbol{a} - \boldsymbol{b}) \quad (3.14)$$

with $W = \mathrm{diag}\,(w_1, \ldots, w_m)$, where $w_i = w(\boldsymbol{x}_i - \boldsymbol{x}; h)$. The minimization of J with respect to $\boldsymbol{a}$ formally yields (if $M^T W M$ is nonsingular)

$$\boldsymbol{a} = (M^T W M)^{-1} (M^T W) \boldsymbol{b}. \quad (3.15)$$

3.2 Least Squares Method for Solving Elliptic Equations

We now consider the following linear second order differential model equation, which represents all equations in the above presented projection scheme

$$A\psi + \boldsymbol{B} \cdot \nabla\psi + C\Delta\psi = f, \tag{3.16}$$

where the coefficients $A, \boldsymbol{B}, C$ are given and real and $f = f(\boldsymbol{x})$ is a given real valued function. We solve this equation with Dirichlet $\psi = \phi$ or Neumann boundary conditions

$$\frac{\partial\psi}{\partial\boldsymbol{n}} = \phi \quad \text{on} \quad \Gamma. \tag{3.17}$$

In the following, we demonstrate the method to solve (3.16-3.17). To our knowledge, there are two types of methods of directly solving elliptic equations in a given meshfree configuration. The first one is presented in [LO80], which can be directly derived from the equation (3.15). The second one is presented in [TK01], where equations (3.16) and (3.17) are added as constraints in the least squares approximation. The comparisons of both methods are presented in [IT02]. It is found that the method presented in [TK01] is more stable and the Neumann boundary condition can be easily included in the approximation. In this paper, we give a short overview about the method presented in [TK01].

We consider $\boldsymbol{x}$ as a central particle and its set of neighbors $P(\boldsymbol{x}, h) = \{\boldsymbol{x}_i : i = 1, 2, \ldots, m\}$. Furthermore, we consider the above Taylor's expansions (3.12). In (3.12) we have assumed that $\psi(\boldsymbol{x}) = \psi$ is a known discrete function value at $\boldsymbol{x}$. Now, let us assume that ψ is not known and denote it by a_0.

We add equations (3.16) and (3.17) as constraints into the m Taylor's expansions (3.12) . These two additional equations is rewritten in the following forms

$$Aa_0 + B_1a_1 + B_2a_2 + B_3a_3 + C(a_4 + a_7 + a_9) = f \tag{3.18}$$

$$n_1a_1 + n_2a_2 + n_3a_3 = \phi, \tag{3.19}$$

where $\boldsymbol{B} = (B_1, B_2, B_3), \boldsymbol{n} = (n_1, n_2, n_3)$. Note that, for the Dirichlet boundary condition, we have only the $m + 1$ equations, where we directly prescribe the boundary conditions on the boundary particles. The matrix M and vectors $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{e}$ are slightly different from above. They are given by

$$\tilde{M} = \begin{pmatrix} 1 & dx_1 & dy_1 & dz_1 & \frac{1}{2}dx_1^2 & dx_1dy_1 & dx_1dz_1 & \frac{1}{2}dy_1^2 & dy_1dz_1 & \frac{1}{2}dz_1^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & dx_m & dy_m & dz_m & \frac{1}{2}dx_m^2 & dx_mdy_m & dx_mdz_m & \frac{1}{2}dy_m^2 & dy_mdz_m & \frac{1}{2}dz_m^2 \\ A & B_1 & B_2 & B_3 & C & 0 & 0 & C & 0 & C \\ 0 & n_1 & n_2 & n_3 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

and by

$$\tilde{\boldsymbol{a}} = (a_0, a_1, a_2, \ldots, a_9)^T, \ \tilde{\boldsymbol{b}} = (\psi_1, \ldots, \psi_m, f, \phi)^T, \\ \text{and} \qquad \tilde{\boldsymbol{e}} = (e_1, \ldots, e_m, e_{m+1}, e_{m+2})^T.$$

Now, we minimize the functional

$$\tilde{J} = \sum_{i=1}^{m+2} w_i e_i^2, \tag{3.20}$$

where $e_{m+1} = (A\psi + \boldsymbol{B} \cdot \nabla\psi + C\Delta\psi - f)$, $e_{m+2} = \left(\frac{\partial\psi}{\partial \boldsymbol{n}} - \phi\right)$ and $w_{m+1} = w_{m+2} = 1$.

Similarly, the minimization of $\tilde{J}$ yields

$$\tilde{\boldsymbol{a}} = (\tilde{M}^T\tilde{W}\tilde{M})^{-1}(\tilde{M}^T\tilde{W})\tilde{\boldsymbol{b}} \tag{3.21}$$

with $\tilde{W} = \text{diag}\,(w_1, \ldots, w_m, 1, 1)$.

The vector $(\tilde{M}^T\tilde{W})\tilde{\boldsymbol{b}}$ is explicitely given by

$$\begin{aligned}(\tilde{M}^T\tilde{W})\tilde{\boldsymbol{b}} = \Bigg(&\sum_{i=1}^m w_i\psi_i + Af,\ \sum_{i=1}^m w_i dx_i\psi_i + B_1 f + n_1\phi,\\ &\sum_{i=1}^m w_i dy_i\psi_i + B_2 f + n_2\phi,\ \sum_{i=1}^m w_i dz_i\psi_i + B_3 f + n_3\phi,\\ &\frac{1}{2}\sum_{i=1}^m w_i dx_i^2\psi_i + Cf,\ \sum_{i=1}^m w_i dx_i dy_i\psi_i,\ \sum_{i=1}^m w_i dx_i dz_i\psi_i,\\ &\frac{1}{2}\sum_{i=1}^m w_i dy_i^2\psi_i + Cf,\ \sum_{i=1}^m w_i dy_i dz_i\psi_i,\ \frac{1}{2}\sum_{i=1}^m w_i dz_i^2\psi_i + Cf\Bigg)^T.\end{aligned}$$

Let $\beta_0, \beta_1, \ldots, \beta_9$ be the first row of the matrix $(\tilde{M}^T\tilde{W}\tilde{M})^{-1}$. We are looking for the function $\psi = a_0$, therefore, equating the first components of vectors on both sides of (3.21), we obtain

$$\begin{aligned}\psi = \beta_0&\left(\sum_{i=1}^m w_i\psi_i + Af\right) + \beta_1\left(\sum_{i=1}^m w_i dx_i\psi_i + B_1 f + n_1\phi\right) +\\ \beta_2&\left(\sum_{i=1}^m w_i dy_i\psi_i + B_2 f + n_2\phi\right) + \beta_3\left(\sum_{i=1}^m w_i dz_i\psi_i + B_3 f + n_3\phi\right) +\\ &\beta_4\left(\frac{1}{2}\sum_{i=1}^m w_i dx_i^2\psi_i + Cf\right) + \beta_5\left(\sum_{i=1}^m w_i dx_i dy_i\psi_i\right) +\\ &\beta_6\left(\sum_{i=1}^m w_i dx_i dz_i\psi_i\right) + \beta_7\left(\frac{1}{2}\sum_{i=1}^m w_i dy_i^2\psi_i + Cf\right) +\\ &\beta_8\left(\sum_{i=1}^m w_i dy_i dz_i\psi_i\right) + \beta_9\left(\frac{1}{2}\sum_{i=1}^m w_i dz_i^2\psi_i + Cf\right).\end{aligned}$$

Rearranging the terms, we have

$$\psi - \sum_{i=1}^{m} w_i \left(\beta_0 + \beta_1 dx_i + \beta_2 dy_i + \beta_3 dz_i + \beta_4 \frac{dx_i^2}{2} + \beta_5 dx_i dy_i + \right.$$
$$\left. + \beta_6 dx_i dz_i + \beta_7 \frac{dy_i^2}{2} + \beta_8 dy_i dz_i + \beta_9 \frac{dz_i^2}{2} \right) \psi_i =$$
$$(\beta_0 A + \beta_1 B_1 + \beta_2 B_2 + \beta_3 B_3 + \beta_4 C + \beta_7 C + \beta_9 C)\, f +$$
$$+ (\beta_1 n_1 + \beta_2 n_2 + \beta_3 n_3)\, \phi.$$

Hence, if we consider $\boldsymbol{x}_j$ an arbitrary particle and $\boldsymbol{x}_{j_i}$ its neighbors of number $m(j)$, then we have the following sparse system of equations for the unknowns $\psi_j, j = 1, \ldots, N$

$$\psi_j - \sum_{i=1}^{m(j)} w_{j_i} \left(\beta_0 + \beta_1 dx_{j_i} + \beta_2 dy_{j_i} + \beta_3 dz_{j_i} + \beta_4 \frac{dx_{j_i}^2}{2} + \beta_5 dx_{j_i} dy_{j_i} + \right.$$
$$\left. \beta_6 dx_{j_i} dz_{j_i} + \beta_7 \frac{dy_{j_i}^2}{2} + \beta_8 dy_{j_i} dz_{j_i} + \beta_9 \frac{dz_{j_i}^2}{2} \right) \psi_{j_i} =$$
$$[\beta_0 A + \beta_1 B_1 + \beta_2 B_2 + \beta_3 B_3 + (\beta_4 + \beta_7 + \beta_9) C]\, f_j +$$
$$(\beta_1 n_1 + \beta_2 n_2 + \beta_3 n_3)\, \phi_j. \qquad (3.22)$$

We can represent the above sparse system in compact matrix-vector-form as

$$A\boldsymbol{\Psi} = \boldsymbol{b}. \qquad (3.23)$$

Hence, (3.23) is a big sparse linear system of equations and can be solved using iterative methods. In this paper we have used SOR method.

Now back to the projection method for the Navier-Stokes equations, we have $d + 2$ such iterative systems, where d is the number of space dimension. As an initial guess for the iterative solvers at time level $n + 1$, we assume the corresponding value at the time level n. This saves a lot of iteration steps.

4 Numerical Tests

4.1 Solutions of Second Order Linear PDEs

Example 1

Consider the second order partial differential equation of the type

$$\psi + \psi_x + \psi_y + \psi_{xx} + \psi_{yy} = f \qquad \text{in} \qquad [0,1] \times [0,1]. \qquad (4.1)$$

The exact solution is given by

$$\psi = (x - \frac{1}{2})(y - \frac{1}{2})(1 - \frac{x^2}{2} - \frac{y^2}{2}). \qquad (4.2)$$

We consider the Dirichlet boundary value problems, where the boundary conditions can be directly obtained from the exact solution.

Table 4.1 shows the maximum error between the exact and the numerical solutions and shows second order convergence.

Table 4.1. Convergence results for Example 1.

N	Smoothing length	Maximum Error
676	0.1	0.00011
2601	0.05	2.7867e-5
10201	0.025	6.9778e-6

Example 2

Consider the following equation with discontinuous coefficient

$$\psi + \nabla \cdot (k \nabla \psi) = f \qquad \text{in} \qquad [0,1] \times [0,1], \tag{4.3}$$

where

$$k = \begin{cases} 1000, & \text{if } y \geq 0.5 \\ 1, & \text{else.} \end{cases}$$

Consider the exact solution

$$\psi = \frac{1}{k}(x - \frac{1}{2})(y - \frac{1}{2})(1 - \frac{x^2}{2} - \frac{y^2}{2}). \tag{4.4}$$

We again consider the Dirichlet boundary conditions. In this example, the source is given by

$$f = -(3x - \frac{1}{2})(y - \frac{1}{2}) - (x - \frac{1}{2})(3y - \frac{1}{2}) \tag{4.5}$$

and we have smoothed three times the coefficient k in the vicinity of the interface $y = 0.5$. The smooth coefficient k is denoted by $\tilde{k}$. Hence the above equation (4.3) is given by

$$\psi + \nabla \tilde{k} \cdot \nabla \psi + \tilde{k} \Delta \psi = f. \tag{4.6}$$

In Table 4.2 we have presented the maximum error between the exact and numerical solutions and we see that the numerical solution converges with of order one. If we use the interface conditions, we get the second order convergence [IT02].

Table 4.2. Convergence results for Example 2.

N	Smoothing length	Maximum Error
676	0.1	0.0691
2601	0.05	0.0377
10201	0.025	0.0203

4.2 Compressible Flows

Consider the 1d compressible flow. We note that, we do not consider smoothed density $\tilde{\rho}$ here. The viscosity is considered to be a constant number. Suppose the flow is ideal gas where the equation of state is given by

$$p = \rho RT = \frac{R}{c_v}\rho e,$$

where R is the gas constant and e the internal energy. Hence, we have

$$\nabla \cdot \boldsymbol{v} = \frac{1}{T}\frac{dT}{dt} - \frac{1}{p}\frac{dp}{dt}. \tag{4.7}$$

We have further assumed the dynamic viscosity μ and the heat conductivity coefficient κ to be constant. The scheme is tested by solving the Sod problem [Sod78], where we let the heat conductivity and viscosity tend to zero such that the solution of the Navier-Stokes equations converges to those of the Euler equations.

In this case, equation (3.2) is given by

$$v^* - \Delta t\frac{4}{3}\frac{\mu}{\rho}\frac{\partial^2 v^*}{\partial x^2} = v^n \tag{4.8}$$

with boundary condition $\boldsymbol{v}^* = 0$. Equation (3.7) is given by

$$-\frac{\rho^n}{\Delta t^2 p^n}p^{n+1} - \frac{1}{\rho^n}\frac{\partial \rho^n}{\partial x}\frac{\partial p^{n+1}}{\partial x} + \frac{\partial^2 p^{n+1}}{\partial x^2} = -\frac{\rho^n}{\Delta t^2} + \rho^n\frac{\nabla \cdot v^*}{\Delta t} - \frac{\rho^n}{T\Delta t}\frac{dT}{dt}, \tag{4.9}$$

where $\frac{dT}{dt}$ is replaced by the energy equation (2.4). Equation (4.9) is solved together with the Neumann boundary condition (3.9). After obtaining the pressure, we correct the velocity $\boldsymbol{v}^*$ according to (3.3).

Equation (3.10) is given by

$$T^{n+1} - \frac{\Delta t}{c_v}\frac{\kappa}{\rho^n}\frac{\partial^2 T^{n+1}}{\partial x^2} = T^n + \frac{\Delta t}{\rho^n c_v}\left(-p^n\frac{\partial v^n}{\partial x} + \frac{4}{3}\mu(\frac{\partial v^n}{\partial x})^2\right) \tag{4.10}$$

with boundary conditions $T = T_0$ on Γ. Finally, we update the density according to (2.5).

Let the domain $\Omega = (0, 1)$ with boundary points 0 and 1. We consider the discretization over $[0, 1]$ with N particles at x_i for $i = 1, 2, \ldots, N$ and constant time step Δt.

The initial conditions for Sod's problem are [Sod78]

$$\rho_i^0 = 1, \;\; v_i^0 = 0, \;\; e_i^0 = 2.5 \;\text{ for }\; 0 \le x_i < 0.5$$

$$\rho_i^0 = 0.125, \;\; v_i^0 = 0, \;\; e_i^0 = 2.0 \;\text{ for }\; 0.5 \le x_i \le 1$$

completed by the following boundary conditions

$$v(t) = 0,\ e(t) = 2.5 \ \text{ at } \ x = 0 \ \text{ and } \ v(t) = 0,\ e(t) = 2 \ \text{ at } \ x = 1.$$

We note that the initial and boundary conditions for the temperature T is obtained from the relation

$$T = \frac{1}{c_v} e.$$

The initial spacing of the particles is given by $dx = 1/N$, where N is the total number of particles considered and the size of the support h is equal to 3 times the initial spacing of the particles.

Since the scheme is of central difference type, there are some oscillations for small viscosity. Therefore, we choose $\mu = \mu(N),\ \kappa = \kappa(N)$. For example, for $N = 100$, we considered $\mu = 0.001$ and $\kappa = 0.001$. The heat coefficient does not play big role for the stability of the scheme. It can be set to zero.

Since we solve the conservation equations implicitly, we need the restriction of time step only for the motion of the particles. The time step is should be chosen such that the particles cannot move more than a partition of h in each time step.

The numerical solutions are obtained for 100, 400 and 2000 particles and are compared with the exact solutions of the compressible Euler equation at the fixed time $t = 0.2$. The values for *mu* and *kappa* are for 400 and 2000 particles are four and twenty times smaller than thoose for 100 particles. The time step for 100 particles is chosen 0.002. Similarly, for 400 and 2000 particles the time smaller step is taken by corresponding factors.

In figures 4.1 we plot the exact and numerical results, like the density, velocity and the pressure. It is clear that the scheme is stable and the solutions of the Navier-Stokes equations tend to the Euler solutions when the number of particles tends to infinity and the viscosity and the heat conduction coefficient tend to zero.

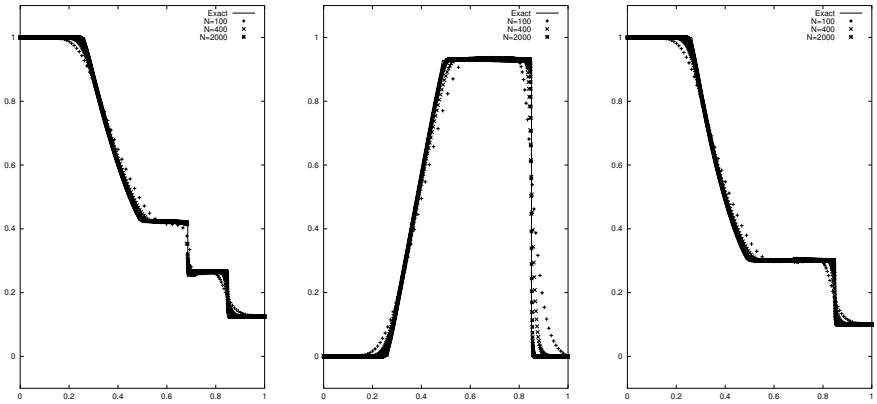


Figure 4.1. Density(left), Velocity(center) and Pressure (right) at $t = 0.2$. Solid lines represent the exact solutions, dots represent the numerical solutions.

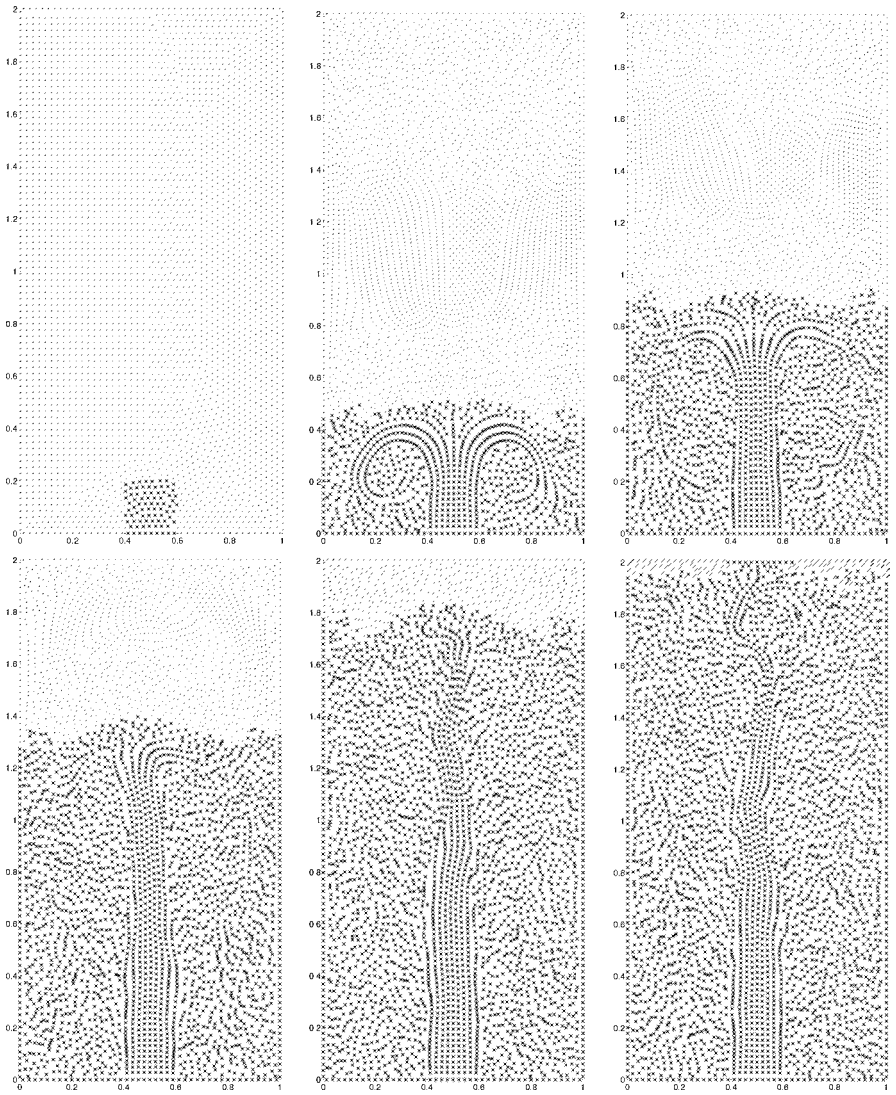


Figure 4.2. Level of water at $t = 0.0, 1.0, 2.0$ (top row, left to write) and at $t = 3.0, 4.0, 4.4$ (bottom row), star are water particles and dots are gas particles

4.3 Two Phase Flows

In this case we express all quantities in dimensionless form. So, they can be interpreted as being in SI-units. Consider a cavity of $[0, 1] \times [0, 2]$ initially filled with air. On the center of the lower boundary we place a hole as inflow

boundary. Similarly, there is an outflow hole in the top boundary. The width of the inflow and outflow holes is 0.2. The rest of the boundaries are solid walls with no slip conditions. The inflow velocity is 2. There is gravity acting downwards with $g = 9.81$. The densities of air and water are 1 and 1000 respectively. The dynamic viscosity of air is $1.81 * 10^{-5}$ and $1.005 * 10^{-3}$ for water. We consider the air as compressible and the equation of state is given by

$$\rho = \rho_0 + \frac{1}{c^2}(p - p_0) \tag{4.11}$$

with speed of sound $c = 5$ and the reference density ρ_0 and reference pressure p_0 are initial density and pressure, equal to 1 and 0, respectively. The time step $\Delta t = 0.002$ is considered. The theoretical fill time for this particular cavity is 5. Since fill time for initially replaced liquid is 0.1, therefore, the fill time for this case is 4.9.

In Figure 4.2 we have plotted the filling process for different times. At time 4.4 we stopped the simulation since the liquid particles started to leave the outflow boundary. The numerical result shows close approximation of the theoretical fill time.

References

[BKZ92] Brackbill J. U., Kothe D. B. and Zemach C.:: A continuum method for modeling surface tension, J. Comput. Phys., 100, 335–354 (1992)

[Cho68] Chorin A.: Numerical solution of the Navier-Stokes equations, J. Math. Comput. vol. 22, 745–762 (1968)

[CR99] Cummins S. J. and Rudmann M.: An SPH Projection Method, J. Comput. Phys., 152, 284–607 (1999)

[Dil96] Dilts G. A.: Moving least squares particle hydrodynamics I, consistency and stability, Hydrodynamics methods group report, Los Alamos National Laboratory (1996)

[GM97] Gingold R. A., Monaghan J. J.: Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. Roy. Astron. Soc., 181, 375–389 (1997)

[HJKT03] Hietel D., Junk M., Kuhnert J. and Tiwari S.: Meshless methods for Conservation Laws, preprint ITWM (2003)

[GW01] Ginzburg I., Wittum G.: Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolant, J. Comp. Phys., 166, 302–335 (2001)

[HN81] Hirt C. W., Nichols B. D.: Volume of fluid (VOF) method for dynamic of free boundaries, J. Comput. Phys., 39, 201 (1981)

[HW65] Harlow F. H., Welch J. E.: Numerical study of large amplitude free surface motions, Phys. Fluids, 8 (1965)

[IT02] Iliev O., Tiwari S.: A generalized (meshfree) finite difference discretization for elliptic interface problems, Numerical Methods and Applications, Lecture notes in Computer Sciences, I. Dimov, I. Lirkov, S. Margenov, Z. Zlatev (Eds), Springer Verlag, 480–489 (2002)

[KP97] Kelecy F. J., Pletcher R. H.: The development of free surface capturing approach for multi dimensional free surface flows in closed containers, J. Comput. Phys., 138, 939 (1997)

[Ku99] Kuhnert J.: General smoothed particle hydrodynamics, Ph.D. thesis, Kaiserslautern University, Germany, (1999)

[Ku02] Kuhnert J.: An upwind finite pointset method for compressible Euler and Navier-Stokes equations, Springer LNCSE: Meshfree methods for Partial Differential Equations, Vol. 26, M. Griebel, M. A. Schweitzer (Eds) (2002)

[KTU00] Kuhnert J., Tramecon A., Ullrich P.: Advanced Air Bag Fluid Structure Coupled Simulations applied to out-of Position Cases, EUROPAM Conference Proceedings, ESI group, Paris, France (2000)

[LO80] Liszka T., Orkisz J.: The finite difference method on arbitrary irregular grid and its application in applied mechanics, Computers & Structures, 11, 83–95 (1980)

[Luc77] Lucy L. B.: A numerical approach to the testing of the fission hypothesis, Astron. J., 82, 1013 (1977)

[Mon94] Monaghan J. J.: Simulating free surface flows with SPH, J. Comput. Phys., 110, 399 (1994)

[Mor00] Morris J. P.: Simulating Surface Tension with Smoothed Particle Hydrodynamics, Int. J. Numer. Methods Fluids, 33, 333–353 (2000)

[MFZ97] Morris J. P., Fox P. J., Zhu Y.: Modeling Low Reynolds Number Incompressible Flows Using SPH, J. Comput. Phys., 136, 214–226 (1997)

[Sod78] Sod G. A: A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, J. Comp. Phys. 27, 1–31, (1978)

[TK01] Tiwari S., Kuhnert J.: Grid free method for solving Poisson equation, preprint, Berichte des Fraunhofer ITWM, Kaiserslautern, Germany, Nr. 25 (2001)

[TK102] Tiwari S., Kuhnert J.: Finite pointset method based on the projection method for simulations of the incompressible Navier-Stokes equations, Springer LNCSE: Meshfree methods for Partial Differential Equations, Vol. 26, M. Griebel, M. A. Schweitzer (Eds) (2002)

[TK202] Tiwari S., Kuhnert J.: A meshfree method for incompressible fluid flows with incorporated surface tension , revue aurope'enne des elements finis, Volume 11–nř 7–8, (Meshfree and Particle Based approaches in Computational Mechanics) (2002)

[TK03] Tiwari S., Kuhnert J.: Particle method for simulations of free surface flows, Hyperbolic Problems: Theory, Numerics, Application, Proceedings of the Ninth International Conference on Hyperbolic Problems, T. Y. Hou and E. Tadmor (Eds), Springer Verlag (2003)

[Tiw00] Tiwari S.: A LSQ-SPH approach for compressible viscous flows, International series of numerical mathematics Vol. 141, H. Freistuehler, G. Warnecke (Eds), Birkhaueser (2000)

[TM03] Tiwari S., Manservisi S.: Modeling incompressible Navier-Stokes flows by LSQ-SPH, Nepal Mathematical Sciences Report Vol 20 No 1 & 2 (2003)

SPH Renormalized Hybrid Methods for Conservation Laws: Applications to Free Surface Flows

Jean Paul Vila*

INSA MIP, 135 Avenue de Rangueil, F–31077 Toulouse Cedex 4, France.
ONERA, Avenue Edouard Belin, BP 4025, F–31055 Toulouse Cedex 4, France.

1 Introduction

The SPH method (Smooth Particle Hydrodynamics) is our starting point. It has been "discovered" in 1977 by Lucy ([15]) a British astrophysicist. At this time his computation only requires 100 particles. The effective development of the method is due to J. Monaghan, an Australian applied mathematician ([7], [8], [16]), who also developed most of the extensions of the original techniques to multifluid equations, MHD, etc.

Until 1985, the SPH method was specialized in Astrophysics applications, W. Benz (yet an astrophysicist) is among the first which use SPH methods for complex applications such as high velocity impacts problems with damage models ([4],[5]). Actually, a lot of research center use SPH methods ([19], [11]) as an efficient alternative to Finite Element Lagrangian codes in the field of high velocity impacts. Industrial codes using SPH methods have been available only recently. This is partly due to some difficulties in the theoretical and numerical basis for handling with boundary conditions.

In this paper, I first give an overview of classical recipes for designing SPH methods. I then introduce a weak discrete formulation, which provides an efficient tool for understanding and solving problems related with the global conservation property. I show how Renormalization ([12], [20]), a new efficient tools in the field of SPH methods can be used in this context. I also introduce new hybrid SPH - Godunov type schemes. Combining this new approach with renormalization I overcome usual restrictions on the ratio of the smoothing length to the size of the mesh, which needs to be large enough (or equal to some specific value depending on the kernel function).

I have originally presented this approach, which really mixes Finite Difference type Riemann solver and SPH in [24]. At this time I only use standard SPH (i.e. without renormalization), note that this approach leads to a com-

* Jean-Paul.Vila@gmm.insa-tlse.fr

plete mathematical analysis of the nonlinear scalar case (see [3]); the renormalized case has been studied later on ([2]). J. Monaghan ([18]) also introduced a Riemann problem related with the interaction of neighboring particles. He proposed to use it as a guide line for designing of efficient numerical viscosity, in this sense his approach is different of the one proposed here.

In ([25]) I also introduce a second order version of the method based on MUSCL type extension of the formalism, it seems that recently a similar approach ([10]) has been revisited.

In this paper I review all these recent results and also give some typical numerical experiments enhancing the behavior of these new methodologies.

2 Classical Recipes

The design and analysis of weighted particle methods for transport equations and Euler inviscid equations is a well documented field. Thus we briefly discuss the main tools necessary for a comprehensive study of our results. We refer for example to the book of P. A. Raviart ([21]) for an exhaustive study of the different points reviewed here.

2.1 Particle Approximation of Functions

The first tool is to obtain a quadrature formula on a set of moving particles. Let us take such a set $(x_i(t), w_i(t))_{i\in P}$, indexed by $i \in P$, where $x_i(t)$ is the position of the particle and $w_i(t)$ its weight. Let $\boldsymbol{v}$ a regular vector field in $\mathbb{R}^d$, we classically move the particles along the characteristic curves of the field $\boldsymbol{v}$ and also modify the weights in order to take account of deformations due to the field $\boldsymbol{v}$:

$$\begin{array}{ll} \dfrac{d}{dt}x_i = \boldsymbol{v}(x_i,t) & \dfrac{d}{dt}w_i = \text{div }(\boldsymbol{v}(x_i,t))\, w_i \\ x_i(0) = \xi_i & w_i(0) = \omega_i \end{array}$$

(ξ_i, t) are the Lagrange coordinates of the particle i. We thus have the following quadrature formula where $J(\xi, t)$ is the Jacobean of $(\xi, t) \longrightarrow (x, t)$:

$$\int_{\mathbb{R}^d} f(x)dx = \int_{\mathbb{R}^d} f(x(\xi,t))J(\xi,t)d\xi \approx \sum_{j\in P} w_j(t)f(x_j(t)) \qquad (2.1)$$

Weighted particle Approximation

The previous quadrature formula, together with the following tools:

- Smoothing kernel,
- Convolution,

leads in 3 steps to the smoothed (or regularized) particle approximation of a function:

Smoothing kernel $W(x,h)$

$$W(x,h) = \frac{1}{h^d}\theta(\frac{\|x\|}{h})$$

We take usually θ as a positive function with compact support $\subset [0,2]$ (d is the space dimension), for example:

$$\theta(y) = C \times \begin{cases} 1 - \frac{3}{2}y^2 + \frac{3}{4}y^3 & \text{if } 0 \leq y \leq 1, \\ \frac{1}{4}(2-y)^3 & \text{if } 1 \leq y \leq 2 \end{cases}$$

$C = \frac{2}{3}, \frac{10}{7\pi}, \frac{1}{\pi}$ (d=1,2 or 3). $W(x,h) \longrightarrow \delta$ when $h \longrightarrow 0$. h is the "smoothing length".

Regularization of f

Convolution of f with the kernel W:

$$< f(x) >= f * W(x) = \int f(y)W(x-y,h)dy \tag{2.2}$$

Quadrature of (2.2)

We provide a quadrature of the integral in (2.2) with the help of (2.1), we thus define $\Pi^h(f)$ the smoothed (or regularized) particle approximation of a function f as:

$$\Pi^h(f)(x) = \sum_{j \in P} w_j f(x_j) W(x - x_j(t), h)$$

We shall use the following notations:

$$W_{ij} \equiv W(x_j - x_i, h) \qquad \nabla W_{ij} \equiv grad_x[W(x_i - x_j, h)]$$

Particle approximation of derivatives is easily handled by taking direct derivation of smoothed particle approximations, which gives at the point x_i:

$$\nabla \Pi^h(f)_i = \sum_{j \in P} w_j f(x_j) \nabla W_{ij}$$

Remark 2.1. Here we use the standard notations in SPH literature. The smoothing length is h, the kernel is W and we have chosen Δx as the characteristic size of the mesh. In most of the mathematical papers related to particle weighted methods, the smoothing length is denoted ε, the kernel or cut-off function is ζ^ε and h is the characteristic size of the mesh. The reader has to take account of that in some of the references quoted in this paper.

In classical discretization methods such as Finite Differences, Finite Volumes or Finite Elements we have a unique discretization parameter which is Δx the characteristic size of the mesh. Here we get an additional parameter h, the so-called "smoothing length", which is the characteristic size of the regularizing kernel W. The combined effect of these 2 parameters can be studied accurately. We refer to the book of P.A. Raviart [21] for a detailed analysis of interpolation errors in various Sobolev norms and semi-norms. We just recall some results of [21] which will be useful for our analysis. We restrict ourselves to compactly supported symmetric kernels such that $W \in \mathcal{C}^{m+1}$, $m \geq 2$. Then, there exists a constant $C > 0$ just depending on the transport field $\boldsymbol{v}$ (supposed regular enough) such that:

$$\begin{aligned} &\forall u \in W^{\mu,p}(I\!R^d), \;\; s \geq 0, \mu = \max(r+s, m), \\ &\qquad r = 1, 2, \frac{d}{m} \leq p \leq \infty, \qquad q = \frac{p-1}{p} \\ &|u - \Pi^h(u)|_{s,p,I\!R^d} \leq \\ &\qquad C\left(h^r |u|_{r+s,p,I\!R^d} + (1 + \frac{\Delta x}{h})^{\frac{d}{q}} \frac{(\Delta x)^m}{h^{m+s}} ||u||_{m,p,I\!R^d}\right) \end{aligned} \tag{2.3}$$

Note that we need that the ratio $\frac{(\Delta x)}{h} \longrightarrow 0$, in order that $\Pi^h(u) \longrightarrow u$. In these formulae Δx is the characteristic scale of the mesh. That means - taking account that W is compactly supported - that the number of neighbors of any particles, i.e. the number of particles located at a distance less than a length of order h, needs to go to infinity when h and Δx go to 0.

2.2 Particle Approximation of a Model Partial Differential Equation

We consider the following model PDE in conservation form:

$$L_v(\Phi) + \text{div } F(x, t, \Phi) = S \tag{2.4}$$

where $L_v(\Phi)$ is the transport operator:

$$L_v(\Phi) = \frac{\partial \Phi}{\partial t} + \sum_{l=1,d} \frac{\partial}{\partial x^l}(v^l \Phi)$$

For the mathematical analysis $\boldsymbol{v}$ is supposed to be known and not depending on Φ. Such a dependency is possible and classical in applications. We thus look for an approximation $(\Phi_i(t))_{i \in P}$ of (2.4) defined on the particles moving along the characteristic curves of $\boldsymbol{v}$:

$$\Phi_i := \Phi(x_i(t), t)$$

We thus have

$$\frac{1}{w_i} \frac{d}{dt} (w_i \Phi_i) = L_v(\Phi)_i$$

In order to compute an approximation of div $F(x,t,\Phi)$, we naturally smooth F, which leads to:

$$L_v(\Phi)_i + \nabla \Pi^h(F)_i + F_i \nabla \Pi^h(1)_i = S_i \tag{2.5}$$

or equivalently

$$\frac{d}{dt}(w_i \Phi_i) + w_i \sum_{j \in P} w_j (F_j + F_i) \nabla W_{ij} = w_i S_i \tag{2.6}$$

We have added $\nabla \Pi^h(1)_i$ in order to satisfy the global conservation. We first remark that this term is small (due to(2.3)), we then use $\nabla W_{ij} = -\nabla W_{ji}$ to prove easily that:

$$\frac{d}{dt}\left(\sum_{i \in P} w_i \Phi_i\right) = \sum_{i \in P} w_i S_i, \tag{2.7}$$

discrete version of the continuous conservation of Φ:

$$\frac{d}{dt}\left(\int_{\mathbb{R}^d} \Phi dx\right) = \int_{\mathbb{R}^d} S dx$$

Note that (2.6) is a centered scheme, we thus need some upwinding or equivalently some artificial viscosity in order to stabilize the method. We practically introduce Π_{ij}, an artificial viscosity term satisfying $\Pi_{ij} = \Pi_{ji}$ - we need to satisfy conservation- and the scheme is:

$$\frac{d}{dt}(w_i \Phi_i) + w_i \sum_{j \in P} w_j (F_j + F_i + \Pi_{ij}) \nabla W_{ij} = w_i S_i \tag{2.8}$$

We then perform a suitable time discretization to get an efficient numerical scheme.

Applying this formalism to Euler equations we recover the standard SPH equations (see e.g. [25] for details).

2.3 Gap Theoretical Analysis – Practise

In practical computations, the parameters h and Δx are chosen so that the number of neighbors of any particles, i.e. the number of particles located at a distance less than a length of order h, is almost constant all other the computational domain (≈ 25 for 2D computations, and ≈ 50 in 3D). We thus have $\frac{h}{\Delta x} \simeq \mathcal{O}(1)$, which does not agree with theoretical estimates (2.3), which require at least that $\frac{\Delta x}{h} \longrightarrow 0$. The renormalization techniques - see section 3.2 - overcome this difficulty. As a characteristic behavior of the method let us present the approximation $\nabla \Pi^h(x)$ and $\Pi^h(1)$ obtained by using a set of particles uniformly distributed on a uniform 2D mesh. In such a situation the result is only function of the ratio $\frac{h}{\Delta x}$, we get:

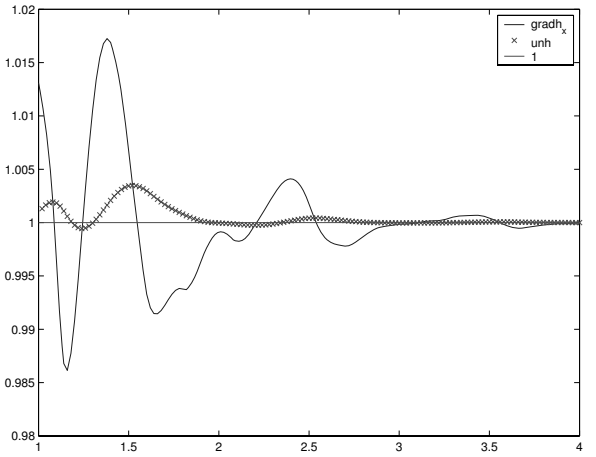


Figure 2.1. $\nabla \Pi^h(x)$ and $\Pi^h(1)$ function of $\frac{h}{\Delta x}$.

The curves in Figure 2.1 represent the value of $\nabla \Pi^h(x)$ and $\Pi^h(1)$ (vertical axis) as a function (horizontal axis) of $\frac{h}{\Delta x}$, it really proves that unless we choose very particular value of the ratio we don't respect consistency in the usual sense! Note that errors concerning evaluation of a derivative may be stronger than 1.5%.

3 Weak Formulation

We now introduce a new formulation of these methods. It is based upon the weak formulation, and combined with renormalization gives an efficient tool for solving the difficulties quoted in the previous comments.

3.1 Basic Principle

We look for an approximation of the system (2.4), given by its weak formulation:

$$\begin{aligned} &\forall \varphi \in C_0^2(I\!R^d \times I\!R^{+,*}) \\ &\int_{I\!R^d \times I\!R^+} (\Phi L_v^*(\varphi) + F(x,t,\Phi)\nabla(\varphi) + S\varphi)\, dxdt = 0 \end{aligned} \tag{3.1}$$

where $-L_v^*$ is the adjoin operator of L_v:

$$L_v^*(\varphi) = \frac{\partial \varphi}{\partial t} + \sum_{l=1,d} v^l \frac{\partial \varphi}{\partial x^l}$$

To get uniqueness we need to introduce the notion of entropy solution, classical in this field, but it is not essential here since we only want to introduce

the main concepts (we refer to [3] for a more detailed analysis). In order to provide a better understanding of particle schemes like those of the previous sections, let us now introduce a general setup for particle approximations of (2.4).
We proceed in 3 steps:

- We provide the space with a discrete scalar product:

$$\int_{I\!R^d} fg\, dx \longrightarrow (f,g)_h := \sum_{j\in P} w_j f_j g_j$$

  which is clearly an approximation of the scalar product in $L^2(I\!R^d)^m$ thanks to the quadrature formula.
- We also introduce the linear operator $D_{h,S}$, which is supposed to approximate strongly the derivative, i.e. for any φ regular enough

$$\sup_{i\in P} \|D_{h,S}\varphi_i - D\varphi_i\| \to 0 \text{ as } h \to 0$$

  and let us define $-D^*_{h,S}$ as the adjoin operator of $D_{h,S}$. We thus have

$$(D_{h,S}\varphi, \Psi)_h = -\left(\varphi, D^*_{h,S}\Psi\right)_h \tag{3.2}$$

- A discrete version of (2.4) is provided by just replacing the integration over $I\!R^d$ by the discrete scalar product $(.,.)_h$ and the derivative $\nabla(\varphi)$ by its approximation $D_{h,S}\varphi$:

$$\int_{I\!R^+} \left[(\Phi, L^*_v(\varphi))_h + (F(\Phi), D_{h,S}\varphi)_h + (S,\varphi)_h\right] dt = 0$$

We thus get after some discrete and continuous integration by parts

$$\frac{d}{dt}(w_i\Phi_i) + w_i D^*_{h,S}(F)_i = w_i S_i \tag{3.3}$$

The schemes (3.3) satisfy automatically a result similar to Lax-Wendroff ([13]) theorem for Finite Difference schemes:

Theorem 3.1. *If* $\overline{\Phi}^{0,h}(x,t) = \sum_{j\in P} w_j\Phi_j(t)\chi_{B_i(t)}(x) \longrightarrow \Phi$ *a.e. where* $(\Phi_j)_{j\in P}$ *is solution of (3.3) and if* $D_{h,S}f \longrightarrow \nabla f$ *for* f *regular when* h *and* Δx *go to* 0*, then* Φ *is a weak solution of (2.4).*

It remains to define the operator $D_{h,S}$. With the same computation as above we easily see that the scheme (3.3) satisfy global conservation (i.e.$\dfrac{d}{dt}(\Phi,1)_h = (S,1)_h$) if $1(x)$ is in the null space of $D_{h,S}$ (i.e.$D_{h,S}f = \nabla f$ for $f \in P0$).

We recover the standard scheme by taking $D_{h,S}f = \nabla I\!I^h(f) - f\nabla I\!I^h(1)$. We thus have

$$D_{h,S}f_i = \sum_{j\in P} w_j(f_j - f_i)\nabla W_{ij}$$

with $P0 \subset \ker(D_{h,S})$ and

$$D^*_{h,S}f_i = \sum_{j\in P} w_j(f_j + f_i)\nabla W_{ij} = \nabla \Pi^h(f)_i + f_i \nabla \Pi^h(1)_i$$

I refer to [25] for a detailed analysis. We now discuss the technical features in relation with use of Renormalization.

3.2 Renormalization

Renormalization is a technique recently appeared in SPH literature ([12], [20]), it is supposed to improve accuracy of the method. We prove here that, with the help of the general setup of section 3.1 it is also conservative in the sense of (2.7). All the approximation and convergence results can be extended by using renormalized particle weighted approximation, this is precisely studied in [2] and [14]. In particular we are able to relax the assumption that the ratio $\frac{\Delta x}{h}$ goes to zero, and we just need that $\frac{\Delta x}{h} = \mathcal{O}(1)$. Numerical tests in [23] and [14] prove that classical SPH scheme converge to a wrong solution unless we take well chosen value of the ratio $\frac{\Delta x}{h}$.

Formally, renormalization is a tool, which provides new formulae for $D_{h,S}f(x)$ with the help of a weight matrix (the renormalization matrix) in the following way:

$$D_{h,S}f(x) = B(x)\nabla^h \Pi^h_g(f)(x) - f(x)B(x)\nabla^h \Pi^h_g(1)(x)$$

We aim to increase the accuracy, thus instead of $P0 \subset \ker(D_{h,S})$ we ask for $P1 \subset \ker(D_{h,S})$. It can be easily proved that

Proposition 3.1. *We have $P1 \subset \ker(D_{h,S})$ or equivalently $D_{h,S}f = \nabla f$ for any polynomial f of in $P1$, if and only if $B(x) = E(x)^{-1}$ with*

$$E(x)^{\alpha\beta} = \sum_{j\in P} w_j(x_j^\beta - x^\beta)\, \partial^\alpha W(x, x_j)$$

We then have $E(x)^{\alpha\beta} = \partial^{\alpha,h}\Pi^h_g(x^\beta) - x^\beta \partial^{\alpha,h}\Pi^h_g(1)$.

Approximation results (2.3) easily prove that $E(x)^{\alpha\beta} \simeq \delta^{\alpha\beta}$ and consequently that $B(x)$ makes sense, if $\frac{(\Delta x)}{h} \longrightarrow 0$. More precisely it can be proved that:

Proposition 3.2. *Let us suppose that B is uniformly bounded (with respect to h_0 and Δx) then we have*

$$\|D_{h,S}\varphi(x) - D\varphi(x)\| \leq Ch_0\|B(x)\|\|D^2\varphi\|_\infty$$

where h_0 is the characteristic scale of the smoothing length (i.e. $\eta_- h_0 \leq h \leq \eta_+ h_0$ with η_- and η_+ two constants > 0)

The consistency of the method is thus satisfied at the only condition that the smoothing length goes to zero. Moreover, it can be proved that, if $\frac{\Delta x}{h}$ is bounded and if the initial distribution of particles is regular enough the matrix $B(x)$ is uniformly bounded and that:

$$\left|(\varphi, D_{h,S}(\varphi))_h\right| = \left|\left(\varphi, D^*_{h,S}(\varphi)\right)_h\right| \leq C\|\varphi\|_h^2.$$

This also insures stability and convergence of the method (at least in the linear case of symmetric first order systems). The discrete operators are defined according to:

$$\begin{aligned} D_{h,S}f_i &= \sum_{j\in P} w_j(f_j - f_i)B_i.\nabla W_{ij} \\ D^*_{h,S}f_i &= \sum_{j\in P} w_j(f_i B_i.\nabla W_{ij} - f_j B_j.\nabla W_{ji}) \end{aligned} \tag{3.4}$$

and the scheme is:

$$\frac{d}{dt}(w_i\Phi_i) + w_i\sum_{j\in P} w_j(f_i B_i.\nabla W_{ij} - f_j B_j.\nabla W_{ji}) = w_i S_i$$

For operators $D_{h,S}$ given by

$$D_{h,S}\varphi_i := \sum_{j\in P} w_j(\varphi_j - \varphi_i)\boldsymbol{A}_{ij}$$

with

$$\begin{aligned} &(i)\sum_{j\in P} w_j\|\boldsymbol{A}_{ij}\| \leq \frac{C}{h_0} \\ &(ii)\|\sum_{j\in P} w_j\boldsymbol{A}_{ij}\| \leq C \\ &(iii)\|\boldsymbol{A}_{ij} + \boldsymbol{A}_{ji}\| \leq Ch_0(\|\boldsymbol{A}_{ij}\| + \|\boldsymbol{A}_{ji}\|) \\ &(iv)\boldsymbol{A}_{ij} = 0 \text{ if } \|x_i - x_j\| \leq Ch_0 \end{aligned} \tag{3.5}$$

where h_0 is the characteristic scale of the mesh, the scheme given by (3.3) is convergent (case of linear symmetric systems). All these approximation results remain true if we use a symmetric variant of the method

$$D_{h,S}f_i = \sum_{j\in P} w_j(f_j - f_i)B_{ij}.\nabla W_{ij} \tag{3.6}$$

where

$$B_{ij} = \frac{1}{2}(B_i + B_j)$$

In this case $A_{ij} = -A_{ji}$ and the numerical scheme reduces to:

$$\frac{d}{dt}(w_i \Phi_i) + w_i \sum_{j \in P} w_j (f_i + f_j) \boldsymbol{A}_{ij} = w_i S_i$$

We also prove that conditions (3.5) are true for (3.4) and (3.6) at the only condition that $\frac{\Delta x}{h} = \mathcal{O}(1)$, which insures the convergence of the method.

The method presented here has a lot of connection with RKPM, PU and other meshless methods. It can be seen as a RKPM or PU method with local approximation of order one, and with nodal integration for quadrature. The formulation proposed here, although less general furnishes a mathematical framework for both stability and accuracy analysis and takes account for numerical quadrature (even if the formal accuracy is limited).

Note also that in the case of non dissipative systems the approach of ([6]) based upon least action principle gives a similar framework.

3.3 Hybrid Schemes Finite Volumes–Weighted Particles

Finally, I report how we can use Godunov type schemes – similar to those used in the field of Finite Volume schemes – inside the methodology of Weighted Particle schemes. We have:

$$\nabla_x W(\boldsymbol{x}_j - \boldsymbol{x}_i) = -D\theta_{ij} \boldsymbol{u}_{ij} \qquad \boldsymbol{u}_{ij} = \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{\|\boldsymbol{x}_j - \boldsymbol{x}_i\|},$$

$$D\theta_{ij} = \frac{1}{h_{ij}^{d+1}} D\theta(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|)$$

where h_{ij} is a local smoothing length. In the symmetric renormalized case we have

$$\boldsymbol{A}_{ij} = -D\theta_{ij} B_{ij} \boldsymbol{u}_{ij}, \qquad \boldsymbol{n}_{ij} = \frac{\boldsymbol{A}_{ij}}{\|\boldsymbol{A}_{ij}\|},$$

Note that in the standard case $\boldsymbol{n}_{ij} = \boldsymbol{u}_{ij}$. In order to simplify, let us take $S = 0$, the renormalized standard scheme is given by

$$\frac{d}{dt}(w_i \Phi_i) - w_i \sum_{j \in P} w_j \left(F_i + F_j\right) \boldsymbol{n}_{ij} D\theta_{ij} = 0$$

which makes appear conservation law in the direction n_{ij}:

$$\frac{\partial}{\partial t}(\Phi) + \frac{\partial}{\partial x}(F(x_{ij}, t, \Phi) \boldsymbol{n}_{ij}) = 0 \tag{3.7}$$

Therefore it is natural to introduce a 1-dimensional finite difference scheme in conservation form associated to (3.7), which brings a sufficient numerical viscosity. Such a scheme consists in replacing the centered approximation $(F(\Phi_i) + F(\Phi_j))\boldsymbol{n}_{ij}$ by the numerical flux of a Finite Difference scheme $2g(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j)$, which is required to satisfy:

$$\begin{array}{l}(i)\ \ g(\boldsymbol{n}, u, u) = F(u)\boldsymbol{n} \\ (ii)\ g(\boldsymbol{n}, u, v) = -g(-\boldsymbol{n}, v, u)\end{array}$$

The numerical viscosity $Q(\boldsymbol{n}, u, v)$ is classically defined in the scalar case (i.e. $\Phi \in I\!R$) as:

$$Q(\boldsymbol{n}, u, v) = \frac{F(u)\boldsymbol{n} - 2g(\boldsymbol{n}, u, v) + F(v)\boldsymbol{n}}{v - u}$$

The new system is written

$$\frac{d}{dt}(w_i\Phi_i) - w_i \sum_{j\in P} w_j 2g(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j) D\theta_{ij} = 0, \quad \Phi_i(0) = \Phi^0(\xi_i) \tag{3.8}$$

or equivalently

$$\begin{array}{l}\dfrac{d}{dt}(w_i\Phi_i) - w_i \displaystyle\sum_{j\in P} w_j (F(x_{ij}, t, \Phi_i) + F(x_{ij}, t, \Phi_j))\boldsymbol{n}_{ij} \\ \qquad\qquad + Q(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j)(\Phi_i - \Phi_j) D\theta_{ij} = 0 \\ \Phi_i(0) = \Phi^0(\xi_i)\end{array}$$

This form is much closer to the one classically used in SPH literature (2.8), where $Q(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j)(\Phi_i - \Phi_j)$ replace here Π_{ij}. We can expect more robustness, as it is the case in the field of F.V. schemes. Mathematical analysis of the convergence is possible (cf. [3] , [2], [14]).

This approach is connected with the notion of Arbitrary Lagrange Euler approximation (A.L.E.), it is also in some sense a generalization of ideas develop by Harten and Hymann ([9]) in their work on self adjusting grid methods for conservation laws.

In the simpler case of a null transport field, the formula 3.8 makes possible a clear comparison with the corresponding formula in the context of finite volume schemes:

$$\frac{d}{dt}(m_i\Phi_i) - \sum_{j\in N(i)} m_{ij} g(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j) = 0$$

where m_i is the volume of the cell i and m_{ij} is the surface of the edge connecting the neighouring cells i and j. Thus the volume w_i acts as m_i and $2w_iw_jD\theta_{ij}$ acts as m_{ij}. One of the main difficulties in the mathematical analysis of the SPH scheme is that the identity

$$\sum_{j\in N(i)} m_{ij} g(\boldsymbol{n}_{ij}, \Phi_i, \Phi_i) = F(\Phi_i) \sum_{j\in N(i)} m_{ij}\boldsymbol{n}_{ij} = 0$$

which is true for Finite Volume methods turns only to

$$\sum_{j\in P} w_j 2g(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j) D\theta_{ij} = 2F(\Phi_i) \sum_{j\in N(i)} w_j \boldsymbol{n}_{ij} D\theta_{ij} \neq 0$$

More precisely it goes strongly to zero with the ratio $\frac{\Delta x}{h}$, or only weakly to zero in the case of the renormalized version of the method. We refer to ([3], [2],[14]) for detailed mathematical analyis.

3.4 Higher Order Version of the Method

In the field of Particle methods a classical tool to increase accuracy is to increase the smoothness of the kernel together with making equal to zero its momentum. This approach has been used in SPH calculations, for example by using a "super Gaussian" kernel. To overcome numerical difficulties due to the use of artificial viscosity - for example the sensitivity of the numerical results to the value of the coefficients α and β -, we have introduced in the previous section an alternative to the classical artificial viscosity. The increase of accuracy by modifying the kernel introduces some instability due to the non positiveness of the kernel. Thus, we propose a different method based upon the well known techniques of MUSCL schemes developed by Van Leer [22] for finite difference schemes in the end of seventies, and extended more recently to finite volume methods (see e.g. Ref.[1] for a mathematical analysis).

The idea is to take account of information given by $\nabla \Pi^h(\Phi)$ to compute a numerical flux, which increases the accuracy of the method: We replace the flux $g(n_{ij}, \Phi_i, \Phi_j)$, which approximates the flux of the conservation law (3.7) located at x_{ij} by $g(\boldsymbol{n}_{ij}, \Phi_{ij}, \Phi_{ji})$. Φ_{ij} is an approximation of Φ at x_{ij} given by a first order Taylor expansion from the point x_i:

$$\Phi_{ij} = \Phi_i + \nabla(\Pi^h(\Phi))_i.(x_{ij} - x_i).$$

It is well known in the field of MUSCL finite difference schemes that such an approximation leads to unstable schemes unless we introduce a limitation of the derivative $\nabla(\Pi^h(\Phi))_i$ of the unknown used to compute the values at x_{ij}. We propose to start from the value:

$$\nabla^h(\Phi^l)_i = \nabla(\Pi^h(\Phi^l))_i = \sum_{j\in P} w_j \left(\Phi_j^l - \Phi_i^l\right) \boldsymbol{A}_{ij},$$

and then to make a loop over all the neighboring particles such that each component of the gradient is reduced in a way that for all neighboring particles j:

$$\begin{cases} \Phi_{ij}^l - \Phi_i^l = \lambda_{ij}^l (\Phi_j^l - \Phi_i^l), \\ \text{with } 0 \leq \lambda_{ij}^l \leq 1, \end{cases}$$

where the interface values at x_{ij} have been computed with help of the limited gradient according to:

$$\Phi_{ij}^l = \Phi_i^l + \nabla^h(\Phi^l)_i.(x_{ij} - x_i).$$

In the scalar case it can be proven that the method is L^∞ stable and convergent with suitable assumptions on the numerical flux.

4 Application to Euler Equations

4.1 Use of Riemann Solvers

Instead of using artificial viscosity, we can use techniques of section 3.3 based upon Godunov type schemes and Riemann solvers. Let us give a regular transport field $\boldsymbol{v}^0(x,t)$. We then consider the following conservative form of Euler equations in 2D (for simplicity)

$$L_{v^0}(\Phi) + \sum_{i=1,d} \frac{\partial}{\partial x^l}(F_E^l(\Phi) - v^{0,l}\Phi) = 0,$$

where the fluxes F_E^l are given by:

$$(i)\ F_E^1(\Phi) = \begin{pmatrix} \rho v^1 \\ p + \rho(v^1)^2 \\ \rho v^1 v^2 \\ v^1(p+E) \end{pmatrix}, \qquad (ii)\ F_E^2(\Phi) = \begin{pmatrix} \rho v^2 \\ \rho v^1 v^2 \\ p + \rho(v^2)^2 \\ v^2(p+E) \end{pmatrix}.$$

We thus have to solve between each particle i and j, the Riemann problem:

$$\begin{cases} \dfrac{\partial}{\partial t}(\Phi) + \dfrac{\partial}{\partial x}((F_E(\Phi)n_{ij} - v^0(x_{ij},t)\boldsymbol{n}_{ij}\Phi)) = 0, \\ \Phi(x,0) = \begin{cases} \Phi_i \text{ if } x < 0, \\ \Phi_j \text{ if } x > 0 \end{cases}, \end{cases} \tag{4.1}$$

Let us consider the classical Riemann problem for Euler equations:

$$\begin{cases} \dfrac{\partial}{\partial t}(\Phi) + \dfrac{\partial}{\partial x}(F_E(\Phi)\boldsymbol{n}_{ij}) = 0, \\ \Phi(x,0) = \begin{cases} \Phi_i \text{ if } x < 0, \\ \Phi_j \text{ if } x > 0 \end{cases}, \end{cases} \tag{4.2}$$

and let us denote by $\Phi_E(\dfrac{x}{t}; \Phi_i, \Phi_j)$ the solution of this problem. An easy calculation proves that the solution of (4.1) is given by:

$$\begin{cases} \Phi = \Phi_E(\dfrac{x + X^0(t)}{t}; \Phi_i, \Phi_j), \\ X^0(t) = \displaystyle\int_0^t v^0(x_{ij},\tau)n_{ij}d\tau. \end{cases} \tag{4.3}$$

It follows that a reasonable choice for the flux $g_E(n_{ij}, \Phi_i, \Phi_j)$ of the Godunov scheme associated with our Smooth Particle approximation is:

$$\begin{cases} \lambda^0_{ij} = v^0(x_{ij}, t)\boldsymbol{n}_{ij}, \\ \Phi_{ij}(\lambda^0_{ij}) = \Phi_E(\lambda^0_{ij}; \Phi_i, \Phi_j), \\ G_E(\Phi_i, \Phi_j) = F_E(\Phi_{ij}(\lambda^0_{ij})) - v^0(x_{ij}, t) \otimes \Phi_{ij}(\lambda^0_{ij}), \\ g_E(\boldsymbol{n}_{ij}, \Phi_i, \Phi_j) = G_E(\Phi_i, \Phi_j)\boldsymbol{n}_{ij}. \end{cases}$$

The resulting particle approximation is given by:

$$\frac{d}{dt}(x_i) = v^0(x_i, t), \qquad \frac{d}{dt}(w_i) = w_i div(v^0(x_i, t)),$$
$$\frac{d}{dt}(w_i\Phi_i) + w_i \sum_{j\in P} w_j 2G_E(\Phi_i, \Phi_j)\boldsymbol{A}_{ij} = 0, \qquad \Phi_i(0) = \Phi^0(\xi_i).$$

The detailed equations for mass, momentum and total energy conservation are:

$$\begin{cases} \dfrac{d}{dt}(w_i\rho_i) + w_i \displaystyle\sum_{j\in P} w_j 2\rho^0_{E,ij}(v^0_{E,ij} - v^0(x_{ij}, t))\boldsymbol{A}_{ij} = 0, \\ \dfrac{d}{dt}(w_i\rho_i v_i) + w_i \displaystyle\sum_{j\in P} w_j 2\left[\rho^0_{E,ij} v^0_{E,ij} \otimes (v^0_{E,ij} - v^0(x_{ij}, t)) + p^0_{E,ij}\right]\boldsymbol{A}_{ij} = 0, \\ \dfrac{d}{dt}(w_i\rho_i E_i) + w_i \displaystyle\sum_{j\in P} w_j 2\left[E^0_{E,ij}(v^0_{E,ij} - v^0(x_{ij}, t)) + p^0_{E,ij} v^0_{E,ij}\right]\boldsymbol{A}_{ij} = 0, \end{cases}$$

where $(\rho^0_{E,ij}, \rho^0_{E,ij} v^0_{E,ij}, E^0_{E,ij})^T = \Phi_{ij}(\lambda^0_{ij})$. We refer to [25] for an extension of this methodology to the case of approximate Riemann solvers.

Comments

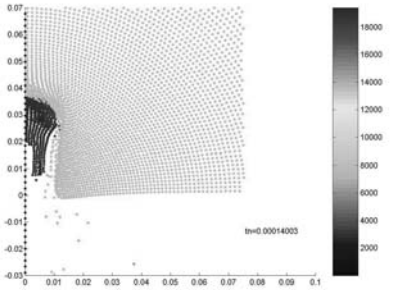
Even if we take the transport field equal to the velocity of the particles, the mass conservation does not keep $w_i\rho_i = m_i$ constant as in the standard method developed in the previous paragraph. Although we loose this nice property of the method, we keep global conservation of mass, momentum and energy. We also expect more robustness since we have convergence results in the scalar case.

We believe that it is sometimes quite essential to move the particles with a smoother velocity field than the exact velocity together for theoretical and computational reasons (the XSPH variant of the method introduced by Monaghan [16] moves the particle with smooth velocity).

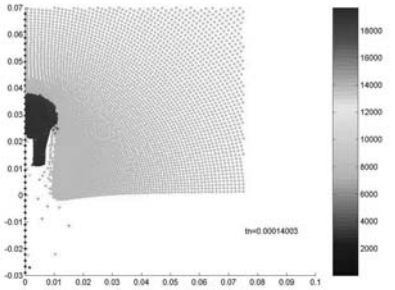
5 Applications

5.1 SPH and High Speed Dynamic

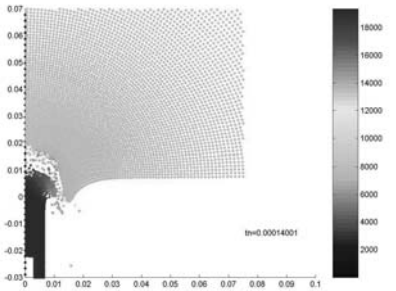
We present here some comparison of the standard and renormalized classical SPH schemes with artificial viscosity on a typical High velocity impact test



(a) Case 1



(b) Case 2



(c) Case 3

Figure 5.2. Impact of Tungsten projectile. See Color Plate 7 on page 298.

case of a tungsten rod at 1300 km/s against a piece of steel. The aim of the test case is to evaluate the influence of the scheme parameters on the observed shape of the crater. (radius and high). The behavior is hypoelastic with a yield criterion for plastic flow. Mie Gruneisen pressure laws are used for both tungsten and steal.

$$p(\rho, u) = (1 - \frac{\Gamma}{2}\frac{\mu}{1+\mu})P_H(\mu) + \Gamma\rho_0 u$$
$$P_H(\mu) = \begin{cases} \rho_0 c_0^2 \mu(1 + (2s-1)\mu + (s-1)(3s-1)\mu^2) \text{ if } \mu > 0 \\ \rho_0 c_0^2 \mu \quad \text{if } \mu \leq 0 \end{cases}$$

ρ_0 is the density at rest, c^2 is the sound velocity. The behavior is perfectly elastoplastic with modulus G_0 and Yield strength Y_0, see Table 5.1. We present

Table 5.1. Material parameters for tungsten and steel.

Material	Tungsten	Steel
pressure law	$\rho_0 = 19300\ kg/m^3$ $c_0 = 4030\ m/s$ $s = 1.237$ $\Gamma = 1.67$	$\rho_0 = 7860\ kg/m^3$ $c_0 = 4610\ m/s$ $s = 1.73$ $\Gamma = 1.67$
mech. behavior	$G_0 = 168.68\ 10^9 Pa$ $Y_0 = 8.10^8 Pa$ and $Y_0 = 30.10^8 Pa$	$G_0 = 81.\ 10^9 Pa$ $Y_0 = 12.10^8 Pa$

in Figure 5.2 results with Standard ($\frac{h}{\Delta x}$=1.2) and renormalized method with two level of discretization. The computations use axisymetric 2D geometry.

Case 1 and case 2 give the final computed state (the projectile is at rest) with the renormalized scheme and with 2 level of grid (the finer one is Case 2). The result is already correct in case 2 with only 10 particles within the radius of the projectile. Case 3 represents the result with Standard SPH and the finer mesh, the computation reveals some unexpected rebound of the projectile. Note that this phenomenon does not occur if we use standard scheme with a less refined distribution of particles.

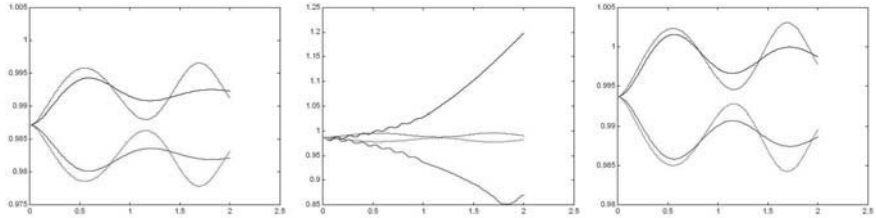
This example is characteristic of the robustness of the new algorithms with respect to mesh sensitivity.

5.2 Free Surface Flow Computation

We present here results concerning simulation of incompressible free surface flow such as those that can arise in sloshing of satellite tanks or launcher tanks. All the computations are performed with an artificial compressibility of the water. Practically we use a linear pressure law, with a sound velocity chosen such that the effective Mach number of the flow is less than 0.1. This procedure is the key point to compute such incompressible flow with SPH explicit in time tools.

A Linear Regime Test Case

We first present a test case in 2D plane geometry in the linear regime. We have a 2D tank filled with water; the length is $L = 1.0$m, the high $H = 1.0$m. At time $t = 0.0$s the water is at rest with an hydrostatic pressure. We suddenly impose a horizontal acceleration $g = 0.0981$. The characteristic Froude number of the surface waves is $F = 0.001$. In the Figure 5.3 we present successively the computed (full line) and exact position (doted line) of the water free surface on the two sides of the tank. In Case 1 we use a 40×40 initial distribution of particles and SPH Renormalized Godunov-Muscl scheme, in Case 2 SPH Renormalized Standard scheme (numerical viscosity $\alpha = 0.5, \beta = 1.$) with 40×40 particles and in Case 3 SPH Renormalized Godunov-Muscl with 80×80 particles.



(a) RGM 40×40 (b) RS 40×40 (c) RGM 80×40

Figure 5.3. Linear sloshing.

Classical SPH scheme with artificial viscosity are not able to compute reasonably this solution, due to the sensitivity of the result to small perturbations (in practice the displacement of any particle is less than 1/100 of the average distance between any particles). It's almost ill conditioned for SPH Lagrangian type scheme. Our new method offers the possibility to produce such a computation.

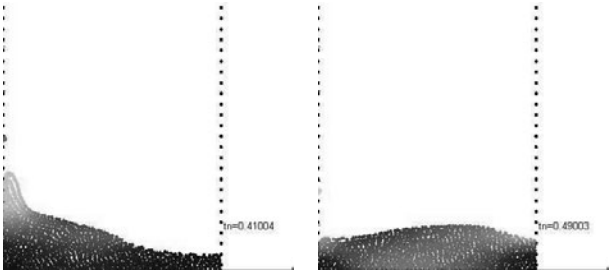
Effect of Dissipation in a Dam Break Type Problem

This test case study the evolution of water in a 2D tank of width L=0.15m. The water is initially located in a "dam" of high 0.15m and width 0.05m. The inner wall is removed at $t = 0.0$s. In Figure 5.4 we compare different simulations

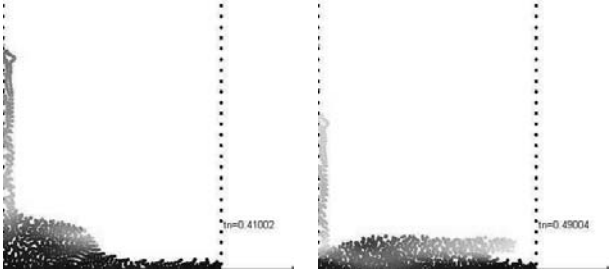
In case 1 and case 2 we use an initial set of 20×60 particles. Case 3 use 40×120 particles. In case 2 and 3 we use our SPH Renormalized Godunov-Muscl (RGM) scheme while in case 1 we use SPH Renormalized Standard (RS) scheme (numerical viscosity $\alpha = 0.5, \beta = 1.$). Clearly SPH RS scheme (case 1) suffers of a great dissipation effect. Comparison of case 2 and 3 proves that even with the thicker mesh the SPH RGM scheme gives accurate results.

Geyser Formation

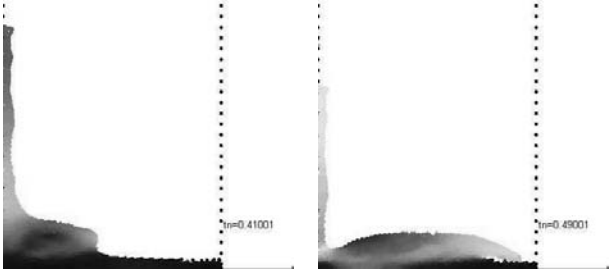
We finally present simulations corresponding to a set of experiments of E. Hopfinger (LEGI Grenoble) dedicated to the formation of geyser and supported by CNES. The experimental set up is made of two cylindrical vessels. The exterior one is filled with some amount of water. At time $t = 0.0$s the inner wall is removed (practically with a given acceleration). We present computation with our new SPH RGM scheme in 2D axisymetric configuration. We investigate the influence of the vertical acceleration a of the wall on the formation of the geyser. In this first sequence of computations the initial high of the water is $H = 0.015$m. The Figure 5.5 represent the free surface configuration at time $t = 0.19$s (left column) and at the occurrence of the maximum wave high (right column)



(a) Case 1 RS 20×60



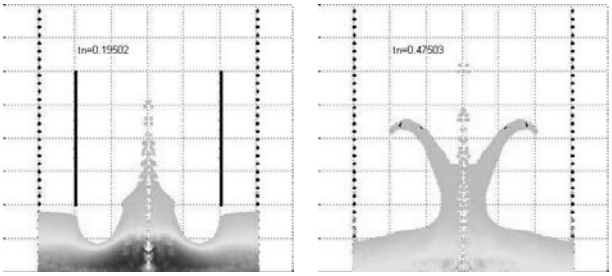
(b) Case 2 RGM 20×60



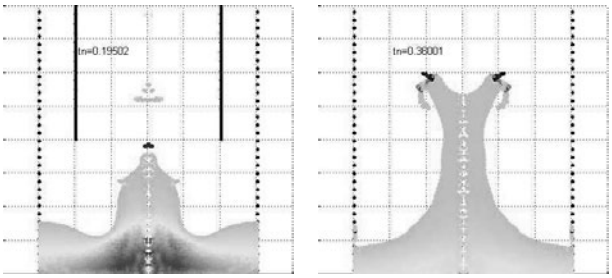
(c) Case 3 RGM 40×120

Figure 5.4. Dam break problem. See Color Plate 8 on page 299.

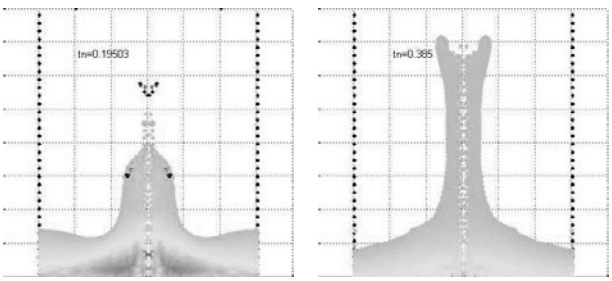
Note that initially we expected that the velocity of the gate was sufficient to proceed as in a dam break simulation. This hypothesis leads to an overestimated high of the geyser and a wrong shape, the introduction of a moving boundary in the numerical computation allow us to give a simple explanation to the mushroom shape formation.



(a) Case 1 a=3.5 ms^{-2}



(b) Case 2 a=7.0 ms^{-2}



(c) Case 3 a=14.0 ms^{-2}

Figure 5.5. Geyser formation ($h = 0.015$m). See Color Plate 9 on page 300.

Note that with a lower filling ratio ($H = 0.005$m) we do not observe this dependency with respect to the wall acceleration. In this case computation with $a = \infty$ (see Figure 5.7) gives results in accordance with experimental results.

Figure 5.6. Experimental results ($h = 0.015$m).

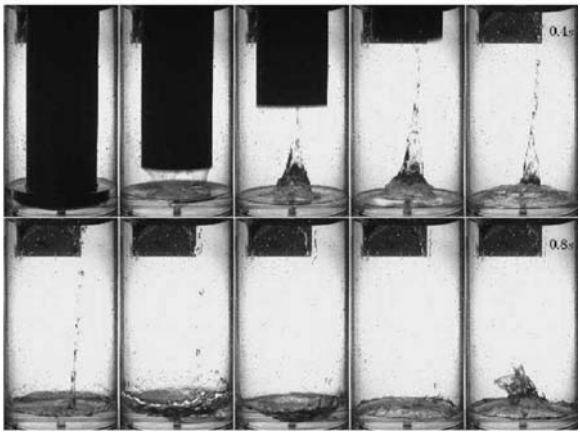


Figure 5.7. Experimental results ($h = 0.005$m).

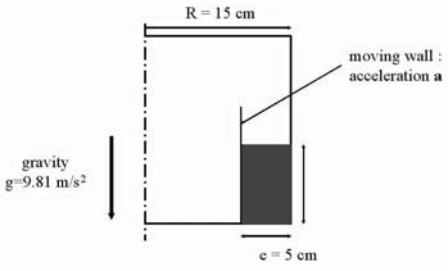
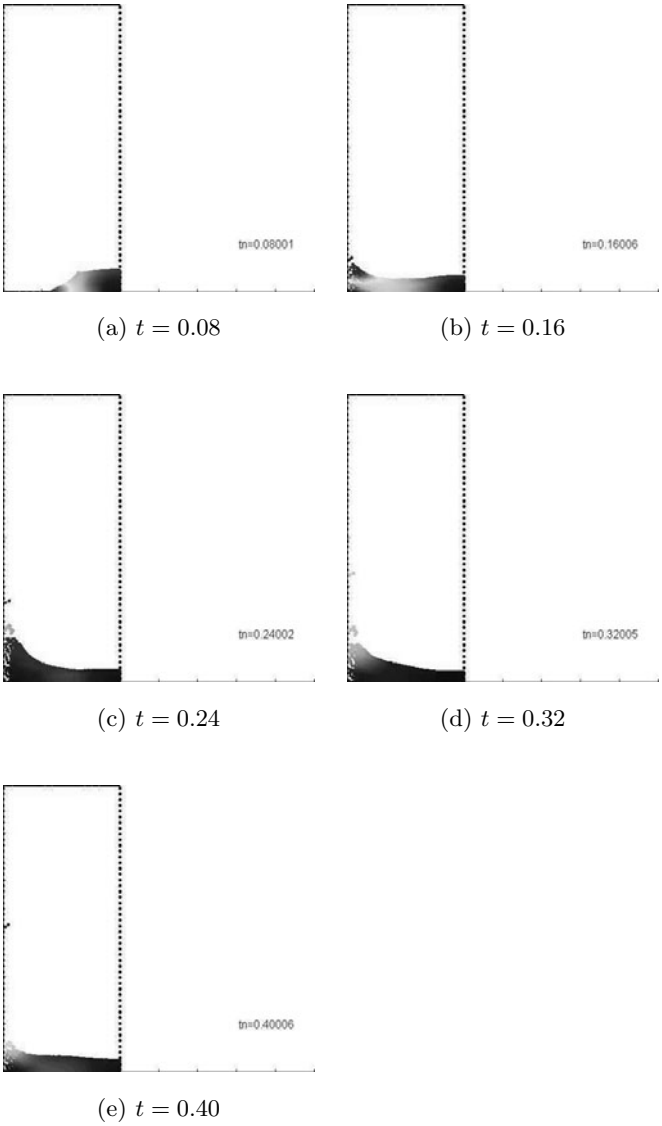


Figure 5.8. Experimental setup.

Note that both on the numerical and experimental results we capture a small precursor prior to the formation of the main wave at the center of the cylinder. At a first glance, the numerical one (which corresponds at list



(a) $t = 0.08$ (b) $t = 0.16$

(c) $t = 0.24$ (d) $t = 0.32$

(e) $t = 0.40$

Figure 5.9. Geyser formation ($h = 0.005$m).

to 5 particles) could be seen as a numerical artifact, our experience is that we always keep it in numerical simulation, and surprisingly we observe it in real experiments. Anyway such details certainly require a finer small scale modeling. This last example shows the ability of the method to compute free surface flow with low dissipation effects.

References

1. Benharbit S., Chalabi A., Vila J.P., Numerical viscosity, and convergence of finite volume Methods for conservation laws with Boundary conditions, S.I.A.M Journal of Num. Analysis, (1995).
2. Ben Moussa B., Lanson N., Vila J.P. "Convergence of Meshless methods for Conservation laws. Applications to Euler Equations.", VII conf. on Hyp. problems, Zurich, 1998.
3. Ben Moussa B., Vila J.P. "Convergence of SPH method for scalar nonlinear conservation laws", SIAM Num. analysis1999
4. Benz W. "Smooth Particle Hydrodynamics: A Review", Harvard-Smithsonian Center for Astrophysics preprint 2884, 1989
5. Benz W., Asphaug A. "Impact Simulations with Fracture: I. Methods and Tests"Icarus, 1993
6. Bonet, J. and Lok, T.-S. L. "Variational and momentum preservation aspects of smooth particle hydrodynamics formulations", Comput. Methods Appl. Mech. Engrg., 180, 97–115, 1999.
7. Gingold R.A., Monahghan, 1977, MNRAS, 181–375
8. Gingold R.A., Monahghan J.J., "Shock simulation by the particle method S.P.H." J.C.P. 52, 374–389 (1983)
9. Harten A., Hyman J.M. "Self adjusting grid methods for one-dimensional hyperbolic conservation laws" J. Comp. Phys (1983)
10. Inutsuka S. "Reformulation of SPH with Riemann solvers" J.C.P. 179, 238–267 (2002)
11. Johnson G. R. "Artificial viscosity effects for SPH computations" Int. J. Impact Engng Vol 18, 5, pp. 477–488, 1996
12. Johnson G.R., Beissel S.R., Normalized Smoothing functions for impact computations, Int. Jour. Num. Methods Eng., (1996)
13. Lax P.D.,Wendroff B., Systems of Conservation Laws CPAM, 13, (1960), pp.217–237.
14. Lanson N. PHD thesis INSA Toulouse 2000
15. Lucy L. 1977 ,Astrono. J., 82,1013
16. Monaghan J.J. "Smooth Particle Hydrodynamics" Annu. Rev. Astron. Astrop., 1992, 30,543–74
17. Monaghan J.J., Kocharyan "SPH simulation of multi-phase flow" Comp. Physics Comm. 87 (1995) 225–235
18. Monaghan J.J., "SPH and Riemann Solvers" J. Comp. Phys. 298–307 (1997)
19. Petschek A.G., L.D. Libersky "Cylindrical Smooth Particle Hydrodynamics" J. Comp. Phys. 109, 76–83 (1993)
20. Randles R.W., Libertsky L.D. "Smoothed Particle Hydrodynamics, Some recent improvements and Applications" Comp. Meth. Appli. Mech. Eng;, 139 (1996) 375–408
21. Raviart P.A. "An analysis of particle methods" in Num. method in fluid dynamics, F. Brezzi ed.. Lecture Notes in Math., vol 1127, Berlin, Springer 1985
22. Van Leer B., Towards the ultimate conservative difference scheme V, J.C.P. , 32, (1979), pp.101–136.
23. Vila JP., Développements récents et Applications des Méthodes particulaires régularisées, ESAIM Proceeding CANUM 97, (1997).

24. Vila JP. "Weighted Particle-Finite Volume Hybrid schemes" in Finite Volumes for Complex Applications ed. F. Benkhaldoun, R. Vilsmeier, HERMES, Paris, 1996.
25. Vila JP. "Particle Weighted methods and Smooth Particle Hydridynamics" M3AS 9, 2, 161–209 (1999)

Discontinuous Radial Basis Function Approximations for Meshfree Methods

Jingxiao Xu* and Ted Belytschko**

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3111, USA.

Abstract Meshfree methods with discontinuous radial basis functions and their numerical implementation for elastic problems are presented. We study the following radial basis functions: the multiquadratic (MQ), the Gaussian basis functions and the thin-plate basis functions. These radial basis functions are combined with step function enrichments directly or with enriched Shepard functions. The formulation is coupled with level set methods and requires no explicit representation of the discontinuity. Numerical results show the robustness of the method, both in accuracy and convergence.

1 Introduction

Radial basis functions (RBF) have been used for a wide range of applications [6, 12, 22, 1], but RBF were only recently applied to solve partial differential equations (PDE): Kansa used RBF to solve the Navier-Stokes equations of fluid dynamics [12], Wendland applied RBF with Galerkin methods to solve partial differential equations [30]. One of the advantages of RBF for the meshfree methods is that it outperforms other interpolation methods for scattered data in accuracy, stability, and the simplicity of the implementation. It takes little time to evaluate the radial basis functions, unlike some other meshless approximations such as moving least squares [3, 9] which require considerable time. However, there are still two issues to be solved [27]: First, multiquadratic RBF and Gaussian RBF are globally supported functions. Globally supported functions result in a full system matrix which is extremely expensive unless special techniques such as multipolar methods are used to solve the equations. Second, their performance for problems with discontinuities in the solution or its gradient degrades significantly.

Compactly supported radial basis functions are now available [29, 31, 7], and the choice of shape parameters has been a hot topic in data fitting [13, 5,

* j-xu1@northwestern.edu

** t-belytschko@northwestern.edu

21]. A recently developed meshfree method based on radial basis functions is the point interpolation meshless method proposed by Wang and Liu [28]. The point interpolation method (PIM) [15] based on Gaussian and multiquadric radial basis functions are applied. In this method the system matrix is sparse and banded.

Several methods for the analysis of discontinuities have been developed both in the area of the meshless methods [20, 10, 26] and finite element methods [18, 4]. In the extended finite element method (XFEM) [18, 2], techniques for approximating discontinuities are based on the local partitions of unity [17, 8]. The goal of extended finite element methods is to expand the approximation function space of standard finite element methods so that they contain features of the solution space that is known. Compared with the standard finite element methods, XFEM can greatly improve accuracy and convergence. Recent developments in the XFEM include the incorporation of level set methods in XFEM [24, 11, 19]. The discontinuity surfaces are defined by implicit functions, i.e. level sets, approximated by finite element interpolants, so that the discontinuity surfaces can be described by nodal values. For moving discontinuities, the level sets are updated by solution of conservation equations or by geometric updates (Ventura et al. [26]), which is very appealing in 3-D problems where the surfaces can be complicated.

Motivated by XFEM, a new meshfree approximation [26] for discontinuous functions and in particular cracks was introduced recently. In this method, a jump function that accounts for the displacement discontinuity along the crack faces is added on nodes whose supports are cut by the discontinuity and a branch function accounting for the neartip crack fields is added on those nodes whose supports are partially cut by the discontinuity. These enrichments can be limited to the nodes surrounding the crack, so the cost is very modest; the methods are easily coupled to level set methods.

In this paper, we develop and examine such methods for introducing discontinuities in radial basis functions. We consider both the enrichment of the radial basis functions by step functions and asymptotic solutions and the combination of radial basis functions with discontinuous Shepard functions. In the former, not all radial basis functions need to be enriched for good accuracy; nevertheless, the treatment of curved discontinuities can be quite awkward. The combined method avoids some of these drawbacks.

The outline of this paper is as follows. In section 2, several different radial basis functions and their completeness conditions are reviewed. In section 3, a method for treating discontinuous functions by radial basis function approximation is proposed; it is coupled with level set functions. The corresponding discrete forms for discontinuous radial basis function approximation are given in section 4. Section 5 reports some elastic solutions to test the accuracy and convergence of the proposed method and illustrates the accuracy of the adopted discontinuous enrichment.

2 Radial Basis Function Approximation

The approximation $u^h(\boldsymbol{x})$ for a function $u(\boldsymbol{x})$ is constructed by a linear combination of translates of a single radially symmetric function $\phi(\|\cdot\|)$ plus a lower-degree polynomial of order m (which is sometimes absent). Let $\boldsymbol{x}_I$ be a set of N scattered nodes (often called particles in meshfree methods) within the domain Ω with boundary Γ. The radial basis function approximation for a vector field, such as a displacement field $u_i(\boldsymbol{x})$, can be written as:

$$u_i^h(\boldsymbol{x}) = c_{i\alpha}x_\alpha + \sum_{I=1}^{N} u_{iI}\phi_I(\boldsymbol{x}) \tag{2.1}$$

where $c_{i\alpha}$ are arbitrary constants and $\phi_I(\boldsymbol{x})$ are the radial basis functions. The range of α is 0 to n^{SD} and $x_\alpha = [1, x, y]$ for 2-D problems. Details on the requirements of the polynomial terms for various radial basis functions for completeness are presented in [6, 16]. Repeated lower case indices are summed in this paper.

For 2-D problems we can write (2.1) as:

$$u_x^h = c_{10} + c_{11}x + c_{12}y + \sum_{I=1}^{N} u_{xI}\phi_I(r) \tag{2.2}$$

$$u_y^h = c_{20} + c_{21}x + c_{22}y + \sum_{I=1}^{N} u_{yI}\phi_I(r) \tag{2.3}$$

where $r = \|\boldsymbol{x} - \boldsymbol{x}_I\|$, $\phi_I(r) \equiv \phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|)$. Some examples of popular choices of the radial basis functions $\phi_I(\boldsymbol{x})$ and their names are given below:

$$\left.\begin{array}{ll} \phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|) = r & \text{Linear} \\ \phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|) = r^2\log(r) & \text{Thin-plate spline} \\ \phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|) = e^{-(r^2/l^2)} & \text{Gaussian} \\ \phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|) = (r^2 + R^2)^q & \text{Multiquadratic} \end{array}\right\} \tag{2.4}$$

where R and q are the shape parameters, l is a characteristic dimension of the domain (the choice of these parameters is presented in [13, 5, 21]).

The partial derivatives for the Gaussian radial basis functions are:

$$\phi_{,x} = -\frac{2r}{l^2}e^{-(r^2/l^2)}r_{,x}, \quad \phi_{,y} = -\frac{2r}{l^2}e^{-(r^2/l^2)}r_{,y} \tag{2.5}$$

where commas denote partial derivatives and:

$$r_{,x} = \frac{(x - x_I)}{r}, \; r_{,y} = \frac{(y - y_I)}{r} \tag{2.6}$$

When a global polynomial basis is included as in equation (2.2,2.3) the coefficients should be constrained so that the approximation is unique. The

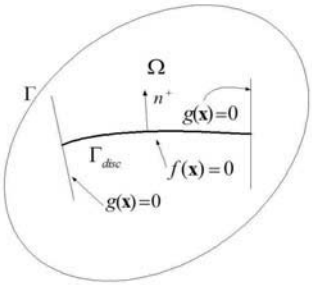


Figure 3.1. Schematic of problem definition and representation discontinuity by two level sets in a two dimensional domain

constraints (the orthogonality conditions) [6] for the radial basis function coefficients of (2.1) are:

$$\sum_{I=1}^{N} u_{iI} = \sum_{I=1}^{N} u_{iI} x_{0I} = 0 \qquad i = 1 \; to \; n^{SD} \tag{2.7}$$

$$\sum_{I=1}^{N} u_{iI} x_{jI} = 0 \qquad i = 1 \; to \; n^{SD}, \; j = 1 \; to \; n^{SD} \tag{2.8}$$

Note that $x_{0I} = y_{0I} = 1$. As stated, for example in [30] and in [27], these conditions are necessary for the completeness of the approximation.

3 Approximation for Discontinuous Functions by Radial Basis Function Interpolation

3.1 Discontinuities in Functions

Consider a two dimensional domain Ω with a boundary Γ, and suppose a discontinuity in the function occurs across the surface denoted Γ_{disc} as shown in Figure 3.1. The discontinuity can be a strong discontinuity (such as a crack) or a weak discontinuity (a discontinuity in gradients). We also consider strong discontinuities of vector fields in which only the tangential component is discontinuous. The position of such a discontinuity will be represented implicitly by a level set function whose zero contours represent the discontinuous surfaces.

The level set method is a general tool for the description of evolving surfaces, and has been used for a wide range of applications. This approach has

been followed in [23, 24], where the evolution of an elliptic plane crack has been analyzed, and in [11, 19] for the general case of 3-D crack evolution. To improve numerical efficiency, these functions are not defined on the whole domain but only on narrow bands surrounding the discontinuities. The vector level set method of Ventura et al. [26] is used here. The update of the nodal values of the vector level set is performed by geometric operations on the data, and no evolution equation is explicitly introduced. One of the advantages of using level set functions to represent the surfaces of discontinuity is that they are described by the nodal values of the level set function and the nodal arrangement can be completely independent of the geometry of the discontinuities.

To represent discontinuous solutions, like the ones in fracture mechanics, the visibility criterion has been proposed in the Element Free Galerkin method [20]. In this approach, the boundaries of the body and any internal line of discontinuity are considered opaque when constructing the shape functions, so the supports are truncated by the lines of discontinuity. Recently a new discontinuity approximation for crack growth by EFG was introduced by Ventura et al. [26]. In this method, the displacement discontinuity along the crack faces and Westergard's solution for the crack tip field are introduced by a local partition of unity. We use a similar method to construct a discontinuous approximation for radial basis function in the following sections, even though radial basis functions are not partitions of unity.

3.2 Modelling of Crack By the Radial Basis Function Interpolation with Level Set Method

Consider the crack shown in Figure 3.2. We define a crack surface implicitly by $f(\boldsymbol{x}) = 0$. To specify the edge of the crack we construct another implicit function $g(\boldsymbol{x})$ orthogonal to $f(\boldsymbol{x})$. The crack edge is given by $f(\boldsymbol{x}) = g(\boldsymbol{x}) = 0$. These functions include the sign of the distance.

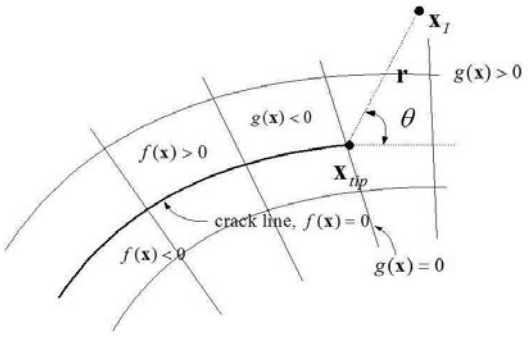


Figure 3.2. A discontinuity represented by level set functions

To construct the discontinuous displacement field, it is split into continuous and discontinuous parts

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}_{cont}(\boldsymbol{x}) + \boldsymbol{u}_{disc}(\boldsymbol{x}) \tag{3.1}$$

where $\boldsymbol{u}_{cont}$ is approximated by standard radial basis functions as in (2.1)

$$u_i^h(\boldsymbol{x}) = c_{i\alpha}x_\alpha + \sum_I u_{iI}\phi_I(\boldsymbol{x}) \tag{3.2}$$

Two methods are studied for introducing the discontinuous enrichment:

1. the discontinuous enrichment is added to the radial basis functions, so that the supports for both the continuous and discontinuous approximations will be global and many of nodes (but not all) need to be enriched.
2. additional Shepard functions with compact support are added at the same nodes.

The Shepard functions satisfy zeroth order completeness, and have a simpler structure than the higher order MLS shape functions. The cubic spline weight function is used in this paper and the support of the weight function is set to be $2.5d_{min}$ and d_{min} is the minimum distance between two neighboring nodes in the scattered node set.

We consider two different enrichments: a step function and a near tip asymptotic field.

The step function part of the approximation is given by

$$\boldsymbol{u}_{disc}(\boldsymbol{x}, t) = \sum_{I \in N_{bis}} \psi_I(\boldsymbol{x})\boldsymbol{a}_I(t)H(f(\boldsymbol{x}))H(g(\boldsymbol{x})) \tag{3.3}$$

where $\boldsymbol{a}_I$ are enrichment variables at node I, $\psi_I(\boldsymbol{x})$ is the RBF or Shepard function for method 1 or method 2, respectively. For method 2, these enrichments are added only for the nodes whose supports are cut by the discontinuity, N_{bis} is the set of nodes whose supports are bisected by the discontinuity. The jump enrichment function is defined by

$$H(x) = \begin{cases} +1 \; if \; x \geq 0 \\ -1 \; if \; x < 0 \end{cases} \tag{3.4}$$

Around the crack tip we approximate the discontinuous displacement field with a branch function which represents the asymptotic displacement field near the crack tip.

$$\boldsymbol{u}_{disc}(\boldsymbol{x}) = \sum_I \psi_I(\boldsymbol{x})(\boldsymbol{a}_{Ii}B_i(\boldsymbol{x})) \tag{3.5}$$

where $\psi_I(\boldsymbol{x})$ is the RBF for method 1 and Shepard functions for method 2.

The branch functions for linear fracture mechanics are defined as

$$B_i(\boldsymbol{x}) = (\sqrt{r}\sin\frac{\theta}{2}, \sqrt{r}\cos\frac{\theta}{2}, \sqrt{r}\sin\frac{\theta}{2}\sin\theta, \sqrt{r}\cos\frac{\theta}{2}\sin\theta) \tag{3.6}$$

where r is the distance of $\boldsymbol{x}$ from the crack tip and θ is given by $\theta = \arctan(f/g)$. These functions span the Westergaard solution for the neartip field. Note that the construction of the cracktip enrichment insures that the discontinuity will coincide with $f(\boldsymbol{x}) = 0$, i.e. the crack, away from the cracktip.

3.3 Discontinuities in Gradient

In this section, we develop a similar methodology for modelling the discontinuity in the gradient of the approximation. These discontinuities occur at interfaces between materials and different phases of materials. The approximation is constructed as follows:

$$\boldsymbol{u}^h(\boldsymbol{x}) = \boldsymbol{c}_\alpha x_\alpha + \sum_I \boldsymbol{u}_I \phi_I(\boldsymbol{x}) + \sum_J \boldsymbol{q}_J \Upsilon_J(\boldsymbol{x}) \tag{3.7}$$

where $\phi(\boldsymbol{x})$ are the standard radial basis functions, $\boldsymbol{q}_J$ are the amplitude parameters of the jumps and $\Upsilon_J(\boldsymbol{x})$ are the enrichment functions.

One form of the enrichment functions we will use is introduced in [14].

$$\Upsilon_J(f) = \mid f(\boldsymbol{x}) \mid - \sum_I \phi_I \mid f_I \mid \tag{3.8}$$

This enrichment function introduces two extra degrees of freedom for 2-D problems, and it does not provide compact supports for the lines of discontinuity other than straight lines.

Sukumar [24] and Belytschko et al. [4] introduced the absolute value of level set functions for the enrichment for discontinuous gradient. This enrichment can be written as

$$\Upsilon_J(\boldsymbol{x}) = \phi_J(\boldsymbol{x})|f(\boldsymbol{x})| \tag{3.9}$$

Since

$$\begin{aligned} [\![\Upsilon_{J,n}(f \neq 0)]\!] &= \phi_J [\![|f(\boldsymbol{x})|]\!] \\ &= \phi_J H(f(\boldsymbol{x})) \end{aligned} \tag{3.10}$$

the displacement derivatives are discontinuous across interfaces.

3.4 Tangential Discontinuities

Consider a vector function $\boldsymbol{u}(\boldsymbol{x})$, such as a displacement, with a tangential discontinuity on $f(\boldsymbol{x}) = 0$. The unit normal to the line of discontinuity is given by

$$\boldsymbol{e}_n = \frac{\nabla f}{\|\nabla f\|} \tag{3.11}$$

Although a signed distance function should have a unit gradient, we normalize it here since this should be done in a computation. The discontinuity in the tangential component in two dimension is obtained by the displacement field

$$\boldsymbol{u}(\boldsymbol{x}) = \sum_I \phi_I(\boldsymbol{u}_I + a_I \boldsymbol{e}_t(\boldsymbol{x}) H(f(\boldsymbol{x}))) + \boldsymbol{c}_\alpha x_\alpha \tag{3.12}$$

where $\boldsymbol{e}_t = \boldsymbol{e}_z \times \boldsymbol{e}_n$ is a vector in the tangent direction and $\boldsymbol{e}_z$ is the unit vector along the z axis.

4 Discrete Equations for Discontinuous Radial Basis Function Approximation

Consider a two-dimensional linear elasto-static problem in the domain Ω bounded by Γ. The equilibrium equation is

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{b} = 0 \quad \text{in} \quad \Omega \tag{4.1}$$

where $\boldsymbol{\sigma}$ is the stress tensor corresponding to the displacement field $\boldsymbol{u}$ and $\boldsymbol{b}$ is the body force. The boundary conditions are:

$$\begin{aligned} \boldsymbol{\sigma} \cdot \boldsymbol{n} &= \bar{\boldsymbol{t}} \text{ on } \Gamma_t \\ \boldsymbol{u} &= \bar{\boldsymbol{u}} \text{ on } \Gamma_u \end{aligned} \tag{4.2}$$

where the superposed bar denotes a prescribed boundary value and $\boldsymbol{n}$ is the unit outward normal to the domain. Γ_t is the boundary where the traction boundary conditions are applied and Γ_u is the boundary where the displacement boundary conditions are applied.

Both the displacement boundary condition and the orthogonality conditions in equations (2.7), (2.8) are imposed by the Lagrange multiplier method. Then the solution of the equilibrium equation is equivalent to finding the stationary points of the constrained potential energy:

$$\begin{aligned} W_L &= W + \textstyle\int_{\Gamma_u} \boldsymbol{\lambda} \cdot (\boldsymbol{u} - \bar{\boldsymbol{u}}) d\Gamma + \sum_I \gamma_{i\alpha} u_{iI} x_{\alpha I} \\ &= W + \textstyle\int_{\Gamma_u} \boldsymbol{\lambda} \cdot (\boldsymbol{u} - \bar{\boldsymbol{u}}) d\Gamma + \gamma_{10} \sum_I u_{xI} + \gamma_{11} \sum_I u_{xI} x_I + \gamma_{12} \sum_I u_{xI} y_I \\ &\quad + \gamma_{20} \textstyle\sum_I u_{yI} + \gamma_{21} \sum_I u_{yI} x_I + \gamma_{22} \sum_I u_{yI} y_I \end{aligned} \tag{4.3}$$

where $\boldsymbol{\lambda} = \{\lambda_I\}$ and $\gamma_{i\alpha}$ are the Lagrange multipliers and the subscript L indicates the modified potential. The elastic potential is given by

$$W = \frac{1}{2} \int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}) : C : \boldsymbol{\varepsilon}(\boldsymbol{u}) d\Omega - \int_\Omega \boldsymbol{b} \cdot \boldsymbol{u} d\Omega - \int_{\Gamma_t} \boldsymbol{u} \cdot \bar{\boldsymbol{t}} d\Gamma \tag{4.4}$$

At the equilibrium points the first variation of W_L vanishes, so

$$\delta W_L = \int_\Omega \delta u_{i,j} C_{ijkl} u_{k,l} d\Omega - \int_\Omega \delta u_i b_i d\Omega - \int_{\Gamma_t} \delta u_i \bar{t}_i d\Gamma + \int_{\Gamma_u} \delta\lambda_i (u_i - \bar{u_i}) d\Gamma$$
$$+ \int_{\Gamma_u} \delta u_i \lambda_i d\Gamma + \delta\gamma_{i\alpha} \sum_I u_{iI} x_{\alpha I} + \sum_I \delta u_{iI} x_{\alpha I} \gamma_{i\alpha} = 0 \tag{4.5}$$

For simplicity, we define matrices $\boldsymbol{U}$ and $\boldsymbol{\Phi}$ by

$$\boldsymbol{U} = \left\{ \begin{array}{c} \{\boldsymbol{u_I}\} \\ \{\boldsymbol{a_I}\} \end{array} \right\} \tag{4.6}$$

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \left\{ \begin{array}{c} \{\boldsymbol{\phi}_I(\boldsymbol{x})\} \\ \{\boldsymbol{\phi}_I(\boldsymbol{x}) \Upsilon(\boldsymbol{x})\} \end{array} \right\} \tag{4.7}$$

where $\boldsymbol{a}$ is the vector associated with the unknowns from the enrichment functions and $\Upsilon(\boldsymbol{x})$ is the enrichment function, which can be $H(f(\boldsymbol{x}))$, etc.

The Lagrange multipliers $\boldsymbol{\lambda}$ and $\gamma_{i\alpha}$ are approximated by:

$$\lambda_i(\boldsymbol{x}) = N_I(s)\lambda_{Ii}, \quad \gamma_{i\alpha}(\boldsymbol{x}) = \gamma_{i\alpha}(\boldsymbol{x}) \tag{4.8}$$

where $N_I(s)$ are the finite element shape functions, and s is the arc length along the boundary. The final discrete equations can be obtained by substituting the approximation of $\boldsymbol{u}$ and equation (4.8) into (4.5). From the arbitrariness of the variation, we can obtain the matrix equation:

$$\left[\begin{array}{cccc} \boldsymbol{K}_{UU} & \boldsymbol{K}_{Uc} & \boldsymbol{G}_{RB} & \boldsymbol{G}_U \\ \boldsymbol{K}_{cU} & \boldsymbol{K}_{cc} & \boldsymbol{0} & \boldsymbol{G}_c \\ \boldsymbol{G}_{RB}^T & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{G}_U^T & \boldsymbol{G}_c^T & \boldsymbol{0} & \boldsymbol{0} \end{array} \right] \left\{ \begin{array}{c} \boldsymbol{U} \\ \boldsymbol{c} \\ \boldsymbol{\gamma} \\ \boldsymbol{\lambda} \end{array} \right\} = \left\{ \begin{array}{c} \boldsymbol{f}_U \\ \boldsymbol{f}_c \\ \boldsymbol{0} \\ \boldsymbol{q}_U \end{array} \right\} \tag{4.9}$$

where :

$$\boldsymbol{K}_{UU} = \int_\Omega \Phi_{I,j} C_{ijkl} \Phi_{J,l} d\Omega \tag{4.10}$$

$$\boldsymbol{K}_{Uc} = \int_\Omega \Phi_{I,j} C_{ijkl} x_{\beta,l} d\Omega \tag{4.11}$$

$$\boldsymbol{K}_{cc} = \int_\Omega x_{\alpha,j} C_{ijkl} x_{\beta,l} d\Omega \tag{4.12}$$

$$\boldsymbol{G}_U = \int_{\Gamma_u} \Phi_I N_J d\Gamma \tag{4.13}$$

$$\boldsymbol{G}_c = \int_{\Gamma_u} x_\alpha N_J d\Gamma \tag{4.14}$$

$$\boldsymbol{q}_U = \int_{\Gamma_u} N_I \bar{u_i} d\Gamma \tag{4.15}$$

$$\boldsymbol{f}_c = \int_{\Gamma_t} x_\alpha \bar{t} d\Gamma + \int_\Omega x_\alpha b_i d\Omega \tag{4.16}$$

$$\boldsymbol{f}_U = \int_\Omega \Phi_I b_i d\Omega + \int_{\Gamma_t} \Phi_I \bar{t}_i d\Gamma \tag{4.17}$$

$$\boldsymbol{G}_{RB}^T = [\boldsymbol{X_1} \;\; \boldsymbol{X_2} \;\; \boldsymbol{X_3} \;\; \ldots] \tag{4.18}$$

$$\boldsymbol{X_I} = \begin{bmatrix} 1 & 0 & x_I & 0 & y_I & 0 \\ 0 & 1 & 0 & x_I & 0 & y_I \end{bmatrix} \tag{4.19}$$

$$\boldsymbol{c}^T = [c_{10} \;\; c_{20} \;\; c_{11} \;\; c_{21} \;\; c_{12} \;\; c_{22}] \tag{4.20}$$

$$\boldsymbol{\gamma}^T = [\gamma_{10} \;\; \gamma_{20} \;\; \gamma_{11} \;\; \gamma_{21} \;\; \gamma_{12} \;\; \gamma_{22}] \tag{4.21}$$

$$\boldsymbol{\lambda}^T = \begin{bmatrix} \lambda_{u_x 1} & \lambda_{u_y 1} & \lambda_{u_x 2} & \lambda_{u_y 2} & \ldots \end{bmatrix} \tag{4.22}$$

For plane stress problems

$$\boldsymbol{C} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \tag{4.23}$$

5 Example

We first consider a problem in linear elastostatics without discontinuities using radial basis function interpolation. This example examines the accuracy of the radial basis function interpolation. The Gaussian radial basis function (it is globally supported) is used and the effect of parameter l on the results is studied.

5.1 Timosheko Beam Problem

Consider a beam of length L subjected to a parabolic traction at the free end as shown in Figure 5.3. The beam has a unit thickness and plane stress is assumed. The closed-form solution is given by Timoshenko and Goodier [25]:

$$\begin{aligned} u_x &= \tfrac{Py}{6EI}\left[(6L-3x)x + (2+\nu)(y^2 - \tfrac{D^2}{4})\right] \\ u_y &= -\tfrac{Py}{6EI}\left[3\nu y^2(L-x) + (4+5\nu)\tfrac{D^2 x}{4} + (3L-x)x^2\right] \end{aligned} \tag{5.1}$$

where the moment of inertia I of the beam is given by $I = D^3/12$. The corresponding stresses are

$$\sigma_x = \frac{P(L-x)y}{I}, \;\; \tau_{xy} = -\frac{p}{2I}\left[\frac{D^2}{4} - y^2\right], \;\; \sigma_y = 0 \tag{5.2}$$

The parameters of the beam are as follows: $E = 3.0 \times 10^7 \; kPa$, ν=0.3, D=12 m , L=48 m and $P = 1000 \; kN$. We use regularly distributed nodes and the background mesh for numerical integration as in [9]. We define a relative energy error by

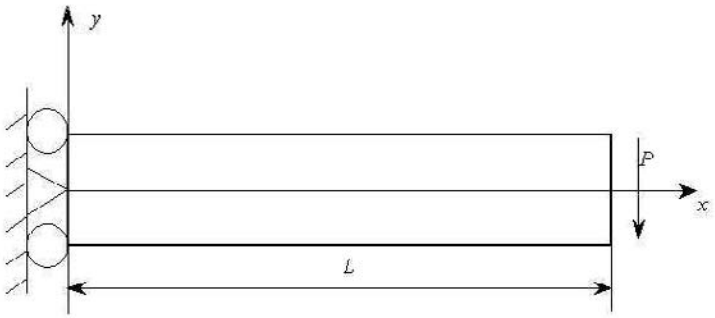


Figure 5.3. Two dimensional cantilever beam example

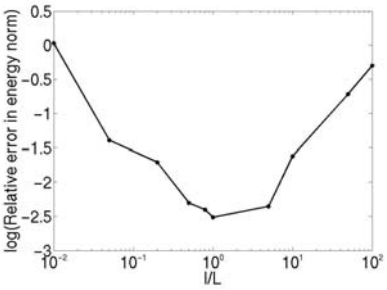


Figure 5.4. Effect of shape parameter l on relative error in energy norm for beam problem

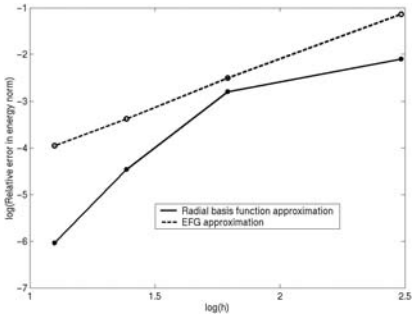


Figure 5.5. Convergence in relative energy norm for the two-dimensional elastic beam problem

$$\textit{Relative energy error} = \frac{\left\|\boldsymbol{u} - \boldsymbol{u}^h\right\|}{\|\boldsymbol{u}\|} = \frac{(\int_\Omega (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^h)\boldsymbol{C}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^h)d\Omega)^{1/2}}{(\int_\Omega \boldsymbol{\varepsilon}\boldsymbol{C}\boldsymbol{\varepsilon}d\Omega)^{1/2}} \quad (5.3)$$

where $\boldsymbol{\varepsilon}$ is the exact strain and $\boldsymbol{\varepsilon}^h$ is the strain from the numerical results.

The effect of parameter l in Gaussian radial basis function on the accuracy of the results is quite marked. Figure 5.4 shows the effect of the parameter l on the relative error in the energy norm for a 16 by 4 background mesh for the beam problem. Reasonable results are obtained when l/L (L is the length of specimen) ranges between 0.1 and 10; the best result is obtained when l/L equals 1. When the value of l/L is too small or close to zero, the effective domain of shape function is very large (the effective domain will be the entire domain when $l/L = 1$); the values of shape function at each node are close to 1. Results deteriorate for smaller values of l/L. The bigger the value of l/L, the smaller the effective domain of the shape function; the shape function tends to the Dirac delta function as l/L increases.

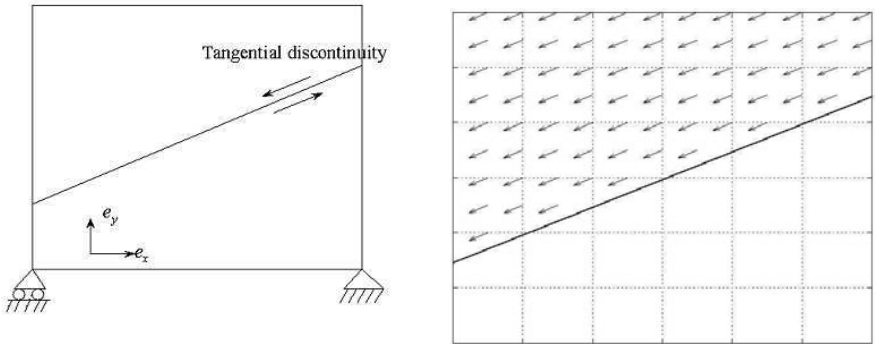
Figure 5.5 compares of the accuracy of the radial basis functions with EFG with $l/L = 1$. The accuracy of radial basis function interpolation exceeds the accuracy of the EFG interpolation. Radial basis function interpolation converges more quickly than the EFG interpolation for the coarse mesh (theoretically the convergence rate for the Gaussian radial basis functions should be exponential). But for the finer mesh, the Gaussian radial basis function becomes unstable and does not converge at all.

5.2 Square Block with Sliding Interface

Consider a block subdivided in two by an inclined frictionless sliding interface as shown in Figure 5.6(a), where we introduce a straight line tangential discontinuity in the two-dimensional displacement field. The straight line is defined by $\boldsymbol{a}_\beta x_\beta = 0$ and $\boldsymbol{n} = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|}$; $\boldsymbol{a} = [a_0, a_1, a_2]$.

The unit tangent vector is given by

$$\boldsymbol{e}_t = \frac{2}{\sqrt{5}}\boldsymbol{e}_x + \frac{1}{\sqrt{5}}\boldsymbol{e}_y \tag{5.4}$$



(a) (b)

Figure 5.6. a) Straight line tangential discontinuity in a square block; b) The lowest eigenmode of the square block with a tangential discontinuity

The Gaussian radial basis functions are used in this case and the parameter $l/L = 1$ (L is the length of specimen). Figure 5.6(b) shows the lowest eigenmode of the system. The upper part of the block moves in a translational rigid mode in the $\boldsymbol{t}$ direction. The smallest eigenvalue is $\lambda_1 = 5.626 \times 10^{-10}$, i.e. zero to machine precision.

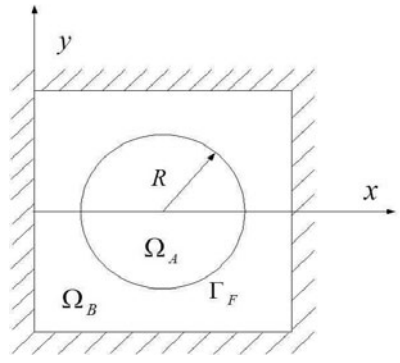


Figure 5.7. Model of journal bearing

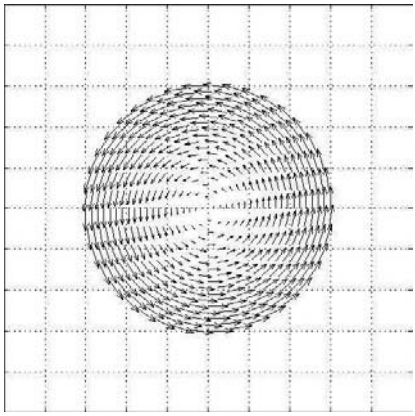


Figure 5.8. Displacement mode corresponding to the lowest eigenvalue for the journal bearing problem

5.3 Journal Bearing

The model is shown in Figure 5.7. Four sides of the specimen are fixed and on the circle marked Γ_F a discontinuity in the tangential displacement is introduced by adding enrichment based on the nodal values of tangent. The enriched displacement approximation is

$$\boldsymbol{u}(\boldsymbol{x}) = \sum_I \phi_I(\boldsymbol{x})(\boldsymbol{u}_I + \boldsymbol{a}_I H(f(\boldsymbol{x}))\boldsymbol{v}(\boldsymbol{x}_I)) + \boldsymbol{c}_\alpha x_\alpha \tag{5.5}$$

where

$$\boldsymbol{v} = -(y - y_0)\boldsymbol{e}_x + (x - x_0)\boldsymbol{e}_y \tag{5.6}$$

where x_0, y_0 are the co-ordinates of the center of the shaft of radius R and

$$f(\boldsymbol{x}) = (x - x_0)^2 + (y - y_0)^2 - R^2 \tag{5.7}$$

We examine how closely the model can capture the lowest eigenvalue and the eigenvector corresponding to rigid-body rotation of the shaft Ω_A inside the bearing Ω_B. The lowest eigenvalue should be zero since the shaft Ω_A should be free to rotate.

In this example, the specimen is 4.8 by 4.8, and the radius of the circular shaft $R = 0.3$. Young's modulus $E = 1$, Poisson's ratio $\nu = 0.3$. Gaussian radial basis functions are used and the parameter $l/L = 1$. The lowest eigenvalue is 1.333×10^{-19}, which is zero to machine precision. The corresponding eigenvector is shown in Figure 5.8; it represents rigid-body rotation of the shaft very accurately.

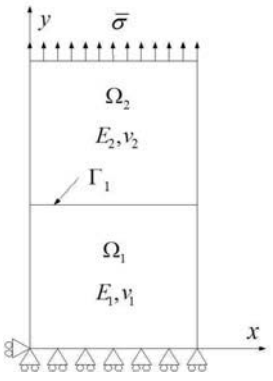


Figure 5.9. Bimaterial bar problem

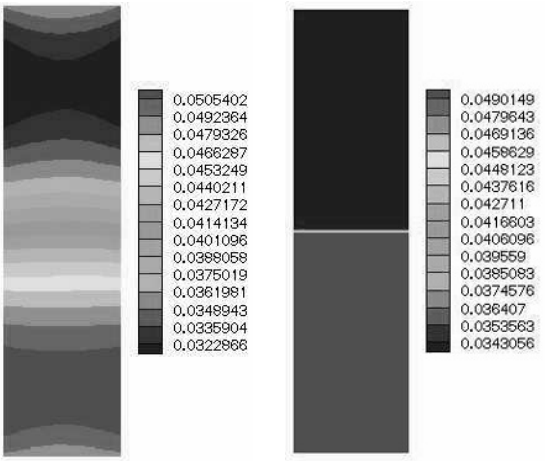
5.4 Bimaterial Bar

Consider a bimaterial bar ($\Omega = \Omega_1 \cup \Omega_2$) as shown in Figure 5.9, with length $L = 12$, height $h = 48$ with material moduli E_1 and Poisson's ratio ν_1 in Ω_1 and E_2, ν_2 in Ω_2. The material constants were chosen so that there is no singularity at the edge of the interface; this condition is $\frac{E_1}{\nu_1} = \frac{E_2}{\nu_2}$ for plane stress. The interface Γ_1 is located in the middle of specimen. For the mixed boundary problem with $u_y = 0$ at $y = 0$ and $\sigma_y = \overline{\sigma}$ at $y = h$ and no body force, the exact displacement solution is:

$$u_y(y) = \begin{cases} \frac{\overline{\sigma}}{E_1} y & 0 \le y \le h/2 \\ \frac{\overline{\sigma}}{E_2}(y - h/2) + \frac{\overline{\sigma} h}{E_1 2} & \frac{h}{2} \le y \le h \end{cases} \tag{5.8}$$

Since the exact displacement solution is piecewise linear for the bimaterial bar and the interface is a straight line, we use the enrichment functions from equation (3.8) which introduces only two extra degrees of freedom globally. Let Young's modulus and Poisson's ratios in Ω_1 and Ω_2 to be 20, 0.2 and 30, 0.3 respectively. We use the same radial basis functions as in the last example and the parameter l/L is set to be 1.

Figure 5.10 shows contour plots of ε_{yy} for both the standard radial basis function solution without enrichment and radial basis function solution with enrichment. As we can see from the figure, the solution with the standard radial basis function approximation without enrichment captures the solution quite well but it exhibits oscillations in the gradient field near the interface. The enriched radial basis function approximation gives the exact solution without oscillations. The relative errors in the energy norm for the bimaterial bar problem are 4.20×10^{-6}(almost zero) for the enriched radial basis functions and 0.0078 for the radial basis functions without enrichment.



(a) (b)

Figure 5.10. Plots of strain ε_{yy} for the bimaterial bar. (a) Standard RBF approximation without enrichment, (b) Standard RBF approximation with enrichment

5.5 Bimaterial Boundary-value Problem

In this example, an enriched radial basis function approximation for the elastostatic response of a circular material inhomogeneity under radially symmetric loading as show in Figure 5.11 is considered. The material properties are constant in Ω_1 and Ω_2, but there is a discontinuity in the material constants across the interface Γ_1 $(r = a)$. The Lamé constants for domain Ω_1, Ω_2 are $\lambda_1 = \mu_1 = 0.4$ and $\lambda_2 = 5.7692$, $\mu_2 = 3.8461$, respectively. These correspond to $E_1 = 1$, $\nu_1 = 0.25$, and $E_2 = 10$, $\nu_2 = 0.3$. We impose a linear displacement field: $u_x = x$, $u_y = y$ $(u_r = r, u_\theta = 0)$ on the outside circle. Navier's equation in polar coordinates reduces to:

$$\frac{d}{dr}\left[\frac{1}{r}\frac{d}{dr}(ru_r)\right] = 0 \tag{5.9}$$

By considering displacement and traction continuity across the interface, the exact displacement solution can be written as:

$$u_r(r) = \begin{cases} [(1 - \frac{b^e}{a^2})\alpha + \frac{b^2}{a^2}]r & 0 \le r \le a, \\ (r - \frac{b^2}{r})\alpha + \frac{b^2}{r} & a \le r \le b, \end{cases} \tag{5.10}$$

$$u_\theta = 0, \tag{5.11}$$

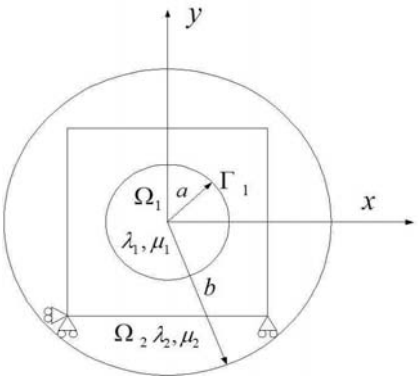


Figure 5.11. Bimaterial boundary-value problem

where

$$\alpha = \frac{(\lambda_1 + \mu_1 + \mu_2)b^2}{(\lambda_2 + \mu_2)a^2 + (\lambda_1 + \mu_1)(b^2 - a^2) + \mu_2 b^2} \tag{5.12}$$

The radial (ε_{rr}) and hoop ($\varepsilon_{\theta\theta}$) strains are given by:

$$\varepsilon_{rr}(r) = \begin{cases} (1 - \frac{b^2}{a^2})\alpha + \frac{b^2}{a^2}, \ 0 \le r \le a, \\ (1 + \frac{b^2}{r^2})\alpha - \frac{b^2}{r^2}, \ a \le r \le b, \end{cases} \tag{5.13}$$

$$\varepsilon_{\theta\theta}(r) = \begin{cases} (1 - \frac{b^2}{a^2})\alpha + \frac{b^2}{a^2}, \ 0 \le r \le a, \\ (1 - \frac{b^2}{r^2})\alpha + \frac{b^2}{r^2}, \ a \le r \le b, \end{cases} \tag{5.14}$$

The radial and hoop stress are:

$$\begin{aligned} \sigma_{rr}(r) &= 2\mu\varepsilon_{rr} + \lambda(\varepsilon_{rr} + \varepsilon_{\theta\theta}), \\ \sigma_{\theta\theta}(r) &= 2\mu\varepsilon_{\theta\theta} + \lambda(\varepsilon_{rr} + \varepsilon_{\theta\theta}), \end{aligned} \tag{5.15}$$

In the numerical model, we consider a square plate (2×2) with a circular inclusion (Ω_1) of radius $a = 0.4$. The exact tractions from equation (5.15) with $b = 2.0$ are imposed on the four sides of square plate, and three degrees of freedoms are fixed on two nodes to remove rigid body motions. In this case, the enrichment (3.9) is used to capture the strain discontinuity across the circular interface. The level set function is given by $f(\boldsymbol{x}) = \sqrt{x^2 + y^2} - a^2$. A convergence study is conducted by using equally spaced rectangular arrangements of nodes: 2×2, 4×4, 8×8, 16×16. We include the coarse models to show that the radial basis function can provide very good results with very few unknowns.

We first consider two cases: (a) radial basis functions with enrichment on all nodes. (b) element free Galerkin method with enrichment. These results are compared with the results from X-FEM [24] in Figure (5.12). To eliminate the

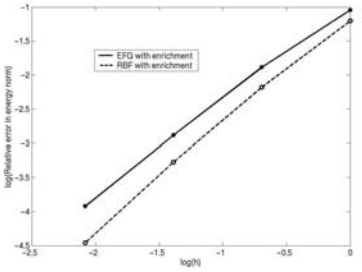


Figure 5.12. Rates of convergence in energy norm for the bimaterial boundary-value problem

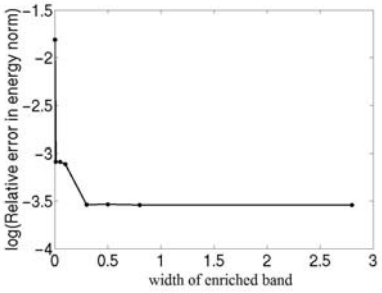


Figure 5.13. Accuracy of solutions for 8x8 nodes for different widths of enriched band

effects of the radial basis function shape parameters, we use thin-plate spline radial basis functions; which have no shape parameters. Both the enriched radial basis function interpolation and the enriched element free Galerkin method have similar accuracy and rates of convergence ($R = 1.56$ for enriched radial basis function interpolation and $R = 1.43$ for the enriched element free Galerkin method). The accuracy and the rates of convergence of both enriched methods are significantly better than those without enrichment, and also much better than those from X-FEM with linear 3-node elements with the same enrichment functions ($R = 0.75$ in [24]). However, the enriched RBF method does not achieve the accuracy seen in the previous examples. We do not know the source of this error.

The enrichment schema for the globally supported radial basis function approximation is quite expensive. An interesting question arises: how many nodes need to be enriched with the discontinuity to obtain good accuracy? We investigated this question by enriching only those nodes that fall in a band about the interface. Figure 5.13 shows the accuracy versus the width of the band of enriched nodes for a 8x8 model. We can see that there is a big increase in accuracy with enrichment only on those nodes that are closest to the interface as compared to no enrichment. Subsequently a small improvement in accuracy is seen as we increase the width of the enriched band of nodes.

5.6 Static Mode I Crack

A mode I crack problem for which the exact solution is known can be constructed by using the well-known near-tip stress field and prescribing the corresponding tractions at the boundary. The rigid body motions are suppressed by constraints at two nodes as illustrated in Figure (5.14). A square patch with unit sides and a crack of length 0.5 is considered. This problem is used to compare the performance of the radial basis functions against the element free Galerkin method. The near-tip stress field for a mode I crack is

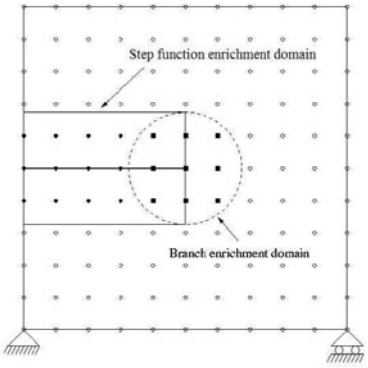


Figure 5.14. Discretization and enriched nodes for crack. Filled circles and squares are jump and branch enriched nodes, respectively

$$\sigma_x = \frac{K_{\mathrm{I}}}{\sqrt{2\pi r}} \cos(\vartheta/2)\,[1 - \sin(\vartheta/2)\sin(3\vartheta/2)] \tag{5.16}$$

$$\sigma_y = \frac{K_{\mathrm{I}}}{\sqrt{2\pi r}} \cos(\vartheta/2)\,[1 + \sin(\vartheta/2)\sin(3\vartheta/2)] \tag{5.17}$$

$$\tau_{xy} = \frac{K_{\mathrm{I}}}{\sqrt{2\pi r}} \sin(\vartheta/2)\cos(\vartheta/2)\cos(3\vartheta/2) \tag{5.18}$$

Three approaches are studied here: case 1, radial basis functions in both continuous and discontinuous approximations; case 2, enrichment with discontinuous Shepard functions combined with radial basis functions for the continuous approximation; case 3, the element free Galerkin method for both continuous and discontinuous parts. The Shepard functions with the supports of $2.5d_{min}$ (d_{min} is the minimum distance between two neighboring nodes) are used for the second case. Gaussian radial basis functions with global support are used in this example; $l/L = 1$. For the normal element free Galerkin method, the sizes of the support domains are $2.5d_{min}$. The traction is prescribed so that the stress intensity factor $K_{\mathrm{I}} = 1.0\ \mathrm{Nm}^{-3/2}$, and the relative error in energy is computed according to equation (5.3). The jump function is introduced on those nodes whose supports are cut by the crack in the second and third cases, while around the crack tip we enrich the approximation with a branch function (see equation (3.6)) as shown in Figure (5.14).

All three methods show good accuracy. Figure (5.17) and Figure (5.16) show the stress contours obtained by the enriched radial basis functions (case 1) and the enriched element free Galerkin method (case 3). As we can see, the stress contours from the enriched radial basis functions are smoother than those of the enriched element free Galerkin method. Figure (5.15) shows the convergence for the three methods. The convergence rates for the three meth-

ods are: $R = 0.83$ for method 1; $R = 0.8240$ for method 2; $R = 0.832$ for method 3, so the convergence rates are very similar but the results from the enriched radial basis functions are more accurate but at more computational cost than the other two cases.

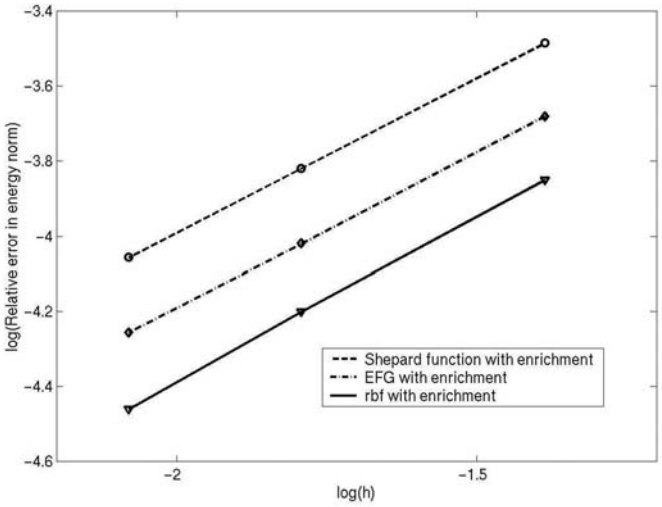


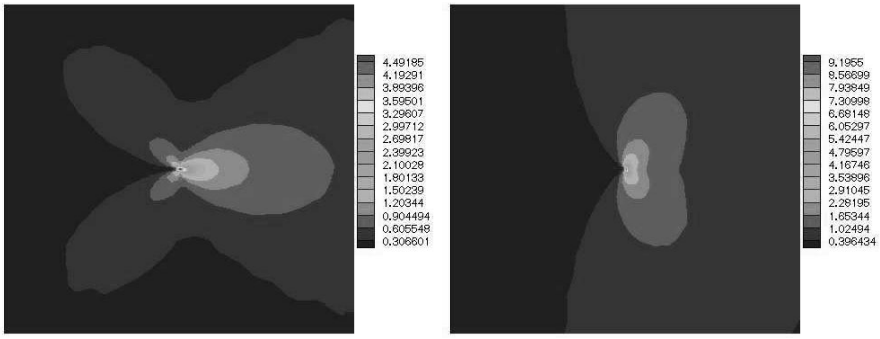
Figure 5.15. Relative errors in energy for mode I crack for various enrichment methods

6 Conclusions

Radial basis function methods for discontinuous approximations and their numerical implementation for the elastic problems have been presented. Methods for both discontinuities in the approximation and the gradient of the approximation have been studied. The method is coupled with level set methods and thus requires no explicit representation of the discontinuity surface.

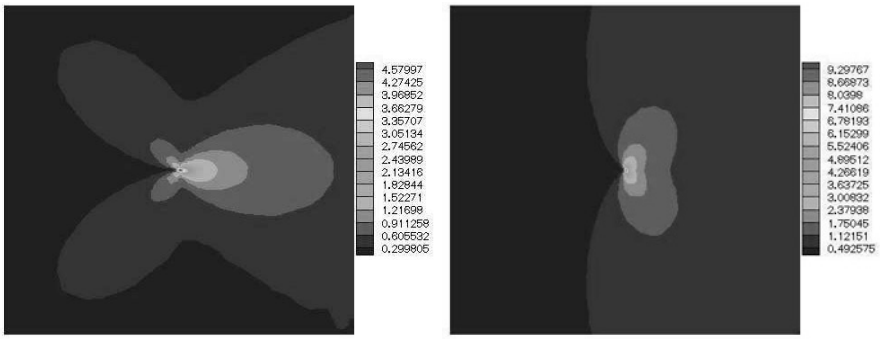
Since the radial basis functions considered here have global support, the enrichment with discontinuous functions is not as straightforward as for compactly supported approximants such as EFG. Therefore two methods have been studied to alleviate the computational burden:

1. adding the discontinuous enrichment only to nodes in the vicinity of the discontinuity.
2. combining continuous radial basis function approximations with discontinuously enriched Shepard functions with local support.



(a) (b)

Figure 5.16. Stress contours for static crack by the enriched EFG. (a) σ_{xx}, (b) σ_{yy}. See Color Plate 10 on page 301.



(a) (b)

Figure 5.17. Stress contours for static crack by the enriched RBF. (a) σ_{xx}, (b) σ_{yy}. See Color Plate 11 on page 301.

For straight discontinuities, exceptional accuracy has been obtained. For curved discontinuities in the gradient, as in the circular inclusion problem, the accuracy and rates of convergence are quite a bit below the expected potential of the method. We do not understand the source of the error yet. Difficulties with gradient enrichment have also been seen in finite element methods and there some remedies have been developed [8].

One of the advantages for the radial basis functions for meshfree methods is that they outperform the other interpolation methods for scattered data sets in accuracy, stability, and the simplicity of the implementation. Also its

high convergence rate (MQ radial basis functions converge exponentially for smooth problems) and higher-order continuity make radial basis functions very promising for meshfree methods.

A drawback for most radial basis functions is that they are not compactly supported. Globally supported radial basis functions are highly expensive for large models unless methods such as multipolar methods are used. We are now studying these applications.

Acknowledgement. The support of the Office of Naval Research and the Army Research Office is gratefully acknowledged.

References

1. R. K. Beatson, J. B. Cherrie, and D. L. Ragozin, *Fast evaluation of radial basis functions: Methods for four-dimensional polyharmonic splines*, Siam J. Math Anal., 32 (2001), pp. 1272–1310.
2. T. Belytschko and T. Black, *Elastic crack growth in finite elements with minimal remeshing*, International Journal for Numerical Methods in Engineering, 45 (1999), pp. 601–620.
3. T. Belytschko, Y. Y. Lu, and L. Gu, *Element-free Galerkin methods*, International Journal for Numerical Methods in Engineering, 37 (1994), pp. 229–256.
4. T. Belytschko, N. Moës, S. Usui, and C. Parimi, *Arbitrary discontinuities in finite elements*, International Journal for Numerical Methods in Engineering, 50 (2001), pp. 993–1013.
5. R. E. Carlson and T. A. Foley, *The parameter r^2 in multiquadric interpolation*, Computers and Mathematics with Applications, 21 (1991), pp. 29–42.
6. J. C. Carr, W. R. Fright, and R. K. Beatson, *Surface interpolation with radial basis functions for medical imaging*, IEEE Transactions on Medical Imaging, 16 (1997), pp. 96–107.
7. C. S. Chen, M. Ganesh, M. A. Golberg, and A. H.-D. Cheng, *Multilevel compact radial functions based computational schemes for some elliptic problems*, Computers and Mathematics with Applications, 43 (2002), pp. 359–378.
8. J. Chessa, H. Wang, and T. Belytschko, *On the construction of blending elements for local partition of unity enriched finite elements*, International Journal for Numerical Methods in Engineering, 57 (2003), pp. 1015–1038.
9. J. Dolbow and T. Belytschko, *An introduction to programming the meshless element-free Galerkin method*, Archives of Computational Methods in Engineering, 5 (1998), pp. 207 – 241.
10. M. Fleming, Y. A. Chu, B. Moran, and T. Belytschko, *Enriched element-free Galerkin methods for crack tip fields*, International Journal for Numerical Methods in Engineering, 40 (1997), pp. 1483–1504.
11. A. Gravouil, N. Moës, and T. Belytschko, *Non-planar 3d crack growth by the extended finite element and level sets. part ii: level set update.*, International Journal for Numerical Methods in Engineering, 53 (2002), pp. 2569–2586.
12. E. J. Kansa, *A scattered data approximation scheme with application to computational fluid-dynamics–i and ii*, Computers and Mathematics with Applications, 19 (1990), pp. 127–161.

13. E. J. Kansa and R. E. Carlson, *Improved accuracy of multiquadric interpolation using varible shape parameters*, Computers and Mathematics with Applications, 24 (1992), pp. 99–120.
14. Y. Krongauz and T. Belytschko, *EFG approximation with discontinuous derivatives*, International Journal for Numerical Methods in Engineering, 41 (1998), pp. 1215–1233.
15. G. R. Liu and Y. T. Gu, *A point interpolation method for two-dimensional solids*, International Journal for Numerical Methods in Engineering, 50 (2001), pp. 937–951.
16. W. R. Madych and S. A. Nelson, *Multivariate interpolation and conditionally positive definite functions*, Approx. Theory and its Appl., 4 (1988), pp. 77–89.
17. J. M. Melenk and I. Babuška, *The partition of unity finite element method: Basic theory and applications*, Computer Methods in Applied Mechanics and Engineering, 39 (1996), pp. 289–314.
18. N. Moës, J. Dolbow, and T. Belytschko, *A finite element method for crack growth without remeshing*, International Journal for Numerical Methods in Engineering, 46 (1999), pp. 131–150.
19. N. Moës, A. Gravouil, and T. Belytschko, *Non-planar 3d crack growth by the extended finite element and level sets. part i: Mechanical model.*, International Journal for Numerical Methods in Engineering, 53 (2002), pp. 2549–2568.
20. D. Organ, M. Fleming, T. Terry, and T. Belytschko, *Continuous meshless approximations for nonconvex bodies by diffraction and transparency*, Computational Mechanics, 18 (1996), pp. 1–11.
21. S. Rippa, *An algorithm for selecting a good value for the parameter c in radial basis function interpolation*, Advances in Computational Mathematics, 11 (1999), pp. 193–210.
22. M. Sharan, E. J. Kansa, and S. Gupta, *Application of the multiquadric method for numerical solution of elliptic partial differential equations*, Applied Mathematics and Computation, 84 (1997), pp. 275–302.
23. M. Stolarska, D. L. Chopp, N. Moës, and T. Belytschko, *Modelling crack growth by level sets in the extended finite element method*, International Journal for Numerical Methods in Engineering, 51 (2001), pp. 943–960.
24. N. Sukumar, D. L. Chopp, N. Moës, and T. Belytschko, *Modeling holes and inclusions by level sets in the extended finite element method*, Computer Methods in Applied Mechanics and Engineering, 190 (2001), pp. 6183–6200.
25. S. P. Timoshenko and J. N. Goodier, *Theory of Elaticity (Third ed.)*, New York, McGraw Hill, 1970.
26. G. Ventura, J. X. Xu, and T. Belytschko, *A vector level set method and new discontinuity approximations for crack growth by EFG*, International Journal for Numerical Methods in Engineering, 54 (2002), pp. 923–944.
27. J. G. Wang and G. R. Liu, *On the optimal shape parameters of radial basis functions used for 2-d meshless methods*, Computer Methods in Applied Mechanics and Engineering, 191 (2002), pp. 2611–2630.
28. J. G. Wang and G. R. Liu, *A point interpolation meshless method based on radial basis functions*, International Journal for Numerical Methods in Engineering, 54 (2002), pp. 1623–1648.
29. H. Wendland, *Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree*, Advances in Computational Mathematics, 4 (1995), pp. 389–396.

30. H. Wendland, *Meshless Galerkin methods using radial basis functions*, Mathematics of Computation, 68 (1999), pp. 1521–1531.
31. Z. Wu, *Compactly supported positive definite radial functions*, Advances in Computational Mathematics, 4 (1995), pp. 283–292.

Treating Moving Interfaces in Thermal Models with the C-NEM

Julien Yvonnet*, David Ryckelynck, Philippe Lorong, and Francisco Chinesta

Laboratoire de Mécanique des Systèmes et des Procédés, UMR 8106 CNRS ENSAM-ESEM, 151 boulevard de l'Hôpital, F-75013 Paris, France.

Abstract This paper deals with the description of a new numerical simulation technique based in the constrained natural element method, a novel meshless method, able to compute multiphase thermal problems with moving interfaces without requiring the frequent mesh updating characteristics of interfaces tracking finite element techniques. This strategy combines some of the ideas of the natural element method with a particular treatment of the moving boundaries and interfaces involving discontinuities of some fields.

1 Introduction

Phase boundaries represent material interfaces across several fields which may exhibit sharp gradients, and even discontinuities. A wide range of numerical methods have been developed for treating these problems according to the pertinent physics and assumptions about the interface [LER00]. When a sharp interface is considered, its motion is governed by the jump in the temperature gradient normal to the phase boundary and is accompanied by the latent heat effects (Stefan condition). In order to satisfy these conditions the most common approach lies in tracking explicitly the interface motion. Within the interface tracking approach two main alternatives exist: the moving mesh methods and the mixed Eulerian-Lagrangian methods. Moving finite element mesh methods conform element boundaries to the interfaces as it evolves. Although these methods are very accurate, they are limited by the severe mesh distortion. Thus, frequent remeshing is needed, with the associated field projections between successive meshes. Moreover, remeshing is, even today, a delicate task in the 3D case. To alleviate remeshing efforts a number of Eulerian-Lagrangian methods have been developed recently that track the interface while solving the equations on a fixed grid [UMS99]. Many of these methods effectively smear the discontinuity over a few grid cells, and are therefore not capable of representing the true discontinuity across the interface.

* julien.yvonnet@paris.ensam.fr

A new approach for representing localized behaviours has recently emerged in the field of the finite element method, known as the partition of unity method [MBA96]. The main idea is to extend the classical approximation considering the product of the standard shape functions and local enrichment functions. The extended finite element method (X-FEM) is a variation on this framework. Recently, the X-FEM has been coupled to the Level Set Method [SSO94] to represent interface topologies [SCM01]. In this way, the discontinuity evolution can be properly represented on a fixed background mesh, just by adding an appropriate enrichment in the functional approximation in the elements concerned by the moving discontinuity [JCD02]. However, when the material in which the interface is moving, is subjected to large displacements, an updated Lagrangian description could be a better choice. When the background mesh evolves, remeshing will be required also to avoid too high background mesh distortions.

To alleviate dependance to the mesh and to provide smoother shape functions, the use of the meshfree or meshless methods is investigated. The meshless methods discretize a continuum body by a finite number of particles (or nodes) and the field of interest is interpolated under these nodes without the aid of an explicit mesh. Many meshless methods have been proposed, including the smooth particle hydrodynamics (SPH) [LUC77], the diffuse element method (DEM) [NTV92], the element free Galerkin (EFG) [BLG94], the reproducing kernel particle method (RKPM) [LJZ95], the HP clouds DOD96 and the partition of unity method (PUM) [MBA96].

The introduction of moving discontinuities in these meshless methods can present difficulties for the following reasons: (i) The quality of the approximation as well as the conditioning of the global system is pathologically dependant from the size of the support of the shape functions; (ii) Imposition of essential boundary conditions needs particular treatments; (iii) Integration is not accurately defined, and (iv) The physical discontinuity across the interfaces must be introduced accurately. In order to overcome these different problems, we propose the use of the constrained natural element method (C-NEM) [YRa03] [YRb03] for treating thermal models involving moving interfaces. This approach is an extension of the natural element method [SMB98] in which both trial and test functions are constructed on the basis of the Voronoi based interpolants [SIB80] [HSU02], verifying the Kronecker delta property and whose support is defined by the union of the Delaunay spheres passing through the visible nodes. In the C-NEM, the introduction of a visibility criterion and its related constrained Voronoi diagram preserves the appealing properties of the NEM in any geometry (convex or not) and allows easily the introduction of material discontinuities. In a former paper the C-NEM has been sucessfully applied in some problems involving non convex domains, discontinuities and cracks [YRb03].

The structure of this work is as follow: in section 2, we introduce a simple mathematical model of a thermal problem involving a moving interface. Actually, it turns out that it results the standard Stefan problem [VUI93]. In

section 3, the C-NEM is summarized, and it will be applied in section 4 to discretize the weak formulation of the non-linear Stefan problem. A numerical benchmark is presented in section 5, which allows us to conclude about the accuracy of the proposed technique by comparing the numerical results with the exact solution of the problem.

2 Problem Formulation

Let $\Omega \subset \Re^2$ be a bounded domain and T the temperature field. On the domain boundary $\Gamma \equiv \partial\Omega$ the temperature or the thermal flux are prescribed. We will denote by Γ_1 the domain boundary where the temperature is known $T(\mathbf{x} \in \Gamma_1, t) = \overline{T}(\mathbf{x}, t)$ and by Γ_2 the domain boundary where the heat flux $\overline{q}$ is imposed. The thermal model is defined in the time interval $[0, t_{max}]$. The initial temperature $T(\mathbf{x}, t = 0) = T_0$, where T_0 is assumed to be higher than the material melting temperature T_m. At time, $t = 0$, a part of the domain boundary Γ_1 is suddenly submitted to a temperature $T_1 < T_m$. A moving solidification front Γ_I is then generated, whose position evolves in time, i.e. $\Gamma_I(t)$, dividing the domain Ω in two regions $\Omega_1(t)$ (containing the solid phase at time t) and $\Omega_2(t)$ (which contains the liquid phase) as shown in Figure 2.1. For a sake of simplicity we will consider, from now on, a homogeneous and isotropic thermal model in both phases.

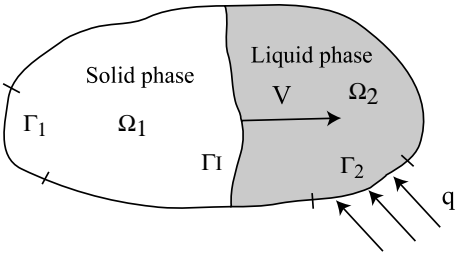


Figure 2.1. Two phases problem.

The heat transfer model is defined in each phase, neglecting volumetric source terms, by:

$$\begin{cases} c_1 \frac{\partial T(\mathbf{x},t)}{\partial t} = \nabla \cdot (k_1 \nabla T) & in\ \Omega_1(t) \\ c_2 \frac{\partial T(\mathbf{x},t)}{\partial t} = \nabla \cdot (k_2 \nabla T) & in\ \Omega_2(t) \end{cases} \tag{2.1}$$

where c_1 and c_2 are the volumetric heat capacities of both phases, being k_1 and k_2 their thermal conductivities. The associated initial and boundary conditions are:

$$\begin{cases} T(\mathbf{x}, t=0) = T_0 & \forall \mathbf{x} \in \Omega \\ T(\mathbf{x}, t) = \overline{T}(\mathbf{x}, t) & \forall \mathbf{x} \in \Gamma_1, \forall t \in [0, t_{max}] \\ -k\nabla T(\mathbf{x}, t) \cdot \mathbf{n} = \bar{q}(\mathbf{x}, t) & \forall \mathbf{x} \in \Gamma_2, \forall t \in [0, t_{max}] \end{cases} \tag{2.2}$$

The evolution of the interface $\Gamma_I(t)$ is described by a Stefan condition:

$$\mathbf{V}(\mathbf{x} \in \Gamma_I(t)) = \frac{|[q]|}{L}\, \mathbf{n}_I(\mathbf{x}) \tag{2.3}$$

where $\mathbf{V}$ is the interface velocity, L is the volumetric latent heat of fusion, $\mathbf{n}_I(\mathbf{x})$ is the normal vector to the interface at point $\mathbf{x}$ which is assumed to point into the liquid phase, and $[[q]]$ the thermal flux jump across the interface $\Gamma_I(t)$, i.e.

$$|[q]| = \left(k_1 \nabla T|_{\Gamma_I^-(t)} - k_2 \nabla T|_{\Gamma_I^+(t)} \right)\, \mathbf{n}_I \tag{2.4}$$

The additional constraint prescribed on the interface $\Gamma_I(t)$ is:

$$T(\mathbf{x}, t) = T_m; \quad \forall \mathbf{x} \in \Gamma_I(t) \tag{2.5}$$

where T_m is the melting temperature.

3 The Constrained Natural Element Method (C-NEM)

In this section, the utility of the C-NEM to describe moving interfaces and discontinuities in a fixed cloud of nodes is discussed. After a brief review of the Voronoi-based interpolants, we introduce the constrained Voronoi diagram which is used for computing the shape functions in any domain.

3.1 Natural Neighbor Interpolation

We briefly touch upon the foundation of Sibson's natural neighbor coordinates (shape functions) that are used in the natural element method. For a more in-depth discussion on the Sibson interpolant and its application for solving second-order partial differential equations, the interested reader can refer to Braun and Sambridge [SBM95], and Sukmar *et al.* [SMB98]. The NEM interpolant is constructed on the underlying Voronoi diagram. The Delaunay tesselation is the topological dual of the Voronoi diagram.

Consider a set of nodes $S = \{n_1, n_2, \ldots, n_N\}$ in $\Re^2$. The Voronoi diagram is the subdivision of $\Re^2$ into regions T_i (Voronoi cells) defined by:

$$T_i = \{\mathbf{x} \in \Re^2 : d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j), \forall j \neq i\}, \quad \forall\, i \tag{3.1}$$

The Sibson coordinates of $\mathbf{x}$ with respect to a natural neighbor n_i (see Figure 3.2) is defined as the ratio of the overlap area (volume in 3D) of their Voronoi cells to the total area (volume in 3D) of the Voronoi cell related to point $\mathbf{x}$:

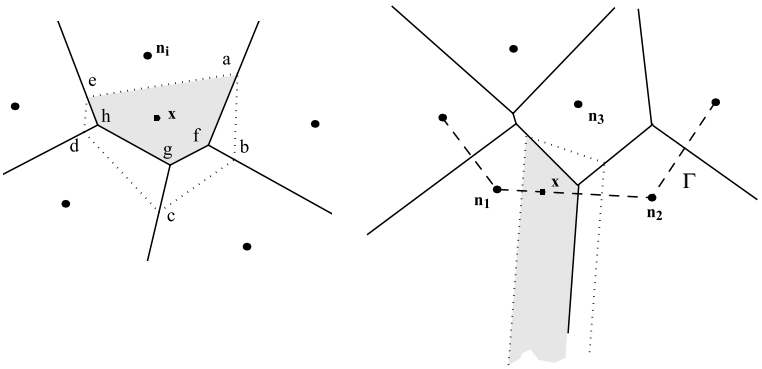


Figure 3.2. Construction of the Sibson shape functions.

$$\phi_i(\mathbf{x}) = \frac{Area(afghe)}{Area(abcde)} \tag{3.2}$$

If the point $\mathbf{x}$ coincides with the node n_i, i.e. $(\mathbf{x} = \mathbf{x}_i)$, $\phi_i(\mathbf{x}_i) = 1$, and all other shape functions are zero, i.e. $\phi_j(\mathbf{x}_i) = \delta_{ij}$ (δ_{ij} being the Kroenecker's delta). The properties of positivity, interpolation, and partition of unity are then verified [SMB98]:

$$\begin{cases} 0 \leq \phi_i(\mathbf{x}) \leq 1 \\ \phi_i(\mathbf{x}_j) = \delta_{ij} \\ \sum_{i=1}^{n} \phi_i(\mathbf{x}) = 1 \end{cases} \tag{3.3}$$

The natural neighbor shape functions also satisfy the local coordinate property [SIB80], namely:

$$\mathbf{x} = \sum_{i=1}^{n} \phi_i(\mathbf{x}) \mathbf{x}_i \tag{3.4}$$

which combined with Eq. (3.3), implies that the natural neighbor interpolant spans the space of linear polynomials (linear completeness).

Natural neighbor shape functions are C^∞ at any point except at the nodes, where they are only C^0, and on the boundary of the Delaunay circles (spheres in 3D) where they are only C^1, because of the discontinuity in the neighbors nodes across these boundaries.

Another important property of this interpolant is the ability to reproduce linear functions over the boundary of convex domains. The proof can be found in Sukumar *et al.* [SMB98]. An illustration is depicted in Figure 3.2 (b): as the areas associated to points on the boundary become infinite, the contribution of internal points vanish in the limit when the point approaches the convex boundary, and the shape functions associated with nodes n_1 and n_2 become linear on the segment $(n_1 - n_2)$. This is not true in the case of non convex

boundaries, and the next section focuses in an approach to circumvert this difficulty.

Consider an interpolation scheme for a scalar function $T(\mathbf{x}) : \Omega \subset \Re^2 \rightarrow \Re^2$, in the form:

$$T^h(\mathbf{x}) = \sum_{i=1}^{n} \phi_i(\mathbf{x})\ T_i \tag{3.5}$$

where T_i are the nodal temperatures at the n natural neighbor nodes, and $\phi_i(\mathbf{x})$ are the shape functions associated with each neighbor node. It is noted that Eq. (3.5) defines a local interpolation scheme. Thus, the trial and test functions used in the discretization of the variational formulation describing the thermal problem treated in this paper take the form of Eq. (3.5).

3.2 The Constrained Natural Element Method

Constrained Voronoi Diagram. It was proved in [YRb03] [SMB98] and [CCC02] that spurious influences between "non-visible" nodes and lost of linearity in the interpolation along boundaries of non convex domains appear in the framework of the NEM. In order to avoid this drawback and to recover all properties of the method for any geometry (including non convex domains containing cracks or involving field discontinuities), a visibility criterion is introduced in order to restrict influent nodes among natural neighbors. The computation of the shape functions is done on the basis of the so-called constrained (or extended) Voronoi diagram (CVD), introduced by Seidel in [SEI88]. The constrained Voronoi cells are defined formally by:

$$\begin{aligned} T_i^C \ &= \ \{\mathbf{x} \in \Re^n : d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j), \\ &\forall j \neq i,\ S_{x \rightarrow n_i} \cap \Gamma = \emptyset,\ S_{x \rightarrow n_j} \cap \Gamma = \emptyset\} \end{aligned} \tag{3.6}$$

where Γ is the domain boundary, composed with segments $l_i \in L$, L being a set of segment in the plane, and $S_{a \rightarrow b}$ denotes the segment between the points a and b.

A generalization of the constrained Delaunay triangulation to 3D doesn't exist without adding new nodes, as shown in [SCH28]. Nevertheless, some techniques for constructing 3D constrained Delaunay tesselations are available and provided in [SMB98],[SHE00] by addition of Steiner points.

The Constrained Natural Element Approximation. In order to solve partial differential equations defined in non convex domains, or to reproduce some functional discontinuities, we consider the following approximation of both the trial and the test functions:

$$T^h(\mathbf{x}) \ = \sum_{i=1}^{V} \phi_i^C(\mathbf{x}) T_i \tag{3.7}$$

where V is the number of natural neighors visible from point $\mathbf{x}$ and ϕ_i^C is the constrained natural neighbor shape function related to the i-*th* node at point x. The computation of the C-n-n (constrained natural neighbor) shape functions is similar to the natural neighbor shape function, when one proceed using the constrained Voronoi diagram introduced previously. It was shown in [YRa03] and [YRb03] that the use of the constrained Voronoi diagram does not affect the properties of the NEM interpolation, allowing the extension of the linearity of the shape functions on the convex domains boundaries, to any geometry, convex or not.

The ability of the C-NEM for treating problems involving cracks has been illustrated in [YRb03]. In the present paper, we focus on its application in the context of a moving interface defining two domains with different thermal properties. Thus, defining at time t two CVD (constrained Voronoi diagrams) of $\Omega_1(t)$ and $\Omega_2(t)$, both with respect to the interface $\Gamma_I(t)$, it can be proved that the interpolated temperature field is C^1 everywhere, except at the nodes and on the interface $\Gamma_I(t)$ where it is only C^0. Thus, this interpolation seems to be appropriate to simulate the Stefan problem considered in this paper.

To illustrate this behavior, we consider the situation depicted in Figure 3.3, where the point $\mathbf{x}$ moves from Ω_1 to Ω_2. If $\mathbf{x}$ is in Ω_1, the interpolated field is constructed from Eq. (3.7) using the neighbor visible nodes from point $\mathbf{x}$ (Γ_I is assumed opaque). If $\mathbf{x}$ is on Γ_I, according to the previous discussion, the interpolated field is strictly linear because it only depends on the two neighbor nodes located on Γ_I. Finally, when $\mathbf{x}$ is in Ω_2, the interpolated field is defined using the visible neighbor and visible nodes from point $\mathbf{x}$ (Γ_I being opaque). The continuity of the interpolated field is then guaranteed, but a discontinuity appears in the field derivatives, because of a sudden change in the neighbor nodes across the interface. We can then reproduce the temperature field continuity, as well as the expected flux discontinuity on the interface.

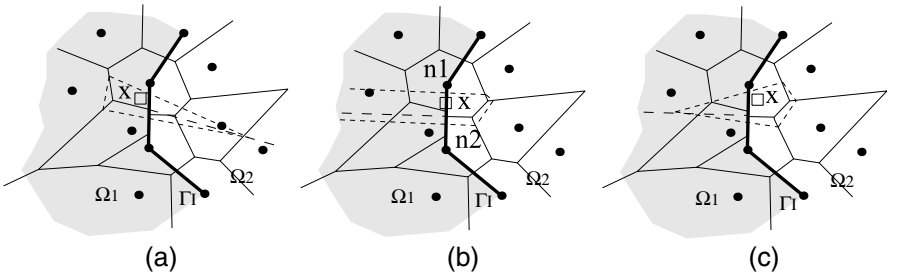


Figure 3.3. Reproducing discontinuous derivatives using the constrained Voronoi diagram.

4 C-NEM Discretization

Assuming that in our example $\Gamma_2 \equiv \Gamma_I$, the weak formulation associated with Eq. (2.1) results:

Find $T \in H^1(\Omega)$ *verifying* $T = \bar{T}$ *on* Γ_1 *such that:*

$$\int_\Omega c\frac{\partial T}{\partial t}\delta T d\Omega = -\int_\Omega k\nabla T\cdot\nabla\delta T d\Omega + \int_{\Gamma_I(t)} |[q]|\delta T d\Gamma, \quad \forall\ \delta T\ \in\ H_0^1(\Omega) \tag{4.1}$$

Where $H^1(\Omega)$ and $H_0^1(\Omega)$ are the usual Sobolev functional spaces. Substituting the trial and test functions (both approximated in the C-NEM framework) in the above equation and using the arbitrariness of the field δT, the following system of equations is obtained:

$$\mathbf{C}\dot{\mathbf{T}} + \mathbf{K}\mathbf{T} = \mathbf{F} \tag{4.2}$$

where $\mathbf{T}$ is the vector containg the unknown nodal temperatures. For time discretization, we consider the solution on the time interval $[0, t_{max}]$, partitioned into steps as $[t^n, t^{n+1}]$ and the generalized trapezoidal time stepping algorithm characterized by the parameter α:

$$\frac{\partial T^{n+1}}{\partial t} = \frac{T^{n+1} - T^n - (1-\alpha)\Delta t\frac{\partial T^n}{\partial t}}{\alpha\Delta t} \tag{4.3}$$

where $\frac{\partial T^0}{\partial t}$ is initialized by setting:

$$\int_\Omega c\frac{\partial T^0}{\partial t}\delta T d\Omega = -\int_\Omega k\nabla T^0\cdot\nabla\delta T d\Omega \quad \forall\ \delta T\ \in\ H_0^1(\Omega) \tag{4.4}$$

which leads, after applying the discretization scheme described in Eq. (3.5), to:

$$\mathbf{C}\frac{\partial \mathbf{T}^0}{\partial t} = -\mathbf{K}\mathbf{T}^0 \tag{4.5}$$

The stabilized conforming nodal integration proposed by Chen et al. in [CWY01] is employed for the numerical integration of $\mathbf{K}$ (see our former work [YRb03] for more details). This technique, based on the assumed strain method, has been used to reduce significantly the integration errors, and allows to satisfy the patch test exactly in the context of the natural neighbor interpolation, which is not the case if a standard Gauss quadrature scheme is used [CCC02]. In the context of the C-NEM, the representative domains related to each node used in such techniques to define the assumed gradient, are the constrained Voronoi cells depicted in figure 3.3, which are accurately defined everywhere, and especially in the interface neighborhood.

A lumped mass matrix $\tilde{\mathbf{C}}$ is computed making use of the constrained Voronoi cells areas as nodal weights.

The iteration procedure is defined as:

Knowing $\mathbf{T}^n$ *and* $|[q]|^n$ *at time* t^n*, the non-linear problem associated with Eq. (4.1) results in finding* $\mathbf{T}^{n+1}$ *and* $|[q]|^{n+1}$ *such that Eqs. (2.5) and (4.1) are satisfied.* For this purpose we proceed as follows:

1. Compute the interface velocity $\mathbf{V}^n(\mathbf{x})$ using Eq. (2.3) and update the interface position at time t^{n+1} using the forward Euler formula:

$$\mathbf{x}_J^{n+1} = \mathbf{x}_J^n + \Delta t \mathbf{V}^n(x_J^n) \tag{4.6}$$

   where $\mathbf{x}_J$ are the nodes defining the interface.
2. Update locally the constrained Voronoi diagram and the shape functions associated with integration points in the interface neighborhood. Then, we compute $\tilde{\mathbf{C}}^{n+1}$ and $\mathbf{K}^{n+1}$.
3. Solve Eq. (4.1) using a Newton-Raphson procedure where the tangent matrix is computed numerically.
4. Repeat while $t^{n+1} < t_{max}$.

An alternative scheme using the Latin method [LAD98] in the extended finite element framework can be found in Merle and Dolbow [MED02].

5 Numerical Example

In this section, we illustrate the potentiality of the proposed technique simulating a two-phases Stefan problem. The problem is essentially one-dimensional, but we solve it here in two dimensions to underline the outstanding features of the method.

The Stefan problem models the one-dimensional freezing of a semi-infinite domain ($x \geq 0$). The initial temperature T_0 is assumed constant in the whole domain, being higher than the melting temperature T_m. At time $t = 0$ the temperature at the left boundary $x = 0$ is suddenly prescribed to a value T_1 lower than the melting point, originating a solidification front that progresses from the boundary $x = 0$ in the x direction. The exact flow front position $x_f(t)$ is given by:

$$x_f(t) = 2\lambda\sqrt{\beta_s t} \tag{5.1}$$

where $\beta_s = k_s/c_s$ is the thermal diffusivity of the solid phase, and the constant λ satisfies the following relationship:

$$\frac{e^{-\lambda^2}}{erf(\lambda)} = \frac{k_l\sqrt{\eta}(T_0 - T_m)e^{-\eta\lambda^2}}{k_s(T_m - T_1)erfc(\lambda\sqrt{\eta})} + \frac{\lambda L\sqrt{\pi}}{c_s(T_m - T_1)} \tag{5.2}$$

with $\eta = \beta_s/\beta_l$ being the ratio of the thermal diffusivities and where k_l represents the liquid phase conductivity. The temperature field in the solid phase $0 \leq x \leq x_f(t)$ is then:

$$T(x,t) = T_1 + \frac{T_m - T_1}{erf(\lambda)} erf\left(\frac{x}{2\sqrt{\beta_s t}}\right) \tag{5.3}$$

and in the liquid phase $x \geq x_f(t)$:

$$T(x,t) = T_0 - \frac{T_0 - T_m}{erfc(\lambda\sqrt{\eta})} erfc\left(\frac{x}{2\sqrt{\beta_l t}}\right) \tag{5.4}$$

In the present investigation, we use the water-satured sand thermal properties provided in [LYO81] and listed in Table I. T_1 and T_0 were set to -10 and 4.0 ^0C respectively ($\lambda = 0.3073$). We simulate the evolution of the temperature field in $\Omega = [0,1] \times [0,0.5]$ cm. In order to use the inifinite domain solution as reference solution, the temperature at $x = 1$ is prescibed to its expected value according to Eq. (5.4)

Table 5.1. Thermal properties of the water saturated sand.

Properties	Solid	Liquid	
Volumetric heat capacity (cal.$^0C^{-1}cm^{-3}$)	0.49	0.62	
Thermal conductivity ($cal.cm^{-1}s^{-1}.^0C^{-1}$)	$9.6.10^{-3}$	$6.9.10^{-3}$	
Melting temperature (^0C)			0.0
Volumetric latent heat of fusion (cal.cm^{-3})			19.2

In a first test, we consider in the domain Ω a 20×10 uniform grid and Δt = 2s. Figure 5.4 compares the computed interface position and the exact one. An excellent accuracy can be noticed, as depicted in figure 5.5, where the error in the front position is represented. Figure 5.6 shows the temperature profile at different times. We can point out that the discontinuity in the temperature gradient is accurately obtained, being in excellent agreement with the exact solution. The external boundary conditions are given by: $T(t) = T_1$ on $(x = 0)$, $T(t) = T_{ex}$ given by Eq. (5.4) on $(x = 1)$ and $-k\nabla T \cdot \mathbf{n} = 0$ on (y=0) and (y=0.5). The initial condition is given by $T(t = 0) = T_0$ in Ω.

In a second test, we consider the domain Ω containing 200 nodes distributed at random. The purpose of the present test is to investigate the meshless feature of the technique, in which due to its meshless character no geometrical restrictions concerning the relative nodal positions are involved. Thus, neither the background nodal distribution nor the relatinve position of the nodes defining the moving interface with respect to the background nodes, induce a lack of accuracy when high distortions in the Delaunay mesh, used to compute the Voronoi diagram, takes place. This is the main difference between the proposed strategy and the standard finite element method whose accuracy depends significantly on the geometrical quality of the mesh. Moreover, this test approaches the situations encountered when the material is also moving, inducing highly irregular nodal densities and high background mesh

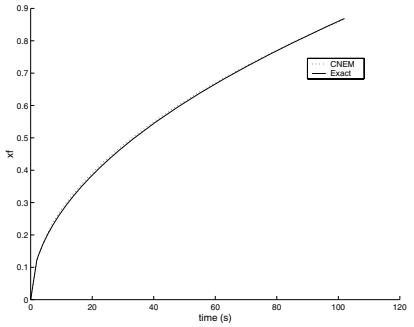


Figure 5.4. Computed C-NEM front position versus the exact solution using a 20×10 regular grid.

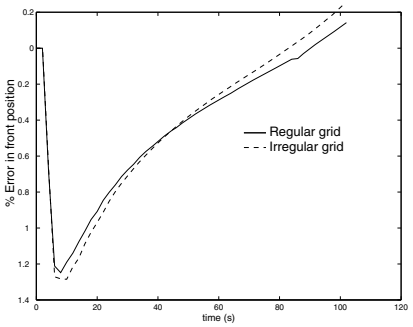


Figure 5.5. Error in the front position.

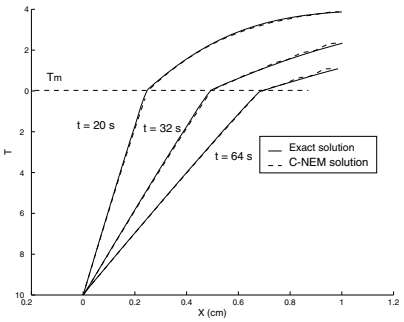


Figure 5.6. Temperature profiles along the line y=0.25 using a regular grid.

distortions. Figure 5.7 depicts the cloud of nodes and the interface position as well as the associated constrained Voronoi cells. Remarkably, despite of the very irregular nodal distribution and density, we can notice that the interface is not distorted as it moves through the domain. From Figs. 5.5 and 5.8 we can conclude that the accuracy is not affected significantly by the resgularity on the nodal distribution. In Figure 5.9 some temperature profiles along the line $y = 0.25$ are depicted, from which an excellent accuracy can be noticed.

6 Conclusion

In this paper, the salient features of the C-NEM meshless method are used for treating thermal problems involving moving interfaces. The use of the constrained Voronoi diagram allows to introduce the desired discontinuities in the gradient of the solution without any enrichment. In the C-NEM framework, essential boundary conditions can be imposed directly, due to the strict linearity of the shape functions over the boundaries (convex or not) and the

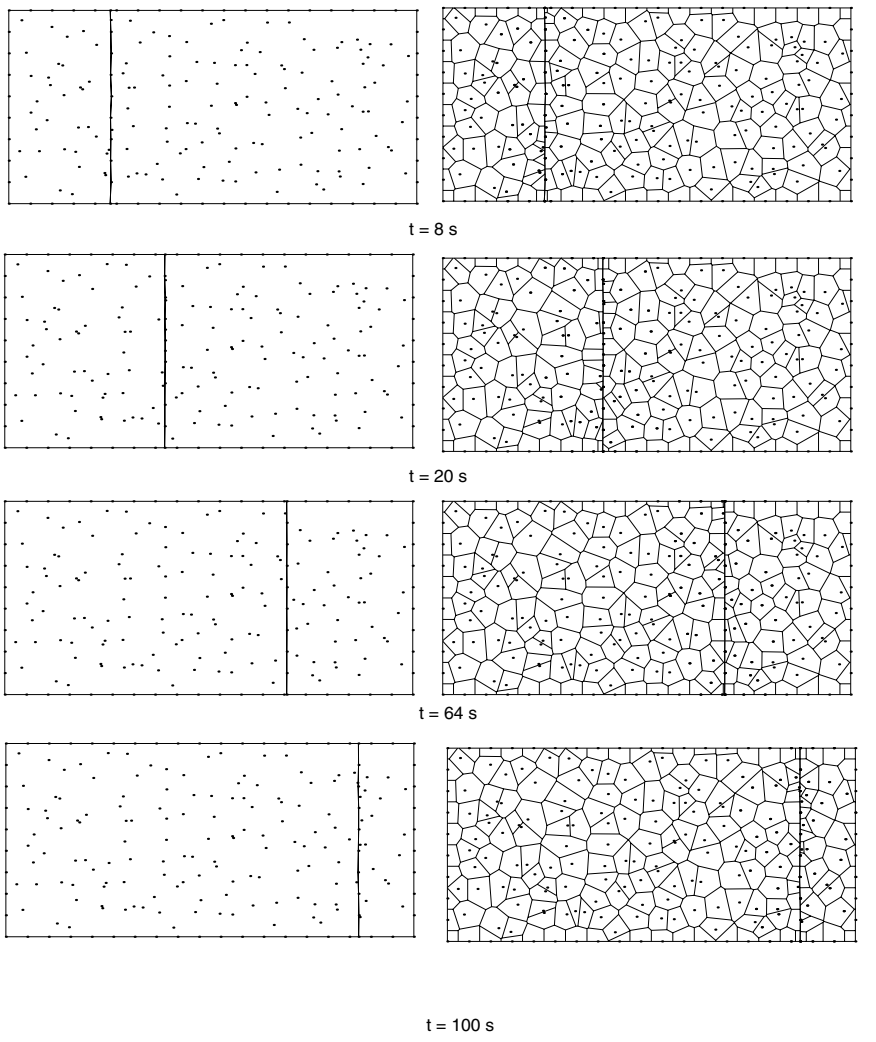


Figure 5.7. Computed interface position using an irregular cloud of nodes: (a) Cloud of nodes and interface position; (b) Constrained Voronoi cells.

respect of the Kronecker delta property. No user-defined parameter is involved in the shape functions support size, the support being defined like the union of the Delaunay spheres passing through a node and its neighbor visible nodes. A stabilized conforming nodal integration is computed over the constrained Voronoi cells to enhance accuracy. The most outstanding quality of this technique is the ability for introducing discontinuities located on a line (surface in 3D) defined by another set of nodes that move through a fixed (or also moving) cloud of nodes which constitutes a background scattered of nodes.

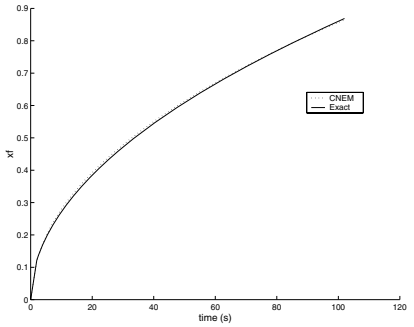


Figure 5.8. C-NEM interface position versus the exact solution using an irregular grid.

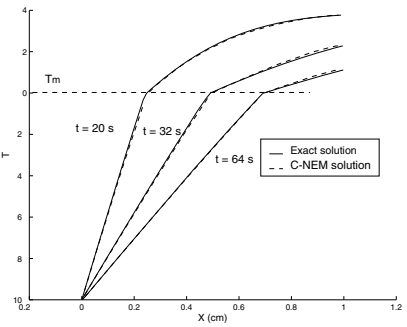


Figure 5.9. Temperature profiles along the line $y = 0.25$ using an irregular grid.

The meshless character of this technique allows to proceed without remeshing (in a finite element sense), even for highly irregular nodal distributions. The method is relatively simple and delivers similar level of accuracy than r-adaptative or the partition of unity finite element schemes. The technique seems promising for the simulation of arbitrary and moving discontinuities over an unstructred fixed or also moving set of nodes.

References

[BLG94] Belytschko T., Lu Y.Y., Gu L.: Element-free Galerkin methods. International Journal for Numerical Methods in Engineering **37**, 229–256 (1994)

[CWY01] Chen J.S., Wu C.T., Yoon Y.: A stabilized conforming nodal integration for Galerkin mesh-free methods. International Journal for Numerical Methods in Engineering, **50**, :435–466 (2001)

[CCC02] Cueto E., Cegoñino J., Calvo B., Doblaré M.: On the imposition of essential boundary conditions in Natural neighbor Galerkin Methods. Communications in Numerical Methods in Engineering, **19**, 361–376 (2002)

[DOD96] Duarte C.A., Oden J.T.: An H-p adaptative method using clouds. Computer Methods in Applied Mechanics and Engineering **139**, 237–262 (1996)

[HSU02] Hiyoshi H., Sugura K.: Improving continuity of Voronoi-based interpolation over Delaunay spheres. Computational Geometry, **22**, 167–183 (2002)

[JCD02] Ji H., Chopp D., Dolbow J.E.: A hybrid finite element/level set method for modeling phase transformations. International journal for numerical methods **54**, 1209–1233 (2002)

[LAD98] Ladevèze P.: Non Linear Computational Structural Mechanics. Springer:, New York (1998)

[LER00] Lewis, R., Ravindran, K.: Finite element simulation of metal casting. International journal for numerical methods in engineering, **47**, 93–102 (2000)

[LJZ95] Liu W.K., Jun S., Zhang Y.F.: Reproducing Kernel Particle Methods. Int. J. Numer. Methods Fluids **21**, 1081–1106 (1995)

[LUC77] Lucy L.B.: A numerical approach to the testing of fusion process. The astronomic journal **88**, 1013–1024 (1977)

[LYO81] Lynch D., O'Neill K.: Continuously deforming finite elements for the solution of parabolic problems, with and without phase change. International Journal for Numerical Methods in Engineering, **17**, 81–96 (1981)

[MBA96] Melenk, J.M., babuška, I.: The partition of unity finite element method: basic theory and applications. Computer Methods in Applied Mechanics and Engineering, **139**, 289–314 (1996)

[MED02] Merle R., Dolbow J.E.: Natural neighbor Galerkin methods. Solving thermal and phase change with the extended finite element method, **28**(5), 339–350 (2002)

[NTV92] Nayroles B., Touzot G., Villon P.: Generalizing the finite element method: diffuse approximation and diffuse elements. Computational mechanics **10**, 307–318 (1992)

[UMS99] Udaykumar, H., Mittal, R, Shyy W.: Computation of solid-liquid phase fronts in the sharp interface limit on fixed grids. Journal of computational physics, **153**, 535–574 (1999)

[SBM95] Sambridge M., Braun J., McQueen M.: Geophisical parameterization and interpolation of irregular data using natural neighbors. Geophys. i. J. Int, **122**,

[SCH28] Schönhardt E.: Uber die zerlegung von dreieckspolyedern in tetraeder. Math. Annalen, **98**, (1928) 837–857 (1995)

[SEI88] Seidel R.: Constrained Delaunay triangulations and Voronoi diagrams with obstacles In: "1978-1988 Ten Years IIG" 178–191. (1988)

[SHE98] Shewchuck J.R.: Tetrahedral mesh generation by delaunay refinement. In: Proceedings of the fourteenth annual symposium on computational geometry, Minneapolis, Minnesota, june 1998; pp. 86–95, association for computing machinery (1988)

[SHE00] Shewchuck J.R.: Sweep algorithms for constructing higher-dimensional constrained Delaunay triangulations. In: Proceedings of the sixteenth annual symposium on computational geometry, Hong-Kong, june 2000; pp. 350–359, association for computing machinery (2000)

[SIB80] Sibson R.: A vector Identity for the Dirichlet tesselations. Math. Proc. Camb. Phil. Soc., **87**, 151–155 (1980)

[SCM01] Sukumar N., Chop D., Moës N., Belytschko T.: Modeling holes and inclusions by level sets in the extended finte element method. Computer methods in appled mechanics and engineering **190**, 6183–6200 (2001)

[SMB98] Sukumar N., Moran B., Belytschko T.: The natural elements method in solid mechanics. International Journal for Numerical Methods in Engineering, **43**, 839–887 (1998)

[SMSB01] Sukumar N., Moran B., Semenov Y., Belikov V.V.: Natural neighbor Galerkin methods. International Journal for Numerical Methods in Engineering, **50**, 207–219 (2001)

[SSO94] Sussman M., Smereka P., Osher S.: A level set approach for computing solutions to incompressible two-phase flows. Journal of computational physics **114**, 146–159 (1994)

[VUI93] Vuik C.:Some historical notes about the Stefan problem. In: CWI-tract 90 (CWI, Amsterdam) (1993)

[YRa03] Yvonnet J., Ryckelynck D., Lorong P., Chinesta P.: Interpolation naturelle sur les domaines non convexes par l'utilisation du diagramme de Voronoi contraint - MŐthode des ŐlŐments C-Naturels. Revue EuropŐenne des éléments Finis **12**(4), 487–509 (2003)

[YRb03] Yvonnet J., Ryckelynck D., Lorong P., Chinesta P.: A new extension of the natural element method for non convex and dscontinuous problems, the constrained natural element method (C-NEM). International Journal for Numerical Methods in Engineering, accepted.

Bridging Scale Particle and Finite Element Methods

Wing Kam Liu[1⋆], Lucy T. Zhang[2⋆⋆], Eduard G. Karpov[1⋆⋆⋆], Hiroshi Kadowaki[1†], and Harold Park[1‡]

[1] Mechanical Engineering Department, Northwestern University, 2145 Sheridan Rd., Evanston, IL 60208, USA.
[2] Mechanical Engineering Department, Tulane University, 6823 Saint Charles Ave., New Orleans, LA 70118, USA.

Abstract We summarize the strengths and limitations of currently available multiple-scale techniques, where the emphasis is made on the latest prospective approaches, such as the bridging scale method, multiscale boundary conditions, and multiscale fluidics. Example problems, in which multiple-scale simulation methods yield equivalent results to full atomistic simulations at fractions of the computational cost, are shown. We conclude by discussing future research directions and needs in multiple-scale analysis, and also discuss the ramifications of the integration of current nanoscale research into education.

1 Introduction

The rapid advances in nanotechnology, nanomaterials and nanomechanics offer huge potentials in national defense, homeland security, and private industry. An emphasis on nanoscale entities will make our manufacturing technologies and infrastructure more sustainable in terms of reduced energy usage and environmental pollution. Recent advances in the research community on this topic have stimulated ever-broader research activities in science and engineering devoted to their development and their applications. With the confluence of interest in nanotechnology, the availability of experimental tools to synthesize and characterize systems in the nanometer scale, and computational tools widely accessible to model microscale systems by coupled continuum/molecular/quantum mechanics, we are poised to unravel the traditional

⋆ w-liu@northwestern.edu
⋆⋆ ltzhang@tulane.edu
⋆⋆⋆ ekarpov@northwestern.edu
† h-kadowaki@northwestern.edu
‡ hpark@northwestern.edu

gap between the atomic and the macroscopic world in mechanics and materials. This in turn opens up new opportunities in education and research. Over the past three decades, we have acquired new tools and techniques to synthesize nanoscale objects and to learn their many incredible properties. Today's high resolution electron microscopes can routinely see individual atoms. Scanning probe techniques allow us to manipulate atoms one at a time. Advanced materials synthesis provides the technology to tailor-design systems from as small as molecules to structures as large as the fuselage of a plane. We now have the technology to detect single molecules, bacteria or virus particles. We can make protective coatings more wear-resistant than diamond and fabricate alloys and composites stronger than ever before.

In most of these applications, nanoscale materials will be used in conjunction with other components that are larger and have different response times, thus operating at different time and length scales. Single scale methods such as "ab initio" quantum mechanical methods or molecular dynamics (MD) will have difficulty in analyzing such hybrid structures due to the limitations in terms of the time and length scales that each method is confined to. Because of the availability of accurate interatomic potentials for a range of materials, classical MD simulations have become prominent as a tool for elucidating complex physical phenomena. However, the length and time scales that can be probed using MD are still fairly limited. For the study of nanoscale mechanics and materials, we must model up to a scale of several microns, consisting of billions of atoms, which is too large for MD simulations to date. Hence, we need to develop multiscale approaches for this class of problems. One possible approach that can be applied to many problems is to use MD only in localized regions in which the atomic scale dynamics are important, and a continuum simulation method, such as the finite element ([16], [43]) or meshfree method ([5], [6], [13], [14], [23],[27], [28], [24], [25], [26]), everywhere else. This general approach has been taken by several different groups using methods that have had varying degrees of success. In particular, these methods do not satisfactorily address the issue of disparate time scales in the two regions, and provide a rather simplified treatment of the interface between the atomistic and continuum regimes. Current research in engineering is just beginning to impact molecular scale mechanics and materials and would benefit from interaction with basic sciences. For solids, research in the area of plasticity and damage has experienced some success in advancing microscale component design. Development of carbon nanotubes ([8], [34], [36], [41], [42]) is also an area in which nanoscale research has clearly played a major role. Other areas of opportunities include nanocomposites and nanoalloys. Electrophoresis and electro-osmotic flows coupled with particulate motion in a liquid have been important research areas that have had great impact in the homeland security area. Microfluidic devices often comprise components that couple chemistry, and even electrochemistry, with fluid motion. Once the physics-based models are determined for the solids and fluids, computational approaches will need to be employed or developed to capture the coupled physics phenomena. The

paper's outline is given by the following. In Section 2, we will review the hierarchical and concurrent coupling of the atomistic and continuum simulations, and multiscale boundary conditions. Section 3 focuses on the bridging scale method for quasi-static and dynamic problems along with several examples and applications. Bridging scale computations for localized failure is explained in Section 4. Section 5 concludes the paper by discussing future research needs in multiple-scale analysis.

2 Review of Multiscale Simulation Methods

Over the past few decades, continuum methods have dominated materials modeling research. This approach of predicting material deformation and failure by implicitly averaging atomic scale dynamics and defect evolution over time and space, however, is valid only for large enough systems that include a substantial number of defects. As a result, numerous experimental observations of material behavior cannot be readily explained within the continuum mechanics framework: dislocation patterns in fatigue and creep, surface roughening and crack nucleation in fatigue, the inherent inhomogeneity of plastic deformation, the statistical nature of brittle failure, plastic flow localization in shear bands, and the effects of size, geometry, and stress state on yield properties. Thus, there is a considerable effort to find fundamental descriptions for strength and failure properties of nanoscale materials, taking into account their atomic structures. The use of MD simulations has provided useful information of chemical interactions at the nanoscale. However, MD simulations have their own limitations, as discussed in the previous section. Typical atomistic simulations are still restricted to very small systems consisting of several million atoms or less and timescales on the order of picoseconds. Thus, even for nanoscale structures and materials, atomistic modeling would be computationally prohibitive. The limitations of atomistic simulations and continuum mechanics, along with practical needs arising from the heterogeneous nature of engineering materials, have motivated research on multiscale simulations that bridge atomistic simulations and continuum modeling ([1], [9], [33], [17], [35], [32], [40], [31]). In order to make the computations tractable, multiscale models generally make use of a coarse-fine decomposition. An atomistic simulation method, such as MD, is used in a small subregion of the domain in which it is crucial to capture the individual atomistic dynamics accurately. A continuum simulation is used in all other regions of the domain in which the deformation is considered to be homogeneous and smooth. Since the continuum region is usually chosen to be much larger than the atomistic region, the overall domain of interest can be considerably large. A purely atomistic solution is normally not affordable on this domain, though the multiscale solution would presumably provide the detailed atomistic information only when and where it is necessary. The key issue is then the coupling between the coarse and fine scales. Depending on the method of information exchange between

the coarse and fine regions, multiscale methods can be classified into three groups: hierarchical, concurrent, and multiscale boundary conditions.

Hierarchical approaches embed the intrinsic atomistic properties of the solid in the continuum formulation according to the Cauchy-Born rule, so that small scales depend on large scales in some predictable way. Hierarchical techniques are based on the assumption of homogeneous lattice deformation; therefore they are more effective for elastic single-phase problems. Difficulties typically arise from modeling defects in atomic lattices, dislocations, and failure phenomena.

Within concurrent methods, the behavior at each length scale depends strongly on the others. An appropriate model is solved at each length scale simultaneously (continuum mechanics for macro elastic media, molecular dynamics for large groups of atoms and quantum mechanics for bond breaking), while a smooth coupling is introduced between the different scales. The inter-scale dependence is complicated, and it is not preassigned. Concurrent approaches are more relevant for studying complicated problems, involving inhomogeneous lattice deformation, fracture in multiphase macroscopic materials, and nano fluidics. However, two arguable issues do typically arise: a) how to separate the scales, and b) what is the adequate mechanism of coupling the atomistic and continuum simulations.

Multiscale boundary conditions for molecular dynamic simulations is an emerging approach not to involve the explicit continuum model, so that the issues of separating the scales and coupling the simulations do not arise. In this case, the coarse grain behavior is taken into account on the fine/coarse grain interface at the atomistic level through the lattice impedance techniques. These methods, though, may appear to be more effective than the concurrent methods only for a particular class of problems with linear coarse grains in solids.

3 Concurrent Bridging Scale Method

To avoid information loss when passing information from one length scale to another, researchers have expended a great deal of effort in developing concurrent methods to achieve a seamless bridging between different length scales. In a concurrent method, simulations at different length scales are performed simultaneously and the information interfaces between different length scales continually transmit information from one simulation to the other.

In overcoming the requirement of grading finite-element mesh down to lattice size, as in QC and MAAD ([1], [2]) methods, a concurrent coupling method has been recently developed by Liu and co-workers ([22], [39], [17], [35], [32], [40], [31]). A unique characteristic of this method is that it is formally assumed that the FE and MD solutions exist simultaneously in the entire computational domain and MD calculations are performed only in the regions that are necessary. The basic idea is to decompose the total displacement field

$\mathbf{u}(\mathbf{x})$ into coarse and fine scales

$$\mathbf{u}(x,t) = \bar{\mathbf{u}}(x,t) + \mathbf{u}'(x,t) \tag{3.1}$$

where $\bar{\mathbf{u}}(x,t)$ is the coarse scale solution and $\mathbf{u}'(x,t)$ is the fine scale solution, corresponding to the part that has a vanishing projection onto the coarse scale basis functions. The coarse scale solution can be interpolated by basic finite element shape functions as $\bar{\mathbf{u}} = \mathbf{Nd}$, where $\mathbf{d}$ is the FE solution and $\mathbf{N}$ is the shape function evaluated at atomic locations. Wagner and Liu ([40]) demonstrated that $\mathbf{u}' = \mathbf{Qq}$, where $\mathbf{q}$ is the MD solution,$\mathbf{Q} = \mathbf{I} - \mathbf{P}$, $\mathbf{P}$ is a projection operator that depends on both shape functions and the properties of the atomic lattice, and $\mathbf{I}$ is the identity matrix.

Within concurrent methods, the behavior at each length scale depends strongly on the others. An appropriate model is solved at each length scale simultaneously (continuum mechanics for macro elastic media, molecular dynamics for large groups of atoms and quantum mechanics for bond breaking), while a smooth coupling is introduced between the different scales. The inter-scale dependence is complicated, and it is not preassigned. Concurrent approaches are more relevant for studying complicated problems, involving inhomogeneous lattice deformation, fracture in multiphase macroscopic materials, and nanofluidics. However, two arguable issues do typically arise: a) how to separate the scales, and b) what is the adequate mechanism of coupling the atomistic and continuum simulations.

3.1 Quasistatic Problems

Within the bridging scale method, the governing equations are obtained, employing the principle of virtual work, to give

$$\mathbf{N^T f^{int}}(\mathbf{d},\mathbf{q}) = \mathbf{N^T f^{ext}}(\mathbf{d},\mathbf{q}) \tag{3.2}$$

$$\mathbf{Q^T f^{ext}}(\mathbf{d},\mathbf{q}) = \mathbf{Q^T f^{ext}}(\mathbf{d},\mathbf{q}) \tag{3.3}$$

where $\mathbf{f^{int}} = -\partial U(\mathbf{u})/\partial \mathbf{u}$ is the internal force, and $\mathbf{f^{ext}}$ is the external force. The first equation is solved over the entire domain, while the solution of Eq. 3.3 is equivalent to that from MD simulation, and is hence only solved in localized region. Note that these two equations are coupled. A Newton's method can be used to iteratively solve the coupled equations. Bridging between the coarse and fine scale is realized by transparently exchanging information between coarse and fine scale regions.

It remains to be shown how the internal forces are calculated for the coarse scale simulation. Note that the energy density associated with α is directly related to the bond vector over an effective domain, ΔV_α in the deformed configuration. The total potential energy in the system can be written as a discrete summation over all the atoms: coarse scale simulation provides boundary conditions for the fine scale simulation, while the fine scale simulation provides an accurate approximation of the internal force that enriches the coarse scale:

$$U = \sum_{\alpha} \sum_{\beta \neq \alpha} w(\mathbf{r}_{\alpha,\beta}) \Delta V_{\alpha} \tag{3.4}$$

where β runs over all the neighbors of atom α within a prescribed cut-off radius, and w is an energy density function. In the coarse scale, the atomic bond vector is deformed according to the coarse scale deformation field. Thus it can be interpolated by the nodal displacement $\mathbf{d}$. By definition, $\mathbf{Nf^{int}} = \partial w(\mathbf{r}_{\alpha\beta})/\partial \mathbf{d}$, one has

$$\mathbf{Nf^{int}} = \sum_{\alpha} \sum_{\bar{\beta} \neq \alpha} \frac{\partial w(\mathbf{r}_{\alpha\beta})}{\partial \mathbf{r}_{\alpha\bar{\beta}}} [\mathbf{N}(\mathbf{X}_{\bar{\beta}}) - \mathbf{N}(\mathbf{X}_{\alpha})] \Delta V_{\alpha} \tag{3.5}$$

The discrete summation of the right-hand side of the above equation makes the evaluation computationally intensive. In practice, the strain energy can be considered as a smooth function. Thus, the discrete summation over the atoms in equation 3.5 can be replaced by evaluations at quadrature points, as

$$\mathbf{Nf^{int}} = \sum_{\alpha} \sum_{\beta \neq \alpha} \lambda_{\alpha} \frac{\partial w(\mathbf{r}_{\alpha\beta})}{\partial \mathbf{r}_{\alpha\beta}} [\mathbf{N}(\mathbf{X}_{\beta}) - \mathbf{N}(\mathbf{X}_{\alpha})] \Delta V_{\alpha} \tag{3.6}$$

The advantages of the bridging scale method over other concurrent methods are the following. First, it does not involve calculation of any high-order tensor, such as the Piola-Kirchoff stress, which makes it computationally more efficient. Secondly, no mesh gradation is required. Finally, the bridging scale method can be extended to the dynamic case ([32], [40]), as outlined in the following section, while the quasi-continuum method has currently be shown only for quasi-static processes. All these factors have made the bridging scale method an increasingly popular approach to simulating nanostructured materials.

Mechanics of Multi-walled Carbon Nanotubes. The bridging scale method has been successfully used in modeling buckling of multi-walled carbon nanotubes ([35]). In these simulations, a 15-walled Carbon nanotube (CNT) is considered with the outermost shell being a $(140, 140)$ nanotube, and all inner shells of the (n, n) type; from the outer most shell, n reduces by 5 every layer. The length of the tube is $90nm$ and the original MD system contains about 3 million atoms. This is replaced with a system of $27,450$ particles. In addition to the particles, two sections along the tube are enriched with molecular structures of multi-walled nanotubes. The position of the enrichment region is determined by a multi-resolution analysis of the coarse scale simulation. Therefore, the scheme is adaptive. The length of each enrichment region is $3.6nm$. Each section contains $49,400$ atoms and this adds $296,400$ more atomic degrees of freedom. A bending angle with increment of 0.25 degree/step is imposed on both ends of the tube for a total of 100 steps. The multiscale configuration is illustrated in Figure 3.1. Figure 3.2 shows the buckling pattern approximated by meshfree approximation at the final stage of loading, followed by the energy density contour plot for each layer of CNT.

Two distinctive buckling patterns can be seen from the meshfree approximation, while the contour plot shows clearly the strain energy concentration at the buckling point. A unique advantage of using the multiscale method is that we are able to reveal the details of the molecular structure at the kinks, which cannot be resolved by the coarse scale representation alone. The atomic structure of the buckling region for each layer of multi-walled CNT are plotted on the right hand side of Figure 3.2.

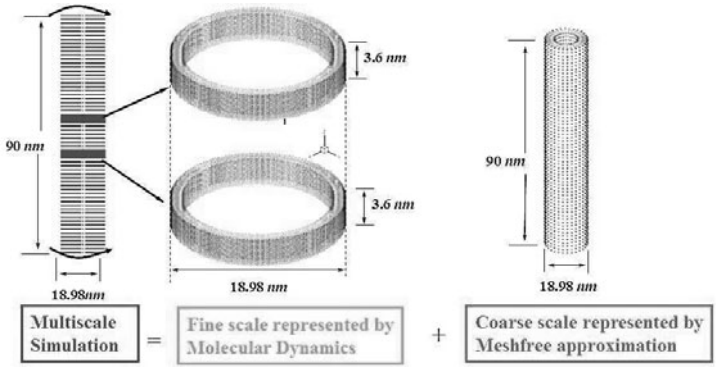


Figure 3.1. Multiscale analysis of a 15-walled CNT by a bridging scale method. The multiscale simulation model consists of 10 rings of carbon atoms (with 49,400 atoms each) and a meshfree continuum approximation of the 15-walled CNT by 27,450 nodes.

Grain and Grain Boundaries. Multiscale simulations of grain and grain boundaries are studied to observe the mechanical behaviors of the effects and behaviors in both coarse (homogeneous grain) and fine (grain boundaries) scales. This method uses the idea of being able to project the detailed (fine) grain boundary solution, which is assumed to be inhomogeneous, onto a continuum that consists of homogeneous grains. While maintaining the accuracy in the total solution, the computational cost is greatly reduced and the algorithm implementation is enhanced. The material properties used in finite element can be easily extracted from the free energy or the equilibrium state of the material. The multiscale solution is to be compared with full atomistic solution for simple numerical mechanical testing examples such as shear or compression. The mechanical effects are to be captured in both global and local areas. The results from both methods are expected to be almost identical, though the work is still in progress.

3.2 Dynamic Simulations

The dynamic formulation for the bridging scale method is obtained according to the Lagrangian formalism. Importantly, the decomposition Eq. (3.1) leads

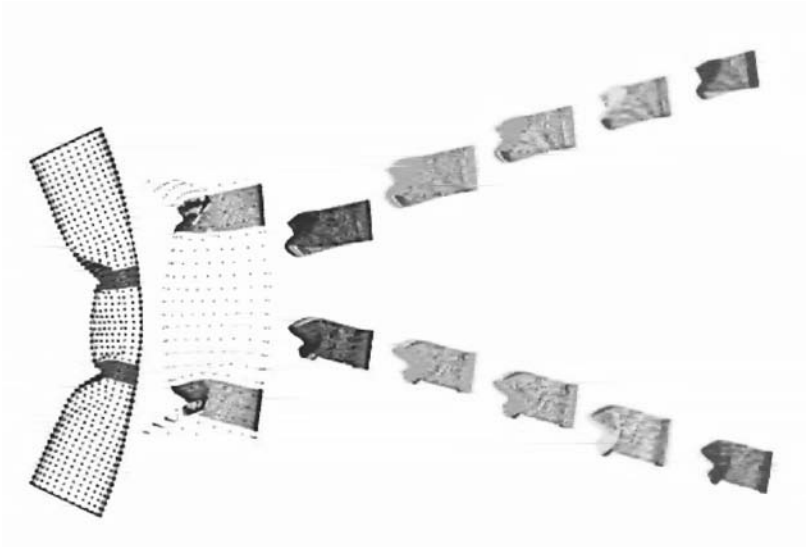


Figure 3.2. The global buckling pattern captured by meshfree method whereas the detailed local buckling of the ten rings of atoms are captured by a concurrent bridging scale molecular dynamic simulation. See Color Plate 12 on page 302.

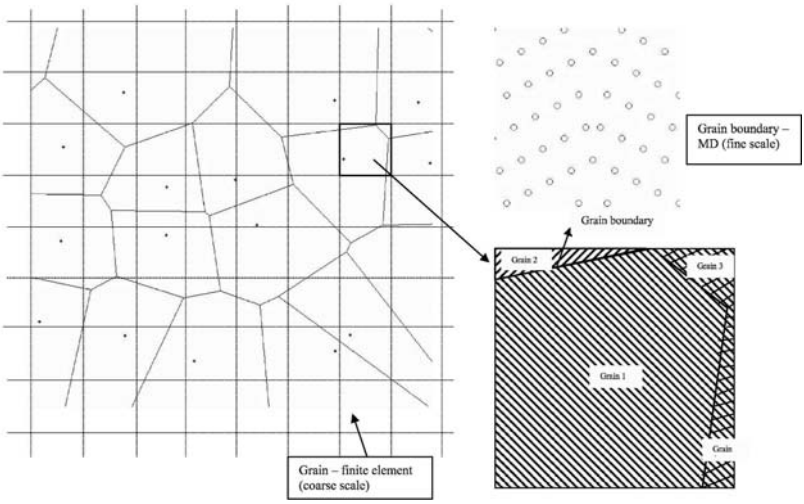


Figure 3.3. Grains are modelled using finite element (coarse scale and grain boundaries are modelled using molecular dynamics (fine scale).

to a Lagrangian of the system in which the kinetic energies of the two scales are uncoupled ([40]):

$$\mathcal{L} = \frac{1}{2}\dot{\mathbf{d}}^{\mathbf{T}}\mathbf{M}\dot{\mathbf{d}} + \frac{1}{2}\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{A}\dot{\mathbf{q}} - U(\mathbf{u}) \tag{3.7}$$

where $\mathbf{A} = \mathbf{Q}^{\mathbf{T}}\mathbf{M}_{\mathbf{A}}\mathbf{Q}$. This in turn provides a convenient form of force coupling between the coarse and fine scale equations of motion:

$$\begin{aligned} \mathbf{M}_A^f \ddot{\mathbf{q}}^f &= \mathbf{f}^f(\mathbf{u}) \\ \mathbf{M}\ddot{\mathbf{d}} &= \mathbf{N}^T\mathbf{f}(\mathbf{u}) \end{aligned} \tag{3.8}$$

where $\mathbf{M}$ and $\mathbf{M}_A$ are the FE and MD mass matrices, respectively. This form of the equations of motion formally implies that the atomistic MD and continuum FE solutions $\mathbf{q}$ and $\mathbf{d}$ exist simultaneously and everywhere in the computational domain; the upper script index "f" in the first equation stands for the full MD displacement and force vectors, and the atomic mass matrix. This concept is illustrated in Figure 3.4.

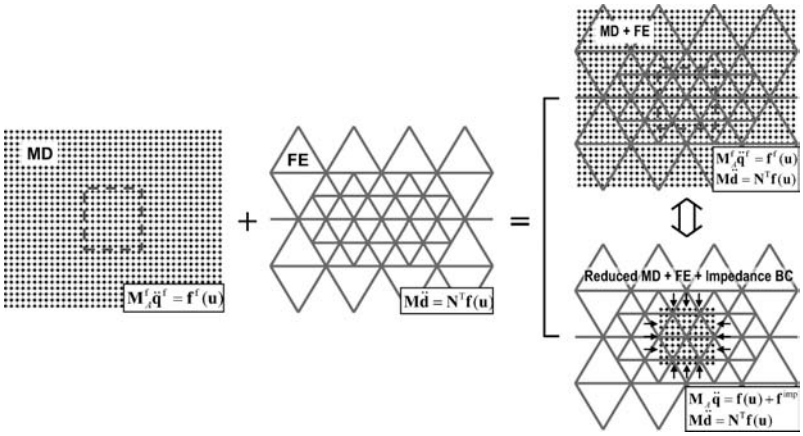


Figure 3.4. Illustration of the bridging scale approach: the MD and FE solutions are coupled through the projection technique. The ubiquitous atomistic resolution is replaced with a reduced MD region by utilizing the impedance boundary conditions. The dashed red line shows boundary of the domain of interest.

From the practical point of view, the explicit Lagrangian formulation Eq. (3.8) is of little merit, because it requires solving the MD equations throughout the continuum; that is in most cases not affordable computationally. Therefore, it is next assumed that the atomistic processes of interest

are localized on a small region that features large amplitudes of the relative atomic motion (the fine scale). The rest of the domain (the coarse scale) displays nonlinear elastic behavior. The atomistic degrees of freedom within the coarse scale are then eliminated from the formulation, and their cumulative effect upon the boundary atoms in the fine scale is taken into account through the impedance force, $\mathbf{f}^{imp}$, incorporated into the right-hand side of the reduced system of MD equations of motion:

$$\begin{aligned} \mathbf{M}_A\ddot{\mathbf{q}} &= \mathbf{f}^f(\mathbf{u}) + \mathbf{f}^{imp} \\ \mathbf{M}\ddot{\mathbf{d}} &= \mathbf{N}^T\mathbf{f}(\mathbf{u}) \end{aligned} \tag{3.9}$$

Here, the MD equation, as compared with Eq. (3.8), involves reduced force and displacement vectors and the mass matrix.

The impedance force $\mathbf{f}^{imp}$ is derived by first writing the fine scale equation of motion in component form as

$$\ddot{\mathbf{u}}'_{l,m} = \sum_{l'=l-1}^{l+1} \sum_{m'=m-\mu}^{m+\mu} \mathbf{M}_h^{-1}\mathbf{K}_{l-l',m-m'}\mathbf{u}'_{l',m'}(t) + \mathbf{M}_h^{-1}\mathbf{f}^{ext}_{l,m}(t) \tag{3.10}$$

where $\mathbf{f}^{ext}_{l,m}(t)$ is the external force acting upon unit cell (l, m), the matrices $\mathbf{K}$ relate the displacements in cell $(-l', -m')$ to the forces in cell (l, m), μ is the largest connectivity number for the index m, i.e. $m+\mu$ and $m-\mu$ are the most distant cells interacting with the current cell m, and l' and m' represent the range of the forces in the l and m coordinate directions.

By taking the Laplace and Fourier transforms of (3.10), an expression can eventually be reached which expresses the force exerted on the remaining fine scale degrees of freedom by the mathematically eliminated fine scale degrees of freedom. The structure of this external, or impedance force, resembles the generalized Langevin equation ([3], [4]),

$$\mathbf{f}^{imp}_m(t) = \sum_{m'} \int_0^t \theta_{m-m'}(t-\tau)(\mathbf{q}_{0,m'}(\tau) - \bar{\mathbf{u}}_{0,m'}(\tau))d\tau + \mathbf{R}(t) \tag{3.11}$$

where θ is a memory kernel function that describes renormalization of the atomic interaction along the boundary of reduced MD domain, and $\mathbf{R}$ is a random function that accounts for thermal atomic motion in the coarse scale. Importantly, the impedance force for the bridging scale formulation is obtained by utilizing the multiscale boundary conditions for lattice structures; therefore it involves analytically exact structural response at the atomistic level. This is a distinctive feature of the present method, as compared with the "ghost" atom technique to provide the interface displacements and forces based on the FE shape function interpolation. According to Eq. (3.11), the physical nature of the force Eq. (3.11) is due to inertia of the atomic lattice and thermal effects outside the MD domain. The coupled system of equations Eqs. (3.9) and (3.11) can be solved using existing FE and MD codes along with

suitable techniques for exchanging information about the internal forces. The kernel function θ can also be viewed as a lattice impedance matrix, since it encapsulates the impedance of the atomic structure in the coarse grain. The idea beneath the formulation Eqs. (3.9 - 3.11) is illustrated in Figure 3.4. As can be seen from this figure, the coarse scale FE model covers the entire model, and the atomistics are solved explicitly only on a selected region of interest. The atomic motion outside this domain is represented by the impedance boundary conditions according to Eqs. (3.9) and (3.11). The reduced MD domain then evolves in time similarly, as if it was still a part of the full MD domain.

One issue relating to the formulation Eqs. (3.9) and (3.11) is the large size of matrix θ, which is typically $m \times m$, where m is number of atomistic degrees of freedom along the MD domain boundary. Each element of this matrix has to be calculated for a time large enough to capture the impedance effects accurately. This also requires a numerical inversion of the Laplace transform that can itself be computationally expensive. However, recent works by Karpov et al [22] and Wagner et al [39] have shown that exploiting the intrinsic symmetry and spatial repetitiveness of the atomistic structure in crystalline solids can drastically simplify and reduce the size and computational expense associated with these matrices. Based on Fourier analysis of periodic structures ([20], [37], [21]), this approach provides the impedance boundary conditions in terms of the lattice response functions, also known as the lattice dynamics Green's function in solid-state physics ([21]). These functions provide a compact memory function, similar to θ of Eq. (3.11), to describe the re-normalization of the interatomic interaction at the boundaries of the simulated domain. The matrices in the memory kernel are only $m_B \times m_B$, where m_B is at most the number of degrees of freedom in a Bravais lattice ([30], [18]), i.e. one repetitive lattice cell. Though the original formalism Eq. (3.11) was derived assuming harmonic character of the atomic motion along the fine/coarse grain interface, the current form of the impedance force Eq. (3.11) provides a reliable first order approximation that treats satisfactory also moderately non-linear interfaces ([33], [32]). Karpov and Liu ([18]) have shown that for stronger non-linearities, the performance of the impedance boundary conditions can be improved with an update based on the perturbation approach. Finally, the quantum mechanical enrichment of the bridging scale formulation can be expressed, according to the MD/QM equations $\hat{H}\Psi = (U + \sum_\alpha E_\alpha)\Psi$ and $m_i \ddot{\mathbf{r}}_i = -\frac{\partial U}{\partial \mathbf{r}_i}$, and Eqs. (3.9) and (3.11) as:

$$\begin{aligned} \hat{H}\Psi &= (U + \sum_\alpha E_\alpha)\Psi, \quad \mathbf{f}(\mathbf{u}) = -\frac{\partial U}{\partial \mathbf{u}} \\ \mathbf{M_A}\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{u}) + \int_0^t \theta(t-\tau)(\mathbf{q}(\tau) - \bar{\mathbf{u}}(\tau))d\tau \\ \mathbf{M}\ddot{\mathbf{d}} &= \mathbf{N}^T\mathbf{f}(\mathbf{u}) \end{aligned} \tag{3.12}$$

This set of equations stands for the concurrent coupling of simulations at the three scales: sub-atomic, atomistic and continuum; in more details, see reference ([19]).

Wave Propagation. The dynamic bridging scale method with impedance boundary conditions on the MD/continuum interface was applied by Park et al. [32] in studying wave propagation, and crack initiation and growth in the (111) plane of a face-centered cubic lattice structure governed by a two-body Lennard-Jones potential $V = 4\epsilon[(^{\sigma}\ \ _{r})^{6}]$. Figure 3.5 shows a comparison of the MD region displacements after a Gaussian-type wave with a combination of high and low frequency components that originated in the MD region has propagated into the surrounding continuum region.

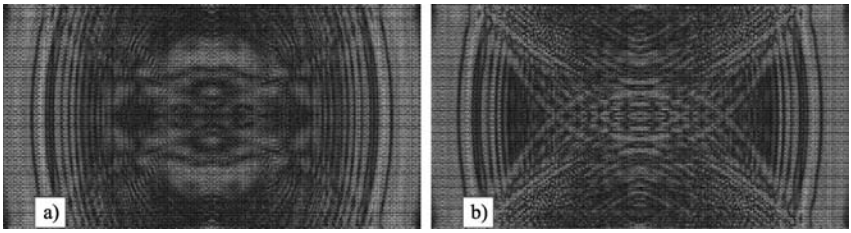


Figure 3.5. Wave propagation through the atomistic domain in the FCC lattice structure: a) impedance boundary conditions are involved at the MD/continuum interface, b) continuity boundary conditions. See Color Plate 13 on page 302.

By comparing the Figures 3.5a and 3.5b, it is evident that if the high frequency waves emitted from the MD region are not treated in a mathematically rigorous manner, they reflect back from the MD/continuum (finite element) interface. The high frequency reflections are seen in the wake of the wave that has departed the MD region. These high frequency waves must be properly accounted for, particularly if they represent the majority of the initial energy of the MD system.

Crack Propagation. In the crack propagation problem, Figure 3.6, considered by the same authors ([32]), the central part of the simulated domain contains a pre-crack that initiates and later propagates due to the tensile boundary conditions applied to the coarse scale. These loads are modeled with velocity boundary conditions applied to the boundaries of the coarse scale. Atomistic resolution is introduced in the area close to the crack. The coarse scale information eventually propagates into the MD region, and consequently the crack begins to propagate. Two snapshots of this process are given in Figures 3.6a and 3.6b; the first snap shot captures initiation of the crack, and the second shows the configuration just before complete separation of the structure. Accuracy of the bridging scale simulation was verified by comparing it with a benchmark MD simulation, where the atomistic resolution was set throughout the entire coarse grain. The authors observed a very

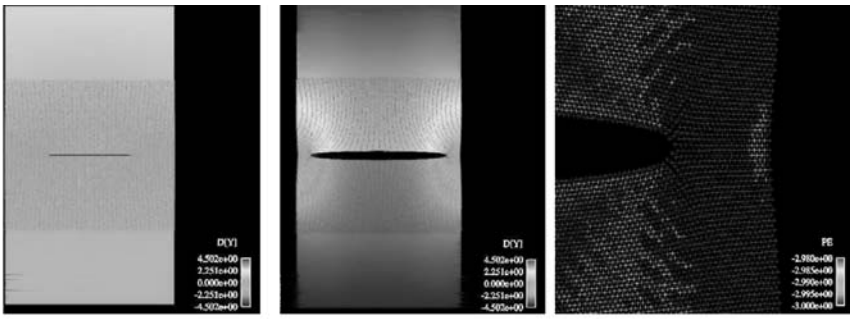


Figure 3.6. Bridging multiscale modeling of crack propagation: a) initiation of a crack in the MD region, b) pre-separation phase, c) lattice dislocation pattern at the crack tip. See Color Plate 14 on page 303.

good agreement of the results: 1) the crack propagation speed was virtually identical in both cases; 2) the bridging scale model captures all specific details of the process within the fine grain, such as initiation and emission of the lattice dislocations away from the crack tip, see Figure 3.6.

For sufficiently large initial velocities, the authors observed complete fracture of the atomic lattice into two separate sections, depicted in Figure 3.7.

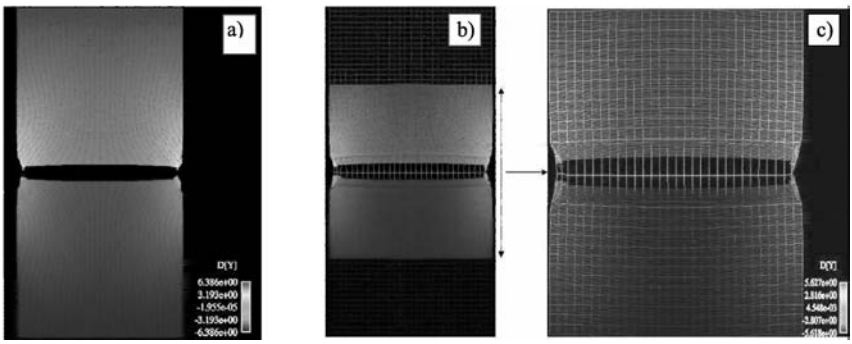


Figure 3.7. Comparison of crack simulations at the lattice separation stage: a) the full MD model, b) the bridging multiscale model. c) The subdomain of coexisting of the MD and FE solutions (zoom in). See Color Plate 15 on page 303.

The multiscale simulation of the entire region is shown in Figure 3.7b, where the FE mesh is present everywhere, and MD model in a smaller subsection of the total domain. The zoom Figure 3.7c shows only the region in which both the FE and MD models exist, as opposed to the entire domain. The bridging scale simulation can be seen to agree very well with the benchmark

MD simulation, Figure 3.7a. It is also noteworthy that complete fracture of the underlying MD lattice is allowed in the coupled simulation - this is because the finite element simulation in that region is simply carried along by the MD simulation. The crack opening is shown in the Figure 3.7c by magnifying the y-component of the displacement by a factor of three.

4 Bridging Scale Computations for Localized Failure

In the analysis of failure phenomena with rate-independent strain-softening material, one needs a regularization method to avoid spurious mesh dependence [7]. Also, in order to capture the detail deformation inside the localized region, very fine discretization is required. In this study, the micropolar continuum [11] is employed to regularize the numerical solution, and the bridging scale computation [29, 38, 15, 40, 39, 22, 17, 33, 32] is utilized in the coupling of two FEM calculations of different resolutions to achieve accuracy and computational efficiency at the same time.

4.1 Governing Equations of the Micropolar Continuum

In addition to the displacement fields u_i, a micro-rotation field ω_{ij} is involved in the micropolar continuum. Following the approach of Germain [12] with the principle of virtual power, the governing equations can be written as follows:

$$\begin{aligned}\int_\Omega (\rho\ddot{u}_i\delta\dot{u}_i + \rho I\ddot{\omega}_{ij}\delta\dot{\omega}_{ij})\,\mathrm{d}\Omega = \\ -\int_\Omega \left[\sigma^S_{ji}\delta D_{ij} + \sigma^A_{ji}\left(\delta\dot{\omega}_{ij} - \delta W_{ij}\right) + \tau_{kji}\delta\dot{\omega}_{ij,k}\right]\mathrm{d}\Omega \\ +\int_\Omega (b_i\delta\dot{u}_i + B_{ji}\delta\dot{\omega}_{ij})\,\mathrm{d}\Omega + \int_\Gamma (t_i\delta\dot{u}_i + T_{ji}\delta\dot{\omega}_{ij})\,\mathrm{d}\Gamma\end{aligned} \tag{4.1}$$

for all $\delta\dot{u}_i$ and $\delta\dot{\omega}_{ij}$, where Ω and Γ are the domain and its boundary, respectively. Superposed dots represent the material time derivatives. Inside each integral in (4.1), the first terms are identical to those of the conventional continuum. Material constants ρ and ρI represent the density of mass and the density of inertia of micro-rotation, respectively. The symmetric and the skew-symmetric part of the virtual velocity gradient $\frac{\partial\delta\dot{u}_i}{\partial x_j}$ are denoted by δD_{ij} and δW_{ij}. The conventional symmetric Cauchy stress is denoted by σ^S_{ji}. A comma in a subscript of a tensor denotes the spatial derivative. A skew-symmetric stress is denoted by σ^A_{ji} which is called micro-stress. A third-order tensor τ_{kji} is called couple-stress and is skew-symmetric in terms of the second and the third subscripts. The conventional body force and surface traction are denoted by b_i and t_i. The body-couple and the surface moment are denoted by B_{ji} and T_{ji}.

The plastic response of this model is described by the generalized J_2-flow theory of de Borst [10] with small modifications. The generalized stress invariant J_2 and the time derivative of the generalized plastic strain invariant γ are defined as follows:

$$J_2 = (a_1 + a_2)\, s_{ij} s_{ij} + (a_1 - a_2)\, \sigma^A_{ij} \sigma^A_{ij} + 2a_3 \tau_{kji} \tau_{kji} / \ell^2 \tag{4.2}$$

$$\dot{\gamma} = \left[(b_1 + b_2)\, \dot{e}^p_{ij} \dot{e}^p_{ij} + (b_1 - b_2) \left(\dot{\omega}^p_{ij} - W^p_{ij} \right) \left(\dot{\omega}^p_{ij} - W^p_{ij} \right) + \frac{b_3}{2} \dot{\omega}^p_{ij,k} \dot{\omega}^p_{ij,k} \ell^2 \right]^{1/2} \tag{4.3}$$

where s_{ij} denotes the deviatoric portion of σ^S_{ij}. The superscript p on the strain-rate components indicates their plastic contribution, and $\dot{e}^p_{ij}$ denotes the deviatoric portion of the plastic contribution of D_{ij}. The material length scale parameter is denoted by ℓ. The yield stress $\bar{\sigma}$ is a function of γ. It decreases from the initial value $\bar{\sigma}_0$ at the beginning of the plastic deformation and stays constant after the strain invariant reaches a specified value γ_c.

4.2 Bridging Scale Computations

Consider a problem where we are mainly interested in the deformation inside a very small region compared to the entire domain. This region is called *the domain of interest*. This method starts with the decomposition of the displacement field into the coarse part and the fine part as follows (Fig. 4.8):

$$\mathbf{u} = \boldsymbol{\mathcal{N}} \mathbf{d} + \mathbf{Q} \mathbf{q} \tag{4.4}$$

where operators $\boldsymbol{\mathcal{N}}$ and $\mathbf{Q}$ are determined so that the shared information between two FEM computations is correctly subtracted [40]. This decomposition splits the equation of motion into the coarse-scale part and the fine-scale part:

$$\begin{aligned} \mathbf{M} \ddot{\mathbf{d}} &= \boldsymbol{\mathcal{N}}^T \left(-\mathbf{f}^{\text{int}} (\boldsymbol{\mathcal{N}} \mathbf{d} + \mathbf{Q} \mathbf{q}) + \mathbf{f}^{\text{ext}} \right) \\ \mathbf{M}_A \ddot{\mathbf{q}} &= -\mathbf{f}^{\text{int}} (\mathbf{q}) + \mathbf{f}^{\text{ext}} \end{aligned} \tag{4.5}$$

In order to reduce the computational cost, the calculation of the degrees of freedom (DOFs) of the fine-scale FEM outside the domain of interest is avoided. The effects of these eliminated DOFs are substituted by dynamic force applied along the interfaces. It is called *impedance force* and is derived analytically by the Fourier analysis of a repetitive structures [20, 21](Fig. 4.9). Thus the governing equations for the fine-scale FEM are reduced to

$$\mathbf{M}_{Aa} \ddot{\mathbf{q}}_a = -\mathbf{f}^{\text{int}}_a \left(\boldsymbol{\mathcal{N}} \mathbf{d} + \mathbf{Q} \begin{bmatrix} \mathbf{q}_a \\ \mathbf{0} \end{bmatrix} \right) - \mathbf{f}^{\text{imp}} + \mathbf{f}^{\text{ext}}_a \tag{4.6}$$

where $\mathbf{f}^{\text{int}}_a, \mathbf{f}^{\text{ext}}_a$, and $\mathbf{q}_a$ denote the nodal internal force, external force, and displacement inside the domain of interest, respectively. The lumped mass matrix for the nodes inside this region is denoted by $\mathbf{M}_{Aa}$. The impedance force applied along the edge of the domain of interest is denoted by $\mathbf{f}^{\text{imp}}$.

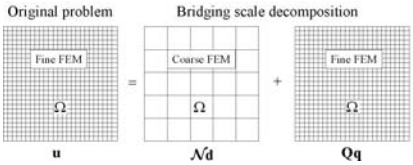


Figure 4.8. Bridging scale decomposition.

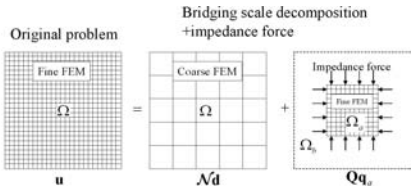


Figure 4.9. Reduction of the model size.

4.3 Numerical Example

A two-dimensional shear failure problem is solved with the micropolar continuum and the proposed bridging multi-scale method. A bar of the size of 0.4m × 0.04m is compressed in the longitudinal direction by the prescribed velocity at both ends (Fig. 4.10). The entire domain is discretized by 80 × 8 coarse-scale finite elements, while the central portion of 0.155m is discretized by the fine-scale finite elements. The fine-scale mesh is aligned to the coarse-scale mesh near the interfaces, while it is rotated 45 degrees in the middle of the bar in order to align to the shear bands. When the mesh was not aligned to the shear band, deformation did not localize sharply. The fine-scale mesh has 17920 elements and 8 × 8 fine-scale elements occupy the same area as one coarse-scale element near the interface. All the boundaries are moment free, and the top and the bottom boundaries are stress free. Fig. 4.11 shows the accumulated effective plastic shear strain distribution which is regularized by the micropolar continuum. Fig. 4.12 and Fig. 4.13 show the total shear-strain profile inside the domain of interest just after the waves created by the dynamic failure have passed away from this region. The proposed dynamic interface prevents most of the reflection of high frequency waves, while one can clearly see some reflections when the continuity of displacement is enforced at the interfaces.

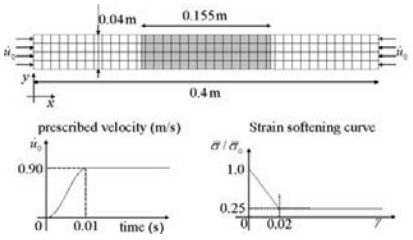


Figure 4.10. Two-dimensional shear localization problem (plane strain).

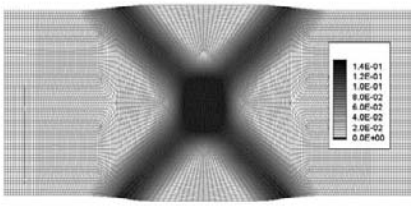


Figure 4.11. Regularized solution with the micropolar continuum.

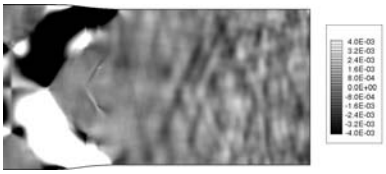


Figure 4.12. Reflected high frequency waves when continuity of displacement is enforced at the interfaces.

Figure 4.13. Reflected high frequency waves of the proposed dynamic interface condition.

5 Conclusions

We have reviewed the bridging scale method utilized in computational nano mechanics and materials, including the relevant underlying principles and concepts. In great generality, three major issues are still to be challenged by future researchers in the area of multiple scale simulations. The first is to correctly account for the non-harmonic high frequency information that emanates from the molecular simulation when the information reaches the continuum. The second is the temperature dependent multiscale formulations; in more details this issue is discussed in [32]. And the third is extending the time range currently available in standard MD simulations, so that the continuum and atomistic simulations may each evolve naturally on its natural time scale. It is noted that while the multi-scale methods reviewed in this work reduce the spatial computational requirements, they are still limited by the fact that MD simulations cannot be run for arbitrarily long periods of time. One of the efforts ongoing to relieve this restriction is the work by Voter and co-workers on the so-called hyper molecular dynamics approach. This method appears well suited for problems in which the physical phenomena of interest occurs infrequently over long periods of time, with diffusion being a prime example. Fundamental nanoscale research is being performed all around the world, and as this research is more and more being turned into viable engineering applications, our ability to model the performance of nanoscale structures remains limited. Continuum-based computational approaches are clearly not applicable over the full range of operational conditions for these nanoscale devices, as non-continuum behavior is observed in the large deformation behavior of carbon nanotubes, ion deposition processes, material mechanics, amongst many others. More crucially, nanoscale components will likely be used in conjunction with components that are larger, and therefore have a mechanical response that is on much larger length and time scales than the nanoscale component. In such hybrid systems, typical single scale simulation methods such as molecular dynamics or quantum mechanics may not be applicable due to the disparity in length and time scales of the structure. For such systems, the computer-aided engineering tools must be able to span length scales from nanometers to microns, and time scales from femtoseconds to microseconds. Therefore,

these systems cannot be modeled by continuum methods alone, because they are too small, or by molecular methods alone, because they are too large. To support the design and qualification of nano-structured materials, a range of simulation tools must be available to designers just as they are today available at the macroscopic scales in general purpose software. However, considerable research is still required to establish the foundations for such software and to develop computational capabilities that span the scales from the atomistic to continuum. These capabilities should include a variety of tools, from finite elements to molecular dynamics and quantum mechanical methods, in order to provide powerful multiscale methodologies. We hope that this work will be viewed as a step in the right direction in making the multi-scale goal a reality.

References

1. F. F. Abraham, J. Broughton, N. Bernstein, and E. Kaxiras. Spanning the continuum to quantum length scales in a dynamic simulation of brittle fracture. *Europhysics Letters*, 44:783–787, 1998.
2. F. F. Abraham, J. Q. Broughton, N. Bernstein, and E. Kaxiras. Spanning the length scales in dynamic simulation. *Computers in Physics*, 12:538–546, 1998.
3. S. Adelman and J. Doll. Generalized langevin equation approach for atom/solid-surface scattering - collinear atom/harmonic chain model. *Journal of Chemical Physics*, 61(10):4242–4245, 1974.
4. S. Adelman and J. Doll. Generalized langevin equation approach for atom-solid-surface scattering - general formulation for classical scattering off harmonic solids. *Journal of Chemical Physics*, 64(6):2375–2388, 1976.
5. T. Belytschko, Y.Y. Lu, and L. Gu. Element-free galerkin methods. *International Journal for Numerical Methods in Engineering*, 37:229–256, 1994.
6. T. Belytschko, Y.Y. Lu, L. Gu, and M. Tabarra. Element-free galerkin methods for static and dynamic fracture. *International Journal of Solids and Structures*, 32(17):2547–2570, 1995.
7. Ted Belytschko, Wing Kam Liu, and Brian Moran. *Nonlinear Finite Elements for Continua and Structures*. John Wiley, Chichester ; New York, 2000.
8. J. Bernholc, C. Brabec, M.B. Nardelli, A. Maiti, C. Roland, and B.I. Yakobson. Theory of growth and mechanical properties of nanotubes. *Applied Physics A-Materials Science and Processing*, 67(1):39–46, 1998.
9. J.Q. Broughton, F.F. Abraham, N. Bernstein, and E. Kaxiras. Concurrent coupling of length scales: methodology and application. *Physical Review B*, 60(4):2391–2403, 1999.
10. R. de Borst. A generalization of j2-flow theory for polar continua. *Computer Methods in Applied Mechanics and Engineering*, 103(3):347–362, 1993.
11. A. C. Eringen and E. S. Suhubi. Nonlinear theory of simple micro-elastic solids. *International Journal of Engineering Science*, 2:189–203, 1964.
12. P. Germain. Method of virtual power in continuum mechanics .2. microstructure. *Siam Journal on Applied Mathematics*, 25(3):556–575, 1973.
13. F.C. Günther and Wing Kam Liu. Implementation of boundary conditions for meshless methods. *Computer Methods in Applied Mechanics and Engineering*, 163(1-4):205–230, 1998.

14. F.C. Günther, Wing Kam Liu, and M.A. Christon. Multiscale meshfree parallel computations for viscous, compressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190:279–303, 2000.
15. W. M. Han, G. J. Wagner, and Wing Kam Liu. Convergence analysis of a hierarchical enrichment of dirichlet boundary conditions in a mesh-free method. *International Journal for Numerical Methods in Engineering*, 53(6):1323–1336, 2002.
16. T.J. R. Hughes. *The finite element method: linear static and dynamic finite element analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1987.
17. H. Kadowaki and Wing Kam Liu. Bridging multiscale method for localization problems. *Computer Methods in Applied Mechanics and Engineering*, accepted, 2003.
18. E.G. Karpov and Wing Kam Liu. A non-linear operator for impedance and multiscale boundary conditions in solids. in preparation, 2003.
19. E.G. Karpov and Wing Kam Liu. On the concurrent multiscale coupling of quantum mechanical and molecular dynamics simulations of nanostructured materials. In preparation, 2003.
20. E.G. Karpov, N.G. Stephen, and D.L. Dorofeev. On static analysis of finite repetitive structures by discrete fourier transform. *International Journal of Solids and Structures*, 39(16):4291–4310, 2002.
21. E.G. Karpov, N.G. Stephen, and Wing Kam Liu. Initial tension in randomly disordered periodic lattices. *International Journal of Solids and Structures*, 40(20):5371–5388, 2003.
22. E.G. Karpov, G.J. Wagner, and Wing Kam Liu. A green's function approach to deriving wave-transmitting boundary conditions in molecular dynamics simulations. *Comp. Materials Science*, submitted, 2003.
23. S. Li and Wing Kam Liu. Meshfree and particle methods and their applications. *Applied Mechanics Reviews*, 55:1–34, 2002.
24. Wing Kam Liu and Y. Chen. Wavelet and multiple scale reproducing kernel methods. *International Journal for Numerical Methods in Fluids*, 21:901–931, 1995.
25. Wing Kam Liu, Y. Chen, C.T. Chang, and T. Belytschko. Advances in multiple scale kernel particle methods. *Computational Mechanics*, 18(2):73–111, 1996.
26. Wing Kam Liu, S. Jun, D.T. Sihling, Y. Chen, and W. Hao. Multiresolution reproducing kernel particle method for computational fluid dynamics. *International Journal for Numerical Methods in Fluids*, 24:1391–1415, 1997.
27. Wing Kam Liu, S. Jun, and Y.F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20(8-9):1081–1106, 1995.
28. Wing Kam Liu, S. Li S. Jun, J. Adee, and T. Belytschko. Reproducing kernel particle methods for structural dynamics. *International Journal for Numerical Methods in Engineering*, 38:1655–1679, 1995.
29. Wing Kam Liu, R. A. Uras, and Y. Chen. Enrichment of the finite element method with the reproducing kernel particle method. *Journal of Applied Mechanics-Transactions of the Asme*, 64(4):861–870, 1997.
30. T.C.W. Mak and G.D. Zhou. *Crystallography in modern chemistry: a resource book of crystal structures*. Wiley, New York, 1992.
31. H.S. Park, E.G. Karpov, and Wing Kam Liu. A temperature equation for coupled atomistic/continuum simulations. *Computer Methods in Applied Mechanics and Engineering: Special Issue on Multiscale Nano Mechanics and Materials*, accepted, 2003.

32. H.S. Park, E.G. Karpov, Wing Kam Liu, and P.A. Klein. The bridging scale for two-dimensional atomistic/continuum coupling. *Philosophical Magazine A*, submitted, 2003.
33. H.S. Park and Wing Kam Liu. An introduction and tutorial on multiple scale analysis in solids. *Computer Methods in Applied Mechanics and Engineering: Special Issue on Multiscale Nano Mechanics and Materials*, accepted, 2003.
34. D. Qian, Wing Kam Liu, and R.S. Ruoff. Mechanics of c60 in nanotubes. *Journal of Physical Chemistry B*, 105:10753–10758, 2001.
35. D. Qian, G.J. Wagner, and Wing Kam Liu. A multiscale projection method for the analysis of carbon nanotubes. *Computer Methods in Applied Mechanics and Engineering*, Accepted, 2003.
36. D. Qian, G.J. Wagner, Wing Kam Liu, M.F. Yu, and R.S. Ruoff. Mechanics of carbon nanotubes. *Applied Mechanics Review*, 55(6):495–533, 2002.
37. M. Ryvkin, M.B. Fuchs, and B. Nuller. Optimal design of infinite repetitive structures. *Structural Optimization*, 18(2-3):202–209, 1999.
38. G. J. Wagner and Wing Kam Liu. Hierarchical enrichment for bridging scales and mesh-free boundary conditions. *International Journal for Numerical Methods in Engineering*, 50(3):507–524, 2001.
39. G.J. Wagner, E.G. Karpov, and Wing Kam Liu. Molecular dynamics boundary conditions for regular crystal lattices. *Computer Methods in Applied Mechanics and Engineering: Special Issue on Multiscale Nano Mechanics and Materials*, 2003.
40. G.J. Wagner and Wing Kam Liu. Coupling of atomistic and continuum simulations using a bridging scale decomposition. *Journal of Computational Physics*, 190:249–274, 2003.
41. B.I. Yakobson, C.J. Brabec, and J. Bernholc. Nanomechanics of carbon tubes: instabilities beyond linear response. *Physical Review Letters*, 76(14):2511–2514, 1996.
42. M.F. Yu, M.J. Dyer, J. Chen, D. Qian, Wing Kam Liu, and R.S. Ruoff. Locked twist in multiwalled carbon nanotube ribbons. *Physical Review B, Rapid Communications*, 64:241403R, 2001.
43. O.C. Zienkiewicz and R.L. Taylor. *The finite element method.* McGraw-Hill, London, New York, 1991.

Appendix

Color Plates

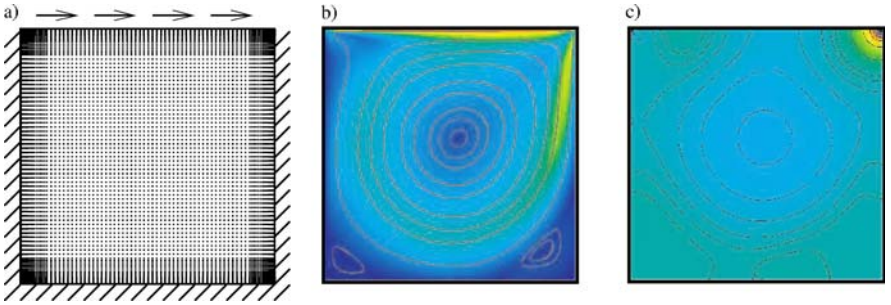


Plate 1. (Figure 4.6 on page 50) a) Data of the driven cavity test case with irregular particle distribution. Velocity b) and pressure c) fields for $Re = 1000$.

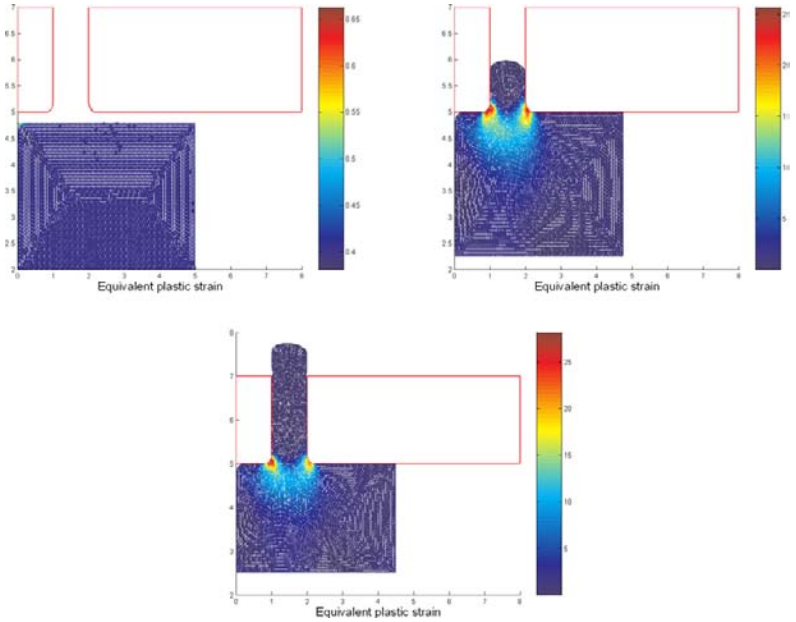
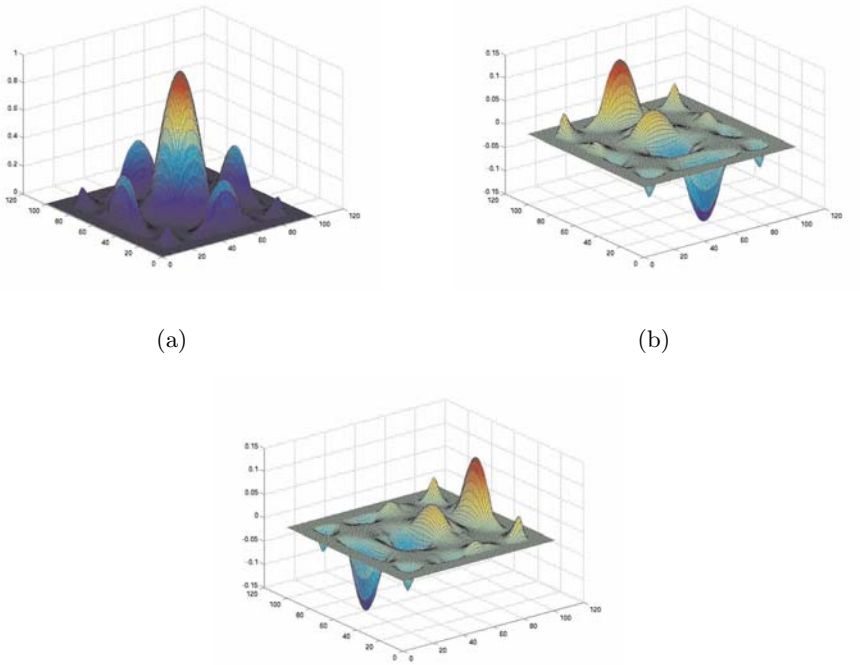


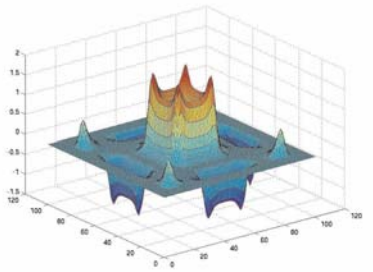
Plate 2. (Figure 5.10 on page 66) Equivalent plastic strain for time steps 1, 100 and 200.



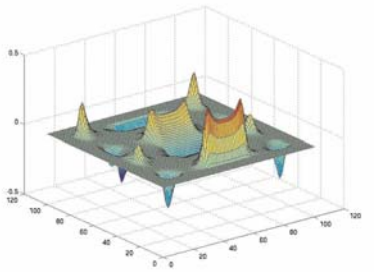
(a) (b)

(c)

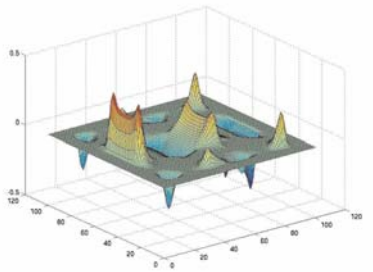
Plate 3. (Figure 3.3 on page 117) Smooth quadrilateral I^1/C^4 RKEM interpolant: (a) the 1st shape function, $\Psi_I^{(00)}(\boldsymbol{x})$, (b) the 2nd shape function $\Psi_I^{(10)}(\boldsymbol{x})$, (c) the 3rd shape function $\Psi_I^{(01)}(\boldsymbol{x})$.



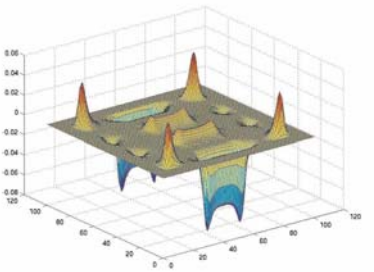
(a)



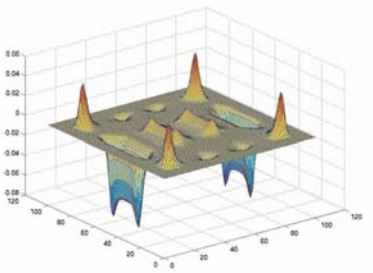
(b)



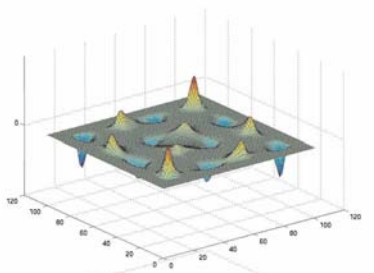
(c)



(d)



(e)



(f)

Plate 4. (Figure 3.4 on page 119) Compatible quadrilateral $I^2/C^4/P^2$ RKPM interpolants: (a) the 1st shape function, $\Psi_I^{(00)}(\boldsymbol{x})$, (b) the 2nd shape function $\Psi_I^{(10)}(\boldsymbol{x})$, (c) the 3rd shape function $\Psi_I^{(01)}(\boldsymbol{x})$ (d) the 4th shape function, $\Psi_I^{(20)}(\boldsymbol{x})$, (e) the 5th shape function $\Psi_I^{(02)}(\boldsymbol{x})$, (f) the 6th shape function $\Psi_I^{(11)}(\boldsymbol{x})$.

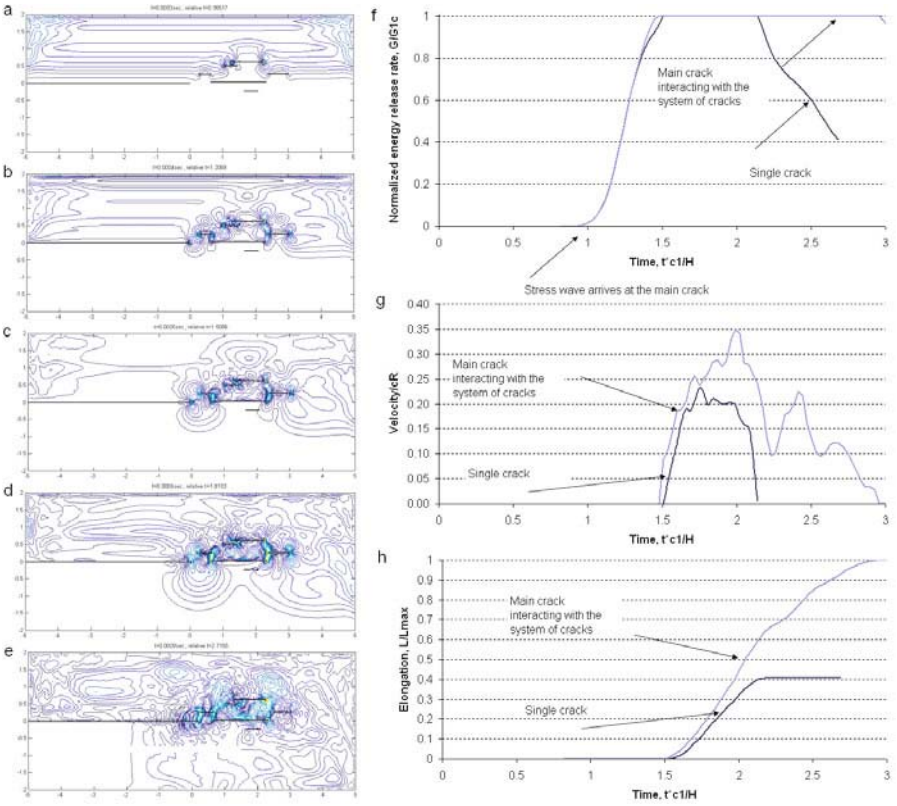


Plate 5. (Figure 5 on page 165) (a)-(e) Von-Mises stress distribution for the main crack interacting with the system of micro-crack and (f) energy release rate, (g) velocity and (h) elongation of the propagating main crack interacting with a system of micro-cracks compared with a single main crack.

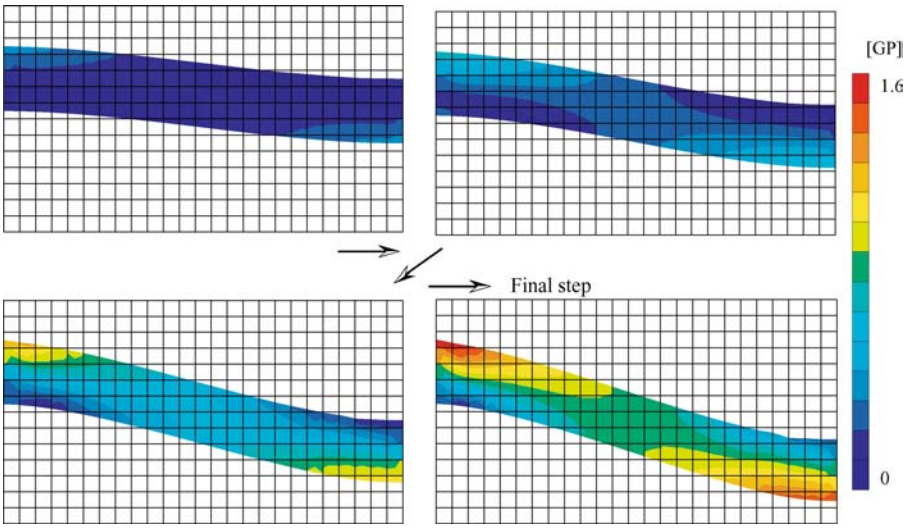
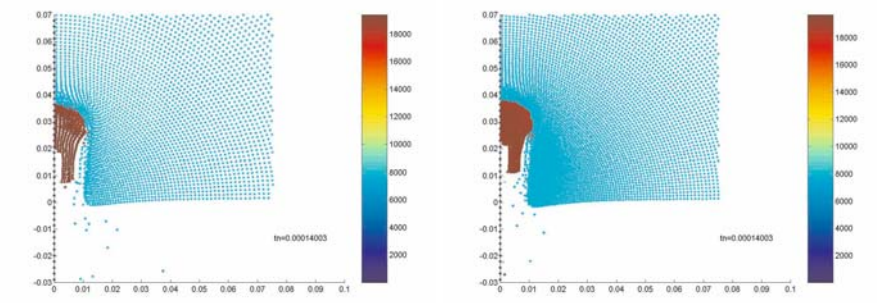
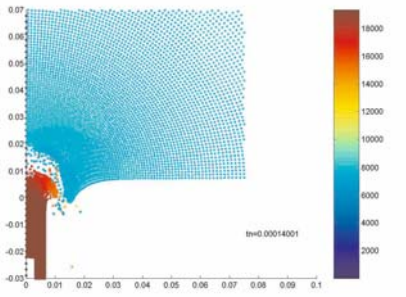


Plate 6. (Figure 4.18 on page 188) Deformed configuration and stress distribution along with the spatially fixed mathematical mesh.

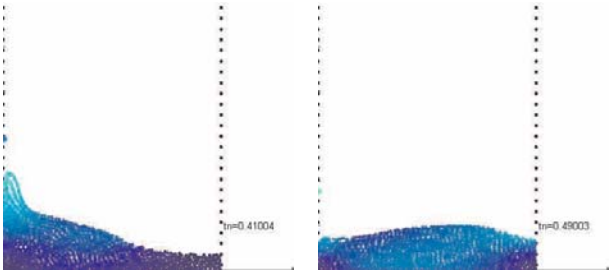


(a) Case 1 (b) Case 2

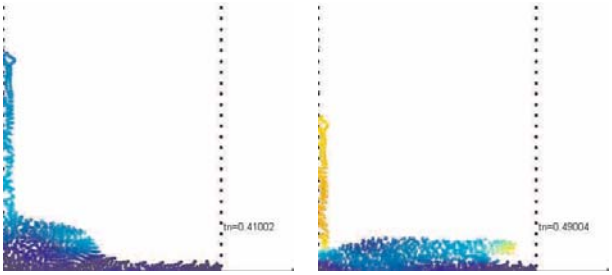


(c) Case 3

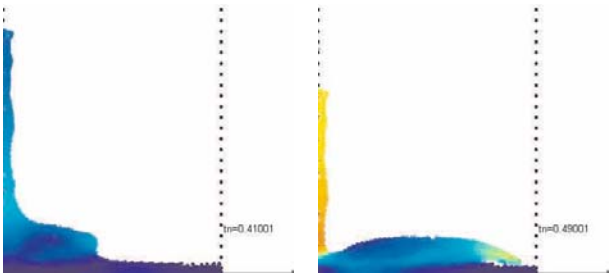
Plate 7. (Figure 5.2 on page 221) Impact of Tungsten projectile.



(a) Case 1 RS 20x60

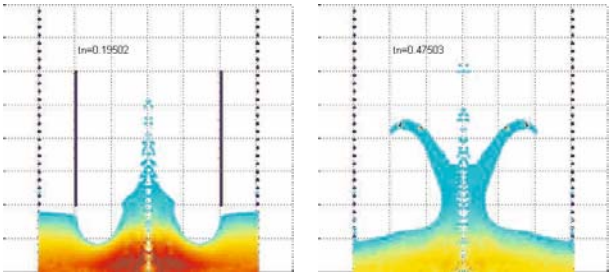


(b) Case 2 RGM 20x60

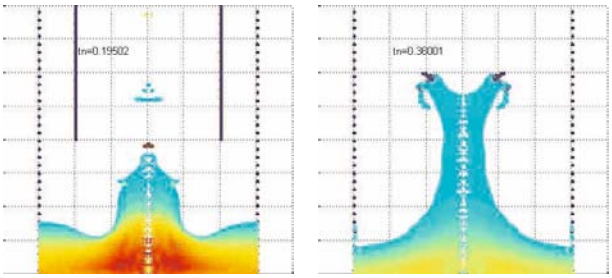


(c) Case 3 RGM 40x120

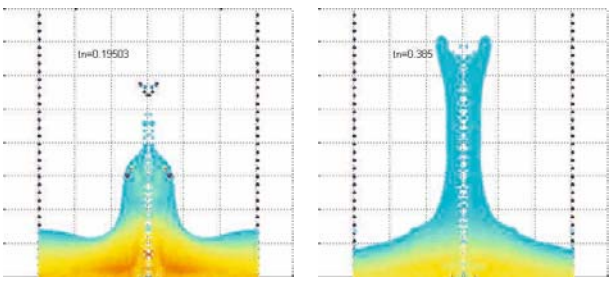
Plate 8. (Figure 5.4 on page 224) Dam break problem.



(a) Case 1 a=3.5 ms^{-2}

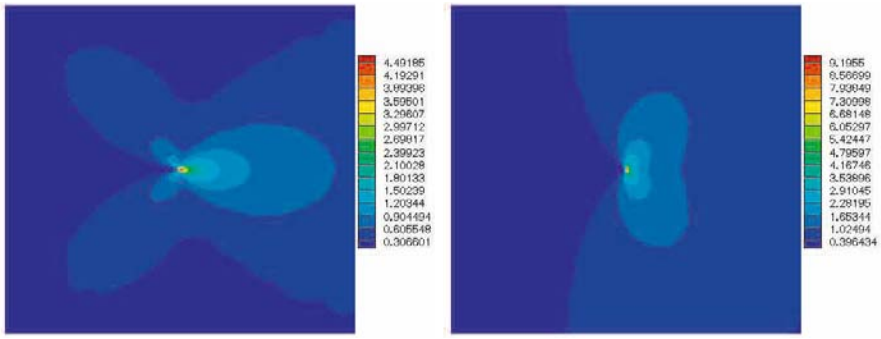


(b) Case 2 a=7.0 ms^{-2}



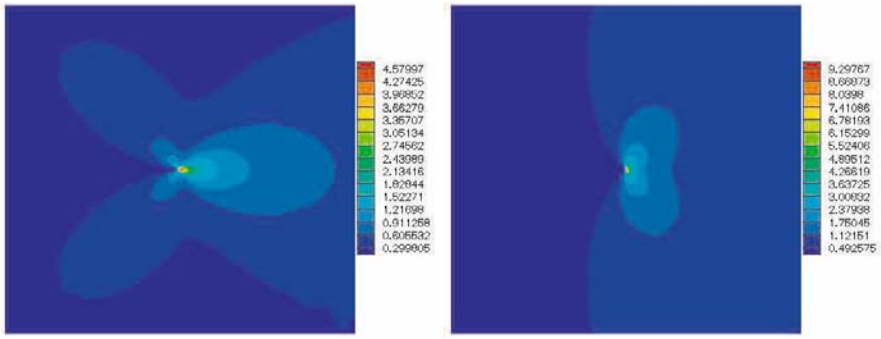
(c) Case 3 a=14.0 ms^{-2}

Plate 9. (Figure 5.5 on page 225) Geyser formation ($h = .015$ m).



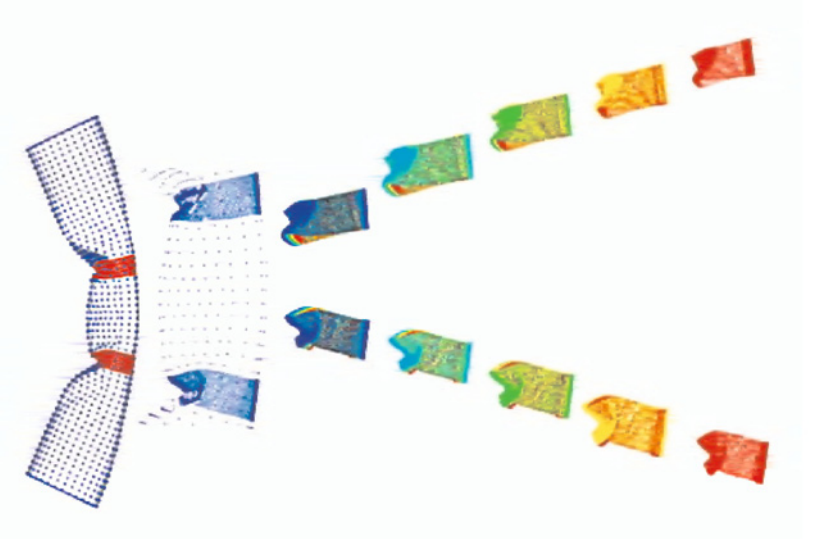
(a) (b)

Plate 10. (Figure 5.16 on page 250) Stress contours for static crack by the enriched EFG. (a) σ_{xx}, (b) σ_{yy}.



(a) (b)

Plate 11. (Figure 5.17 on page 250) Stress contours for static crack by the enriched RBF. (a) σ_{xx}, (b) σ_{yy}.



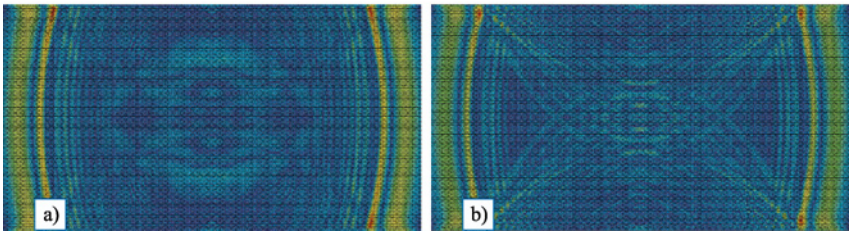


Plate 13. (Figure 3.5 on page 282) Wave propagation through the atomistic domain in the FCC lattice structure: a) impedance boundary conditions are involved at the MD/continuum interface, b) continuity boundary conditions.

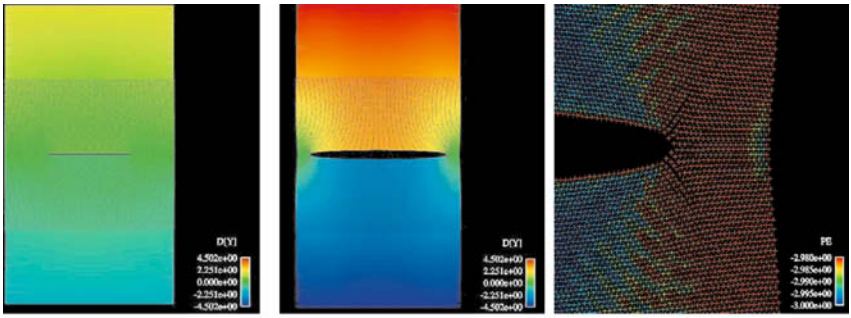


Plate 14. (Figure 3.6 on page 283) Bridging multiscale modeling of crack propagation: a) initiation of a crack in the MD region, b) pre-separation phase, c) lattice dislocation pattern at the crack tip.

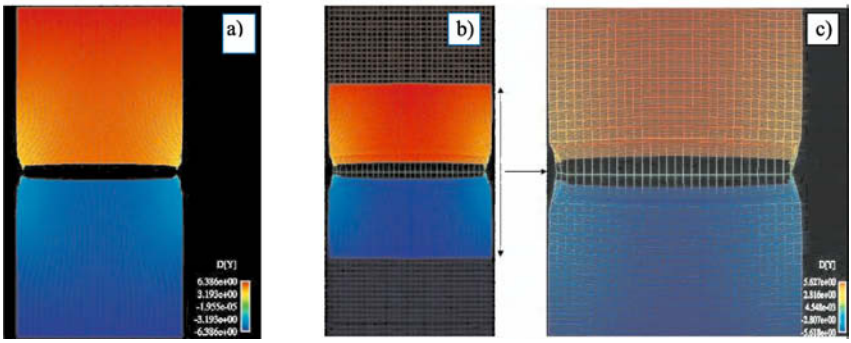


Plate 15. (Figure 3.7 on page 283) Comparison of crack simulations at the lattice separation stage: a) the full MD model, b) the bridging multiscale model. c) The subdomain of coexisting of the MD and FE solutions (zoom in).

Editorial Policy

§1. Volumes in the following three categories will be published in LNCSE:

i) Research monographs
ii) Lecture and seminar notes
iii) Conference proceedings

Those considering a book which might be suitable for the series are strongly advised to contact the publisher or the series editors at an early stage.

§2. Categories i) and ii). These categories will be emphasized by Lecture Notes in Computational Science and Engineering. **Submissions by interdisciplinary teams of authors are encouraged.** The goal is to report new developments – quickly, informally, and in a way that will make them accessible to non-specialists. In the evaluation of submissions timeliness of the work is an important criterion. Texts should be well-rounded, well-written and reasonably self-contained. In most cases the work will contain results of others as well as those of the author(s). In each case the author(s) should provide sufficient motivation, examples, and applications. In this respect, Ph.D. theses will usually be deemed unsuitable for the Lecture Notes series. Proposals for volumes in these categories should be submitted either to one of the series editors or to Springer-Verlag, Heidelberg, and will be refereed. A provisional judgment on the acceptability of a project can be based on partial information about the work: a detailed outline describing the contents of each chapter, the estimated length, a bibliography, and one or two sample chapters – or a first draft. A final decision whether to accept will rest on an evaluation of the completed work which should include

– at least 100 pages of text;
– a table of contents;
– an informative introduction perhaps with some historical remarks which should be accessible to readers unfamiliar with the topic treated;
– a subject index.

§3. Category iii). Conference proceedings will be considered for publication provided that they are both of exceptional interest and devoted to a single topic. One (or more) expert participants will act as the scientific editor(s) of the volume. They select the papers which are suitable for inclusion and have them individually refereed as for a journal. Papers not closely related to the central topic are to be excluded. Organizers should contact Lecture Notes in Computational Science and Engineering at the planning stage.

In exceptional cases some other multi-author-volumes may be considered in this category.

§4. Format. Only works in English are considered. They should be submitted in camera-ready form according to Springer-Verlag's specifications.
Electronic material can be included if appropriate. Please contact the publisher.
Technical instructions and/or TEX macros are available via
http://www.springeronline.com/sgw/cda/frontpage/0,10735,5-111-2-71391-0,00.html
The macros can also be sent on request.

General Remarks

Lecture Notes are printed by photo-offset from the master-copy delivered in camera-ready form by the authors. For this purpose Springer-Verlag provides technical instructions for the preparation of manuscripts. See also *Editorial Policy*.

Careful preparation of manuscripts will help keep production time short and ensure a satisfactory appearance of the finished book.

The following terms and conditions hold:

Categories i), ii), and iii):
Authors receive 50 free copies of their book. No royalty is paid. Commitment to publish is made by letter of intent rather than by signing a formal contract. Springer-Verlag secures the copyright for each volume.

For conference proceedings, editors receive a total of 50 free copies of their volume for distribution to the contributing authors.

All categories:
Authors are entitled to purchase further copies of their book and other Springer mathematics books for their personal use, at a discount of 33,3 % directly from Springer-Verlag.

Addresses:

Timothy J. Barth
NASA Ames Research Center
NAS Division
Moffett Field, CA 94035, USA
e-mail: barth@nas.nasa.gov

Michael Griebel
Institut für Angewandte Mathematik
der Universität Bonn
Wegelerstr. 6
53115 Bonn, Germany
e-mail: griebel@iam.uni-bonn.de

David E. Keyes
Department of Applied Physics
and Applied Mathematics
Columbia University
200 S. W. Mudd Building
500 W. 120th Street
New York, NY 10027, USA
e-mail: david.keyes@columbia.edu

Risto M. Nieminen
Laboratory of Physics
Helsinki University of Technology
02150 Espoo, Finland
e-mail: rni@fyslab.hut.fi

Dirk Roose
Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200A
3001 Leuven-Heverlee, Belgium
e-mail: dirk.roose@cs.kuleuven.ac.be

Tamar Schlick
Department of Chemistry
Courant Institute of Mathematical
Sciences
New York University
and Howard Hughes Medical Institute
251 Mercer Street
New York, NY 10012, USA
e-mail: schlick@nyu.edu

Springer-Verlag, Mathematics Editorial IV
Tiergartenstrasse 17
D-69121 Heidelberg, Germany
Tel.: *49 (6221) 487-8185
Fax: *49 (6221) 487-8355
e-mail: Martin.Peters@springer-sbm.com

Lecture Notes in Computational Science and Engineering

Vol. 1 D. Funaro, *Spectral Elements for Transport-Dominated Equations.* 1997. X, 211 pp. Softcover. ISBN 3-540-62649-2

Vol. 2 H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 1999. XXIII, 682 pp. Hardcover. ISBN 3-540-65274-4

Vol. 3 W. Hackbusch, G. Wittum (eds.), *Multigrid Methods V.* Proceedings of the Fifth European Multigrid Conference held in Stuttgart, Germany, October 1-4, 1996. 1998. VIII, 334 pp. Softcover. ISBN 3-540-63133-X

Vol. 4 P. Deuflhard, J. Hermans, B. Leimkuhler, A. E. Mark, S. Reich, R. D. Skeel (eds.), *Computational Molecular Dynamics: Challenges, Methods, Ideas.* Proceedings of the 2nd International Symposium on Algorithms for Macromolecular Modelling, Berlin, May 21-24, 1997. 1998. XI, 489 pp. Softcover. ISBN 3-540-63242-5

Vol. 5 D. Kröner, M. Ohlberger, C. Rohde (eds.), *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws.* Proceedings of the International School on Theory and Numerics for Conservation Laws, Freiburg / Littenweiler, October 20-24, 1997. 1998. VII, 285 pp. Softcover. ISBN 3-540-65081-4

Vol. 6 S. Turek, *Efficient Solvers for Incompressible Flow Problems.* An Algorithmic and Computational Approach. 1999. XVII, 352 pp, with CD-ROM. Hardcover. ISBN 3-540-65433-X

Vol. 7 R. von Schwerin, ***Multi Body System SIMulation.*** Numerical Methods, Algorithms, and Software. 1999. XX, 338 pp. Softcover. ISBN 3-540-65662-6

Vol. 8 H.-J. Bungartz, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.* Proceedings of the International FORTWIHR Conference on HPSEC, Munich, March 16-18, 1998. 1999. X, 471 pp. Softcover. 3-540-65730-4

Vol. 9 T. J. Barth, H. Deconinck (eds.), *High-Order Methods for Computational Physics.* 1999. VII, 582 pp. Hardcover. 3-540-65893-9

Vol. 10 H. P. Langtangen, A. M. Bruaset, E. Quak (eds.), *Advances in Software Tools for Scientific Computing.* 2000. X, 357 pp. Softcover. 3-540-66557-9

Vol. 11 B. Cockburn, G. E. Karniadakis, C.-W. Shu (eds.), *Discontinuous Galerkin Methods.* Theory, Computation and Applications. 2000. XI, 470 pp. Hardcover. 3-540-66787-3

Vol. 12 U. van Rienen, *Numerical Methods in Computational Electrodynamics.* Linear Systems in Practical Applications. 2000. XIII, 375 pp. Softcover. 3-540-67629-5

Vol. 13 B. Engquist, L. Johnsson, M. Hammill, F. Short (eds.), *Simulation and Visualization on the Grid.* Parallelldatorcentrum Seventh Annual Conference, Stockholm, December 1999, Proceedings. 2000. XIII, 301 pp. Softcover. 3-540-67264-8

Vol. 14 E. Dick, K. Riemslagh, J. Vierendeels (eds.), *Multigrid Methods VI.* Proceedings of the Sixth European Multigrid Conference Held in Gent, Belgium, September 27-30, 1999. 2000. IX, 293 pp. Softcover. 3-540-67157-9

Vol. 15 A. Frommer, T. Lippert, B. Medeke, K. Schilling (eds.), *Numerical Challenges in Lattice Quantum Chromodynamics.* Joint Interdisciplinary Workshop of John von Neumann Institute for Computing, Jülich and Institute of Applied Computer Science, Wuppertal University, August 1999. 2000. VIII, 184 pp. Softcover. 3-540-67732-1

Vol. 16 J. Lang, *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems.* Theory, Algorithm, and Applications. 2001. XII, 157 pp. Softcover. 3-540-67900-6

Vol. 17 B. I. Wohlmuth, *Discretization Methods and Iterative Solvers Based on Domain Decomposition.* 2001. X, 197 pp. Softcover. 3-540-41083-X

Vol. 18 U. van Rienen, M. Günther, D. Hecht (eds.), *Scientific Computing in Electrical Engineering.* Proceedings of the 3rd International Workshop, August 20-23, 2000, Warnemünde, Germany. 2001. XII, 428 pp. Softcover. 3-540-42173-4

Vol. 19 I. Babuška, P. G. Ciarlet, T. Miyoshi (eds.), *Mathematical Modeling and Numerical Simulation in Continuum Mechanics.* Proceedings of the International Symposium on Mathematical Modeling and Numerical Simulation in Continuum Mechanics, September 29 - October 3, 2000, Yamaguchi, Japan. 2002. VIII, 301 pp. Softcover. 3-540-42399-0

Vol. 20 T. J. Barth, T. Chan, R. Haimes (eds.), *Multiscale and Multiresolution Methods.* Theory and Applications. 2002. X, 389 pp. Softcover. 3-540-42420-2

Vol. 21 M. Breuer, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.* Proceedings of the 3rd International FORTWIHR Conference on HPSEC, Erlangen, March 12-14, 2001. 2002. XIII, 408 pp. Softcover. 3-540-42946-8

Vol. 22 K. Urban, *Wavelets in Numerical Simulation.* Problem Adapted Construction and Applications. 2002. XV, 181 pp. Softcover. 3-540-43055-5

Vol. 23 L. F. Pavarino, A. Toselli (eds.), *Recent Developments in Domain Decomposition Methods.* 2002. XII, 243 pp. Softcover. 3-540-43413-5

Vol. 24 T. Schlick, H. H. Gan (eds.), *Computational Methods for Macromolecules: Challenges and Applications.* Proceedings of the 3rd International Workshop on Algorithms for Macromolecular Modeling, New York, October 12-14, 2000. 2002. IX, 504 pp. Softcover. 3-540-43756-8

Vol. 25 T. J. Barth, H. Deconinck (eds.), *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics.* 2003. VII, 344 pp. Hardcover. 3-540-43758-4

Vol. 26 M. Griebel, M. A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations*. 2003. IX, 466 pp. Softcover. 3-540-43891-2

Vol. 27 S. Müller, *Adaptive Multiscale Schemes for Conservation Laws*. 2003. XIV, 181 pp. Softcover. 3-540-44325-8

Vol. 28 C. Carstensen, S. Funken, W. Hackbusch, R. H. W. Hoppe, P. Monk (eds.), *Computational Electromagnetics*. Proceedings of the GAMM Workshop on "Computational Electromagnetics", Kiel, Germany, January 26-28, 2001. 2003. X, 209 pp. Softcover. 3-540-44392-4

Vol. 29 M. A. Schweitzer, *A Parallel Multilevel Partition of Unity Method for Elliptic Partial Differential Equations*. 2003. V, 194 pp. Softcover. 3-540-00351-7

Vol. 30 T. Biegler, O. Ghattas, M. Heinkenschloss, B. van Bloemen Waanders (eds.), *Large-Scale PDE-Constrained Optimization*. 2003. VI, 349 pp. Softcover. 3-540-05045-0

Vol. 31 M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne (eds.), *Topics in Computational Wave Propagation*. Direct and Inverse Problems. 2003. VIII, 399 pp. Softcover. 3-540-00744-X

Vol. 32 H. Emmerich, B. Nestler, M. Schreckenberg (eds.), *Interface and Transport Dynamics*. Computational Modelling. 2003. XV, 432 pp. Hardcover. 3-540-40367-1

Vol. 33 H. P. Langtangen, A. Tveito (eds.), *Advanced Topics in Computational Partial Differential Equations*. Numerical Methods and Diffpack Programming. 2003. XIX, 658 pp. Softcover. 3-540-01438-1

Vol. 34 V. John, *Large Eddy Simulation of Turbulent Incompressible Flows*. Analytical and Numerical Results for a Class of LES Models. 2004. XII, 261 pp. Softcover. 3-540-40643-3

Vol. 35 E. Bänsch (ed.), *Challenges in Scientific Computing - CISC 2002*. Proceedings of the Conference *Challenges in Scientific Computing*, Berlin, October 2-5, 2002. 2003. VIII, 287 pp. Hardcover. 3-540-40887-8

Vol. 36 B. N. Khoromskij, G. Wittum, *Numerical Solution of Elliptic Differential Equations by Reduction to the Interface*. 2004. XI, 293 pp. Softcover. 3-540-20406-7

Vol. 37 A. Iske, *Multiresolution Methods in Scattered Data Modelling*. 2004. XII, 182 pp. Softcover. 3-540-20479-2

Vol. 38 S.-I. Niculescu, K. Gu (eds.), *Advances in Time-Delay Systems*. 2004. XIV, 446 pp. Softcover. 3-540-20890-9

Vol. 39 S. Attinger, P. Koumoutsakos (eds.), *Multiscale Modelling and Simulation*. 2004. VIII, 277 pp. Softcover. 3-540-21180-2

Vol. 40 R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Wildlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering*. 2005. XVIII, 690 pp. Softcover. 3-540-22523-4

Vol. 41 T. Plewa, T. Linde, V.G. Weirs (eds.), *Adaptive Mesh Refinement – Theory and Applications.* 2005. XIV, 552 pp. Softcover. 3-540-21147-0

Vol. 42 A. Schmidt, K.G. Siebert, *Design of Adaptive Finite Element Software.* The Finite Element Toolbox ALBERTA. 2005. XII, 322 pp. Hardcover. 3-540-22842-X

Vol. 43 M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations II.* 2005. XIII, 303 pp. Softcover. 3-540-23026-2

For further information on these books please have a look at our mathematics catalogue at the following URL: `www.springeronline.com/series/3527`

Texts in Computational Science and Engineering

Vol. 1 H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 2nd Edition 2003. XXVI, 855 pp. Hardcover. ISBN 3-540-43416-X

Vol. 2 A. Quarteroni, F. Saleri, *Scientific Computing with MATLAB.* 2003. IX, 257 pp. Hardcover. ISBN 3-540-44363-0

Vol. 3 H. P. Langtangen, *Python Scripting for Computational Science.* 2004. XXII, 724 pp. Hardcover. ISBN 3-540-43508-5

For further information on these books please have a look at our mathematics catalogue at the following URL: `www.springeronline.com/series/5151`