Solving Partial Differential Equations by Taylor Meshless Method

Jie Yang

UNIVERSITÉ DE LORRAINE

Ecole Doctorale EMMA (Mécanique et énergétique)

Thèse

Présentée et soutenue publiquement pour l'obtention du titre de

DOCTEUR DE L'UNIVERSITE DE LORRAINE

par **Jie YANG**

Solving Partial Differential Equations by Taylor Meshless Method

Soutenue le 22 janvier 2018

Membres du jury:

Rapporteurs:	Prof. **Pierre Villon**	Université de Technologie de Compiegne, France
	Prof. **Zakaria Belhachmi**	Université de Haute Alsace, France
Examinateurs:	Prof. **Pierre Ladevèze**	Université de Paris-Saclay, France
	Dr. **Yao Koutsawa**	Luxembourg Institute of Science and Technology
	Dr. **Renata Bunoiu**	Université de Lorraine, France
Directeur:	Prof. **Michel Potier-Ferry**	Université de Lorraine, France
	Prof. **Heng HU**	Wuhan University, China

Laboratoire d'Étude des Microstructures et de Mécanique des Matériaux
LEM3 UMR CNRS 7239 - Université de Lorraine
7 rue Félix Savart - 57073 Metz Cedex 03 - France

Acknowledgments

This thesis has been written during my study as a joint-PhD candidate in Université de Lorraine and Wuhan University.

First I would like to express my deeply-felt gratitude to my parents who create the space of chasing my dreams freely. I also would like to extend my heartfelt appreciation to my supervisors, Prof. Michel Potier-Ferry and Prof. Heng Hu, whose patient guidance, valuable suggestions and constant encouragement make me successfully finish my PhD thesis. Their enthusiasm for scientific research and the attitude towards lift inspire me all the time both in academic study and daily life. They offer me selfless help that made me who I am today.

My heartfelt appreciate also extends to all my colleges in the team. Thank them for their help not only in the scientific research but also in the daily life. I am so grateful for the learning, companionship and encouragement from each other.

Résumé

Le but de cette thèse est de développer une méthode numérique simple, robuste, efficace et précise pour résoudre des problèmes d'ingénierie de grande taille à partir de la méthode Taylor Meshless (TMM) et fournir de nouvelles idée principale de TMM est d'utiliser comme fonctions de forme des polynômes d'ordre élevé qui sont des solutions approchées de l'EDP. Ainsi la discrétisation ne concerne que la frontière. Les coefficients de ces fonctions de forme sont obtenus en discrétisant les conditions aux limites par des procédures de collocation associées à la méthode des moindres carrés. TMM est alors une véritable méthode sans maillage sans processus d'intégration, les conditions aux limites étant obtenues par collocation.

Les principales contributions de cette thèse sont les suivantes: 1) Basé sur TMM, un algorithme général et efficace a été développé pour résoudre des EDP elliptiques tridimensionnelles; 2) Trois techniques de couplage pour des résolutions par morceaux ont été discutées dans des cas de problèmes à grande échelle: la méthode de collocation par les moindres carrés et deux méthodes de couplage basées sur les multiplicateurs de Lagrange; 3) Une méthode numérique générale pour résoudre les EDP non-linéaires a été proposée en combinant la méthode de Newton, la TMM et la technique de différentiation automatique. 4) Pour résoudre des problèmes avec un bord non régulier, des solutions singulières satisfaisant l'équation de contrôle sont introduites comme des fonctions de forme complémentaires, ce qui fournit une base théorique pour la résolution de problèmes singuliers.

Mots clés: Série de Taylor, méthode sans maillage, résolution par morceaux, différenciation automatique, fonctions de forme singulières, équations aux Dérivées partielles

Abstract

Based on Taylor Meshless Method (TMM), the aim of this thesis is to develop a simple, robust, efficient and accurate numerical method which is capable of solving large scale engineering problems and to provide a new idea for the follow-up study on meshless methods. To this end, the influence of the key factors in TMM has been studied by solving three-dimensional and non-linear Partial Differential Equations (PDEs). The main idea of TMM is to use high order polynomials as shape functions which are approximated solutions of the PDE and the discretization concerns only the boundary. To solve the unknown coefficients, boundary conditions are accounted by collocation procedures associated with least-square method. TMM that needs only boundary collocation without integration process, is a true meshless method.

The main contributions of this thesis are as following: 1) Based on TMM, a general and efficient algorithm has been developed for solving three-dimensional PDEs; 2) Three coupling techniques in piecewise resolutions have been discussed and tested in cases of large-scale problems, including least-square collocation method and two coupling methods based on Lagrange multipliers; 3) A general numerical method for solving non-linear PDEs has been proposed by combining Newton Method, TMM and Automatic Differentiation technique; 4) To apply TMM for solving problems with singularities, the singular solutions satisfying the control equation are introduced as complementary shape functions, which provides a theoretical basis for solving singular problems.

Key words: Taylor series, meshless method, boundary collocation, coupling techniques in piecewise resolution, automatic differentiation, singular shape function, partial differential equation

Contents

Chapter 1

Introduction

Contents

1.1 Research background and significance

The physical quantity of the objective world is generally changed over time and space, and its intrinsic laws to be presented in the form of differential or partial differential equations (PDEs). There are many methods to solve these PDEs, for instance finite difference method (FDM), finite element method (FEM), boundary element method (BEM) and meshless methods.

FEM [1, 2] has been widely used in the field of engineering due to its robustness and universality. It contains the following advantages: 1) the use of equivalent integral weak form equations reduces the requirement of continuity of the interpolation functions; 2) the use of localized interpolation form a narrow bandwidth sparse stiffness matrix and this improves the stability of the calculation; 3) since the Kronecker delta property of the interpolation functions, it is convenient to account for essential boundary conditions.

There also exist the following drawbacks in FEM: 1) creation of a mesh for a complicated domain could be time-consuming; 2) when handling large deformation, considerable accuracy is lost because of the elements distortion; 3) in stress calculations, the stresses obtained by using FEM are discontinuous and less accurate. 4) the re-meshing technique may solve the problem of mesh distortion, however this will reduce the computational efficiency and accuracy; 5) it is very difficult to simulate the breakage of material into a large number of fragments as FEM is essentially based on continuum mechanics.

The main idea of BEM [3] is to transform the PDE into boundary integral equation by employing the fundamental solutions and weighted residual approach. Since the boundary integral equation concerns the boundary, only the boundary needs to be discretized. The main advantages of BEM are as follows: 1) since no integration inside the domain is needed, the dimension and the number of degrees of freedom is strongly reduced; 2) as the fundamental solutions adopted in BEM satisfying the boundary condition at infinite, it is convenient to solve problems with unbounded domain.

There also exist some drawbacks when using BEM: 1) it is difficult to obtain the fundamental solutions for a complicated PDE and to deal with integration of singular fields; 2) when computing the crack propagation, one needs to repeatedly update the boundary grids; 3) normally the global matrix is unsymmetrical and full, which reduces the accuracy and robustness; 4) when dealing with non-linear cases, the integration inside the domain is required.

The main idea of FDM [4] is to use finite differences instead of derivatives. As the advantages of simplicity, flexibility and versatility of FDM, it is been widely used in fields of solid and fluid mechanics. It suffers from a major disadvantage in that it relies on regularly distributed nodes.

These difficulties associated with FEM, BEM and FDM mainly come from the mesh or grid in which a predefined connection between neighbor points is required. Thus the idea of eliminating the elements has evolved naturally. The concept of meshless or mesh free methods has been proposed, in which the domain of the problem is represented by a set of arbitrarily distributed nodes. The meshless framework not only provides a great convenience of pre-processing work, but also can avoid problems with mesh, such as mesh distortion, crack propagation, high velocity impact or explosive mechanics. It can effectively compensate the drawbacks of methods based on a mesh.

The meshless methods consists of two main steps: the approximation of unknown functions and the discretization of the PDE. The latter step has two main categories: Galerkin-based technique and collocation approach. Since background mesh and integration are required in Galerkin-based meshless method, it leads to expensive computational cost. The collocation-based meshless method is sometimes efficient since no integration is needed, but it is difficult to solve large-scale problems due to ill-conditioned matrices.

This thesis aims to discuss a newly proposed collocation-based meshless technique, named Taylor Meshless Method (TMM), for solving three-dimensional non-linear PDEs. The final goal is to develop a simple, robust, efficient and accurate numerical method which is capable of solving large scale engineering problems and to provide a new idea for the follow-up study on meshless methods. To this end, the effect of matrix ill-conditioning and the propagation of round-off errors are analyzed carefully.

1.2 Review of meshless method

Mesh free or meshless method is defined with respect to the word “mesh”. It is a common name of discretization that is different in different numerical methods, for instance, it is called grids in FDM, volumes or cells in FVM and named elements in FEM. The grids, volumes or cells and elements are collectively referred to as meshes since they are used to predefine the connection between nodes. To get rid of the tedious work of meshing and at the same time to avoid the computational difficulty caused by mesh distortion, a class of numerical methods without mesh based on interpolation of scatter points or based on shape functions fitting came into being, is collectively referred to as the meshless or mesh free methods.

1.2.1 Classification

The Smoothed Particle Hydrodynamics (SPH) proposed by Gingold and Monaghan [5] and Lucy et al. [6] in 1977 is considered to be one of the earliest meshfree methods in literature. In the 1990s, a new class of meshfree methods emerged based on Galerkin method. Among which, the first one called diffuse element method (DEM) was proposed by Nayroles et al. [7]. Thereafter in the framework of DEM, Belytschko made some improvements and proposed the Element Free Galerkin Method [8] (EFGM). Then in the following few decades, a variety of new meshless methods have sprung up.

According to the difference of discrete areas, the meshless method can be divided into two categories: the domain-type meshless methods and the boundary-type meshless methods.

1.2.1.1 Domain-type meshless methods

There are two main steps in domain-type meshless methods: 1) approximation of unknown functions; 2) discretization of control equation.

The first step is realized by using the interpolation of the arbitrary and irregular scattered points in the whole domain. There exist the following approaches to approximate the unknown functions in the literature: kernel particle approximation [5, 6], reproducing kernel particle [9], moving least-square [10–12], partition of unity [13], radial basis function [14–18], point interpolation method [19], etc.

Concerning the way to discretize the control equation, there are two major techniques: Galerkin method and collocation method. As for Galerkin-based methods, there are two main types of Galerkin methods. The first one is based on background integration in the whole domain while the second one named local Petrov-Galerkin method takes account of the integration in a rather small local sub-domain and no background mesh is required. Therefore meshless methods based on local Petrov-Galerkin integration are considered to be truly mesh free methods. Galerkin-based meshless methods benefit from their robustness and versatility that ensure the ability of solving large-scale problems. However the existence of integration may lead to expansive computational costs.

An alternative way to discretize the control equation is collocation technique. No background mesh and no integration is required, that makes it very efficient. Generally we believe that the collocation-based meshless methods are truly integration-free meshless method. However, when increasing the scale of considered problems, the ill-conditioned matrices in collocation-based methods may lead to numerical instability and low accuracy.

By combining the above-mentioned ways to approximate the unknown functions with the approaches to discretize control equation, one can obtain a variety of domain-type meshless methods. The main domain-type meshless methods have been collected in Table 1.1. Additional information about meshless methods can be found in several review papers and books, see for instance [20, 21].

Table 1.1: Main domain-type meshless methods.

	Galerkin	Local Petrov-Galerkin	Collocation
KP	/	/	SPH [5, 6]
RKP	RKPM [9], MRKPM [22]	MLPG [23]	PCM [24]
MLS	DEM [7], EFGM [8]	MLPG [23, 25]	FPM [26], LSCM [27]
RBF	MG-RBF [28]	MLPG [23]	RBF [29, 30], BKM [31]
PU	Hp Clouds [13]	MLPG [23]	Hp-meshless clouds [32]
PI	PIM [21]	LPIM [33]	/

1.2.1.2 Boundary-type meshless methods

The boundary-type meshless methods are based on complete families of shape functions that are fully exact solutions of considered problems. They are also known as Trefftz methods and many information about Trefftz methods can be found in several review papers, see for instance [34–37] or in some books [38, 39]. Since the PDEs are automatically satisfied, only the discritization of boundary is needed. There are two types of boundary-type meshless method: boundary integration and boundary collocation.

The first one is more or less similar to BEM that is based on the boundary integral equation. Many such kinds of meshless methods can be found in the literature, see for instance Boundary Node Method [40] (BNM), Boundary Element-Free Method [41] (BEFM), Boundary Point Interpolation Method [42] (BPIM), Local Boundary Integral Equation Method [43] (LBIEM), etc. Although the boundary integral type meshless method benefits from the reduction of dimensions and the number of degrees of freedom, it is difficult to obtain the fundamental solutions of complex PDEs and to compute singular boundary integration. Mover, the requirement of integration leads to a low computational efficiency.

Due to the drawbacks of the first kind of boundary-type meshless methods, another boundary-type meshless methods based on fundamental solutions and collocation technique have been proposed. The computational efficiency is usually high since the discretization concerns only the boundary and no mesh or integration is required in this kind of algorithms. For instance, the Method

of Fundamental Solution [44] (MFS) initialized by Kupradze and Aleksidze, Boundary Knot Method [31] (BKM) proposed by Chen et al., Regularized Meshless Method [45] (RMM), Singular Boundary Method [46] (SBM), etc. The main idea of the previously mentioned methods is to approximate the unknown function by a linear combination of the singular fundamental solutions and then the unknown coefficients are determined by applying the boundary conditions.

The present thesis focuses on a new boundary integration-free meshless method proposed by Zézé et al. [47], named Taylor Meshless Method (TMM), that relies on approximated solutions of the PDEs in the sense of Taylor series. When applied to a linear homogeneous PDE with constant coefficients, it coincides with Trefftz method associated with harmonic polynomials. In the following, few two dimensional applications of TMM are considered to briefly recall the basic properties of TMM, especially exponential convergence, robustness and efficiency.

1.3 Taylor Meshless Method

1.3.1 State of the art

The basic idea of TMM is to use the high order polynomial shape functions that are approximated solutions of the considered PDE. First we consider a simplest Laplace equation:

$$\Delta u(x, y) = 0 \tag{1.1}$$

The harmonic polynomial solutions of the Laplace equation can be denoted by the real and imaginary parts of $(x + iy)^n$, $\mathbf{Re}(x + iy)^n$ and $\mathbf{Im}(x + iy)^n$. The general solution from order zero to p are collected in Table 1.2.

Table 1.2: Polynomial solutions of Laplace equation

n	$\mathbf{Re}(x + iy)^n$	$\mathbf{Im}(x + iy)^n$
0	$P_1 = 1$	
1	$P_2 = x$	$P_3 = y$
2	$P_4 = x^2 - y^2$	$P_5 = 2xy$
$\vdots$	$\vdots$	$\vdots$
$p-1$	P_{2p-2}	P_{2p-1}
p	$P_{2p} = P_{2p-2} \cdot x - P_{2p-1} \cdot y$	$P_{2p+1} = P_{2p-2} \cdot y + P_{2p-1} \cdot x$

Then the approximated solution of Laplace equation can be expressed by

the linear combination of the polynomial solutions shown in Table 1.2.

$$u_p(x,y) = \sum_{i=1}^{2p+1} P_i \cdot v_i = \boldsymbol{P}\boldsymbol{v} \tag{1.2}$$

where v_i represents the unknown coefficients.

Since Laplace equation has been fully satisfied by the approximation u_p, only boundary conditions need to be considered to determine the unknown vector $\boldsymbol{v}$. Here, we consider a mixed boundary conditions as follows:

$$\begin{cases} u(\boldsymbol{x}) = u^d & \boldsymbol{x} \in \Gamma_d, \\ \mathbf{T}u(\boldsymbol{x}) = t^n & \boldsymbol{x} \in \Gamma_n. \end{cases} \tag{1.3}$$

Here, a collocation technique combined with the least-square method [27, 48] is used to apply the boundary conditions. One chooses a set of nodes $\boldsymbol{x}_i$ on Γ_d and another set of nodes $\boldsymbol{x}_j$ on Γ_n, see Fig. 1.1.

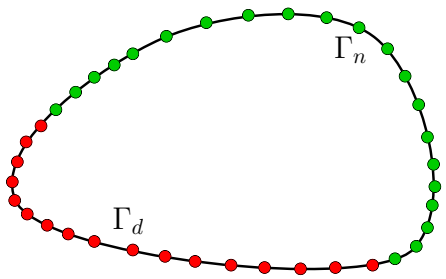


Figure 1.1: Sketch for boundary collocation.

Then one minimizes the error between the approximate value u_p, $\mathbf{T}u_p$ and the given value of u^d, t^n at these points. It comes to minimize the following function:

$$\mathscr{T}(\boldsymbol{v}) = \frac{1}{2}\sum_{\boldsymbol{x}_i \in \Gamma_d} \left|u_p(\boldsymbol{x}_i) - u^d(\boldsymbol{x}_i)\right|^2 + w \cdot \frac{1}{2}\sum_{\boldsymbol{x}_j \in \Gamma_n} \left|\mathbf{T}u_p(\boldsymbol{x}_j) - t^n(\boldsymbol{x}_j)\right|^2 \tag{1.4}$$

The minimization leads to a linear system and solving this system gives the vector $\boldsymbol{v}$ and then the numerical solution of the considered problem.

1.3.2 The influence of the number of collocation points

Here we consider a Helmholtz equation in a rectangular domain:

$$\begin{cases} -\Delta u + u = 0 \quad \text{in } \Omega \\ u|_{y=0,4} = 0 \\ u|_{x=\pm 2.5} = sin(\pi y/4) \end{cases} \tag{1.5}$$

The exact solution is as following:

$$u(x,y) = \frac{\cosh(x\sqrt{1+\pi^2/16})}{\cosh(2.5\sqrt{1+\pi^2/16})}\sin(\pi y/4) \tag{1.6}$$

The influence of the number of collocation points has been illustrated in Fig. 1.2. Three values of the degree are considered, $p = 10$, 15 and 20 respectively. The maximum accuracy is obtained for about $M = 2p+1$ and it remains constant beyond this threshold. Almost the same behavior has been observed in the other 2D cases. In general, one chooses $4p$ collocation points to ensure the convergence.

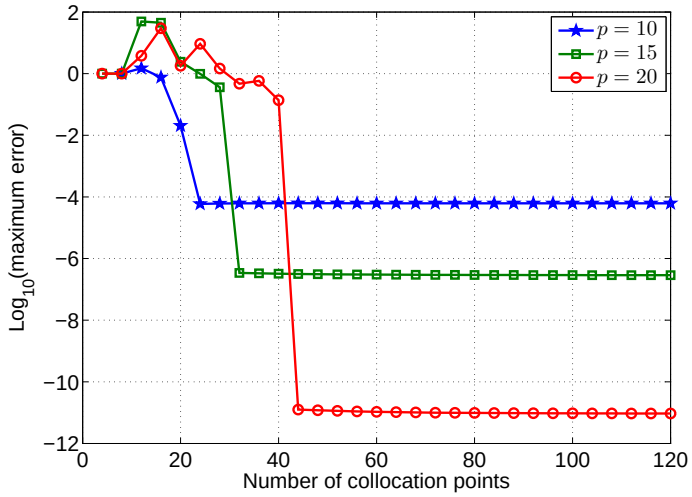


Figure 1.2: The influence of the number of collocation points in problem (1.5).

1.3.3 Compared with FEM

Now we compare the accuracy of the proposed meshless method with classical finite element discretization. Here we consider Laplace equation in a rectangular domain ($\Omega = \{(x,y)|0 \leqslant x \leqslant 10, 0 \leqslant y \leqslant \pi\}$), see Fig. 1.3:

$$\begin{cases} \Delta u(x,y) = 0 \quad \text{in } \Omega \\ u|_{x=0} = \sin(y) \\ u|_{y=0,\pi} = u|_{x=10} = 0 \end{cases} \tag{1.7}$$

The analytical solution is as following:

$$u(x,y) = \sin(y) \cdot \frac{\sinh(10-x)}{\sinh(10)} \tag{1.8}$$

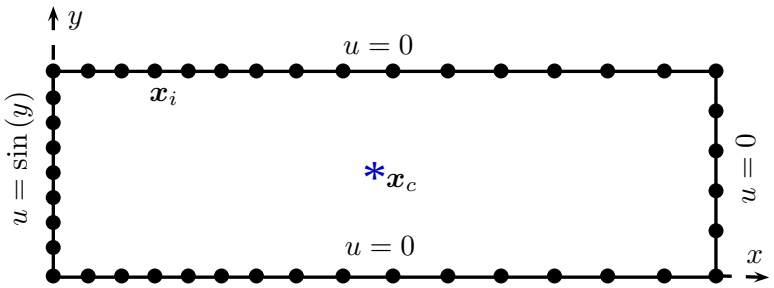


Figure 1.3: Sketch for collocation in a rectangular domain.

In this discussion, we focus on the number of degrees of freedom necessary to get the same accuracy. The results are presented in Table 1.3. First one can see that the TMM converges exponentially with the degree of Taylor series while the FEM (with Q8 elements) converges slowly with the refinement of mesh. To obtain an accuracy of 10^{-6}, only 31 DOFs are necessary with TMM while 9341 DOFs are required in FEM. The presented results show that a significant reduction in the number of DOFs has been obtained.

Table 1.3: For problem (1.7), The comparison between TMM and FEM.

TMM	Order of Taylor series	$p = 10$	$p = 15$	$p = 20$	$p = 30$
	$log_{10}(\max \mathfrak{E})$	-2.8018	-5.9848	-9.4708	-9.4532
	Degrees of freedom	21	31	41	61
FEM	Mesh ($n_x \times n_y$)	30×10	60×20	90×30	120×40
	$log_{10}(\max \mathfrak{E})$	-4.1210	-5.2704	-5.9561	-6.4464
	Degrees of freedom	981	3761	9341	14721

1.3.4 Convergence analysis of linear system

Another property of TMM we care about is the performance on convergence when solving a linear system. Here we consider Stokes equations in a square domain (see Fig. 1.4, $\ell_x = \ell_y = 1$) as follows:

$$\begin{cases} -\mu\Delta u + \partial q/\partial x = f_x \\ -\mu\Delta v + \partial q/\partial y = f_y \\ \partial u/\partial x + \partial v/\partial y = 0 \end{cases} \tag{1.9}$$

where f_x and f_y denote:

$$\begin{cases} f_x = (2\mu\pi^2 - 2\pi)\sin(\pi x)\sin(\pi y) \\ f_y = (2\mu\pi^2 + 2\pi)\cos(\pi x)\cos(\pi y) \end{cases}$$

The boundary conditions are presented in Eq. (1.10).

$$\begin{cases} u = q = 0; & v = -\cos(\pi x); & \Gamma_1 = \{(x,y)|y=1\} \\ u = 0; & v = \cos(\pi x); & \Gamma_2 = \{(x,y)|y=0\} \\ u = 0; & v = \cos(\pi y); & \Gamma_3 = \{(x,y)|x=0\} \\ u = 0; & v = -\cos(\pi y); & \Gamma_4 = \{(x,y)|x=1\} \end{cases} \tag{1.10}$$

The exact solution of linear system Eq. (1.9) reads:

$$\begin{cases} u = \sin(\pi x)\sin(\pi y) \\ v = \cos(\pi x)\cos(\pi y) \\ q = 2\cos(\pi x)\sin(\pi y) \end{cases} \tag{1.11}$$

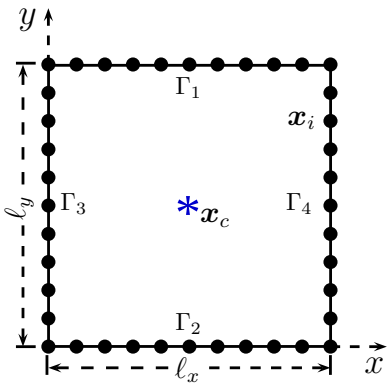


Figure 1.4: Sketch of boundary collocation for Stokes equations.

One should note that there are three unknown functions in the considered case and the number of DOFs is $4p + 2$. The convergence with the degree of Taylor series has been presented in Table 1.4. The maximum relative error decreases exponentially with the degree of Taylor series.

1.3.5 Comments on TMM

In the previous papers [47, 49–51], it has been validated that TMM is robust and efficient for solving two-dimensional elliptic PDEs. One can also use TMM to handle problems with any shape of domain and any boundary condition.

Table 1.4: Convergence analysis with the degree of Taylor series in case of Stokes equations.

Taylor series	p	5	10	15	20	25	30
	u	-0.8196	-3.8717	-6.5825	-11.3130	-11.0942	-10.2116
$log_{10}(\max \mathfrak{E})$	v	-0.8885	-3.6310	-6.3422	-11.0601	-11.0125	-10.3790
	q	0.4908	-2.2985	-4.8797	-9.1742	-8.7675	-7.9215
Total DOFs		22	42	62	82	102	122

For instance, TMM works well for a problem with an amoeba-like boundary shape in [47] and for elasticity problems with displacement and stress boundary conditions in [50]. In practice, TMM needs much less DOFs as compared with other discretization techniques since the PDE is solved analytically.

1.4 The main content of this thesis

In this thesis, a boundary integration-free meshless method based on Taylor series, named Taylor Meshless Method (TMM), is discussed. The main idea of TMM is to compute a family of shape functions that are solutions of the PDE in the sense of Taylor series. And the specificity of TMM is the quasi-exact solution inside the domain of the considered problems, which leads to a reduction of the dimensions and a strong reduction of the number of degrees of freedom. Some one-dimensional and two-dimensional applications of TMM have been proposed in the PhD thesis of Zézé [52] and PhD thesis of Tampango [53], but also to two-dimensional non-linear elasticity in Master thesis of Akpama [54]. TMM is confirmed to be robust and efficient for at least two-dimensional cases with smooth solutions.

The purpose of this thesis is to extend the applications of TMM to three dimensions, large-scale and non-linear range, but also try to initiate the problems with singularities. Along with continued research work, a variety of serious problems arise, for instance the performance of bridging technique to connect several sub-domains, the ill-conditioned matrix and the propagation of round-off errors for large-scale applications, the efficiency for non-linear applications and the handling of singular problems involving cracks, corners or notches. The detailed content of this thesis is as the follows:

• In chapter 2, in the framework of piecewise resolutions of TMM, Lagrange multipliers are applied to account for boundary conditions and least-square collocation is revisited to account for transmission conditions.

• In chapter 3, the applications of TMM are extended to three dimensions. The computation time and the influence of ill-conditioning of matrices on the

accuracy are discussed carefully.

• In chapter 4, TMM is combined with Newton method and automatic differentiation to solve non-linear PDEs in one dimension and three dimensions.

• In chapter 5, few known singular solutions are introduced in TMM as singular shape functions to solve problems with singularities.

Chapter 2

Piecewise resolution of Taylor Meshless Method

Abstract

A recently proposed meshless method is discussed in this paper. It relies on Taylor series, the shape functions being high degree polynomials deduced from the Partial Differential Equation (PDE). In this framework, an efficient technique to couple several polynomial approximations has been presented in [51]: the boundary conditions were applied by using the least-square collocation and the interface was coupled by a bridging technique based on Lagrange multipliers. In this paper, least-square collocation and Lagrange multipliers are applied for boundary conditions respectively and least-square collocation is revisited to account for the interface conditions in piecewise resolutions. Various combinations of these two techniques have been investigated and the numerical results prove their effectiveness to obtain very accurate solutions, even for large scale problems.

Present chapter corresponds to the submitted research paper (Yang et al., Least-square collocation and Lagrange multipliers for Taylor Meshless Method, submitted to Journal, 2017).

Keywords: Taylor series; Meshless; Least-square; Lagrange multiplier; Piecewise resolution.

Contents

2.1 Introduction

Most of the numerical methods for PDEs rely on a priori chosen shape functions. Low degree polynomials are most wide-spread in the Finite Element Method. High-order polynomials permit to get very accurate solutions without re-meshing in the p-version of finite elements [55, 56]. There were many works in the last two decades about radial functions [29, 30], moving least-squares approximations [8, 57, 58] and Method of Fundamental Solution [48, 59, 60], among others. Nevertheless, except for small degree polynomials, it is not easy to manage either the integration procedures [61, 62] in case of a discretisation based on the weak form, or ill-conditioning matrices [63] in case of collocation-based discretisation.

In this chapter, we are interested in the discretisation of PDEs by the method of Taylor series. Taylor series are used quite often for solving ordinary differential equations, see for instance [64–67], but they have been introduced only recently for PDE's in a meshless and integration-less framework [47, 49–51]. The solution is sought in the form of a polynomial with an arbitrary large degree and the size of the polynomial basis is reduced by considering the Taylor series of the PDE. So only the boundary of the domain has to be discretized. In this framework of large degree polynomials, the point-collocation is highly unstable, while the least-squares collocation [48, 68] leads to a reliable numerical technique when applied together with Taylor series [47, 52]. Often the procedure converges exponentially with the degree (p-convergence), as in the p-version of the finite element method, which can lead to very accurate solutions. For instance, numerical tests in [50] have established that it requires much less degrees of freedom (DOFs) than finite element methods. Likely it could be applied to any linear or nonlinear elliptic system, as shown by the solution of hyperelastic boundary value problems in [54]. Nevertheless this convergence depends on the radius of convergence of the series and it is not possible to solve any boundary value problem with a single series.

Thus, numerical methods have to be introduced to connect several high-order polynomial approximations. A least-square approach was proposed in [47], but this piecewise resolution did not yield exponential convergence as with a single Taylor series. Such p-convergence properties have been established in [51] by using two methods based on Lagrange multipliers. The first one is more or less similar with FETI [69–71] or mortar methods [72], the continuity being enforced at some nodes of the interface. The second one defines the connection in a weak sense, which is a discrete version of Arlequin method [73, 74]. In the two cases, the bridging techniques are not perfectly conforming and there are small discontinuities between the two polynomials,

but of course these discontinuities decrease exponentially. With these recent results, an efficient and reliable method is established: it associates Taylor series inside each sub-domain, least-squares collocation for the boundary conditions and Lagrange multipliers at the interfaces between sub-domains. As in [51], it will be called TMM. Thus we have two families of numerical techniques to account for boundary and interface conditions that have been validated in only one case. One may wonder if Lagrange multipliers may also be used for boundary conditions and least-squares at the interfaces. The aim of the chapter is to discuss these questions.

The chapter is organized as follows. In section 2.2, we present the boundary meshless method using Taylor series approximations. In section 5.3, two methods are presented to account for boundary conditions and some applications are considered to assess the validity of the two presented techniques: least-square collocation and Lagrange multiplier method based on radial functions. In section 2.4, least-square collocation is applied for the interface conditions in piecewise resolutions and the boundary conditions are ensured by the two presented methods respectively. In section 2.5, 2D elasticity problems are considered to discuss the behavior of TMM for mechanical problems. Finally in section 2.6, the robustness of the sub-domain techniques is assessed by considering a 3D benchmark that requires more than one million of unknowns within the finite element method.

2.2 State of the art

In this section, a meshless method based on Taylor series approximation is briefly recalled. More detailed descriptions of this method can be found in Zézé et al. [47,52]. To illustrate the technique, we consider a Dirichlet problem:

$$\begin{cases} -\Delta u + c \cdot u = 0 & \text{in} \quad \Omega \\ u(\boldsymbol{x}) = u^d(\boldsymbol{x}) & \text{on} \quad \Gamma \end{cases} \tag{2.1}$$

where c is a constant.

The main idea of this technique is to introduce high degree polynomial shape functions. This technique involves two steps. The first step consists in determining the shape functions by a quasi-exact resolution of the PDE in the domain. The approximate solution of Eq. (2.1) is sought in the form of a polynomial of degree p:

$$u_p(x,y) = \sum_{k=0}^{p} \sum_{i=0}^{k} \widetilde{u}_{i,k-i} x^i y^{k-i} = \sum_{k=0}^{p} \langle \boldsymbol{X}^k \rangle \{ \widetilde{\boldsymbol{u}}^k \} \tag{2.2}$$

where $\langle \boldsymbol{X}^k \rangle = \langle x^k, x^{k-1}y, \cdots, y^k \rangle$, $\quad \{\widetilde{\boldsymbol{u}}^k\}^{\mathrm{t}} = \langle \widetilde{u}_{k,0}, \widetilde{u}_{k-1,1}, \cdots, \widetilde{u}_{0,k} \rangle$.

For each degree k, the unknown is the vector $\{\widetilde{\boldsymbol{u}}^k\} \in \mathbb{R}^{k+1}$. For the complete polynomial Eq. (2.2), there are $(p+1)(p+2)/2$ coefficients to be found. Then if Eq. (2.1) is satisfied, the polynomial $-\Delta u + c \cdot u$ vanishes and this leads to a linear system of equations:

$$-[\Delta^k]\{\widetilde{\boldsymbol{u}}^{k+2}\} + c\{\widetilde{\boldsymbol{u}}^k\} = \{0\} \qquad \forall\, k,\, 0 \leq k \leq p-2 \tag{2.3}$$

For each k, there are $k+1$ equations in Eq. (2.3) for $k+3$ unknowns ($\{\widetilde{\boldsymbol{u}}^{k+2}\} \in \mathbb{R}^{k+3}$). Each vector $\{\widetilde{\boldsymbol{u}}^{k+2}\}$ can then be written as a function of its two first components:

$$\{\boldsymbol{v}^k\} = \left\{ \begin{array}{c} \widetilde{u}_{k,0} \\ \widetilde{u}_{k-1,1} \end{array} \right\} \in \mathbb{R}^2 \tag{2.4}$$

The account of the PDE reduces the general polynomial Eq. (2.2) to a family of $(2p+1)$ polynomials instead of $(p+1)(p+2)/2$:

$$u_p(x,y) = \sum_{i=0}^{2p+1} P_i(x,y) v_i \tag{2.5}$$

For instance in the case of Laplace equation, one recovers the well known result: $P_{2n} = Re(x+iy)^n$ and $P_{2n+1} = Im(x+iy)^n$. A similar procedure can be applied to nonlinear equations or to equations with variable coefficients [52,54], but we will not give more details in this paper that focuses on the treatment of boundary and transmission conditions. At this level, the coefficients $\{\boldsymbol{v}\}$ are unknown.

So the approximate solution of the problem is completely obtained by determining these $2p+1$ variables. That is the goal of the second step. It consists in the application of boundary conditions. A least-square method combined with collocation technique can be used. This technique is rather common within meshless methods and avoids numerical instabilities occurring with pure collocation [48,68]. One chooses a set of nodes $\boldsymbol{x}_j$ on the boundary of the domain and one minimizes the error between the approximate values and the given values of u in these points. It comes to minimize the function:

$$\mathscr{T}(\boldsymbol{v}) = \frac{1}{2} \sum_{j=1}^{M} \left| u_p(\boldsymbol{x}_j) - u^d(\boldsymbol{x}_j) \right|^2 \tag{2.6}$$

This minimization leads to a linear system $[\boldsymbol{K}]\{\boldsymbol{v}\} = \{\boldsymbol{b}\}$ where $[\boldsymbol{K}]$ is an invertible matrix. Solving this system gives the vector $\{\boldsymbol{v}\}$ and so one gets the numerical solution of the problem Eq. (2.1).

2.3 Methods for setting boundary conditions

2.3.1 Two methods to account for boundary conditions

Previous papers about Taylor Meshless Method apply least-square collocation to account for the boundary conditions. Here a new approach based on discrete Lagrange multipliers is presented that looks like the techniques used in [51] for the transmission conditions.

Least-square collocation

The first method to account for the boundary conditions is the least-square collocation as shown in previous section. After solving the PDE inside the domain, see Eq. (2.5), one chooses M collocation points $\boldsymbol{x}_j$ on the boundary of the domain and minimizes the error between the approximate values and the given values of u in these points, according to Eq. (2.34). Thus this first procedure is completely defined, first by the point where the Taylor series are developed ($x = [0, 0]$ in section 2.2) that will be called development point, second by those boundary collocation points. Previous results have established that the procedure is quite robust and converges provided that the number of collocation points is large enough. This question of robustness will be re-discussed in this paper. Other boundary conditions can be studied by functions similar as Eq. (2.34); an example will be discussed in section 2.5 within linear elasticity.

Lagrange multipliers

The second method to account for the boundary conditions is based on Lagrange multipliers and it has many common characteristics with the method introduced in [51] for the discretization of the transmission conditions. One starts from the boundary conditions in a discrete weak form and the Lagrange multiplier is discretized by radial functions that are wide-spread within meshless methods [29, 30].

For simplicity, let us consider only the Dirichlet boundary condition:

$$u(\boldsymbol{x}) = u^d(\boldsymbol{x}) \quad \text{on } \Gamma \tag{2.7}$$

The weak form of Eq. (2.7) involves a Lagrange multiplier $\mu(\boldsymbol{x})$ defined on the boundary:

$$\int_\Gamma \mu(\boldsymbol{x})[u(\boldsymbol{x}) - u^d(\boldsymbol{x})]d\Gamma = 0 \quad \forall\, \mu(\boldsymbol{x}) \,\text{defined on}\, \Gamma \tag{2.8}$$

Two sets of discretization points have to be chosen on the boundary: a cloud of M collocation points ($\boldsymbol{x}_k \in \Gamma$) and a cloud of N points ($\boldsymbol{x}_i \in \Gamma$) to define

the discrete form of the Lagrange multiplier $\mu(\boldsymbol{x})$. Then the continuous weak form Eq. (2.8) is replaced by a discrete one:

$$\sum_{k=1}^{M} \mu(\boldsymbol{x}_k)[u(\boldsymbol{x}_k) - u^d(\boldsymbol{x}_k)] = 0 \quad \forall\, \mu(\boldsymbol{x}) \text{ defined on } \Gamma. \tag{2.9}$$

As in [51], we introduce a family of Gaussians, associated to the centers $\boldsymbol{x}_i$ and depending on a characteristic distance d:

$$\Phi_i(\boldsymbol{x}_k) = exp(\frac{-\|\boldsymbol{x}_k - \boldsymbol{x}_i\|^2}{d^2}) \tag{2.10}$$

These Gaussians are chosen as weighting functions in the variational equation (2.9):

$$\mu(\boldsymbol{x}_k) = \sum_{i=1}^{N} \mu_i \Phi_i(\boldsymbol{x}_k) \quad or \quad \mu(\boldsymbol{x}_k) = \Phi_i(\boldsymbol{x}_k) \tag{2.11}$$

The number of discrete Lagrange multipliers μ_i must be the same as the number of the remaining degrees of freedom in Eq. (2.5) in order to obtain a square matrix $N = 2p + 1$. Hence the proposed method can be seen as a boundary-collocation method, where the collocation conditions are satisfied in a mean sense. By combining Eqs. (2.5), (2.9) and (2.11), we arrive at a linear problem in R^N:

$$[\boldsymbol{K}]\{\boldsymbol{v}\} = \{\boldsymbol{b}\} \tag{2.12}$$

where

$$K_{ij} = \sum_{k=1}^{M} \Phi_i(\boldsymbol{x}_k) P_j(\boldsymbol{x}_k) \quad \text{and} \quad b_i = \sum_{k=1}^{M} \Phi_i(\boldsymbol{x}_k) u^d(\boldsymbol{x}_k) \tag{2.13}$$

2.3.2 Application in a rectangular domain

In this part, an application will be discussed to assess the validity of the two presented techniques accounting for boundary condition. The first goal is to show the accuracy of the solutions. The second goal is to discuss the influence of each parameter, check the robustness of the technique and get the optimal values of the parameters. Here we consider the Dirichlet problem in a rectangular domain, $0 \leqslant x \leqslant 10, 0 \leqslant y \leqslant \pi$.

$$\begin{cases} -\Delta u = 0 \qquad \text{in} \quad \Omega \\ u(x,0) = u(x,\pi) = u(10,y) = 0 \\ u(0,y) = \sin(y) \end{cases} \tag{2.14}$$

The exact solution of Eq. (2.14) is

$$u(x,y) = \sin(y) \cdot \frac{\sinh(10-x)}{\sinh(10)} \tag{2.15}$$

This problem is solved by TMM with a development point at $c = [0, 0]$ in the following. Throughout this paper, the relative error is the difference between exact and approximate solutions, divided by the maximum value of the solution:

$$\mathfrak{E} = \frac{max|u(\boldsymbol{x}) - u^h(\boldsymbol{x})|}{max|u(\boldsymbol{x})|}. \tag{2.16}$$

TMM with least-square collocation

The Taylor meshless method with least-square collocation has been previously discussed in past papers [47,50,51]. The numerical solution obtained by TMM with least-square collocation depends on two parameters: the degree p of the polynomials and the number of collocation points M. It has been established in [47,51,52] that the minimal number of collocation points is a little higher than $2p$. This will be re-discussed here, but generally we shall choose $M = 4p$ as recommended in [47,50].

The p-convergence illustrated in Fig. 2.1 shows that one can get accurate results by increasing the degree. The error decreases more or less exponentially with the degree (p-convergence). The accuracy is improved until $p = 25$ where it becomes stationary. Next, due to the numerical noise related to the accuracy limit of the computer, a small infection is observed, but the accuracy stands around 10^{-8}.

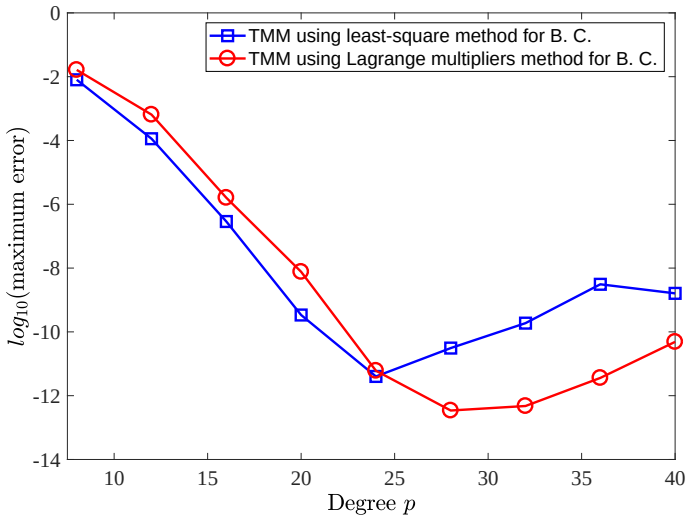


Figure 2.1: The p-convergence of TMM models for Laplace Eq. (2.14).

TMM with Lagrange multipliers

The numerical solution obtained by TMM with Lagrange multipliers mainly depends on three parameters: the degree p of the polynomials, the number of collocation points M and the effective radius d of the radial functions. The p-convergence is illustrated in Fig. 2.1. The same convergence is obtained as in TMM with least-square collocation with a very good accuracy.

The influence of the number of collocation points in TMM with Lagrange multipliers has been illustrated in Fig. 2.2 for three values of degree $p = 5$, $p = 10$ and $p = 15$. The maximal error decreases with M until an optimal number where it becomes stable. In this case, the optimal number is a little higher than $3p$. It can be seen in Fig. 2.2 that $4p$ is large enough to ensure the best convergence.

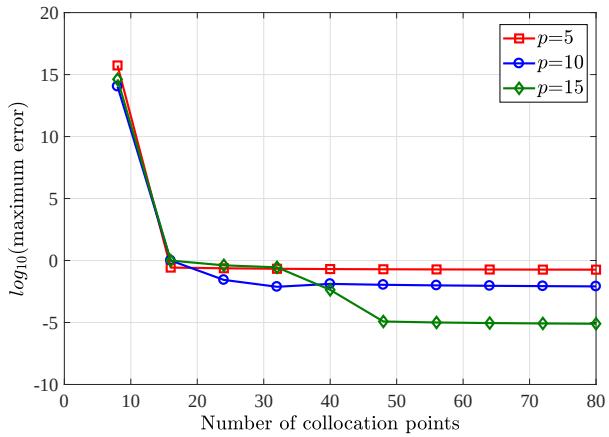


Figure 2.2: The influence of the number of collocation points on the convergence for the Laplace problem (2.14).

The parameter d is the radius of the influence zone around each Lagrange multiplier node. Four collocation points will be located in the influence zone of each Lagrange multiplier node when $d = 2 * \delta d$ is chosen, where the δd represents the minimum distance between two adjacent collocation points. The results are presented in Fig. 2.3. These results are obtained with $M = 4p$ collocation points and different degrees from $p = 8$ to $p = 44$. The results show that the influence of parameter d is increasing with the degree of the polynomial. One can get always an optimal convergence when the chosen parameter d is less than $2 * \delta d$.

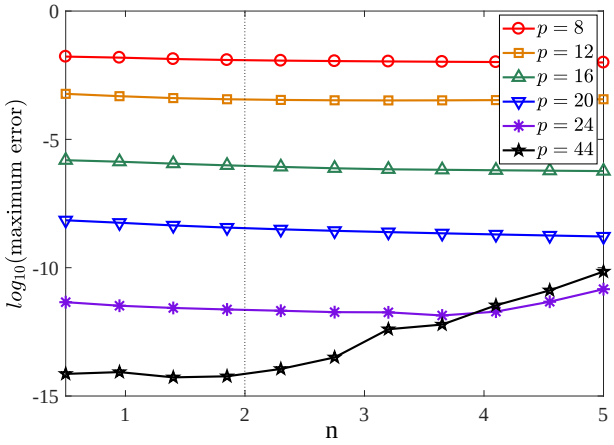


Figure 2.3: The influence of the effective radius of radial functions in TMM with Lagrange multipliers for problem (2.14). $d = n * \delta d$. Δd represents the minimum distance between two adjacent collocation points. $M = 4p$.

2.3.3 Application to Laplace equation in an unit circle

Here, we consider problem Eq. (2.1) in a unit disk $x^2 + y^2 \leqslant 1$ with $c = 0$:

$$\begin{cases} -\Delta u(\boldsymbol{x}) = 0 & \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}) = u^d(\boldsymbol{x}) & \boldsymbol{x} \in \Gamma \end{cases} \tag{2.17}$$

where $u^d(x, y) = (x - x_0)/[(x - x_0)^2 + (y - y_0)^2]$ and the singular point is set to be $\boldsymbol{x}_0 = [1, 1]$. The exact solution of Eq. (2.17) reads:

$$u(\boldsymbol{x}) = \frac{(x - x_0)}{(x - x_0)^2 + (y - y_0)^2} \tag{2.18}$$

The same case solved by TMM with least-square collocation has been presented in Tampango et al. [51]. The influence of the degree of the polynomials shape functions p and the number of collocation points M have been well discussed. Here, we will only analyze the influence of the parameters on the algorithm with Lagrange multipliers.

Table 2.1: The p-convergence of TMM with Lagrange multipliers for Eq. (2.17). $\boldsymbol{x}_0 = [1, 1]$, $M = 4p$ and $d = 3 * \delta d$.

Degree p	10	15	20	25	30	40	60
$log_{10}(\mathfrak{E})$	-1.609	-2.398	-3.016	-3.783	-4.639	-6.032	-8.952

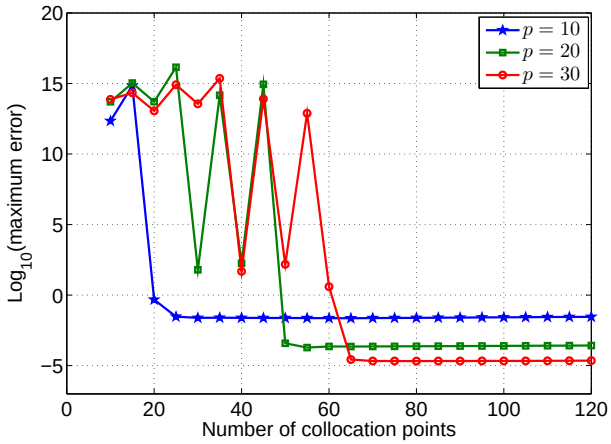


Figure 2.4: The influence of the number of collocation points on the convergence for Laplace problem Eq. (2.17).

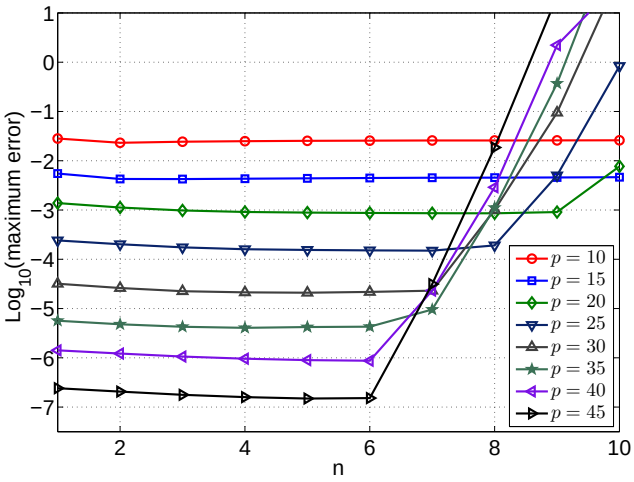


Figure 2.5: The influence of the effective radius d for Eq. (2.17). $d = n * \delta d$ and $M = 4p$, where δd represents the minimum distance between two adjacent collocation points.

The p-convergence is illustrated in Tab. 2.1. It can be seen that one converges exponentially by increasing the degree.

The influence of the number of collocation points is illustrated in Fig. 2.4 for three values of degree $p = 10$, 20 and 30. First one checks that the method fails if the number of collocation points is too small. Next the maximal error decreases with M until an optimal number where it becomes stable. The optimal number is a little higher than $2p$. It can be seen in Fig. 2.4 that $4p$ is large enough to ensure the best convergence. Here, one gets the same conclusion as in section 2.3.2.

The influence of the parameter d is illustrated in Fig. 2.5. The convergence keeps stable with a good accuracy by increasing the value of the radius d of the radial functions until a value $d = 6 * \delta d$ where the maximal error increases rapidly. The influence of d becomes sensitive with high degree polynomials. Anyway, TMM with Lagrange multipliers always works well with small values of parameter d.

2.3.4 Comments of applying boundary conditions

A new technique based on Lagrange multiplier and radial functions was presented to account for boundary conditions and assessed by several numerical tests. Contrarily to the pure collocation method, it converges, is robust and the error decreases exponentially with the degree in all the considered examples. It has about the same efficiency as the least-square collocation previously used within TMM. It depends on three parameters: the degree p, the number of collocation points M and d the characteristic distance appearing in the gaussian functions. A safe carrying out is obtained for $M = 4p$ and $d/\delta d$ between 1 and 4, δd being the distance between neighbor collocation points. With these choices, the algorithm depends only on the degree p that controls the convergence. As in previous versions of TMM, very high degrees (say 20, 40) can be chosen, which can yield very accurate solutions without huge computational cost unlike high degree finite elements. Nevertheless the previously used least-square collocation method is as efficient as the new one and has the advantage to need only one family of discretization points.

2.4 Using least-square collocation to connect various sub-domains

It has not been proved yet that the least-square collocation method can be applied to the transmission conditions by keeping the property of exponential convergence with the degree. By comparison with Lagrange multipliers, it has the drawback to lead to full matrices, but the number of degrees of freedom is smaller and it is more easily defined. Only a set of collocation points is needed and their number and position can be chosen very freely. Of course this can be associated with a treatment of the boundary conditions and the two methods discussed in section 5.3 will be investigated.

2.4.1 Full least-square method

To complete the solution of the PDE by Taylor series obtained in section 2.2, a numerical procedure requires a treatment of boundary and transmission conditions. Let us begin by the simplest procedure where least-square collocation is applied in the two cases.

2.4.1.1 Statement

Let us consider an elliptic equation in a domain Ω

$$\begin{cases} \mathbf{L}u = f & \text{in} \quad \Omega \\ u(\boldsymbol{x}) = u^d(\boldsymbol{x}) & \text{on} \quad \Gamma \end{cases} \tag{2.19}$$

where $\mathbf{L}$ is a second order differential operator.

In order to make a piecewise resolution, an interface (Γ_r) is introduced by splitting the domain Ω into two sub-domains Ω_1 and Ω_2 (see Fig. 2.6). A splitting in several sub-domains can be achieved in the same way.

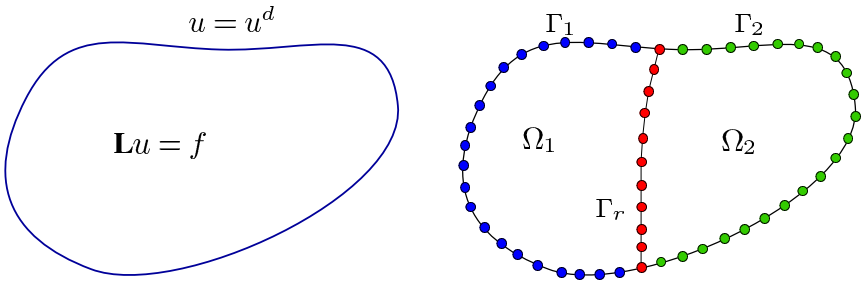


Figure 2.6: Two sub-domains for the Laplace problem (2.19).

The discretization of the domain is generated by two types of nodes:
- nodes on the external boundaries Γ_1 and Γ_2 for boundary conditions.
- nodes at the interface Γ_r for transmission conditions.

The two conditions will be accounted by using a least-square method as shown in the previous section. Thus a set of nodes is chosen on each sub-boundary and on the interface:
- a set of M_1 collocation nodes on the boundary Γ_1.
- a set of M_2 collocation nodes on the boundary Γ_2.
- a set of M_r collocation points on the interface Γ_r.

With the initial formulation of the TMM, it is possible to build polynomial shape functions for problems (2.19) in each sub-domain:

$$u^1(\boldsymbol{x}) = P_0^1(\boldsymbol{x}) + \sum_{i=1}^{2p_1+1} P_i^1(\boldsymbol{x}) v_i^1 \tag{2.20}$$

$$u^2(\boldsymbol{x}) = P_0^2(\boldsymbol{x}) + \sum_{i=1}^{2p_2+1} P_i^2(\boldsymbol{x}) v_i^2 \tag{2.21}$$

where the $P_0^1(\boldsymbol{x})$ and $P_0^2(\boldsymbol{x})$ are used to balance the right-hand side analytical function $f(\boldsymbol{x})$ of Eq. (2.19).

The function accounting for boundary conditions on Γ_1 and Γ_2 is exactly the same as in previous papers:

$$\mathscr{T}_1(\boldsymbol{v}^1) = \frac{1}{2} \sum_{j_1=1}^{M_1} \left| P_0^1(\boldsymbol{x}_{j1}) + \sum_{i=1}^{2p_1+1} P_i^1(\boldsymbol{x}_{j_1}) v_i^1 - u^d(\boldsymbol{x}_{j_1}) \right|^2 \tag{2.22}$$

$$\mathscr{T}_2(\boldsymbol{v}^2) = \frac{1}{2} \sum_{j_2=1}^{M_2} \left| P_0^2(\boldsymbol{x}_{j2}) + \sum_{i=1}^{2p_2+1} P_i^2(\boldsymbol{x}_{j_2}) v_i^2 - u^d(\boldsymbol{x}_{j_2}) \right|^2 \tag{2.23}$$

In the same way, the two transmission conditions will be ensured via the following coupling function:

$$\mathscr{C}(\boldsymbol{v}^1, \boldsymbol{v}^2) = \frac{1}{2} \sum_{j_r=1}^{M_r} \left[\left| u^1(\boldsymbol{x}_{j_r}) - u^2(\boldsymbol{x}_{j_r}) \right|^2 + \left| \frac{\partial u^1}{\partial n}(\boldsymbol{x}_{j_r}) - \frac{\partial u^2}{\partial n}(\boldsymbol{x}_{j_r}) \right|^2 \right] \tag{2.24}$$

Then for the whole problem, searching the variables $\{v^{(1)}\}$ and $\{v^{(2)}\}$ is equivalent to minimizing the following function:

$$\mathscr{T}(\boldsymbol{v}^1, \boldsymbol{v}^2) = \mathscr{T}_1(\boldsymbol{v}^1) + \mathscr{T}_2(\boldsymbol{v}^2) + \mathscr{C}(\boldsymbol{v}^1, \boldsymbol{v}^2) \tag{2.25}$$

The minimization of the function $\mathscr{T}(\boldsymbol{v}^1, \boldsymbol{v}^2)$ leads to a linear symmetric and invertible system as follows:

$$\begin{bmatrix} \boldsymbol{K}_{11} & \boldsymbol{K}_{12} \\ \boldsymbol{K}_{21} & \boldsymbol{K}_{22} \end{bmatrix} \begin{Bmatrix} \boldsymbol{v}^1 \\ \boldsymbol{v}^2 \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \end{Bmatrix} \tag{2.26}$$

2.4.1.2 Application to Laplace equation in a disk

In this section, the numerical solutions obtained by this piecewise approach are presented to study the validity of the TMM proposed in previous section. The Laplace problem Eq. (2.17) is reconsidered with a singularity at $\boldsymbol{x}_0 = [1.0, 1.0]$. The domain is subdivided in two parts as shown on Fig. 2.7. The parameters of the discretization are the following ones:

- approximation degrees in the sub-domains (p_1 and p_2).
- number of collocation on the boundary of each sub-domain (M_1 and M_2).
- center of the Taylor expansion in each sub-domain ($\boldsymbol{x}_{c_1}$ and $\boldsymbol{x}_{c_2}$).
- number of collocation nodes on the interface (M_r).

The development points of the approximate series are chosen at $\boldsymbol{x}_{c_1} = [-0.5, 0]$ for the sub-domain on the left-hand side and at $\boldsymbol{x}_{c_2} = [0, 0]$ on the right-hand side. The influence of the other parameters will be discussed in details.

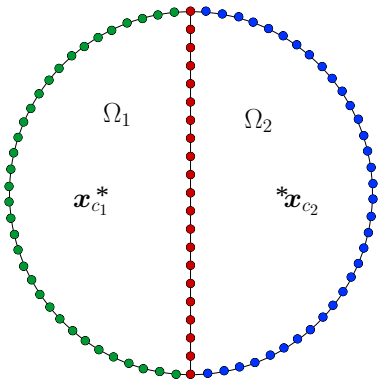


Figure 2.7: Two sub-domains for the Laplace problem (2.17).

We present the error on the inner circle of radius $r = 0.8$ in Fig. 2.8. This circle encounters the interface at $\theta = \pi/2$ and $\theta = 3\pi/2$. The Fig. 2.8 presents the approximate solution and the exact solution for $p_1 = p_2 = p = 10$, $M_1 = M_2 = 2p$, $M_r = p$. The figure shows a good correspondence between the exact and the approximate solution. Thus the transmission conditions have been well accounted for.

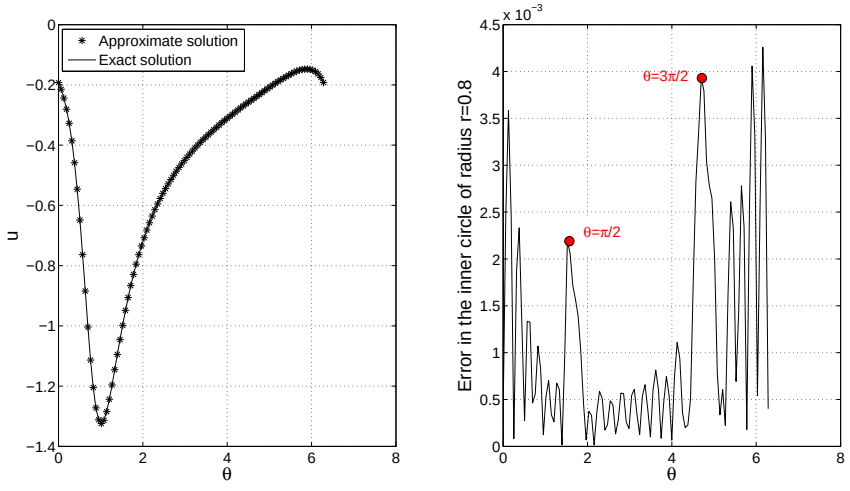


Figure 2.8: Laplace equation (2.17) in the unit disk. Solutions on the inner circle of radius 0.8, $p_1 = p_2 = p = 10$, $M_1 = M_2 = 2p$, $M_r = p$.

The optimal number of boundary collocation points was previously discussed [47]: a choice $M_1 = 2p$, $M_2 = 2p$ is sufficient to ensure the convergence. Beyond this limit, the error is stable. The method works well with more points, but this does not improve the convergence.

We are now interested in the influence of the number of coupling nodes (M_r) on the quality of the solution. It defines the quality of the information transmitted along the interface. The Tab. 2.2 presents the error of the whole domain for several approximation degrees and for different number of collocation points M_r. As for the boundary collocation, the error decreases when M_r increases and becomes constant from some threshold, around $M_r = p$. From this value, the convergence is stabilized for larger M_r. Furthermore, the convergence is rapidly improved by increasing the degree. The above results prove the robustness of the least-square technique applied for both boundary and interface condition.

Table 2.2: Laplace problem (2.17) split into two sub-domains. The influence of the number of coupling nodes M_r. $p_1 = p_2 = p$. The maximal error of the whole domain.

Degree	M_r	log_{10}(maximum error)
$p = 10$	6	-0.8724
	8	-1.8752
	20	-1.7609
$p = 20$	12	-1.9941
	16	-3.6059
	40	-3.6191
$p = 30$	22	-4.7267
	28	-5.2471
	60	-5.3946

In the previous case, the polynomials on both sides of the interface had the same degree. Now we are interested in the case with different degrees in each sub-domain, always in order to test the robustness of the coupling technique. The results, presented in the Tab. 2.3, are similar to the previous approximation with a common degree. The error is stabilized from some threshold around $M_r = (p_1 + p_2)/2$. One sees that a coarse discretization is sufficient in the left part, in which the gradient is smaller: indeed one gets about the same accuracy with $(p_1, p_2) = (7, 15)$ as with $p_1 = p_2 = 15$. This also establishes the robustness of the least-square coupling technique that works well with different degrees.

Table 2.3: Laplace problem (2.17) split into two sub-domains. Coupling polynomials with different degrees by least-square method. The maximal error of the whole domain. $M_1 = 2p_1$ and $M_2 = 2p_2$.

p_1	p_2	M_r	log_{10}(maximal error)	p_1	p_2	M_r	log_{10}(maximal error)
5	15	6	-0.1261	12	15	6	-0.0261
		8	-1.5376			10	-2.3342
		10	-2.3962			12	-2.7563
		20	-2.3150			30	-2.6764
9	15	6	-0.0368	15	15	8	-0.9729
		10	-2.4050			10	-2.2286
		12	-2.7529			12	-2.7885
		28	-2.6682			30	-2.7231

2.4.1.3 Application to Poisson equation in a crown

Here we consider a Poisson's problem:

$$\begin{cases} -\triangle u(x,y) = -\dfrac{4}{(x^2+y^2)^2} & \text{in} \quad \Omega \\ u^d(x,y) = \dfrac{1}{x^2+y^2} & \text{on} \quad \Gamma \end{cases} \tag{2.27}$$

The domain is a crown ($\Omega = \{(x,y)/r_1^2 \leq x^2+y^2 \leq r_2^2\}$, with $r_1 = 0.8$ and $r_2 = 1$). The exact solution of Eq. (2.27) is $u(x,y) = 1/x^2+y^2$. This problem has been solved in [51]. The result obtained from TMM with one single domain were always completely wrong. That is due to the singularity at $\boldsymbol{x}_0 = [0,0]$ in the middle of crown. There is no development point that can cover the whole domain by avoiding the singularity (see Fig. 2.9).

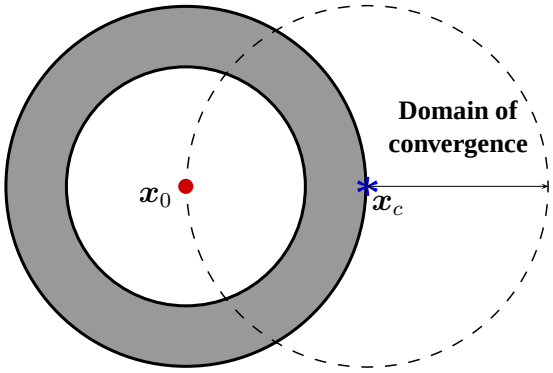


Figure 2.9: Discretization of the crown for a piecewise resolution.

In the paper [51], this problem has been solved by dividing the whole domain into several sub-domains coupled by using Lagrange multipliers and Arlequin method. Here we are interested in the behavior of the least-square collocation approach used both for boundary and interface conditions.

To validate TMM coupled with the least-square technique, we split the crown into several sectors (Fig. 2.10). In each sector, the development point of the approximate series ($\boldsymbol{x}_{c_i}$) has been chosen in the middle of the outer arc.

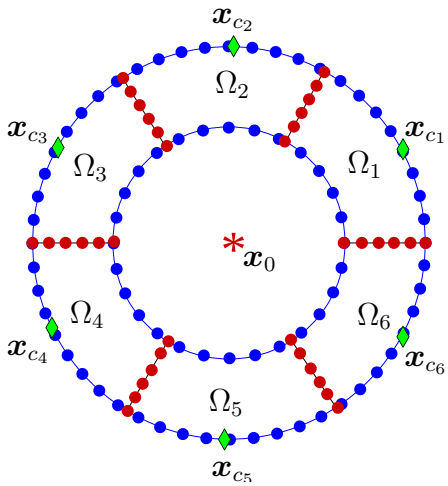


Figure 2.10: Discretization of the crown for a piecewise resolution.

The Fig. 2.11 presents the maximal error of the TMM model with different numbers of sub-domains. We choose p collocation points on each interface and $2p$ collocation points on the boundary. This result shows that the convergence is improved by increasing the number of sub-domains. The main reason is that each Taylor series can widely cover its sub-domain by avoiding the singularity.

The convergence with the degree is illustrated in Fig. 2.12. Here the crown is split into six sub-domains. The results show that the convergence is improved by increasing the degree until a value $p = 20$ where it becomes stable, with a very good accuracy. Likely such a plateau is due to numerical noise related to the accuracy limit of the computer. The obtained accuracy ($10^{-4.19}$ for $p = 10$, $10^{-7.49}$ for $p = 20$) is almost the same as those of Reference [51] where various Lagrange multipliers techniques were applied for the interface conditions.

2.4.1.4 Outcome of full least-square method

The full least-square method works well. It is robust and does not depend on the number of collocation points. It converges exponentially with the degree,

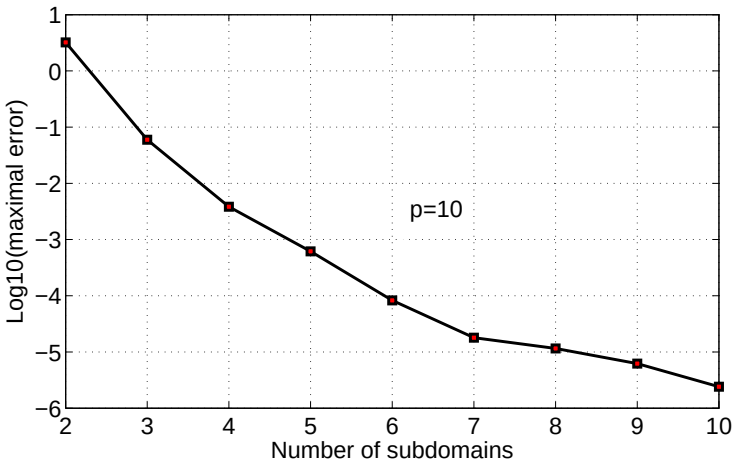


Figure 2.11: The convergence with the number of sub-domains ($p = 10$).

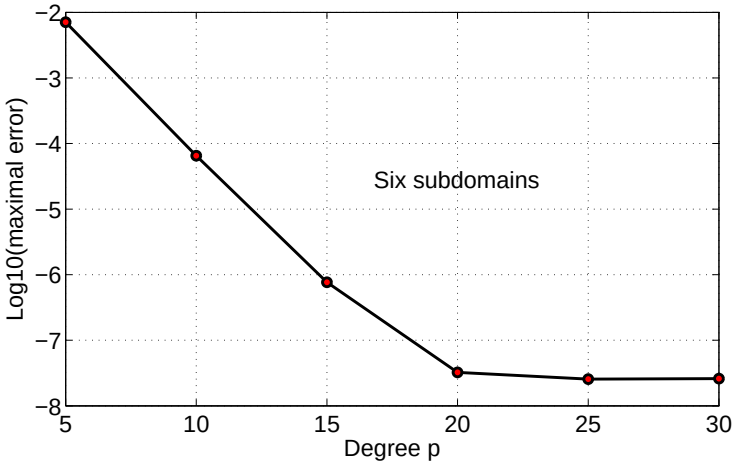


Figure 2.12: The p-convergence solved by TMM with six sub-domains.

about as the previous technique least-square/Lagrange multipliers [51]. Thus it provides an interesting alternative.

2.4.2 A mixed Lagrange/least-square method

We now check that transmission least-square collocation is also compatible with a Lagrange multiplier discretization of boundary conditions. This will be complementary to the method in [51] where the interface condition was treated by Lagrange multipliers and the boundary condition by least-square collocation.

2.4.2.1 Statement

In this section, Lagrange multiplier method is applied for boundary condition in a piecewise resolution and least-square collocation for interface condition. Let us consider a Poisson problem with Dirichlet boundary condition:

$$\begin{cases} -\triangle u(\boldsymbol{x}) = q(\boldsymbol{x}) & \text{in} \quad \Omega \\ u(\boldsymbol{x}) = g(\boldsymbol{x}) & \text{on} \quad \Gamma \end{cases} \tag{2.28}$$

An interface (Γ_r) is introduced by splitting the domain Ω into two sub-domains Ω_1 and Ω_2 (see Fig. 2.13).

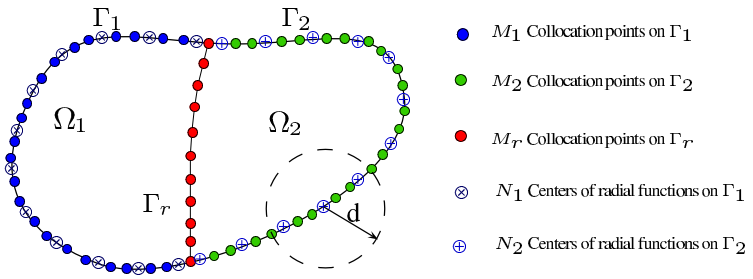


Figure 2.13: Two sub-domains for Laplace problem (2.28).

Here, transmission conditions will be considered as the target function and discretized by the least-square collocation. For this purpose a set of M_r collocation points is chosen on the interface and transmission conditions would be satisfied in a mean sense via Eq. (2.24). The weak form of the boundary condition involving Lagrange multipliers $\mu^s(\boldsymbol{x})(s \in [1,2])$ reads:

$$\int_{\Gamma_s} \mu^s(\boldsymbol{x})[u^s(\boldsymbol{x}) - g^s(\boldsymbol{x})]d\Gamma = 0 \quad \forall\, \mu^s(\boldsymbol{x}) \,\text{defined on}\, \Gamma_s. \tag{2.29}$$

Then for the whole problem, searching the variables $\boldsymbol{v}$ and $\boldsymbol{\mu}$ is equivalent to vanishing the gradient of the following function:

$$\mathscr{L}(\boldsymbol{v},\boldsymbol{\mu}) = \mathscr{C}(\boldsymbol{v}^1,\boldsymbol{v}^2) + \sum_{s=1}^{2}\int_{\Gamma_s}\mu^s(\boldsymbol{x})[u^s(\boldsymbol{x}) - g^s(\boldsymbol{x})]d\Gamma \tag{2.30}$$

where $\mathscr{C}(\boldsymbol{v}^1,\boldsymbol{v}^2)$ is the same as in Eq. (2.24).

As in Section 5.3, two sets of discretization points have to be chosen on the boundary in each sub-domain:

- M_s collocation points ($\boldsymbol{x}_m \in \Gamma_s$) to discrete the boundary condition.
- N_s points ($\boldsymbol{x}_n \in \Gamma_s$) to define the Lagrange multiplier.

Then the discretization form of Eq. (2.30) reads:

$$\mathscr{L}(\boldsymbol{v},\boldsymbol{\mu}) = \frac{1}{2}{}^t\{\boldsymbol{v}\}[\boldsymbol{K}_r]\{\boldsymbol{v}\} - {}^t\{\boldsymbol{b}_r\}\{\boldsymbol{v}\} + {}^t\{\boldsymbol{\mu}\}[\boldsymbol{C}]\{\boldsymbol{v}\} - {}^t\{\boldsymbol{\mu}\}\{\boldsymbol{b}\} \tag{2.31}$$

where

$$[\boldsymbol{K}_r] = \begin{bmatrix} K_{r11} & -K_{r12} \\ -K_{r21} & K_{r22} \end{bmatrix}, \quad [\boldsymbol{C}] = \begin{bmatrix} C^1 & 0 \\ 0 & C^2 \end{bmatrix}$$

$$[K_{rst}]_{ij} = \sum_{m=1}^{M_r}\left(P_i^s(\boldsymbol{x}_m)P_j^t(\boldsymbol{x}_m) + \frac{\partial P_i^s(\boldsymbol{x}_m)}{\partial n}\frac{\partial P_j^t(\boldsymbol{x}_m)}{\partial n}\right).$$

$$[C^s]_{ij} = \sum_{n=1}^{M_s}\Phi_i^s(\boldsymbol{x}_n)P_j^s(\boldsymbol{x}_n).$$

$$[\boldsymbol{b}] = \begin{bmatrix} b^1 \\ b^2 \end{bmatrix}, \quad [\boldsymbol{b}_r] = \begin{bmatrix} b_r^1 \\ b_r^2 \end{bmatrix}.$$

$$[b^s]_i = \sum_{n=1}^{M_s}\Phi_i^s(\boldsymbol{x}_n)\left(g^s(\boldsymbol{x}_n) - P_0^s(\boldsymbol{x}_n)\right).$$

$$[b_r^s]_i = \sum_{m=1}^{M_r}P_i^s(\boldsymbol{x}_m)\left(P_0^2(\boldsymbol{x}_m) - P_0^1(\boldsymbol{x}_m)\right) + \frac{\partial P_i^s(\boldsymbol{x}_m)}{\partial n}\left(\frac{\partial P_0^2(\boldsymbol{x}_m)}{\partial n} - \frac{\partial P_0^1(\boldsymbol{x}_m)}{\partial n}\right).$$

(Note that: $s,t = 1,2$ and $\boldsymbol{x}_m \in \Gamma_r$ and $\boldsymbol{x}_n \in \Gamma_s$.)

The stationarity of the function $\mathscr{L}(\boldsymbol{v},\boldsymbol{\mu})$ leads to a linear symmetric and invertible system as follows:

$$\begin{bmatrix} \boldsymbol{K}_r & {}^t\boldsymbol{C} \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix}\begin{Bmatrix} \boldsymbol{v} \\ \boldsymbol{\mu} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{b}_r \\ \boldsymbol{b} \end{Bmatrix} \tag{2.32}$$

2.4.2.2 Application to Laplace equation in a disk

Let us consider the Laplace problem (2.17) with a singularity at $[1,1]$. We solve this problem in a piecewise way by the method described in section

2.4.2.1. The domain is split into two sub-domains. The development points of the Taylor series are chosen at $\boldsymbol{x}_{c_1} = [-0.5, 0]$ for the sub-domain on the left-hand side and at $\boldsymbol{x}_{c_2} = [0, 0]$ on the right-hand side.

Sufficiently large numbers of collocation points and of Lagrange multipliers have been chosen in such a way that the accuracy is optimal. The radial functions needed to discretize the Lagrange multipliers depend on a length d that is compared with the minimal distance between neighbor collocation points, called δd. In Tab. 2.4, the maximal error is reported as a function of the degree and of the ratio $d/\delta d$ that was varied in the range $[1, 6]$. Clearly the efficiency of the method depends very little on this ratio that can be chosen rather freely.

Table 2.4: The p-convergence for Laplace Eq. (2.17) solved by using two sub-domains. δd represents the minimum distance between to adjacent collocation points. $M_1 = M_2 = 2p$, $N_1 = N_2 = p$ and $M_r = 2p$.

d	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$
$1 * \delta d$	-1.5550	-3.2542	-5.0940	-6.3572	-6.2538
$3 * \delta d$	-1.6715	-3.4039	-5.1856	-6.3524	-7.6927
$6 * \delta d$	-1.6236	-3.3400	-4.9338	-5.4374	-5.5595

These results can be compared with the single domain solution presented in section 2.3.3, to the pure least-square method of section 2.4.1 or the method of reference [51] that combines Lagrange multipliers and least-square collocation, but in the opposite way, see Tab. 2.5. Clearly all these methods give almost the same results and all converge exponentially with the degree. This means that the discretization of boundary and transmission conditions can be done by either method, without significant influence on the effectiveness.

2.4.2.3 Application to Poisson equation in a crown

Poisson Eq. (2.27) is reconsidered in this section and solved by the method of section 2.4.2.1. The crown has been split in six sub-domains with the same development points as in section 2.4.1.3. The degree varies and the numbers of discretization nodes are as follows: $M_r = p$, $M_s = 2p$ and $N_s = 2p$. As for the previous example, the influence of the ratio $d/\delta d$ is studied in the range $[1, 6]$. Roughly, this influence is very weak except sometimes for $d = 6 * \delta d$, where the accuracy is slightly lower. The accuracy obtained by this Lagrange/least-square technique has been compared with the same alternative methods as for the previous example. The results illustrated in Tab. 2.7 confirm that all these methods are more or less equivalent. Let us mention the quasi-perfect

Table 2.5: The maximal relative error of the solutions of Laplace problem (2.17) obtained by various methods.

Degree p	$p = 10$	$p = 20$	$p = 30$
Single domain (section 2.3.3)	-1.609	-3.016	-4.639
Least-square/Lagrange (Method of reference [51])	-1.578	-3.081	-4.767
Least-square/Least-square (section 2.4.1.2)	-1.893	-3.667	-5.331
Lagrange/Least-square (Present method)	-1.672	-3.404	-5.186

solution obtained by the present method with a degree $p = 30$ and $d = \delta d$ or $d = 2 * \delta d$: the error is less than 10^{-12}.

Table 2.6: The maximal relative error of the solutions for Poisson problem (2.27) split into six sub-domains. δd represents the minimum distance between two adjacent collocation points. $M_r = p$, $M_s = 2p$ and $N_s = 2p$, where M_r, M_s and N_s represent the number of collocation points on the interface, the number of collocation points on the boundary of each sub-domain and the number of discretization points of Lagrange multipliers μ^s on the boundary of each sub-domain.

d	$p = 10$	$p = 15$	$p = 20$	$p = 25$	$p = 30$
$1 * \delta d$	-4.5857	-7.2258	-9.0945	-10.5019	-12.9376
$3 * \delta d$	-5.3912	-6.8305	-9.6991	-10.5763	-11.2436
$6 * \delta d$	-4.2289	-6.7304	-8.3716	-8.9787	-9.0100

Table 2.7: The maximal relative error of the solutions for Poisson problem (2.27) split into six sub-domains.

	$p = 10$	$p = 20$	$p = 30$
Least-square/Lagrange (Method of reference [51])	-3.615	-8.177	-8.350
Least-square/Least-square (section 2.4.1.3)	-4.186	-7.491	-7.586
Lagrange/Least-square (Present method)	-4.586	-9.095	-12.938

2.4.3 Comments about applying interface conditions

In this section, the discretization of the transmission condition by least-square collocation has been discussed and compared with previous techniques based on Lagrange multipliers. From numerical tests, we can conclude that all these methods have about the same efficiency and robustness. All converge exponentially with the degree and are few sensitive to the discretization parameters.

2.5 Application in 2D elasticity

The previous examples concern a single elliptic equation of Laplace or Poisson type. We now discuss examples in 2D elasticity. In the literature, one can find few applications of TMM in linear elasticity [49, 50] or even in nonlinear elasticity [54]. It seems that the algorithm works about in the same way for a system as for a single PDE, but this could be no longer the case if the sought solution is not a perfectly smooth function. In real life, one needs to represent point-wise forces, corners or cracks, in which cases the solution is more or less singular. Thus the representation of non-smooth solutions by Taylor series is a serious challenge and this research will be initiated here by an example with a square domain.

The reliability is a central point in the assessment of a new numerical method. In the previous examples presented in this chapter, the convergence is exponential with a limit at a very high accuracy of 10^{-7} or 10^{-8}. This limit may be greater, for instance if the development point is not well chosen, see for instance Fig. 4 in [49]. Thus for the sake of reliability, it is important that this limit of accuracy remain sufficiently low.

In this part we apply least-square collocation whose reliability and efficiency were proved in the front part of this paper. Here, we consider 2D elasticity problems in the generic form:

$$\begin{cases} \boldsymbol{G}u(x) = 0 & \text{in} \quad \Omega \\ u(\boldsymbol{x}) = u^d(\boldsymbol{x}) & \text{on} \quad \Gamma_u \\ \boldsymbol{T}u(\boldsymbol{x}) = t^d(\boldsymbol{x}) & \text{on} \quad \Gamma_t \end{cases} \tag{2.33}$$

where $\boldsymbol{G}$ and $\boldsymbol{T}$ are differential operators:

$$\boldsymbol{G} = \frac{E}{1-\nu^2} \begin{bmatrix} \dfrac{\partial^2}{\partial x^2} + \dfrac{1-\nu}{2}\dfrac{\partial^2}{\partial y^2} & \dfrac{1+\nu}{2}\dfrac{\partial^2}{\partial x \partial y} \\ \dfrac{1+\nu}{2}\dfrac{\partial^2}{\partial x \partial y} & \dfrac{\partial^2}{\partial y^2} + \dfrac{1-\nu}{2}\dfrac{\partial^2}{\partial x^2} \end{bmatrix}$$

$$\boldsymbol{T} = \frac{E}{1-\nu^2}\begin{bmatrix} n_x\dfrac{\partial}{\partial x} + n_y\dfrac{1-\nu}{2}\dfrac{\partial}{\partial y} & n_x\nu\dfrac{\partial}{\partial y} + n_y\dfrac{1-\nu}{2}\dfrac{\partial}{\partial x} \\ n_y\nu\dfrac{\partial}{\partial x} + n_x\dfrac{1-\nu}{2}\dfrac{\partial}{\partial y} & n_y\dfrac{\partial}{\partial y} + n_x\dfrac{1-\nu}{2}\dfrac{\partial}{\partial x} \end{bmatrix}$$

in which ν is the Poisson ratio and E is the Young modulus. The n_x and n_y are the components of the normal to the boundary.

Assuming $u^h(\boldsymbol{x})$ has satisfied the control equation $\boldsymbol{G}u(x) = 0$, the only thing need to be considered is the boundary conditions. It comes to minimize the following function:

$$J(v) = \frac{1}{2}\sum_{\boldsymbol{x}_j\in\Gamma_u}\left|u^h(\boldsymbol{x}_j) - u^d(\boldsymbol{x}_j)\right|^2 + w\cdot\frac{1}{2}\sum_{\boldsymbol{x}_j\in\Gamma_t}\left|\boldsymbol{T}u^h(\boldsymbol{x}_j) - t^d(\boldsymbol{x}_j)\right|^2 \quad (2.34)$$

2.5.1 A 2D elasticity problem without singularity

In this section, we consider an elasticity problem (2.33) with a uniformly distributed force $\sigma_x = q_1$ along $x = L$ and $\sigma_y = q_2$ along $y = 0$ and $y = B$ (see Fig. 2.14).

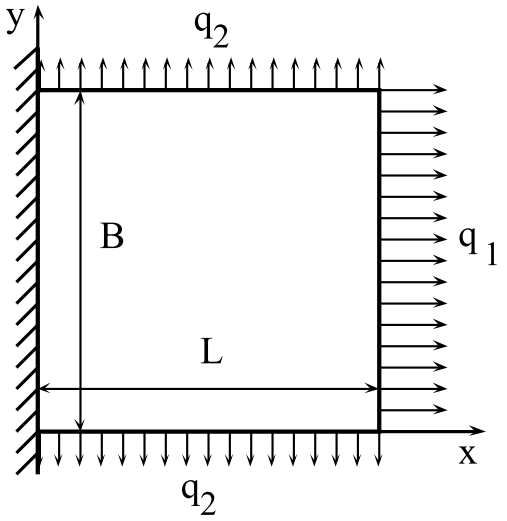


Figure 2.14: A first 2D elasticity problem

The constants are as follows: $E = 1000$, $\nu = 0.3$, $L = 10$, $B = 10$, $q_1 = 100$ and $q_2 = \nu\cdot q_1$. The development point is chosen at the center $X_0 = [5, 5]$. The analytical solution in this case varies linearly:

$$\begin{cases} u(x,y) = \dfrac{x}{E}(q_1 - \nu q_2) \\ v(x,y) = 0 \end{cases} \quad (2.35)$$

where the u and v represent the displacements along x-axis direction and y-axis direction respectively.

The numerical results presented in Tab. 2.8 show that TMM satisfies naturally the consistency condition. It is able to recover a spatially linear function with a very high accuracy. Nevertheless this accuracy slightly decreases for large degrees due to numerical noise, but it remains lower than 10^{-10}.

Table 2.8: 2D elasticity problem shown in Fig. 2.14. The error shown in table is log_{10}(maximal error).

Degree p	1	3	5	6	10	15
error of u	-15.612	-15.135	-14.737	-12.587	-11.670	-10.167
error of v	-16.286	-15.279	-14.525	-12.627	-11.870	-10.557

2.5.2 A 2D elasticity problem with singularity

In this section, we consider a 2D elasticity problem (2.33) with a linear distribution of force σ_x along the end (see Fig. 2.15). In this case, there will be a singularity in the top-left corner of the square. The data are as follows: $E = 1000$, $\nu = 0.3$, $L = 10$, $B = 10$ and $q = 100$. This example has no analytical solution so that the reference solution has to be defined numerically. Several finite element solutions have been computed with linear and quadratic elements and with several meshes. We consider that a 500×500 mesh and four node quadrilateral elements (Q4) leads to a sort of reference solution. Indeed with respect to a 250×250 mesh of $Q4$ elements or a 100×100 mesh of $Q8$ elements, the difference is less than 6.8×10^{-4}.

2.5.2.1 TMM with one domain

A single Taylor series is first considered with a development point at the center of the square $X_0 = [5, 5]$. The numerical results presented in Tab. 2.9 indicate that the convergence is improved slowly by increasing the degree of the polynomials. There is an accuracy limit that is never less than 1%. According to the distribution of the error presented in Fig. 2.16, this maximal error is mainly governed by the singularity located at the left corner of the square [0, 10]. The relative errors at the point $X = [10, 10]$ that is far away from the singularity point is illustrated in Fig. 2.17. One observes also a

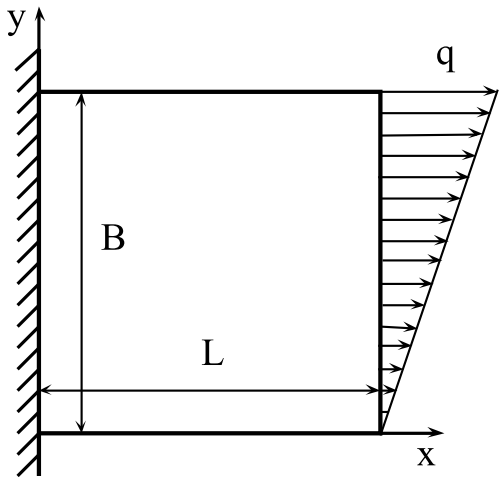


Figure 2.15: A second 2D elasticity problem

plateau from a degree $p = 20$, but with a more satisfactory exactness. One may wonder if the results are improved by a piecewise resolution.

Table 2.9: TMM with a single domain for the 2D elasticity problem shown in Fig. 2.15. The maximal error in the whole domain.

Relative error	$p = 10$	$p = 20$	$p = 30$	$p = 40$
Max error of u	0.009	0.009	0.0055	0.0035
Max error of v	0.080	0.030	0.018	0.012

2.5.2.2 TMM with piecewise resolution

In this section, we split the whole domain into two or three subdomains, as pictured in Fig. 2.18. Least-square method is applied both for boundary and interface conditions. The comparison with the solution obtained by a single domain resolution (Tab. 2.9) shows that the piecewise resolution improves only slightly the convergence, the order of magnitude of the error remaining about 0.8%. Several other tests have been performed, but this did not lead to significant improvements. Hence it seems difficult to represent very accurately more or less singular solutions by Taylor series, even with piecewise resolutions. Other tests involving cracks will confirm this result, see chapter 5. Thus one of the next challenges for TMM is to find new procedures for a better account of singular solutions.

The results presented in Tab. 2.10 and Tab. 2.11 are obtained by using TMM with sub-domains. Least-square method is applied for boundary and

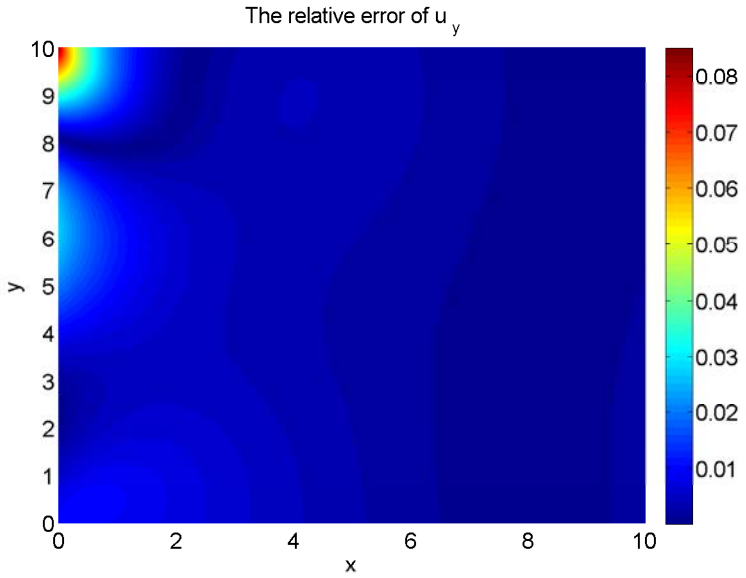


Figure 2.16: Problem of Fig. 2.15 solved with one domain. The relative error of v in the whole domain with the degree p=10.

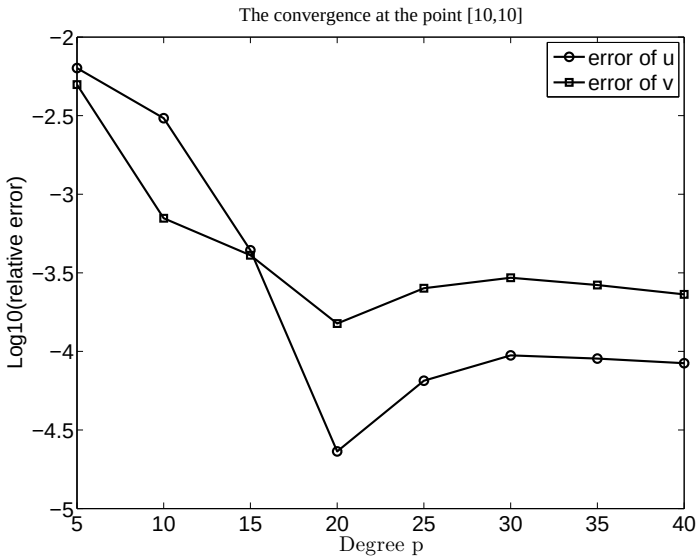


Figure 2.17: Problem of Fig. 2.15 solved with one domain. The p-convergence of the relative error at the point $X = [10, 10]$ far away from the singularity.

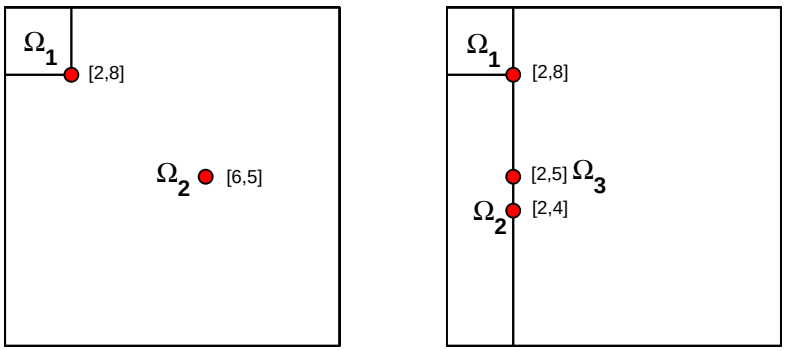


Figure 2.18: TMM model with sub-domains. The red points are the development points of each sub-domain.

interface conditions. The comparison with the solution obtained by a single domain resolution (Tab. 2.9) shows that the piecewise resolution slightly improves the convergence.

Table 2.10: . The p-convergence of TMM with two sub-domains for the 2D elasticity problem shown in Fig. 2.15.

p_1	p_2	Max error of u	Max error of v
5	5	0.018	0.093
10	10	0.011	0.020
15	10	0.012	0.018
20	15	0.0055	0.012
20	20	0.004	0.010

Table 2.11: The p-convergence of TMM with three sub-domains for the 2D elasticity problem shown in Fig. 2.15.

p_1	p_2	p_3	Max error of u	Max error of v
5	5	5	0.016	0.045
10	10	10	0.009	0.02
15	15	15	0.0055	0.012
20	15	15	0.0045	0.009
20	10	15	0.0045	0.008
20	15	20	0.0045	0.009

2.6 A very large scale test

The previous examples concern only two dimensional partial differential equations and have been solved by TMM with a single domain or a few number of sub-domains. It was established that the Lagrange multiplier and least-square collocation are both robust and efficient when accounting for boundary and interface conditions. However the number of degrees of freedom of the previous cases are really small. Hence one considers a three dimensional case divided into many sub-domains to check the robustness and efficiency of TMM when solving large scale problems.

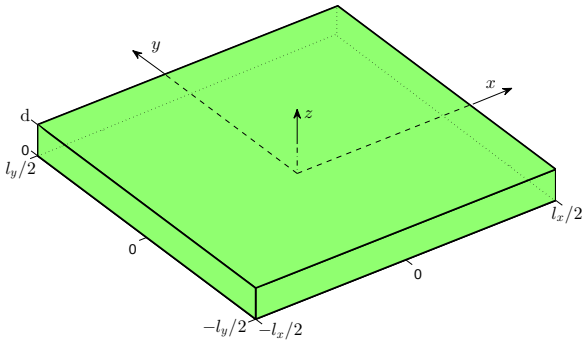


Figure 2.19: Sketch for the shape of domain, problem Eq. (2.36).

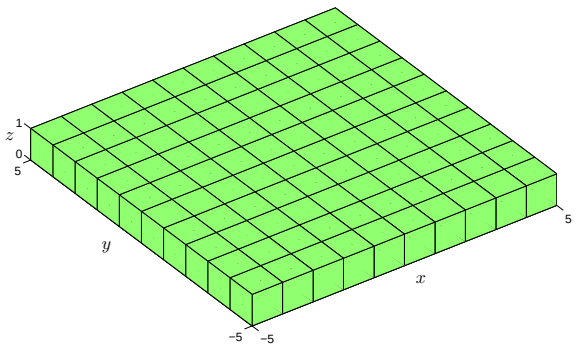


Figure 2.20: Sketch for the plate divided into 100 sub-domains, problem Eq. (2.36).

This benchmark is motivated by wrinkling instabilities of film-substrate systems [75, 76], where one observes many spatial oscillations. Here one accounts only for the substrate and the film is represented by a prescribed sinusoidal displacement. The domain is a rectangular hexahedron with sides

$l_x \times l_y \times d$, see Fig. 2.19. One considers the Laplace equation with Dirichlet boundary conditions:

$$\begin{cases} \triangle u(x,y,z) = 0 \qquad \text{in} \quad \Omega \\ u(x,y,0) = 0 \\ u(x,y,d) = sin(\pi x) sin(\pi y) \end{cases} \tag{2.36}$$

The exact solution is $u(x,y,z) = sin(\pi x) sin(\pi y) sinh(\sqrt{2}\pi z)/sinh(\sqrt{2}\pi d)$.

The domain is split in cubes of size $1 \times 1 \times 1$ that will be called "elements" for simplicity, see Fig. 2.20. This corresponds to one element through the thickness and two elements along each period. The boundary conditions are accounted by using the least-square method and three bridging techniques will be compared to account for the interface conditions: full least-square described in section 2.4.1, Lagrange multipliers and Arlequin methods as introduced in [51].

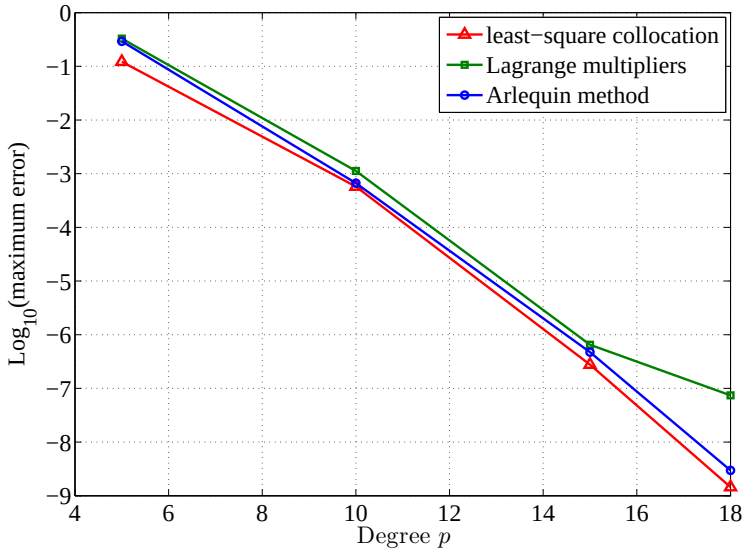


Figure 2.21: The p-convergence for problem Eq. (2.36). The size of the domain is $10 \times 10 \times 1$ and with 100 TMM-elements coupled by three different bridging techniques.

First one considers the domain $l_x \times l_y \times d = 10 \times 10 \times 1$ that is discretized by 100 TMM-elements. The convergence of the algorithm with the degree (p-convergence) is presented in Fig. 2.21. The three bridging techniques lead about to the same results. The error decreases exponentially with the degree up to a very high accuracy, even for this large scale problem. This establishes the robustness of all these procedures with respect to the number of sub-domains. Next these TMM-calculations have been compared the finite

element method. In this respect the code FreeFem++ has been chosen [77]. It is an open source finite element code to solve systems of Partial Differential Equations. Its efficiency relies especially on very rapid multi-frontal linear solvers like UMFPACK [78]. The domain has been discretized by tetrahedral elements with quadratic shape functions (the best 3D-element available in FreeFem++). Various discretizations have been tested and we present the results in Tab. 2.12, the best calculations providing a relative error less than 10^{-3}. In this case, the finite element mesh contains 417501 degrees of freedom, what is 20 times more than with the three TMM-procedures.

Table 2.12: The three versions of TMM are compared with the finite element code FreeFem++. Problem (2.36), $l_x \times l_y \times d = 10 \times 10 \times 1$.

	TMM			FreeFem++
	Least-square	Lagrange multiplier	Arlequin	
Degree	10	11	10	\
DOFs	12100	21600	19300	417501
log_{10}(Error)	-3.1917	-3.4995	-3.1416	-3.3425
Time (s)	5.08	13.34	14.77	10.31
log_{10}(Cond(K))	10.9	14.2	15.9	32.7

Table 2.13: One fixes the size and degree $p = 10$ of each TMM-element and increases the size of the domain. The sub-domains are connected by using least-square collocation.

Domain	Number of elements	DOFs	log_{10}(Error)	log_{10}(Cond(K))
$10 \times 10 \times 1$	100	12100	-3.24	10.7
$20 \times 20 \times 1$	400	48400	idem	idem
$30 \times 30 \times 1$	900	108900	idem	idem

Next we have studied a larger domain $l_x \times l_y \times d = 20 \times 20 \times 1$ that is split in 400 cubes. In this case the FreeFem++ mesh contains 1656441 degrees of freedom to get an error lower than 10^{-3} and the corresponding computation time is 46.44s. With the full least-square TMM and 400 sub-domains, the same accuracy was obtained with 48400 degrees of freedom (degree $p = 10$) and a computation time of only 25.67s. The performance of full least-square TMM with increase of number of TMM-elements has been collected in Table 2.13. The condition numbers of the final linear system remain nearly constant with the increase of the number of sub-domains. This shows the

ability of the present techniques to solve large scale problems with at least the same efficiency as a high-performance finite element code. Up to now, we did not find any limit regarding the number of sub-domains.

2.7 Conclusion

New numerical techniques were assessed in this paper to account for boundary conditions and least-square collocation was reconsidered to account for transmission conditions within Taylor Meshless Method. Because TMM solves quasi-exactly the partial differential equations by the technique of Taylor series, the treatment of boundary and interface conditions becomes a central question in view of a wide application of this method. Sometimes a single Taylor series is sufficient to solve the boundary value problem, but in other cases the domain of convergence of the series imposes a splitting in several sub-domains and therefore transmission conditions. The simplest method is the least-square collocation, previously validated for boundary conditions [47]. In a second technique validated in [51] for transmission conditions, Lagrange multipliers are introduced and discretized by radial functions. This Lagrange multiplier method has been first applied in this paper to account for boundary conditions. These two techniques are rather simple to be implemented and applied and few discretization parameters are needed: the degrees of the polynomials and a set of collocation points on boundary and interfaces; in the case of Lagrange multipliers, the radial functions are also to be defined from a cloud of points and a characteristic distance.

In most of the cases that were tested in this paper, least-square method is very efficient and exponential convergence with the degree has been obtained. Very few DOFs are needed as compared with classical discretization methods, the ratio being typically in the range $10 \sim 100$. It is also quite robust with respect to the parameters of the algorithm and its only drawback is to lead to full matrices when used with few sub-domains. As for Lagrange multiplier method, it converges about in the same way and is quite robust. Its main drawback is to require additional parameters to define radial functions.
In our last 2D elasticity test, we did not find an exponential convergence with the degree, what is very probably due to the fact that the solution is not a smooth function near a corner. The error in this case was much greater than in the previous cases, about 10^{-2} instead of $10^{-7}, 10^{-8}$. Thus the method needs improvements to better capture non-smooth solutions which will be discussed in the chapter 5. Further work is also necessary to compute Taylor series for more complex models like nonlinear elasticity [54].

Finally all these methods have been assessed in a 3D case, whose discretization requires hundreds of sub-domains and more than one million of degrees of freedom in an equivalent finite element simulation. Similar performances were found, whatever be the bridging technique (least-square or Lagrange multipliers). All these TMM-variants seem very efficient and robust according to all the benchmarks discussed in this paper.

Chapter 3

Taylor Meshless Method for large-scale problems

Abstract

A true integration-free meshless method based on Taylor series named Taylor Meshless Method (TMM) has been proposed to solve two dimensional Partial Differential Equations (PDE). In this framework, the shape functions are approximated solutions of the PDE and the discretization concerns only the boundary. In this chapter, the applicability of TMM to solve large-scale problems is discussed under two aspects. First, as in some other meshless methods, ill-conditioned matrices and round-off error propagation could lead to a loss of accuracy when the number of unknowns increases. This point will be investigated in the case of large-scale problems. Second the computation time and its distribution are analyzed from numerical experiments for PDE's in a 3D domain. It is established that the TMM method is efficient and robust, even in the case of large-scale problems while the finite element numerical model involves more than three millions degrees of freedom.

Present chapter corresponds to the published research paper (Yang et al., Solving large-scale problems by Taylor Meshless Method, Int. J. Numer. Methods Eng, 112(2), 103-124, 2017).

Keywords: Taylor series; PDE; large-scale problem; Round-off errors.

Contents

3.1 Introduction

This chapter analyzes the behavior of Taylor Meshless Method (TMM) in the case of large-scale problems, i.e. for Partial Differential Equation (PDE) whose discretization would require millions of degrees of freedom (DOFs) with a standard technique like Finite Element Method (FEM) or Finite Difference Method (FDM). TMM is based on a quasi-exact resolution of the PDE's by the technique of Taylor series so that the resulting matricial problem comes from the discretization of the boundary conditions [47]. Two main techniques have been proposed for boundary conditions: least-square collocation [47] and Lagrange multipliers [79]. The same methods can be also applied for the transmission conditions in piecewise resolutions [51, 79]. The numerical solution is obtained in the form of a single high order polynomial or by a family of polynomials, each one being valid in a sub-domain. An important property of TMM is the exponential convergence with the degree, which permits to obtain very accurate approximations.

There are two challenges in view of applications to large-scale problems: control the computation time, control the condition number of the matrix or at least the effect of ill-conditioning. For the first challenge, controlling computation time, two aspects need to be considered: the way of discretization, the number of DOFs. Two types of discretization techniques, Galerkin method and collocation method are mainly applied in meshless methods. The Galerkin-based meshless methods [8, 80, 81] always need background grid and integration procedures which lead to expensive computation costs, while the collocation-based methods [29, 48, 68, 82, 83], are truly meshless, sometimes very efficient, but difficult to apply for large-scale problems. Nevertheless TMM is quite different from previous meshless techniques due to the convergence properties of Taylor series and to an original analytical solving inside the domain. On the other hand, according to previous numerical experiments, TMM seems competitive in terms of computation time because it needs few DOFs: Tampango et al. [50] were able to solve a Laplace problem with less than 100 DOFs while about 5000 were necessary with quadratic interpolation (P2) finite elements and more than 40000 with linear interpolation (P1) triangles at the same level of accuracy. With account of this ratio 5000/100, a large-scale problem involving 500000 DOFs by finite elements corresponds about to 10000 DOFs within TMM. The numerical cost for this class of problems will be analyzed in this chapter for 3D Laplace and elasticity problems, in order to assess the performance of TMM as compared to other discretization techniques and to locate which steps of the algorithm are time consuming.

The effect of a too large condition number is a challenging problem, but it is well documented in the case of mesh-free methods based on a single packet of

shape functions, i.e. without splitting into sub-domains. It has been early recognized that the Method of Fundamental Solution converges better when the source points are away from the domain, which means that the method works very well with rather flat shape functions [84]. The same behaviour is observed with Radial Basis Functions, especially with multi-quadrics [29, 85–88] that can be made arbitrarily flat. Even though the use of flat shape functions may decrease the error of approximate solutions, the flatness of shape functions makes them indistinguishable which ultimately leads to the ill-conditioning of the matrices [87]. This phenomenon is known as "Schaback's uncertainty principle" [89]: "the error and the condition number can not both be kept small. Either one goes for a small error and gets a bad sensitivity, or one wants a stable algorithm and has to take a comparably large error." A comprehensive study of this phenomenon can be found in [88] by Cheng et al. A similar behaviour has been also observed with Taylor Meshless Method: for small degrees p, the method converges exponentially, generally up to a very high accuracy; then a plateau is reached and beyond this level, the accuracy is progressively affected [51, 79]. Many proposals can be found in the literature to control the effect of ill-conditioned matrices by using localized shape functions [90] or sub-domains techniques [63, 91, 92]. For instance according to recent numerical experiments [93], it appears that the sub-domain technique improves the conditioning, reduces the computation time and delays the loss of stability due to an ill-conditioned matrix, but can not overcome Schaback's uncertainty principle. Such a sensitivity to conditioning is rarely mentioned with other meshless methods based on more localized approximations [8, 57, 58, 94, 95]. In this chapter, the effect of matrix conditioning on the stability of TMM will be discussed carefully from numerical experiments in the case of large-scale problems.

The present study does not fully exhaust the question of applicability of TMM for practical engineering problems. The treatment of non-linear problems has been sketched in [52, 96, 97] by associating Newton-like algorithms with TMM-discretization and Automatic Differentiation [98]. Another question is posed by the performance of TMM for non-smooth solutions that are common in engineering computations because of corners, cracks, concentrated forces and so on. These two questions will be re-discussed in details in the next chapters. On the contrary, previous papers have established that the method works well for any boundary condition and any domain geometry with an easy implementation.

This chapter is organized as follows: section 3.2 states the algorithm for TMM and its piecewise resolutions. Our numerical tests will be presented in section 3.3. We analyze the influence of the number of collocation points in section 3.4.1 and the robustness of convergence within a single domain in

section 3.4.2 and piecewise resolutions in section 3.4.3. The effect of a large condition number of matrix will be discussed in section 3.4.4. The computation time for each step of the algorithm and its comparison with FEM are presented in section 3.5, especially in section 3.5.3 where one discusses a case needing three millions DOFs with FEM.

3.2 Algorithm for Taylor Meshless Method

In this section, the Taylor Meshless Method is adapted in a 3D framework. It is based on a resolution of the PDE's from Taylor series. One seeks the unknowns as rather high degree polynomials by vanishing the residual of the PDE's up to a given order. This leads to a general solution of the PDE's that depends on many parameters, but these unknowns are much less numerous than with other discretization techniques. Typically the general solution of 2D Laplace equation $\Delta u = 0$ is a linear combination of the polynomials $\mathbf{Re}(x+iy)^n$ and $\mathbf{Im}(x+iy)^n$ that are the shape functions of the method. Here we extend in the 3D case the method of [47] that brings together all the shape functions having a same degree. Next the boundary value problem is solved by least-square method as proposed in [47]. If necessary, a piecewise resolution can be done either by introducing Lagrange multipliers [51] or also by least-square method [79]. All these techniques will be recalled shortly for completeness.

3.2.1 Algorithm to compute the shape functions

In this section, we limit ourselves to the case of a linear equation with constant coefficients and a general right hand side $f(\boldsymbol{x})$:

$$-\Delta u(\boldsymbol{x}) + u(\boldsymbol{x}) = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega \tag{3.1}$$

where Ω is a bounded domain in $\mathbb{R}^3$. The unknown function $u(\boldsymbol{x})$ is expanded in Taylor series with a development point $c = [0,0,0]$:

$$u(x,y,z) = \sum_{k=0}^{p} \sum_{i=0}^{k} \sum_{j=0}^{k-i} \widetilde{u}_{j,i,k-i-j} x^j y^i z^{k-i-j} \tag{3.2}$$

The idea is to build the affine subspace of the polynomials that solves the PDE (3.1) in the sense of Taylor series. For convenience, each polynomial is split into homogeneous polynomials. A homogeneous polynomial of degree k combines $(k+2)(k+1)/2$ monomials and it is written in a matricial form

$\langle \boldsymbol{x}^k \rangle \{\widetilde{\boldsymbol{u}}^k\}$, where

$$\{\widetilde{\boldsymbol{u}}^k\} = \begin{Bmatrix} \{\widetilde{\boldsymbol{u}}_0^k\} \\ \{\widetilde{\boldsymbol{u}}_1^k\} \\ \{\widetilde{\boldsymbol{u}}_2^k\} \\ \vdots \\ \{\widetilde{\boldsymbol{u}}_k^k\} \end{Bmatrix}, \quad \{\widetilde{\boldsymbol{u}}_i^k\} = \begin{Bmatrix} \widetilde{u}_{k-i,i,0} \\ \widetilde{u}_{k-i-1,i,1} \\ \widetilde{u}_{k-i-2,i,2} \\ \vdots \\ \widetilde{u}_{0,i,k-i} \end{Bmatrix}.$$

$$\langle \boldsymbol{x}^k \rangle = \left[\langle \boldsymbol{x}_0^k \rangle, \langle \boldsymbol{x}_1^k \rangle, \langle \boldsymbol{x}_2^k \rangle, \cdots, \langle \boldsymbol{x}_k^k \rangle \right]$$

$$\langle \boldsymbol{x}_i^k \rangle = \langle x^{k-i} y^i, x^{k-i-1} y^i z^1, x^{k-i-2} y^i z^2, \cdots, y^i z^{k-i} \rangle$$

In this way, all the polynomials whose degree is lower or equal to p can be written in the following form:

$$u(x,y,z) = \sum_{k=0}^{p} \langle \boldsymbol{x}^k \rangle \{\widetilde{\boldsymbol{u}}^k\} = \sum_{k=0}^{p} u_k(x,y,z) \tag{3.3}$$

The basis of monomials within a homogeneous polynomial of degree k and the corresponding coefficients are organized as follows:

$$\begin{array}{cccccc}
\widetilde{u}_{k,0,0} & & & & & \\
\widetilde{u}_{k-1,1,0} & \widetilde{u}_{k-1,0,1} & & & & \\
\widetilde{u}_{k-2,2,0} & \widetilde{u}_{k-2,1,1} & \widetilde{u}_{k-2,0,2} & & & \\
\widetilde{u}_{k-3,3,0} & \widetilde{u}_{k-3,2,1} & \widetilde{u}_{k-3,1,2} & \widetilde{u}_{k-3,0,3} & & \\
\vdots & \ddots & \ddots & \ddots & \ddots & \\
\widetilde{u}_{0,k,0} & \cdots & \widetilde{u}_{0,3,k-3} & \widetilde{u}_{0,2,k-2} & \widetilde{u}_{0,1,k-1} & \widetilde{u}_{0,0,k}
\end{array}$$

$$\begin{array}{cccccc}
x^k & & & & & \\
x^{k-1}y^1 & x^{k-1}z^1 & & & & \\
x^{k-2}y^2 & x^{k-2}y^1z^1 & x^{k-2}z^2 & & & \\
x^{k-3}y^3 & x^{k-3}y^2z^1 & x^{k-3}y^1z^2 & x^{k-3}z^3 & & \\
\vdots & \ddots & \ddots & \ddots & \ddots & \\
y^k & \cdots & y^3z^{k-3} & y^2z^{k-2} & y^1z^{k-1} & z^k
\end{array}$$

The red color terms represent the terms $\{\boldsymbol{x}_0^k\}$ and $\{\widetilde{\boldsymbol{u}}_0^k\}$ that do not contain y ($i = 0$). The green color items represent the terms $\{\boldsymbol{x}_1^k\}$ and $\{\widetilde{\boldsymbol{u}}_1^k\}$ that contain y^1 ($i = 1$). If the PDE (3.1) is considered as a differential equation with respect to y, its solution can be expressed in terms of the "initial values" $u(x,0,z)$ and $\frac{\partial u}{\partial y}(x,0,z)$. That is why we have distinguished the terms $i = 0$ and $i = 1$. The corresponding coefficients of these terms $i = 0$ and $i = 1$ are chosen as independent variables and collected in the following vector:

$$\{\widetilde{\boldsymbol{v}}^k\} = \begin{Bmatrix} \{\widetilde{\boldsymbol{u}}_0^k\} \\ \{\widetilde{\boldsymbol{u}}_1^k\} \end{Bmatrix} \in \mathbb{R}^{2k+1} \tag{3.4}$$

The other coefficients can be expressed in terms of the independent ones. They are collected in the following vector:

$$\{\widetilde{\boldsymbol{w}}^k\} = \left\{ \begin{array}{c} \{\widetilde{\boldsymbol{u}}_2^k\} \\ \{\widetilde{\boldsymbol{u}}_3^k\} \\ \vdots \\ \{\widetilde{\boldsymbol{u}}_k^k\} \end{array} \right\} \in \mathbb{R}^{k(k-1)/2} \tag{3.5}$$

Two matrices $[\boldsymbol{S}^{vk}]$ and $[\boldsymbol{S}^{wk}]$ can be constructed (see Appendix A) to split the coefficients vector $\{\widetilde{\boldsymbol{u}}^k\}$ into two parts:

$$\{\widetilde{\boldsymbol{u}}^k\} = [\boldsymbol{S}^{vk}]\{\widetilde{\boldsymbol{v}}^k\} + [\boldsymbol{S}^{wk}]\{\widetilde{\boldsymbol{w}}^k\} \tag{3.6}$$

Next, if $u(x, y, z)$ is a homogeneous polynomial of degree $k + 2$, Δu is a homogeneous polynomial of degree k. Hence, there exists a matrix $[\boldsymbol{L}^k]$ such that

$$\Delta u_{k+2}(\boldsymbol{x}) = \langle \boldsymbol{x}^k \rangle [\boldsymbol{L}^k] \{\widetilde{\boldsymbol{u}}^{k+2}\} \tag{3.7}$$

The matrix $[\boldsymbol{L}^k]$ represents the action of the Laplacian on the coefficients of homogeneous polynomials. It is defined in Appendix A Eq. (1). The principle is to vanish the residual of $\Delta u - u + f$, up to the degree $p - 2$. With account of Eq. (3.7), this leads to the following equation:

$$\left\langle \boldsymbol{x}^k \right\rangle \left([\boldsymbol{L}^k]\{\widetilde{\boldsymbol{u}}^{k+2}\} - \{\widetilde{\boldsymbol{u}}^k\} + \{\widetilde{\boldsymbol{f}}^k\} \right) = 0, \quad k \in [0, p-2] \tag{3.8}$$

By inserting Eq. (3.6) into Eq. (3.8), we get

$$\begin{aligned} [\boldsymbol{L}^k][\boldsymbol{S}^{vk+2}]\{\widetilde{\boldsymbol{v}}^{k+2}\} + [\boldsymbol{L}^k][\boldsymbol{S}^{wk+2}]\{\widetilde{\boldsymbol{w}}^{k+2}\} \\ = [\boldsymbol{S}^{vk}]\{\widetilde{\boldsymbol{v}}^k\} + [\boldsymbol{S}^{wk}]\{\widetilde{\boldsymbol{w}}^k\} - \{\widetilde{\boldsymbol{f}}^k\} \end{aligned} \tag{3.9}$$

One checks easily that the square matrix $[\boldsymbol{L}^k][\boldsymbol{S}^{wk+2}]$ given in the Appendix A, is invertible. Assuming that the vector $\{\widetilde{\boldsymbol{v}}^k\}(k \in [0, p])$ is known, the Eq. (3.9) becomes the relation of recurrence between $\{\widetilde{\boldsymbol{w}}^{k+2}\}$ and $\{\widetilde{\boldsymbol{w}}^k\}$. Then one can get the expression of $\{\widetilde{\boldsymbol{w}}^k\}(k \in [0, p])$ as a function of $\{\widetilde{\boldsymbol{v}}^k\}(k \in [0, p])$. Finally, one can get the approximated general solution of the Laplace equation as follows:

$$\begin{aligned} u(\boldsymbol{x}) &= P_0(\boldsymbol{x}) + \sum_{i=1}^{(p+1)^2} P_i(\boldsymbol{x}) v_i \\ &= P_0(\boldsymbol{x}) + \langle \boldsymbol{P}(\boldsymbol{x}) \rangle \{\widetilde{\boldsymbol{v}}\} \end{aligned} \tag{3.10}$$

where the $P_0(x, y, z)$ is used to balance the right-hand side of Eq. (3.1) and the unknown vector $\{\widetilde{\boldsymbol{v}}\}$ is defined by

$$\{\widetilde{\boldsymbol{v}}\} = \begin{Bmatrix} \{\widetilde{\boldsymbol{v}}^0\} \\ \{\widetilde{\boldsymbol{v}}^1\} \\ \vdots \\ \{\widetilde{\boldsymbol{v}}^p\} \end{Bmatrix} \in \mathbb{R}^{(p+1)^2} \tag{3.11}$$

3.2.2 Boundary least-square collocation

Next one has to apply the boundary conditions. It is well established that a pure collocation does not converge. As proposed in [47], the boundary conditions will be accounted by a least-square collocation procedure that was previously used in other meshless context, for instance by Fairweather and Karageorghis [48] and by Zhang et al. [68]. One chooses a set of nodes $\boldsymbol{x}_j$, $j \in [1, M]$, on the boundary of the 3D domain and one minimizes the error between the approximate value and the given value of u in these points. Here we consider a mixed boundary conditions as follows:

$$\begin{cases} u(\boldsymbol{x}) = u^d, & \boldsymbol{x} \in \Gamma_d; \\ \mathbf{T}u(\boldsymbol{x}) = t^n, & \boldsymbol{x} \in \Gamma_n. \end{cases} \tag{3.12}$$

It comes to minimize the following function with respect to the components of the vector $\{\boldsymbol{v}\}$:

$$\mathscr{T}(\mathbf{v}) = \frac{1}{2} \sum_{\boldsymbol{x}_i \in \Gamma_d} \left| u_p(\boldsymbol{x}_i) - u^d(\boldsymbol{x}_i) \right|^2 + w \cdot \frac{1}{2} \sum_{\boldsymbol{x}_j \in \Gamma_n} |\mathbf{T}u_p(\boldsymbol{x}_j) - t^n(\boldsymbol{x}_j)|^2 \tag{3.13}$$

where Γ_d represents the Dirichlet boundary condition, and Γ_n represents the Neumann boundary. This minimization leads to a linear system $[\boldsymbol{K}]\{\boldsymbol{v}\} = \{\boldsymbol{b}\}$ where $[\boldsymbol{K}]$ is an invertible matrix. Solving this system gives the vector $\{\boldsymbol{v}\}$ and therefore the approximate solution of the boundary value problem Eq. (3.1) with boundary Eq. (3.12). Note the simple form of the matrix ($\alpha, \beta \in [1, (p+1)^2]$):

$$[\boldsymbol{K}]_{\alpha\beta} = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i) \cdot P_\beta(\boldsymbol{x}_i) + w \cdot \sum_{j=1}^{M_n} \mathbf{T}P_\alpha(\boldsymbol{x}_j) \cdot \mathbf{T}P_\beta(\boldsymbol{x}_j), \tag{3.14}$$

$$\{\boldsymbol{b}\}_\alpha = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i) \left[u^d(\boldsymbol{x}_i) - P_0(\boldsymbol{x}_i) \right] + w \cdot \sum_{j=1}^{M_n} \mathbf{T}P_\alpha(\boldsymbol{x}_j) \left[t^n(\boldsymbol{x}_j) - \mathbf{T}P_0(\boldsymbol{x}_j) \right]. \tag{3.15}$$

3.2.3 Piecewise resolution

It is not possible to solve any boundary value problem with a single series. Thus, numerical methods have to be introduced to connect several high-order polynomial approximations. Within this aim, several techniques have been presented and well investigated in the previous works for two dimensional problems: discrete Lagrange multipliers or discretized continuous Lagrange multipliers [51] and least-square collocation method [79]. The aim of this section is to briefly recall the main procedures of these bridging techniques in 3D piecewise resolutions.

Here we consider a domain as shown in Fig. 3.1 which has been divided into two parts Ω_1 and Ω_2. For each sub-domain, an independent high-order polynomial approximation based on Taylor series, $u^1(\boldsymbol{x})$ and $u^2(\boldsymbol{x})$, will be obtained as stated in section 3.2.1. The functions accounting for boundary conditions of each sub-domain, $\mathscr{T}_1(v^1)$ and $\mathscr{T}_2(v^2)$, are as in Eq. (3.13).

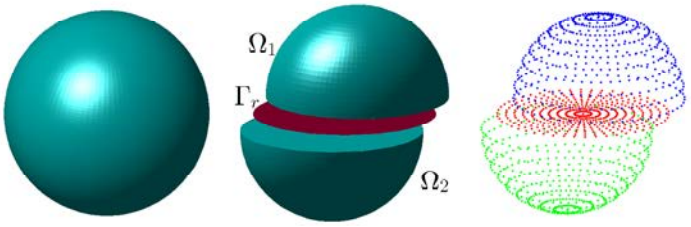


Figure 3.1: Sketch for two sub-domains.

In order to connect the two approximations, a set of M_r collocation points $\boldsymbol{x}_j$ on the interface Γ_r are needed for the discretization of the transmission conditions. The constraints on the interface are as the following:

$$\begin{cases} \psi_j^1(v^1, v^2) = u^1(\boldsymbol{x}_j) - u^2(\boldsymbol{x}_j) = 0 \\ \psi_j^2(v^1, v^2) = \frac{\partial u^1}{\partial n}(\boldsymbol{x}_j) - \frac{\partial u^2}{\partial n}(\boldsymbol{x}_j) = 0 \end{cases} \quad \forall \boldsymbol{x}_j \in \Gamma_r \tag{3.16}$$

In the following, three methods for the transmission conditions are illustrated respectively.

3.2.3.1 Least-square collocation method

By applying the least-square collocation method, the transmission conditions (3.16) will be satisfied in a mean sense via the following coupling function:

$$\mathscr{C}_{ls}(v^1, v^2) = \frac{1}{2} \sum_{j=1}^{M_r} \left[\left| u^1(\boldsymbol{x}_j) - u^2(\boldsymbol{x}_j) \right|^2 + \left| \frac{\partial u^1}{\partial n}(\boldsymbol{x}_j) - \frac{\partial u^2}{\partial n}(\boldsymbol{x}_j) \right|^2 \right] \tag{3.17}$$

Then for the whole problem, searching the variables $\{v^1\}$ and $\{v^2\}$ is equivalent to minimizing the following function:

$$\mathscr{T}(v^1, v^2) = \mathscr{T}_1(v^1) + \mathscr{T}_2(v^2) + \mathscr{C}_{ls}(v^1, v^2) \tag{3.18}$$

The minimization of this function $\mathscr{T}(v^1, v^2)$ leads to a linear system. Solving this system gives the vectors $\{v^1\}$ and $\{v^2\}$ and then the approximate solution for each sub-domain.

3.2.3.2 Discrete Lagrange multipliers

Accounting for the discrete transmission conditions (3.16) by Lagrange multipliers, the constraints will be satisfied in a strong form by using two Lagrange multipliers on each collocation point on the interface via the following coupling function:

$$\mathscr{C}_{lm}(v^1, v^2, \lambda^1, \lambda^2) = \sum_{j=1}^{M_r} [\lambda_j^1 \psi_j^1(v^1, v^2) + \lambda_j^2 \psi_j^2(v^1, v^2)] \tag{3.19}$$

Then for the whole problem, searching the variables $\{v^1\}$ and $\{v^2\}$ is equivalent to setting to zero the gradient of the following function:

$$\mathscr{T}(v^1, v^2, \lambda^1, \lambda^2) = \mathscr{T}_1(v^1) + \mathscr{T}_2(v^2) + \mathscr{C}_{lm}(v^1, v^2, \lambda^1, \lambda^2) \tag{3.20}$$

3.2.3.3 Continuous Lagrange multiplier and its discretization

Another way is to define the Lagrange multiplier at the continuous level. A bilinear form is introduced to account for the transmission conditions:

$$\mathscr{C}_{ar}(\lambda, u) = \sum_{i=1}^{M_r} (\lambda(\boldsymbol{x}_i)\mu(\boldsymbol{x}_i) + l^2 \nabla\lambda(\boldsymbol{x}_i)\nabla\mu(\boldsymbol{x}_i)) \tag{3.21}$$

where l is a characteristic length and $\lambda(\boldsymbol{x})$ is the continuous version of the Lagrange multiplier. In Eq. (3.21) a set of M_r collocation points $\boldsymbol{x}_i$ on the interface has always to be chosen. This bilinear form looks like the bilinear form of Arlequin method [73, 99], but for consistency with the meshless framework, it is defined from points and not from an integral. Numerical experiments in [51] have shown that a distribution of collocation points in a volume is not necessary. So the points are located along the interface between the sub-domains. Next the continuous Lagrange multiplier $\lambda(\boldsymbol{x})$ has to be discretized. As in [51], this discretization is done from two families of radial

functions depending of another set of N_λ discretization nodes $\boldsymbol{x}_j$ (that are the centers of the radial functions):

$$\lambda(\boldsymbol{x}) = \sum_{j=1}^{N_\lambda} (\lambda_j^1 \Phi_j^1(\boldsymbol{x}) + \lambda_j^2 \Phi_j^2(\boldsymbol{x})) \tag{3.22}$$

$$\Phi_j^1(\boldsymbol{x}) = exp(\frac{-\|\boldsymbol{x}-\boldsymbol{x}_j\|^2}{d^2}) \quad and \quad \Phi_j^2(\boldsymbol{x}) = (\boldsymbol{x}-\boldsymbol{x}_j)\cdot \mathbf{n}\, exp(\frac{-\|\boldsymbol{x}-\boldsymbol{x}_j\|^2}{d^2}) \tag{3.23}$$

where d represents the radius of influence of the radial functions and $\mathbf{n}$ is the unit normal to the interface. Then solving the whole problem in the domain is equivalent to taking the stationary values of the following function:

$$\mathscr{T}(v^1, v^2, \Lambda^1, \Lambda^2) = \mathscr{T}_1(v^1) + \mathscr{T}_2(v^2) + \mathscr{C}_{ar}(\Lambda^1, \Lambda^2, u^1 - u^2) \tag{3.24}$$

As compared with the previous discrete Lagrange multipliers method, it permits to use less degrees of freedom ($N_\lambda < M_r$).

3.3 Numerical examples

This paper focuses on the behavior of Taylor Meshless Method for large-scale problems. That is why numerical tests in 3D domains are considered and discussed carefully, what was not done before. Only cases having an exact solution are studied in this chapter, what permits to evaluate the error even it is very small. Throughout this paper, the relative error is the difference between exact and approximate solutions, divided by the maximum value of the solution:

$$\mathscr{E} = max \frac{|u(x) - u_{ex}(x)|}{max|u(x)|}. \tag{3.25}$$

Three boundary value problems will be tested in this paper, two concerning the Laplace equation and the last one the equations of linear isotropic elasticity. In the first case, the exact solution is a polynomial of small degree. This permits to check the consistency of the method: the exact solution lying in the subspace of our shape functions, is it easily recovered within TMM ? In the next two tests, the exact solution is a fundamental solution. According to the location of the singularity, this function can be more or less flat or strongly varying. In the last test, the solution contains many waves so that the discretization involves a very large number of DOFs.

3.3.1 Laplace equation with polynomial solution

The first example is the Laplace equation in a spherical domain ($\Omega = \{x, y, z | x^2 + y^2 + z^2 \leq 1\}$)

$$\begin{cases} -\Delta u = 0 & \text{in} \quad \Omega \\ u(x, y, z) = x^3yz + xy^2z & \text{on} \quad \partial\Omega \end{cases} \tag{3.26}$$

The exact solution is a polynomial of degree 5:

$$u_{ex} = x^3yz + xy^2z + (\frac{1}{3}xyz + \frac{1}{7}xz)(1 - x^2 - y^2 - z^2) \tag{3.27}$$

3.3.2 Laplace equation with singular solution

The second example is always a Laplace equation with alternative Dirichlet boundary conditions in the unit sphere ($\Omega = \{x, y, z | x^2 + y^2 + z^2 \leq 1\}$)

$$\begin{cases} -\Delta u = 0 & \text{in} \quad \Omega \\ u(x, y, z) = \dfrac{1}{r} & \text{on} \quad \partial\Omega \end{cases} \tag{3.28}$$

where $r^2 = (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2$. The exact solution is

$$u_{ex} = \frac{1}{\sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}} \tag{3.29}$$

The solution has a singularity at $X_0 = [x_0, y_0, z_0]$ that must lie outside the domain. If this singularity is far away from the domain, the solution is rather flat. If it is closer, the solution involves large gradients, what shall require finer discretizations.

3.3.3 3D elasticity

The previous examples concern a single elliptic equation. In view of practical applications, it is important to discuss the efficiency of the method for systems of PDEs. Linear isotropic 3D elasticity in the absence of body forces is considered in the unit sphere. The unknown fields are the components u_1, u_2 and u_3 of the displacements. In Cartesian coordinates, the model is based on the Cauchy-Navier equations:

$$\left(\frac{2-2\nu}{1-2\nu}\right)\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_1}{\partial x_3^2} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_2}{\partial x_1 \partial x_2} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_3}{\partial x_1 \partial x_3} = 0 \tag{3.30}$$

$$\left(\frac{2-2\nu}{1-2\nu}\right)\frac{\partial^2 u_2}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_3^2} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_1}{\partial x_1 \partial x_2} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_3}{\partial x_2 \partial x_3} = 0 \tag{3.31}$$

$$\left(\frac{2-2\nu}{1-2\nu}\right)\frac{\partial^2 u_3}{\partial x_3^2} + \frac{\partial^2 u_3}{\partial x_1^2} + \frac{\partial^2 u_3}{\partial x_2^2} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_1}{\partial x_1 \partial x_3} + \left(\frac{1}{1-2\nu}\right)\frac{\partial^2 u_2}{\partial x_2 \partial x_3} = 0 \tag{3.32}$$

The studied benchmark is the Dirichlet problem, where the exact solution is a fundamental solution. For a source located at a point Q acting at a point P, the fundamental solutions of the system (3.30, 3.31, 3.32) are (see, e.g., Refs. [100]):

$$G_{11}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(3-4\nu)r_{PQ}^2 + (x_{1P}-x_{1Q})^2}{r_{PQ}^3}\right] \tag{3.33}$$

$$G_{12}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(x_{1P}-x_{1Q})(x_{2P}-x_{2Q})}{r_{PQ}^3}\right] = G_{21}(P,Q) \tag{3.34}$$

$$G_{13}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(x_{1P}-x_{1Q})(x_{3P}-x_{3Q})}{r_{PQ}^3}\right] = G_{31}(P,Q) \tag{3.35}$$

$$G_{22}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(3-4\nu)r_{PQ}^2 + (x_{2P}-x_{2Q})^2}{r_{PQ}^3}\right] \tag{3.36}$$

$$G_{23}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(x_{2P}-x_{2Q})(x_{3P}-x_{3Q})}{r_{PQ}^3}\right] = G_{32}(P,Q) \tag{3.37}$$

$$G_{33}(P,Q) = \frac{1}{16\pi\mu(1-\nu)}\left[\frac{(3-4\nu)r_{PQ}^2 + (x_{3P}-x_{3Q})^2}{r_{PQ}^3}\right] \tag{3.38}$$

where

$$r_{PQ} = \sqrt{(x_{1P}-x_{1Q})^2 + (x_{2P}-x_{2Q})^2 + (x_{3P}-x_{3Q})^2} \tag{3.39}$$

Poisson ratio and shear modulus are denoted by ν and μ respectively.

The displacement values u_1, u_2 and u_3 in three coordinate directions of point P are approximated by linear combinations of fundamental solutions:

$$\begin{cases} u_1(a,b,c,Q;P) &= aG_{11}(P,Q) + bG_{12}(P,Q) + cG_{13}(P,Q) \\ u_2(a,b,c,Q;P) &= aG_{21}(P,Q) + bG_{22}(P,Q) + cG_{23}(P,Q) \\ u_3(a,b,c,Q;P) &= aG_{31}(P,Q) + bG_{32}(P,Q) + cG_{33}(P,Q) \end{cases} \tag{3.40}$$

where the coefficients a, b and c denote three coordinate components of the point load acting at the source point Q. In the following, we choose values of elastic constants appropriate for steel namely as recommended in [100], $\nu = 0.3$ and $\mu = 1.15$ (10^5 MPa). The coordinate components of the point load acting at the source point Q is set to be $a = b = c = 1$ (10^5 N) and different locations of the source points are tested in the following.

3.3.4 A very large-scale test

This benchmark is issued from the study of instabilities of film-substrate systems [75, 76], where one has to describe many spatial oscillations. Here one accounts only for the substrate and the film is represented by a prescribed sinusoidal displacement. The domain is a cuboid with sides $l_x \times l_y \times d$ ($d = 1$

in any case), see Fig. 3.2. For simplicity we consider only the Laplace problem $\Delta u = 0$ with Dirichlet boundary conditions ($u(x, y, d) = sin(\pi x)sin(\pi y)$ on the top face, $u(x, y, 0) = 0$ on the bottom), the exact solution being $u(x, y, z) = sin(\pi x)sin(\pi y)sinh(\sqrt{2}\pi z)/sinh(\sqrt{2}\pi d)$. The domain is split in sub-domains that are rectangular hexahedra of size $1 \times 1 \times 1$ that will be called "elements" for simplicity, see Fig. 3.3. This corresponds to one element through the thickness and two elements along each period. The connection between sub-domains is achieved by the least-square method and a rather small degree ($p = 10$) is used.

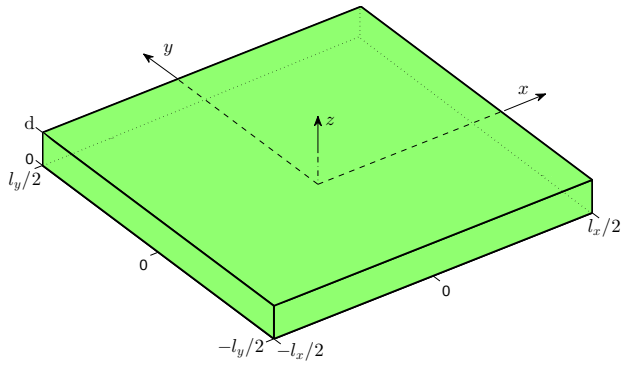


Figure 3.2: Sketch for the shape of domain, problem of section 3.3.4.

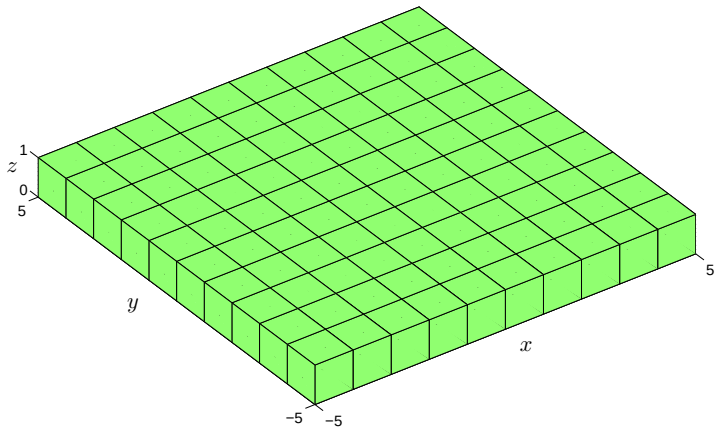


Figure 3.3: Sketch for the plate divided into 100 sub-domains, problem of section 3.3.4.

3.4 Convergence and conditioning

In previous papers [47, 49–51] and in chapters 1 and 2, many results were presented, especially the influence of the number of collocation points, the convergence with the degree, the splitting into sub-domains, the account of various boundary conditions, of various domain shapes and globally the robustness of the procedure. For consistency of the paper, some points will be re-discussed here, but not the last one's: we limit ourselves to Dirichlet boundary conditions and to very simple geometries, the sought conclusion being the ability of the method to deal with large systems. In addition to the issues mentioned above, there exists another key point: the influence of round-off errors and matrix conditioning. Indeed looking for very accurate solutions leads to ill-conditioned matrix and finally to loss of accuracy [88]. Thus it is an important issue to check whether these problems prevent the use of TMM to solve large-scale problems.

3.4.1 Influence of the number of collocation points

The influence of the number of collocation points has been illustrated in Fig. 3.4, Fig. 3.5 and Fig. 3.6 for the polynomial solution of the Laplace problem Eq. (3.26), the singular solution of the Laplace problem Eq. (3.28) and for the elasticity test Eq. (3.30, 3.31, 3.32) respectively, and for three values of the degree $p = 10$, 20 and 30. Let us recall that the number of unknowns is $(p+1)^2$ for a single equation and $3(p+1)^2$ in 3D elasticity. In the three cases, the method diverges for the pure collocation or when the number of collocation points is only slightly larger than the number of unknowns. Next a maximal accuracy is obtained for about $M = 2(p+1)^2$ and it remains constant beyond this threshold. A similar behavior has been previously observed in 2D cases [47]. So the method is remarkably stable with respect to the number of collocation points. In what follows, we choose a number of collocation points $M = 2(p+2)^2$ that is slightly larger than two times of the number of unknowns $(p+1)^2$ necessary for a single equation, what turns out to be sufficient in the considered applications.

Of course the obtained maximal accuracy depends on the problem and of the degree of the shape functions. In the case of a polynomial solution of degree 5, see Fig. 3.4, this accuracy is very large, up to 10^{-15}, what is very close to the unit round-off error $2^{-53} \approx 1.11e^{-16}$ for a single real number within the double-precision floating-point format [101]. Because the exact solution lies in the subspace generated by the shape functions, this very small error is only due to the propagation of these round-off errors. Hence it is natural that the final error increases with the degree because this induces a larger number

of elementary operations. In the two other cases, see Fig. 3.5 and Fig. 3.6, the exact solution is not a polynomial. The maximal accuracy increases with the degree, from about 10^{-3} for $p = 10$ to 10^{-8} for $p = 30$, what corresponds to the exponential convergence of any Taylor series. In those cases, the influence of the round-off errors is too small to be observed in the range $p \in [10, 30]$. This balance between round-off errors and Taylor series errors is the heart of the present discussion.

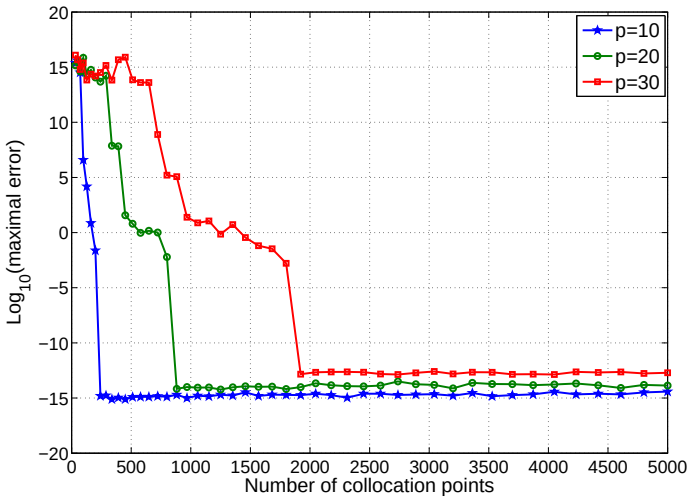


Figure 3.4: Influence of number of collocation points M for degrees $p = 10$, 20, 30 to recover the polynomial solution of problem Eq. (3.26).

3.4.2 Exponential convergence

The convergence with the degree is summarized in Fig. 3.7 for the singular solution of Laplace equation (3.28). Two cases are presented, a first one where the singularity is away from the domain, $X_0 = [2, 2, 2]$, and therefore the solution is rather flat, a second one where the singularity $X_0 = [1, 1, 1]$ is closer and the gradient is larger. The rate of convergence is better if the singularity is far away from the domain, i.e. the convergence is faster for the flat solution, but in the two cases the convergence is exponential for a small degree. A second stage can be seen from a degree p=25, from which the accuracy is no longer improved but it deteriorates slightly. It seems that the convergence is significantly affected by the propagation of round-off errors for large values of the degree, while before this level the behavior is dominated by the convergence of the Taylor series.

This scheme is not new. Within TMM, it had been observed in [50] Figure

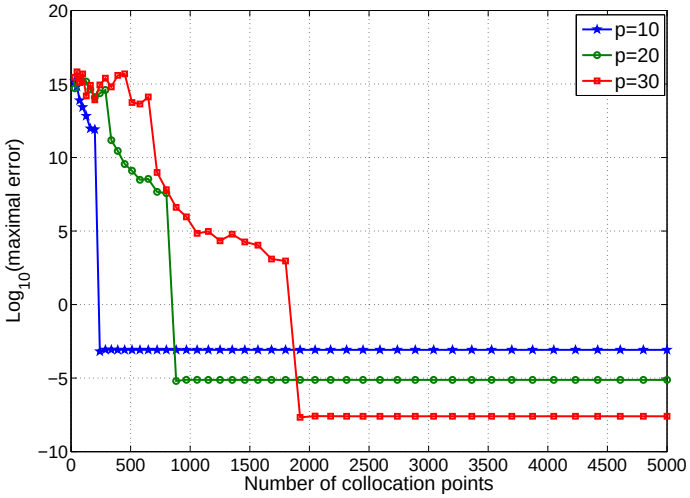


Figure 3.5: Influence of number of collocation points M for degrees $p = 10$, 20, 30 to recover the singular solution of the Laplace problem Eq. (3.28) according to the source point $X_0 = [1, 1, 1]$.

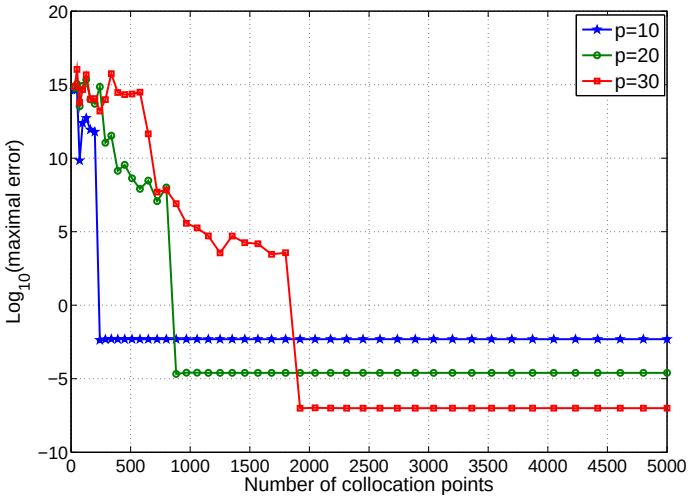


Figure 3.6: Influence of number of collocation points M for degrees $p = 10$, 20, 30 to recover the singular solution of the elasticity problem Eq. (3.30, 3.31, 3.32) according to the source point $X_0 = [1, 1, 1]$.

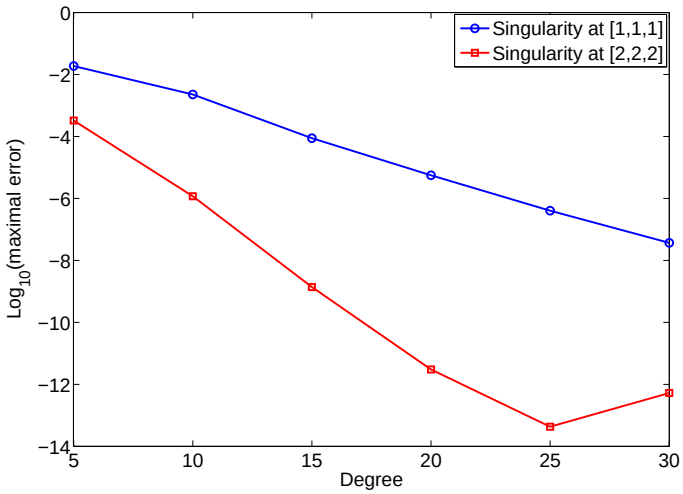


Figure 3.7: Exponential convergence with the degree for Laplace problems Eq. (3.28). Two cases are considered according to the source point X_0.

3, or in [51] Figures 10 and 14, even in cases of piecewise resolutions. It is also similar to "Schaback's uncertainty principle" discussed in [89] for meshless method based on radial basis functions. In the latter paper, this influence of round-off errors becomes also significant beyond some level of accuracy and this had been attributed to the ill-conditioning of the matrix. This point will be discussed below.

Next the elasticity problem with the nearest source point $X_0 = [1, 1, 1]$ is discussed and some features are reported in Table 3.1. Once again, a reversal of the accuracy curve is found for a degree $p = 40$, which corresponds to a matrix condition number of about 10^{28}. Nevertheless the accuracy remains very high after the reversal point ($p = 50$, 60) as well as before ($p = 30$), with an error always lower than $10^{-6.6}$. This illustrates the robustness and the high precision of the method, whether before or after the accuracy reversal.

Table 3.1: Computational features for the 3D elastic problem Eq. (3.30-3.32) with $X_0 = [1, 1, 1]$, according to the degree.

Taylor Meshless Method	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$	$p = 60$
Number of degrees of freedom	363	1323	2883	5043	7803	11163
Total CPU time (s)	0.04	0.39	2.74	14.26	51.97	144.25
Parallel calculation 4 cores (s)	0.37	0.42	2.73	12.11	40.78	108.86
Relative error (Log_{10})	-2.2747	-4.4167	-6.6648	-9.0538	-8.5838	-7.2927
Condition number of $\boldsymbol{K}$ (Log_{10})	8.48	15.69	22.46	27.76	33.56	37.54

3.4.3 Piecewise resolutions

Piecewise resolutions are necessary because Taylor series often diverge [51]. So it is important that the method keeps the same properties of convergence and of robustness in cases of piecewise resolutions. The Laplace problem Eq. (3.28) with singular point $X_0 = [1, 1, 1]$ is solved by the three techniques mentioned in section 3.2.3. The one-domain method works in this case, see Fig. 3.7, and the test will permit to check the influence of the matching technique on the features of the discretization. The sphere is split in two sub-domains by the xy-plane. In the following, we choose $M_i = 2(p+1)^2$ collocation points on the boundary of each sub-domain and $M_r = 2p^2 + 1$ collocation points on the interface. For piecewise resolutions coupled by Lagrange multipliers, the number of Lagrange multipliers is two times the number of the collocation points on the interface ($N_\lambda = 4p^2 + 2$). For the weak sense coupling technique (Arlequin type, section 3.2.3.3), there are $N_\lambda = 4(p/2 - 1)^2 + 2$ Lagrange multipliers on the interface and the radius of influence of the radial functions is chosen to be $d = 0.1$.

The three matching techniques work well, with about the same characteristics as the one-domain method, see Table 3.2, 3.3, 3.4: first there is always a turning point in the accuracy curve for $p \approx 30$, what corresponds to an accuracy in the range $[10^{-7}, 10^{-6}]$; second the condition number of the matrix $[\boldsymbol{K}]$ has about the same value 10^{27} at this accuracy reversal so that the equilibrium point between round-off errors and Taylor-series errors seems very strongly related to the matrix conditioning. They are minor differences between the three matching techniques but they are not important by comparison with other criteria, as the simplicity of least-square and Lagrange multiplier techniques, the lower number of dof's with least-square method or the sparsity of the matrix with the weak sense coupling technique. Anyway the three matching methods are available and none degrades the convergence properties of the one-domain TMM.

Table 3.2: Computational features for the singular solution of the Laplace problem Eq. (3.28) with $X_0 = [1, 1, 1]$, according to the degree. Two sub-domains connected by least-square collocation method.

Taylor Meshless Method	p=10	p=20	p=30	p=40	p=50	p=60
Number of DOFs	242	882	1922	3362	5202	7442
Total CPU time (s)	0.22	0.49	2.53	10.10	32.44	87.43
Relative error (log_{10})	-2.8691	-5.6706	-7.1521	-7.5321	-6.8928	-7.1045
Cond($\boldsymbol{K}$) (log_{10})	10.76	21.40	28.22	31.30	35.21	38.81

Table 3.3: Computational features for the singular solution of the Laplace problem Eq. (3.28) with $X_0 = [1, 1, 1]$, according to the degree. Two subdomains connected by Lagrange multipliers method.

Taylor Meshless Method	p=10	p=20	p=30	p=40	p=50	p=60
Number of DOFs	644	2484	5524	9764	15204	21844
Total CPU time (s)	0.10	0.84	4.83	19.01	62.90	192.60
Relative error (log_{10})	-2.6576	-5.2419	-7.3609	-7.5256	-6.5958	-5.5101
Cond($\boldsymbol{K}$) (log_{10})	21.06	22.56	23.16	27.73	31.67	36.61

Table 3.4: Computational features for the singular solution of the Laplace problem Eq. (3.28) with $X_0 = [1, 1, 1]$, according to the degree. Two subdomains connected by the weak sense coupling method.

Taylor Meshless Method	p=10	p=20	p=30	p=40	p=50	p=60
Number of DOFs	308	1208	2708	4808	7508	10808
Total CPU time (s)	0.15	0.70	4.01	15.69	50.11	142.08
Relative error (log_{10})	-2.8645	-5.7040	-6.6714	-6.6013	-5.1441	-5.9329
Cond($\boldsymbol{K}$) (log_{10})	12.01	19.50	24.17	27.21	30.56	33.28

3.4.4 More about conditioning

To go further in the analysis of conditioning, many calculations have been done, for the three benchmarks described in section 3.3, for many degrees in the range $[5, 60]$. In the cases of singular solutions (Eq. (3.28) and Eq. (3.30, 3.31, 3.32)), many locations of the source points were considered, from very flat solutions for $X_0 = [100, 100, 100]$ up to source points rather close to the domain for $X_0 = [1, 1, 1]$. When the solution is pushed to higher accuracy by increasing the degree p of Taylor series, the matrix condition number increases. For instance in the case of Eq. (3.26), the condition number changes from $10^{3.5}$ to $10^{37.6}$, (Fig. 3.8). Because of the propagation of round-off errors, the accuracy decreases with the degree by starting from a very high accuracy (error $= 10^{-15.1}$ for $p = 5$). A remarkable feature is the quasi-linearity of the curve in the Log-Log plot that can be approximately given as $y \approx 0.206x - 16.4$, where the notations y and x represent the Log_{10}(maximal error) and Log_{10}(condition number) respectively, the value at the origin being very close to the unit round-off error in double precision (1.1×10^{-16}). The very high accuracy for a low degree $p = 5$ is due to fact that the exact solution is a polynomial and the number of elementary calculations is small at this level. By increasing the degree beyond $p = 5$, the number of operations increases and this induces further errors without any compensation due to Taylor series that can be quasi-exact from $p = 5$.

The behavior is different in the general case of non-polynomial solutions, except for very flat solutions ($X_0 = [100, 100, 100]$), in which case the accuracy is also very high from $p = 5$. In the other cases (see Fig. 3.9 and 3.10), one recovers the behavior described in Sections 3.4.1 and 3.4.2. On the left part, one obtains an exponential convergence with the degree and a more or less affine relation between error and conditioning. This corresponds to a convergence dominated by the Taylor series. On the right part of these figures, convergence is deteriorated by the increase of the degree and therefore dominated by the propagation of round-off errors. There is another remarkable feature: all these accuracy curves tend asymptotically to straight lines whose equations are reported in the figures. Moreover these straight lines are very close in the three considered boundary value problems (Laplace equation with a polynomial solution, Laplace equation with source points and 3D elasticity) and this could be approximated by $y = 0.2x - 16$, see Fig. 3.11. Of course the value at the origin (-16) is connected to unit round-off error in double precision, but the slope (0.2) seems also more or less universal to characterize the loss of accuracy due to ill-conditioning. According to this prediction $y = 0.2x - 16$, the accuracy remains fabulous ($y \approx 10^{-8} \sim 10^{-6}$) for condition number of matrix up to $10^{40} \sim 10^{50}$ ($x = 40 \sim 50$) as degree up to $p = 65 \sim 80$. Nevertheless it is not necessary to choose a so large degree, an optimal degree will be at the reversal or before.

In summary, the accuracy of TMM is subjected to two antagonistic effects: the convergence of Taylor series that decreases the error when the degree p increases and the propagation of round-off errors that grows with p. The second effect yields a maximal value of the degree that is very large. The combination of the two effects leads to an accuracy reversal so that an optimal computation requires a degree before this reversal.

3.5 Computation time

The computation time of TMM has not yet been seriously analyzed in the literature. In 2D cases, a comparison with FEM has been done in [50] and it has established that TMM permits a very strong reduction of the number of degrees of freedom as compared with FEM, typically less than 100 DOFs instead of 5000 with FEM at the same level of accuracy. In this section, the origin of the computational cost within TMM is discussed in details. A comparison with the FEM code FreeFem++ [77] will be carried out, mainly to locate the time consuming steps of the TMM computations. This section can be considered as a first attempt to analyze the computational cost of TMM. It has been done with a commercial software (MATLAB R2012b) in

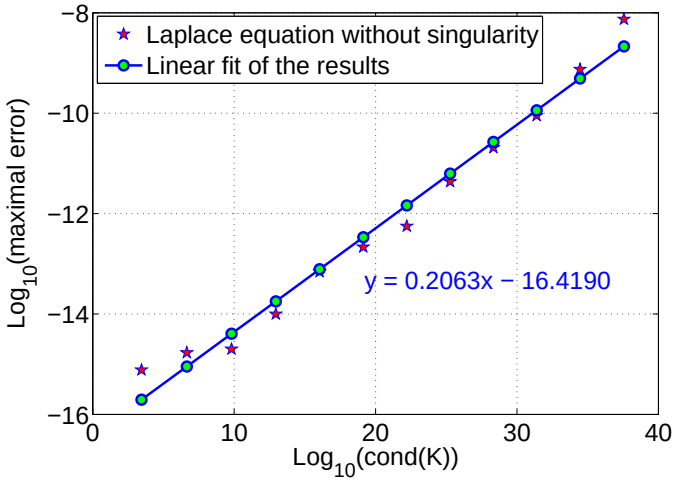


Figure 3.8: The relative error versus the condition number for problem Eq. (3.26).

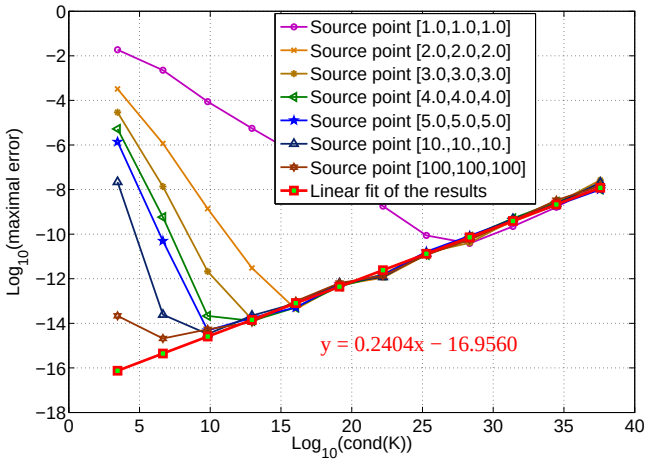


Figure 3.9: The relative error versus the condition number for problem Eq. (3.28).

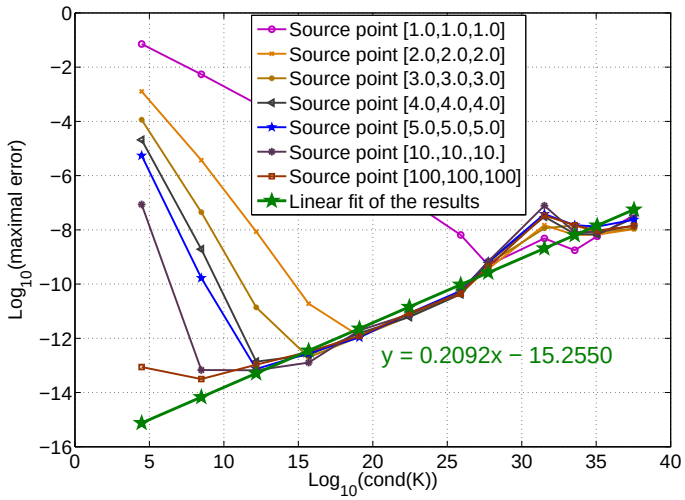


Figure 3.10: The relative error versus the condition number for system (3.30, 3.31, 3.32).

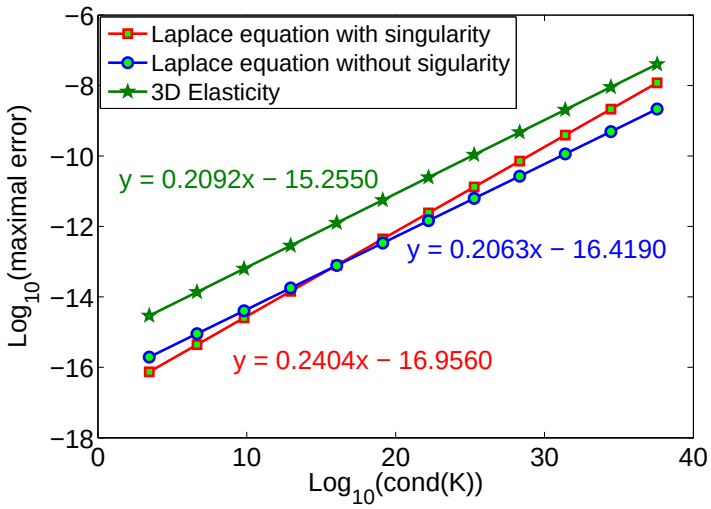


Figure 3.11: Comparing the limits of convergence.

a personal computer under the system of Windows 7 with intel(R) core(TM) i5-2300 type CPU and 8 GB computer memory.

3.5.1 Analysis of the computation time

We considered the Laplace problem Eq. (3.28) with a source point $X_0 = [1, 1, 1]$. This problem is solved by TMM with a development point at $c = [0, 0, 0]$ and a sufficiently large number of collocation points. The convergence analysis is similar with the previous ones, there is an accuracy reversal at $p = 40$ and a high optimal accuracy ($10^{-10.2}$), in Table 3.5. Globally with this new case, one recovers about the same features as with the elasticity problem of Table 3.1 and the multi-domain calculations of Table 3.2, 3.3 and 3.4: same order of magnitude of the errors and of the CPU time, similar accuracy curves with a reversal around $p = 30$ or 40, same growth of the CPU time with the degree. The most significant difference is a larger computation time for the 2 sub-domains cases because they use more DOFs. In the present case, a sufficient accuracy (10^{-5}) is obtained with a rather low degree $p = 20$, few degrees of freedom (441) and a very small computation time ($0.1s$).

Table 3.5: Convergence analysis for problem Eq. (3.28).

Taylor Meshless Method	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$	$p = 60$
Number of DOFs	121	441	961	1681	2601	3721
Collocation points	288	968	2048	3528	5408	7688
Relative error (Log_{10})	-2.6617	-5.1167	-7.6343	-10.2583	-9.6329	-7.8021
Cond($\boldsymbol{K}$) (Log_{10})	6.67	12.95	19.13	25.28	31.39	37.57

Few details about the CPU time are given in Table 3.6. It turns out that a large part of CPU time (71% to 86%) is due to the calculation of the polynomial shape functions, more precisely the numerical values of these polynomials at the cloud of collocation points $P_i(\boldsymbol{x}_j)$. A small part of the time is devoted to the calculation of the matrix $\boldsymbol{K}$, i.e. of products of these quantities $P_i(\boldsymbol{x}_j)$, see Eq. (3.14-3.15). The inversion of the matrix requires few time, less than 6%, even with the largest matrix ($p = 60$, 3721 DOFs). Note that the case p=10 is different because the computer needs a preparing work that requires about 21.5% of the time, but this small size case has little significance in the present discussion. This time distribution is quite different with classical discretization methods where the main cost is due to matrix inversion, what is due to a much larger number of unknowns. A first consequence of these features is an evaluation of the computation time as a function of the degree: the number of polynomials corresponds to the number of unknowns $(p + 1)^2$ and it has been seen that the number of collocation

points is also proportional to this number of DOFs. Hence the number of quantities to be computed $P_i(\boldsymbol{x}_j)$ is of the order of p^4. In Fig. 3.12, we plotted the experimental CPU time as a function of the degree and we found that is about $O(p^{4.04})$, which has a very good agreement compared with the predicted order $O(p^4)$.

Table 3.6: Analysis of CPU time for problem Eq. (3.28).

Taylor Meshless Method	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$	$p = 60$
Number of DOFs	121	441	961	1681	2601	3721
Total CPU time (s)	0.015	0.10	0.56	1.96	5.94	18.66
of which time for $\boldsymbol{P}(\boldsymbol{x})$ (%)	71.00	84.5	85.7	85.1	85.8	84.5
time to $\boldsymbol{K}$ (%)	3.5	5.6	7.7	9.7	10.3	8.7
time for K\F (%)	4.0	2.0	2.5	2.7	2.7	6.0
Parallel calculation (4 cores) (s)	0.05	0.09	0.31	1.15	3.81	10.45

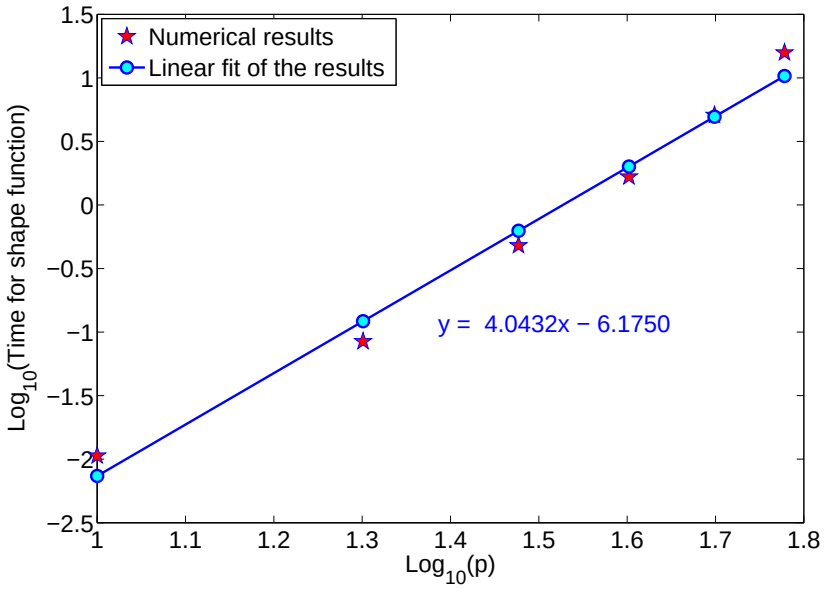


Figure 3.12: CPU time for shape functions versus p.

There is a second interesting consequence of the time distribution: the calculation of all the quantities $P_i(\boldsymbol{x}_j)$ does not require a sequential treatment. Hence this main task in TMM can be achieved in parallel. The used processor has four cores so that first parallel computations have been achieved, the results being reported in Table 3.1 and Table 3.6. Significant time saving have been obtained for the most expansive calculations ($40 \leq p \leq 60$), with a gain between 35 and 45% for the results of Table 3.6 and between 15 and 25% in Table 3.1.

3.5.2 First comparison with FEM

An assessment of the efficiency of TMM requires a first comparison with a classical discretization method. In this respect, we have chosen FreeFem++ that is an open source finite element code to solve systems of Partial Differential Equations [77]. It has a lot of users and offers a large variety of triangular elements, an automatic mesh generator and a large variety of linear solvers. Here tetrahedrons with linear (T1) or quadratic (T2) interpolation will be used to solve the Laplace problem Eq. (3.28) in a spherical domain corresponding to singularities $X_0 = [1, 1, 1]$ and $X_0 = [2, 2, 2]$. To compare TMM and FreeFem++, we look at the number of degrees of freedom and the computation time to get a given accuracy, namely a relative error less than 10^{-2}, 10^{-3} or 10^{-4}. In each case, several finite element meshes have been tested and the most significant two have been reported in Table 3.7. With the linear tetrahedron, we were not able to reach the highest accuracies because of a lack of memory in our computer.

In any case the meshless method is more efficient than the FEM, of course in terms of number of unknowns, but also of computation time. The best results with FreeFem++ correspond to the cases with a weak accuracy requirement (error less than 1%), with a time ratio 32/1 and 100/1 and a DOFs ratio of 6/1 and 13/1. The results with a strong accuracy (error less than 10^{-3} or than 10^{-4}) are more in favor of the meshless method that converges more rapidly with an increase of the degree, the corresponding ratios being respectively 50/1 and 140/1 (time) and 22/1 and 70/1 (DOFs).

Hence Taylor Meshless Method seems a competitive approach to reduce significantly the number of unknowns and the computation time as compared with well established discretization methods. Of course these first results would deserve to be deepened with larger scale problems, other tests with several subdomains, more complex PDEs $\cdots$, a first attempt is presented in the next section.

3.5.3 Large boxes submitted to sinusoidal loading

In most of the previous examples, the focus was on the effect of matrix conditioning, on the robustness of the algorithm and the evaluation of computation cost. It was established that the algorithm still give rather accurate results with very large degrees $(50, 60)$, but the optimal degree lies before the accuracy reversal presented in Fig. 3.8, 3.9 and Fig. 3.10. Moreover large degrees induce increasing computational costs of the order $O(p^4)$. At the opposite, it does not seem a good idea to settle for a small degree ($p = 1, 2, \cdots$) as in the FEM. Indeed, first, the solution by Taylor series and the absence of numerical

Table 3.7: Comparison between the proposed method and FEM: Tetrahedrons with linear and quadratic shape functions (T1 and T2). The table presents the number of NDOFs necessary to get a relative error lower than 10^{-2}, 10^{-3}, 10^{-4}, for problem Eq. (3.28) with two source points X_0. In parentheses, the actually obtained error.

Element Type	Singularity at $X_0 = [1,1,1]$		Singularity at $X_0 = [2,2,2]$	
	NDOFs: 10^{-2}	NDOFs: 10^{-3}	NDOFs: 10^{-2}	NDOFs: 10^{-4}
FEM(T1)	383(-2.0149)	None	215(-2.5099)	None
CPU time(s)	0.57		0.366	
	8242(-2.2166)	None	1220(-2.9442)	None
	12.54		1.85	
FEM(T2)	485(-2.5989)	3767(-3.0271)	349(-3.3227)	4455(-4.0374)
CPU time(s)	0.262	1.15	0.207	1.27
	3380(-2.9704)	36047(-3.3363)	2925(-3.9006)	43468(-4.1868)
	0.98	7.01	0.67	6.83
TMM	64(-2.0859)	169(-3.1406)	16(-2.1455)	64(-4.4096)
CPU time(s)	0.008	0.024	0.002	0.008

integration permits to work with moderately large degrees ($p = 10, 20$) without overgrowth of the computation time like in the p-version of the FEM [102]. Second the accuracy with $p = 1, 2$ should be very poor, which should require a too large number of sub-domains. Hence a good compromise is to choose a degree in the range $p \in [6, 20]$.

The TMM-computations are compared with FreeFem++ calculations using tetrahedral elements and quadratic shape functions. Linear interpolation is available with FreeFem++, but of course the convergence is lower. Elements with higher degrees are not available in 3D. One should mention that FreeFem++ is associated with several multifrontal solvers that are efficient for large-scale problems and we use the default sparse solver UMFPACK [78].

First we concern the domain $l_x \times l_y \times d = 10 \times 10 \times 1$, discretized by 180000 elements and 255025 DOFs within FreeFem++ and by 12100 DOFs with TMM (degree $p = 10$, 100 "elements"), these two discretizations being designed to get an error less than 10^{-3}. The approximate and exact solutions and the relative error on the plane $z = 0.8$ inside the domain obtained by TMM have been illustrated in Fig. 3.13, 3.14 and 3.15. The corresponding CPU-time are $6.39s$ with FreeFem++ and $5.08s$ with TMM. A larger domain ($l_x \times l_y \times d = 25 \times 25 \times 1$) has been considered (second test) by requiring the same accuracy. Several finite element meshes have been tested and the one which is able to reach this accuracy involves more than three millions

DOFs (3213051), the TMM-mesh containing 625 "elements" and 75625 DOFs for a degree $p = 10$. The obtained CPU-times are $98.73s$ for FreeFem++ and $71.93s$ for TMM. Note that the condition number remains relatively small (in the range $10^9 \sim 10^{11}$) for all these large-scale TMM-tests, while it is much larger (about 10^{32}) within FreeFem++ due to a specific procedure to apply the boundary conditions.

In summary the present TMM procedure works well for problems re-

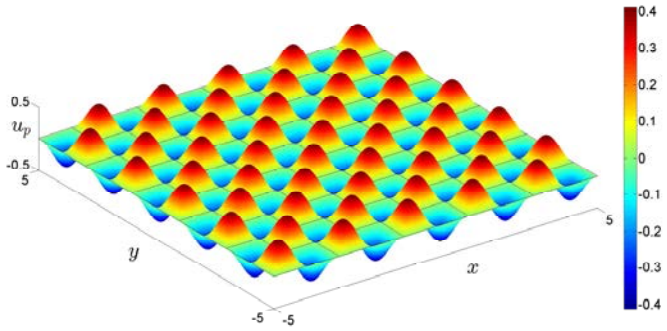


Figure 3.13: Test of section 3.3.4, the approximation on the plane $z = 0.8$.

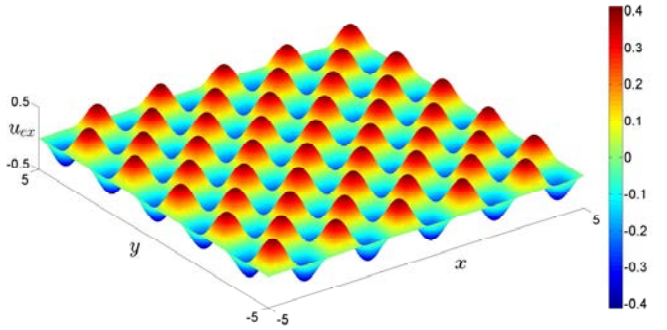


Figure 3.14: Test of section 3.3.4, The exact solution on the plane $z = 0.8$.

quiring more than one million of unknowns by classical finite elements, even with hundreds of TMM-elements. Especially the method converges rapidly with the degree, even with a large number of "elements", which assesses the piecewise resolution techniques presented in section 3.2.3. Of course these first results deserve to be deepened, especially by a consistent use of parallel computations and by an optimal memory management.

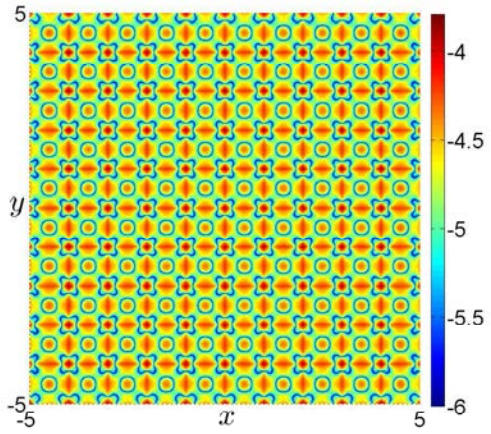


Figure 3.15: Test of section 3.3.4, The relative error on the plane $z = 0.8$.

3.6 Conclusion

A truly integration-free meshless method was assessed in this chapter for the purpose of application to large-scale problems involving three-dimensional domains. The specificity lies in the shape functions that are approximated solutions of the PDEs. So the discretization concerns only the boundary and it is accounted by least-square collocation. Then the control of the computation time and of the ill-conditioning of the matrix become central questions in view of practical applications of this method.

It has been found that the accuracy of the method is governed by two antagonistic effects: first the classical convergence of Taylor series that leads to an exponential convergence with the degree, second the propagation of round-off errors, what grows with the degree and is facilitated by matrix ill-conditioning. According to our few numerical tests, the effect of round-off errors yields a lower bound of the error that seems more or less universal within the double-precision floating-point format. When the two effects are balanced, there is an accuracy reversal point beyond which the error increases. In practice, the method yields very accurate solutions up to a large degree ($p = 60$) and with many degrees of freedom: more than 70000, what would correspond to more than three millions of DOFs with a classical discretization method. This establishes the ability of the method to solve rather large-scale problems. Likely this result is due to the analytical solving inside the domain that limits the number of unknowns. In practice, the optimal degrees seem smaller ($p = 10, 20$) and the last test (section 3.5.3) shows that one can combine a rather large degree and many sub-domains. This allows to account for very large-scale problem, which is difficult to achieve within several other

meshless techniques.

The computation time of the method has been discussed to locate the time consuming steps. In the present state of the algorithm, the largest part of the computation time is spent on calculating the value of the shape functions at the collocation points. The time distribution is quite different from usual methods (FEM, FDM) and this is an advantage in view of parallel computing. Nevertheless by comparing with a well established finite element code, one finds that TMM needs much less degrees of freedom and less computation time.

The numerical tests were based on 3D Laplace equation and linear isotropic elasticity. The application to non-linear problems has been sketched in previous works [52, 54, 96] and a more complete study will be published elsewhere, the only difficulty being the calculation of shape functions for more generic PDEs.

Chapter 4

Taylor Meshless Method for non-linear PDEs

Abstract

A true meshless integration-free method based on Taylor series named Taylor Meshless Method (TMM) has been proposed recently to solve Partial Differential Equations (PDEs), where the shape functions are high degree polynomials and the discretization concerns only the boundary. With high computational efficiency and exponential convergence, the TMM has been confirmed to be very robust in linear problems, including large-scale cases. In this chapter, the TMM and the Automatic Differentiation (AD) are combined with the Newton method to solve non-linear elliptic PDEs, where the AD is used to compute shape functions in a fast manner. The numerical results illustrate that the proposed algorithm is very accurate and efficient in solving non-linear elliptic problems.

Present chapter corresponds to the published research paper (Yang et al., Taylor meshless method for solving non-linear partial differential equations, J. Comput. Phys., 348, 385-400, 2017).

Keywords: Taylor series; Multivariate algorithmic differentiation; Meshless; Newton Method.

Contents

4.1 Introduction

The non-linear Partial Differential Equations (PDEs) play an important role in science and engineering. To solve these non-linear PDEs, plenty of numerical methods have been developed over the past decades, such as finite difference [103,104], finite element, finite volume [105,106] and spectral methods [107–109]. Meshless methods based on fundamental solution [110–113], radial basis function [29–31] and moving least-squares approximations [8, 57, 68,114,115] were also applied to address non-linear PDEs. In these numerical methods, two main classes of discretization techniques were applied: Galerkin procedures and pointwise methods. Galerkin-based methods may work for large-scale problems, but their requirement for integration procedures could definitely lead to high computational costs. Pointwise-based techniques avoid the drawback of integration, but they could cause ill-conditioned matrices and numerical instabilities, which make it difficult to solve large-scale problems.

This paper proposes a new numerical method to solve non-linear elliptic PDEs. The original point of the method is to compute shape functions from Taylor series as proposed in [47]. Since the PDE is solved analytically, a significant reduction in the number of degrees of freedom (DOFs) has been obtained comparing with other discretization techniques, typically a reduction ratio in the range $[20, 100]$, see [50, 116]. This strong reduction has two origins. First, because the PDE is solved analytically inside the domain, the discretization concerns only the boundary. Second, the Taylor series shows exponential convergence properties with respect to its degree and can characterize the unknown field in a large domain using a single series.

In practice, the method is confirmed to be efficient for polynomials with a large degree, generally in the range $[8, 30]$, but the method also works for larger degrees. It is never easy to define an efficient integration procedure with such a class of polynomials. For this reason, a collocation-based discretization method was combined with Taylor series. As the pure collocation may show the numerical instabilities, a least-square minimization proposed in [48,68] was introduced to overcome this drawback. In this way, the method also enjoys many advantages of the collocation-based meshless methods: the discretization is fully defined from clouds of points, what's more, there is no required connection with the neighboring and no integration as well.

As a counterpart, collocation-based methods may be less reliable and lead to very ill-conditioned matrices, making them inapplicable to large-scale problems. This aspect has been well analyzed in the chapter 3. It has been found that the accuracy of the method is dominated by two antagonistic effects. The refinement of discretization results in exponential convergence, while the propagation of round-off errors governed by matrix ill-conditioning grows with

the degree of polynomial shape functions. In practice, the proposed method yields very accurate solutions up to a large degree and with many DOFs.

In the previous papers [47, 49–51, 116, 117], it has been well established that the method always converges exponentially with the degree and can be applied to any boundary condition and any shape of domain. Bridging techniques based on Lagrange multipliers [51] or least-square collocation [117] have been presented to connect several sub-domains. It has been proven that the method also works well (p-convergence) in piecewise resolutions even with hundreds of sub-domains and more than $70,000$ DOFs, while the corresponding finite element numerical model requires over 3 millions DOFs.

The applications of TMM to non-linear PDEs have been attempted before [52–54] but they are limited to simple or specific equations such as ODEs in [52, 53] and 2D-hyperelasticity in [54]. The purpose of this chapter is to define a general procedure for solving non-linear elliptic PDEs. Normally a numerical method for non-linear PDEs combines two steps: a linearization procedure and a discretization procedure. Some linearization techniques are available in literature such as Asymptotic Numerical Method (ANM) [118, 119], Newton method, Newton-Raphson method and Picard iteration [120, 121]. In this paper we use Newton method since it is the simplest and the most commonly used technique to achieve this linearization. The application of Newton method to a non-linear PDE leads to an inhomogeneous linear equation at each iteration. The TMM-solution of the latter linear PDE relies on the computation of Taylor series of a function composed by two analytic functions. This calculation is achieved by Automatic Differentiation (AD) of multivariate Taylor series [98]. Finally a general procedure will be presented to compute the polynomial shape functions for a large class of elliptic equations. This does not necessarily imply an efficient convergence of the non-linear algorithm nor does the convergence of the discretization. For instance it should be non-obvious to capture very localized patterns occurring in contact mechanics [122] or in convection-dominated problems [123].

The layout of this chapter is as the following. In section 4.2.1, the procedure of solving a PDE by the method of Taylor series is shortly recalled. section section 4.2.2 gives a brief description of Newton method. In section 4.2.3, a procedure based on AD is presented to compute the Taylor series of a function composed by two analytic functions. In section 4.2.4, one recalls the main features of TMM including its ability to solve large-scale problems. In section 4.3, a few one-dimensional and three-dimensional applications are discussed to assess the efficiency and robustness of the proposed technique.

4.2 Description of the method

The aim of this section is to present a general computational method based on Taylor series for solving non-linear elliptic PDEs. First we recall how to build the general solution of a linear PDE with variable coefficients, the latter being given by their Taylor series. Roughly, this corresponds to a method presented in previous papers ([47,49–51]). In general, a non-linear equation is solved by a linearization procedure (Newton-Raphson, Newton, etc.) that transforms the non-linear systems into a sequence of linear problems. This leads to compose analytic functions by the method of Taylor series (section 4.2.2). Hence the key point is to establish a numerical technique giving the Taylor series of a function that is the composition of two analytic functions. Such techniques are available in literature concerning algorithmic differentiation (or automatic differentiation) [98]. The chosen algorithm is presented in section 4.2.3. The combination of Newton method and algorithmic differentiation provides a general procedure to solve elliptic PDEs from Taylor series. The main characteristics of TMM are the rapid convergence with the degree (p-convergence), the robustness of the algorithm with respect to several parameters and the ability to deal with rather large degrees. For the sake of completeness, these points are discussed briefly in section 4.2.4.

4.2.1 From PDE to Taylor series

4.2.1.1 Computing the shape functions

The aim of this section is to recall the procedures to build a general solution of a PDE. Here, we consider a linear PDE in Ω, a bounded domain in $\mathbb{R}^3$, with mixed boundary conditions as follows:

$$\begin{cases} -\Delta u(\boldsymbol{x}) + g(\boldsymbol{x})u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = u^d & \boldsymbol{x} \in \Gamma_d, \\ \boldsymbol{T}u(\boldsymbol{x}) = t^n & \boldsymbol{x} \in \Gamma_n. \end{cases} \tag{4.1}$$

where the function $u(\boldsymbol{x})$ is unknown while $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$ are known, and $\boldsymbol{T}$ is a linear differential operator.

The functions $u(\boldsymbol{x})$, $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$ are represented by truncated Taylor series with a development point $\boldsymbol{x}_c = [x_c, y_c, z_c]$. For convenience, the vector $\boldsymbol{x} - \boldsymbol{x}_c$ is denoted by $[\xi, \eta, \gamma]$.

$$u(x, y, z) = \sum_{k=0}^{p} \sum_{i=0}^{k} \sum_{j=0}^{k-i} \widetilde{u}(i, j, k-i-j) \xi^i \eta^j \gamma^{k-i-j} \tag{4.2}$$

$$g(x,y,z) = \sum_{k=0}^{p-2}\sum_{i=0}^{k}\sum_{j=0}^{k-i} \widetilde{g}(i,j,k-i-j)\xi^i\eta^j\gamma^{k-i-j} \tag{4.3}$$

$$f(x,y,z) = \sum_{k=0}^{p-2}\sum_{i=0}^{k}\sum_{j=0}^{k-i} \widetilde{f}(i,j,k-i-j)\xi^i\eta^j\gamma^{k-i-j} \tag{4.4}$$

where $\widetilde{u}(\ ,\ ,\ )$, $\widetilde{g}(\ ,\ ,\ )$ and $\widetilde{f}(\ ,\ ,\ )$ represent the coefficients in the Taylor series of $u(\boldsymbol{x})$, $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$, respectively. Note that only the coefficients $\widetilde{u}(\ ,\ ,\ )$ are unknown. The expansion of $u(\boldsymbol{x})$ contains $(p+1)(p+2)(p+3)/6$ coefficients which will be a large value when a large degree p is chosen.

The principle of TMM is to vanish the Taylor series of the residual $R(\boldsymbol{x}) = \Delta u(\boldsymbol{x}) - g(\boldsymbol{x})u(\boldsymbol{x}) + f(\boldsymbol{x})$ up to order $p-2$. The resulting equations then permit to reduce the number of unknown coefficients. In this respect, we shall distinguish $(p-1)p(p+1)/6$ coefficients that will be expressed in terms of the $(p+1)^2$ other coefficients. After this elimination, $(p+1)^2$ independent unknowns will remain, which will be collected in a vector $\{\boldsymbol{v}\}$.

First $\Delta u(\boldsymbol{x})$ is a polynomial of order $p-2$. The coefficient of the term containing $\xi^m\eta^n\gamma^s$ in $\Delta u(\boldsymbol{x})$ reads:

$$\begin{aligned}\widetilde{\Delta u}(m,n,s) = &(m+2)(m+1)\widetilde{u}(m+2,n,s)\\ &+(n+2)(n+1)\widetilde{u}(m,n+2,s)\\ &+(s+2)(s+1)\widetilde{u}(m,n,s+2).\end{aligned} \tag{4.5}$$

where the expression $\widetilde{\Delta u}(\ ,\ ,\ )$ represents the Taylor coefficients of $\Delta u(\boldsymbol{x})$.

Next we come to the second term $g(\boldsymbol{x})u(\boldsymbol{x})$ in Eq. (4.1). The product of these two Taylor series leads to a function $h(\boldsymbol{x})$. The coefficient $\widetilde{h}(m,n,s)$ of the term containing $\xi^m\eta^n\gamma^s$ in $h(\boldsymbol{x})$ denotes:

$$\widetilde{h}(m,n,s) = \sum_{i=0}^{m}\sum_{j=0}^{n}\sum_{k=0}^{s} \widetilde{g}(i,j,k)\widetilde{u}(m-i,n-j,s-k) \tag{4.6}$$

Then the expression of the residual $R(\boldsymbol{x})$ is as following:

$$R(\boldsymbol{x}) = \sum\sum\sum \left[\widetilde{\Delta u}(m,n,s) - \widetilde{h}(m,n,s) + \widetilde{f}(m,n,s)\right]\xi^m\eta^n\gamma^s \tag{4.7}$$

where $((m,n,s) \in [0,p-2],\ m+n+s \leq p-2)$. The residual vanishes if the following conditions are satisfied:

$$\widetilde{\Delta u}(m,n,s) - \widetilde{h}(m,n,s) + \widetilde{f}(m,n,s) = 0 \tag{4.8}$$

There exist $(p-1)p(p+1)/6$ linear equations in Eq. (4.8) containing $(p+1)(p+2)(p+3)/6$ unknown coefficients $\widetilde{u}(\ ,\ ,\ )$. We choose the coefficients

$\widetilde{u}(0,\,,\,)$ and $\widetilde{u}(1,\,,\,)$ as independent unknowns. If the PDE (4.1) is consider as a differential equation with respect to x, its solution can be expressed in terms of the "initial values" $u(0,y,z)$ and $\partial u/\partial x(0,y,z)$. Then the non-independent coefficients can be expressed in terms of the independent ones, which finally leads to a reduced expression u_p containing $(p+1)^2$ unknowns.

This procedure can be easily applied to a large degree of polynomials p and one can get the approximated solution as following:

$$\begin{aligned} u_p =& P_0(x,y,z) + \sum_{i=1}^{(p+1)^2} P_i(x,y,z) v_i \\ =& P_0(x,y,z) + \langle \boldsymbol{P} \rangle \{\boldsymbol{v}\}. \end{aligned} \tag{4.9}$$

where the first polynomial $P_0(x,y,z)$ balances the right-hand side $f(\boldsymbol{x})$ of Eq. (4.1).

4.2.1.2 Boundary least-square collocation

Since the PDE in Eq. (4.1) has been approximately satisfied by the reduced approximation u_p in Eq. (4.9), only boundary conditions need to be considered to determine the unknown vector $\{\boldsymbol{v}\}$. Here, a collocation technique combined with the least-square method [48,68] is used. One chooses a set of nodes $\boldsymbol{x}_i$ on Γ_d and another set of nodes $\boldsymbol{x}_j$ on Γ_n. Then one minimizes the error between the approximate value u_p, $\boldsymbol{T}u_p$ and the given value of u^d, t^n at these points. It comes to minimize the following function:

$$\mathscr{T}(\boldsymbol{v}) = \frac{1}{2} \sum_{\boldsymbol{x}_i \in \Gamma_d} \left| u_p(\boldsymbol{x}_i) - u^d(\boldsymbol{x}_i) \right|^2 + w \cdot \frac{1}{2} \sum_{\boldsymbol{x}_j \in \Gamma_n} \left| \boldsymbol{T}u_p(\boldsymbol{x}_j) - t^n(\boldsymbol{x}_j) \right|^2 \tag{4.10}$$

where the weight coefficient w is used to balance the two kinds of boundary conditions. This minimization leads to a linear system $[\boldsymbol{K}]\{\boldsymbol{v}\} = \{\boldsymbol{b}\}$ where $[\boldsymbol{K}]$ is an invertible matrix. Solving this system gives the vector $\{\boldsymbol{v}\}$ and therefore the approximate solution of the problem Eq. (4.1). Note the simple form of the matrix $[\boldsymbol{K}]$ and the vector $\{\boldsymbol{b}\}$ writes:

$$[\boldsymbol{K}]_{\alpha\beta} = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i) \cdot P_\beta(\boldsymbol{x}_i) + w \cdot \sum_{j=1}^{M_n} \boldsymbol{T}P_\alpha(\boldsymbol{x}_j) \cdot \boldsymbol{T}P_\beta(\boldsymbol{x}_j), \quad \alpha,\beta \in [1,(p+1)^2] \tag{4.11}$$

$$\{\boldsymbol{b}\}_\alpha = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i)(u^d(\boldsymbol{x}_i) - P_0(\boldsymbol{x}_i)) + w \cdot \sum_{j=1}^{M_n} \boldsymbol{T}P_\alpha(\boldsymbol{x}_j)(t^n(\boldsymbol{x}_j) - \boldsymbol{T}P_0(\boldsymbol{x}_j)) \tag{4.12}$$

where M_d and M_n represent the number of collocation points on the displacement boundary Γ_d and the Neumann boundary Γ_n respectively.

4.2.1.3 Piecewise resolution

It is impossible to solve any boundary value problem with a single Taylor series. Thus, numerical methods are introduced to connect several high-order polynomial approximations. Within this aim, several techniques have been presented and well investigated in the previous works: discrete Lagrange multipliers or discretized continuous Lagrange multipliers [51,99] and least-square collocation method [117]. In this paper, we focus on the least-square collocation for the interface conditions in piecewise resolution which will be briefly recalled in the following.

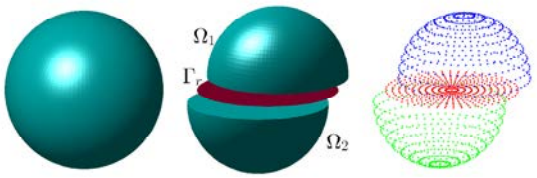


Figure 4.1: Sketch for two sub-domains.

Here we consider a domain as shown in Fig. 4.1. The independent high-order general solution for each sub-domain, u_p^1 and u_p^2, is deduced from Taylor series respectively as stated in section 4.2.1.1. The consideration of the boundary conditions for each sub-domain yields two functions, $\mathscr{T}_1(\boldsymbol{v}^1)$ and $\mathscr{T}_2(\boldsymbol{v}^2)$, as described in section 4.2.1.2. Note that the number of unknowns for each sub-domain is $(p+1)^2$. The constraints on the interface are as follows:

$$\begin{cases} \psi_j^1(\boldsymbol{v}^1, \boldsymbol{v}^2) = u_p^1(\boldsymbol{x}_j) - u_p^2(\mathbf{x}_j) = 0 \\ \psi_j^2(\boldsymbol{v}^1, \boldsymbol{v}^2) = \frac{\partial u_p^1}{\partial n}(\boldsymbol{x}_j) - \frac{\partial u_p^2}{\partial n}(\boldsymbol{x}_j) = 0 \end{cases} \quad \forall \boldsymbol{x}_j \in \Gamma_r \tag{4.13}$$

where $\boldsymbol{x}_j$ represents the collocation points on the interface Γ_r.

By applying the least-square collocation method, the transmission conditions Eq. (4.13) are satisfied in a mean square sense via the following coupling function:

$$\mathscr{C}_{ls}(\boldsymbol{v}^1, \boldsymbol{v}^2) = \frac{1}{2} \sum_{j=1}^{M_r} \left[\left| u_p^1(\boldsymbol{x}_j) - u_p^2(\boldsymbol{x}_j) \right|^2 + w \cdot \left| \frac{\partial u_p^1}{\partial n}(\boldsymbol{x}_j) - \frac{\partial u_p^2}{\partial n}(\boldsymbol{x}_j) \right|^2 \right] \tag{4.14}$$

where the weight coefficient w is used to balance the two kinds of constraints on the interface. For the whole problem, searching the variables $\{\boldsymbol{v}^1\}$ and $\{\boldsymbol{v}^2\}$ is equivalent to minimizing the following function:

$$\mathscr{T}(\boldsymbol{v}^1, \boldsymbol{v}^2) = \mathscr{T}_1(\boldsymbol{v}^1) + \mathscr{T}_2(\boldsymbol{v}^2) + \mathscr{C}_{ls}(\boldsymbol{v}^1, \boldsymbol{v}^2) \tag{4.15}$$

4.2.2 Newton Method

As the method recalled in section 4.2.1 relies on a superposition principle (see Eq. (4.9)), it can be only applied to a linear equation. In this respect, a linearization procedure is required if one wants to solve a non-linear equation. In this paper, we limit ourselves to Newton method that is the simplest way to achieve this linearization. To show this linearization procedure in details, we consider the following non-linear equation:

$$\begin{cases} -\Delta u(\boldsymbol{x}) + h(u(\boldsymbol{x})) = q(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = u^d & \boldsymbol{x} \in \Gamma_d, \\ \boldsymbol{T}u(\boldsymbol{x}) = t^n & \boldsymbol{x} \in \Gamma_n. \end{cases} \tag{4.16}$$

where Ω is a bounded domain and $h(\cdot)$ represents a non-linear function.

The initial solution and the correction of the solution are represented by u_0 and δu respectively. Usually the initial solution is set to be $u_0 = 0$. The $(i+1)^{th}$ Newton iteration is as follows:

$$\begin{cases} -\Delta(\delta u) + h'(u_i)\delta u + R(u_i) = 0, & \boldsymbol{x} \in \Omega, \\ u_{i+1}(\boldsymbol{x}) = u^d & \boldsymbol{x} \in \Gamma_d, \\ \boldsymbol{T}u_{i+1}(\boldsymbol{x}) = t^n & \boldsymbol{x} \in \Gamma_n, \\ u_{i+1} = u_i + \delta u. \end{cases} \tag{4.17}$$

where $h'(\cdot)$ represents the derivative of function $h(u)$ with respect to u and $R(u_i)$ represents the residual of the solution obtained at the previous iteration.

One stops the iteration when one of the following conditions is satisfied:
- $|log_{10}\mathfrak{R}_n - log_{10}\mathfrak{R}_{n-1}| \leq 10^{-2}$,
- The maximum number of iterations is equal to 20,

where $\mathfrak{R}_n$ represents the maximum value of the function $R(u_n)$.

At each iteration of the algorithm, one gets a linear PDE which will be solved by TMM. According to the computing procedure of TMM in section 4.2.1, one needs to expand the known functions $h'(u_i)$ and $R(u_i)$ into Taylor series. This can be realized by AD and will be described in the next section.

4.2.3 Automatic Differentiation

At each step of the non-linear algorithm, one needs to solve a linear PDE with variable coefficients as Eq. (4.1), where the two functions $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$, $\boldsymbol{x} \in \Omega$, depend on the approximation of the unknown function $u(\boldsymbol{x})$ at the

previous iteration:

$$\begin{cases} g(\boldsymbol{x}) & = h'(u_i(\boldsymbol{x})) \\ f(\boldsymbol{x}) & = \Delta(u_i(\boldsymbol{x})) - h(u_i(\boldsymbol{x})) + q(\boldsymbol{x}) \end{cases} \tag{4.18}$$

The computation of TMM-shape functions $\langle \boldsymbol{P}(\boldsymbol{x}) \rangle$ requires the derivatives of composed functions $G(\boldsymbol{x}) = F(U(\boldsymbol{x}))$, where $F(\cdot)$ is a given analytic function and $U(\boldsymbol{x})$ is another analytic function approximated by a truncated Taylor series. The numerical composition of a Taylor series with a given analytic function can be achieved with the standard tools of AD.

First let us recall the statement of AD for univariate functions. The aim is to compute the numerical values of Taylor coefficients (equivalent of the derivatives) of $G(\boldsymbol{x})$ at a point x_c. There are well known recurrence formulae if the function $F(\cdot)$ is simple: product, fraction, exponential, etc. [124]. These elementary functions are included in a small library. As illustrated in Table 4.1, such a recurrence can be extended to more intricate functions by combining these basic operations. AD works for any function that is the composition of elementary functions, which is not restrictive in practice. This can be implemented within Matlab by building a class for Taylor series, including the type and the overloaded operations (i.e. the rules in right column of Table 4.1).

Table 4.1: Successive elementary operations to compute $F(U(\boldsymbol{x})) = U^2/(1 + e^U)$ and its derivatives.

1	$V = U \cdot U$	$V_n = \sum_{k=0}^{n} U_k \cdot U_{n-k}$
2	$W = e^U$	$W_n = \frac{1}{n} \sum_{k=0}^{n-1} (n-k) W_k \cdot U_{n-k}$
3	$X = 1$	$X_0 = 1$, $X_n = 0$ $(n \geq 1)$
4	$Y = X + W$	$Y_n = X_n + W_n$
5	$G = V/Y$	$G_n = (V_n - \sum_{k=0}^{n-1} G_k \cdot Y_{n-k})/Y_0$

The extension to multivariate Taylor series has been addressed since a long time [98, 125, 126]. The best way seems to compute Taylor series of univariate functions $t \to G(\boldsymbol{x}_c + t\boldsymbol{r})$ in a sufficiently large number of directions $\boldsymbol{r}$ and to deduce all the derivatives of the multivariate function $G(\boldsymbol{x})$ up to a given order from all these univariate Taylor series. We refer to the AD literature for the details of the algorithm, including [127].

Neidinger has done an open source Matlab implementation of this theory

in his work on "Directions for computing truncated multivariate Taylor series" [126]. The code sources can be found on his web page [128]. However for our needs in the TMM, we propose an implementation of the truncated power series algebra (TPSA) based on the works of [129–131]. The Mathematica codes of Kaltchev have been translated and adapted in Matlab for the automatic computation of the truncated multivariate Taylor series of any Matlab functions. The TMM Matlab Toolbox contains two classes: "pindexes" and "ndseries". The class "pindexes" is a singleton class and contains all the "boxes" described in [129–131]. Once the space dimension of the problem (i.e. three dimensional problems) and the order of the truncated series are chosen, an instance of "pindexes" is created with all the data to manage the series computation. The class "ndseries" is the main class where the overloading of the operators and elementary functions have been implemented. An instance of "ndseries" has a member called "coeffs" (1D Matlab array) which stores the multivariate Taylor coefficients of the object. This member is manipulated by the operators and elementary functions using the "boxes" defined in the singleton object "pindexes". The following operators and functions are currently overloaded in the TMM Toolbox: acos, asin, asinh, atan, atanh, cos, cosh, exp, log, log10, minus, mpower, plus, sin, sinh, sqrt, tan, tanh, uminus and uplus. Thanks to this Matlab toolbox, the Taylor series of the functions (4.18) can always be computed.

4.2.4 Recalling the basic properties of TMM

In this section, two 3D linear problems are considered to briefly recall the basic properties of TMM, especially exponential convergence, robustness, efficiency and ability to solve large-scale problems. The first case is the Laplace equation in a sphere domain with singular solution and this case will be solved by a single Taylor series. The next one is issued from the study of instabilities of film-substrate systems [75, 76, 132], where one has to describe many spatial oscillations. This second case will be solved with hundreds of sub-domains.

4.2.4.1 Laplace equation with singular solution

The problem considered here reads:

$$\begin{cases} \Delta u(x,y,z) = 0 & in \quad \Omega, \\ u_d(x,y,z) = \dfrac{1}{r} & on \quad \partial\Omega. \end{cases} \tag{4.19}$$

where $\Omega = \{x, y, z | x^2 + y^2 + z^2 \leq 1\}$ and $r = \|\boldsymbol{x} - \boldsymbol{x}_0\|$. The exact solution is as following with a singularity at $\boldsymbol{x}_0 = [x_0, y_0, z_0]$:

$$u_{ex}(x, y, z) = \frac{1}{\|\boldsymbol{x} - \boldsymbol{x}_0\|} \tag{4.20}$$

Throughout this chapter, the technique will be evaluated through the maximum relative error which is defined by:

$$\mathcal{E} = \frac{max\,|u_p(\boldsymbol{x}) - u_{ex}(\boldsymbol{x})|}{max\,|u_{ex}(\boldsymbol{x})|} \tag{4.21}$$

The numerical solution obtained by TMM depends on two parameters: the degree p of the polynomials and the number of collocation points M. The influence of the number of collocation points is illustrated in Fig. 4.2 for three values of the degree $p = 10$, 20 and 30 with a singularity point at $X_0 = [1, 1, 1]$. The maximum error decreases with M until an optimal number $M = 2(p + 1)^2$ where it becomes stable. The divergence for small value of M is easily understood by considering the equivalent 2D problem: in this case, the minimization of Eq. (4.10) is simply an identification of Taylor series that cannot work for a small M. For larger value of M, it is observed that the identified solution does not depend on the collocation points and the remaining error is only due to the truncation of Taylor series. The convergence with the polynomial degree p is presented in Fig. 4.3 for different singular points $X_0 = [1, 1, 1]$ and $X_0 = [2, 2, 2]$ with $2(p + 2)^2$ collocation points to ensure the convergence. The results illustrated in Fig. 4.3 show that this technique converges exponentially with the degree and the rate of convergence depends on each specific problem. This 3D example illustrates the rapid convergence with the degree, as well as the robustness of the method with respect to the number of discretization points and to the degree. For more assessments, we refer to [51, 116].

4.2.4.2 A large-scale test

This benchmark is motivated by wrinkling instabilities of film-substrate systems [75, 76, 132], where one observes many spatial oscillations. Here one accounts only for the substrate, and the film is represented by a prescribed sinusoidal displacement.

$$\begin{cases} \triangle u(x, y, z) = 0 \qquad \text{in} \quad \Omega \\ u(x, y, 0) = 0 \\ u(x, y, d) = sin(\pi x) sin(\pi y) \end{cases} \tag{4.22}$$

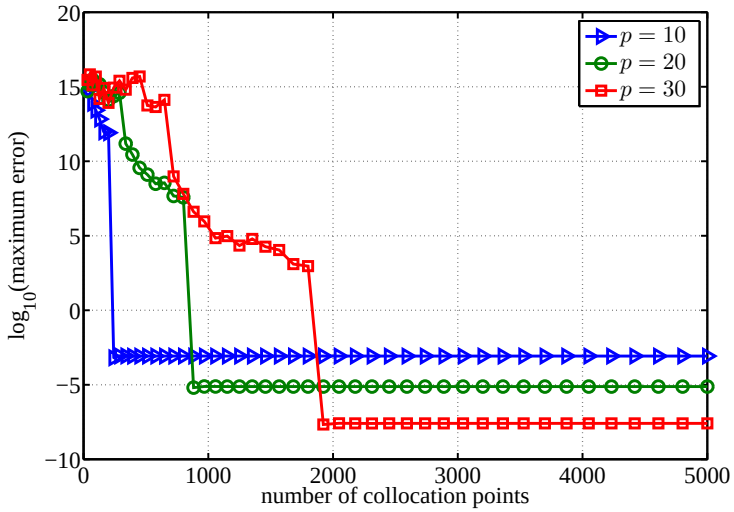


Figure 4.2: Influence of number of collocation points for problem Eq. (4.19).

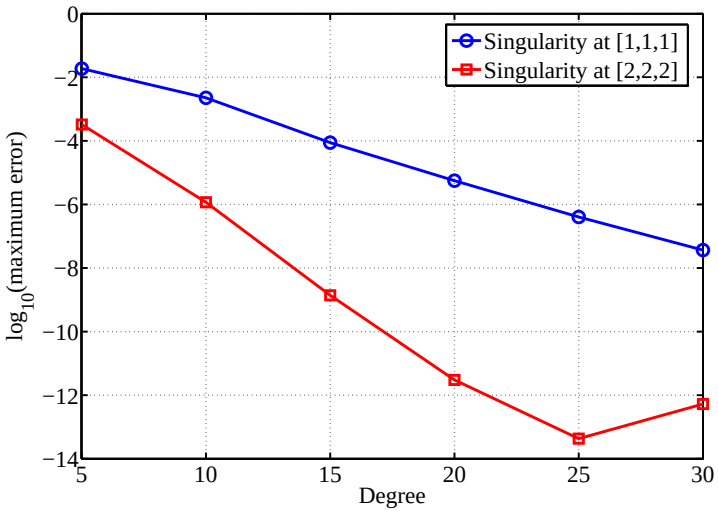


Figure 4.3: p-convergence for two Laplace problems Eq. (4.19) that differ by the position of the singularity.

The domain is a rectangular hexahedron with sides $\ell_x \times \ell_y \times d = 10 \times 10 \times 1$. We split this domain into cubes of size $1 \times 1 \times 1$ that will be called "elements" for simplicity. This results in one element through the thickness and two elements along each period. In this case, Dirichlet boundary conditions are applied and the exact solution is $u(x, y, z) = sin(\pi x)sin(\pi y)sinh(\sqrt{2}\pi z)/sinh(\sqrt{2}\pi d)$. On each surface of the cube "element", p^2 collocation points for the boundary conditions and $p^2/4$ collocation points for the transmission conditions are chosen to ensure the convergence.

Table 4.2: The p-convergence of problem Eq.(4.22).

Degree of polynomial	$p = 5$	$p = 10$	$p = 15$	$p = 18$
Number of DOFs	3600	12100	25600	36100
log_{10}(max error)	-0.9140	-3.2407	-6.5568	-8.8391

The results presented in Table 4.2 show that the error decreases rapidly with the degree (p-convergence). This demonstrates the robustness and rapid convergence of the piecewise resolution even with a large number of subdomains. A comparison between the TMM and FEM has been done for this example. The efficient open source code FreeFem++ [77] is used to perform the finite element simulation. It contains rapid multi-frontal linear solvers such as UMFPACK [78]. The domain is discretized by tetrahedral elements with quadratic shape functions (elements T2). To obtain a relative error less than $10^{-3.2}$, the finite element mesh requires 417501 degrees of freedom. The corresponding CPU-time is $10.31s$ with FreeFem++ and $5.08s$ with TMM. We have also implemented the standard Finite Difference Method (FDM) and tested various regular meshes, especially with steps $h = 0.0625$, $h = 0.05$ and $h = 0.03875$. The FDM converges rather slowly, with respective errors of $10^{-2.64}$, $10^{-2.86}$ and $10^{-3.21}$. The last case requires a huge computational time because of the large number of DOFs (2338588). Finally our most significant FDM result corresponds to a step $h = 0.05$, 752419 DOFs, an error $10^{-2.86}$, and a CPU-time of 77.3s, as compared with FEM (417501 DOFs, 10.31s) and TMM (12100 DOFs, 5.08s). Remark that such comparisons are never perfectly fair because they can be computer-dependent, implementation-dependent and even author-dependent, FreeFem++ benefits from an optimized implementation and from elements T2 (with linear interpolation elements (T1), it converged also more slowly). A larger domain ($\ell_x \times \ell_y \times d = 20 \times 20 \times 1$) has been considered (second test) by requiring the same accuracy. Several finite element meshes have been tested and the one which is able to reach this accuracy involves more than one million DOFs (1656441), the TMM-mesh containing 400 "elements" and 48400 DOFs for a degree $p = 10$. The obtained CPU-times

are 46.44s for FreeFem++ and 25.67s for TMM. The results about CPU-time are obtained by using a personal computer with inter(R) core(TM) i5-2300 type CPU and 8 GB computer memory.

4.2.4.3 Comments for TMM

In the previous papers [47, 49–51], it has been validated that TMM is robust and efficient for solving linear PDEs, even with large numbers of DOFs or hundreds of sub-domains. One can also use TMM to handle problems with any shape of domain and any boundary condition. For instance, TMM works well for a problem with an amoeba-like boundary shape in [47] and for elasticity problems with displacement and stress boundary conditions in [49]. In practice, TMM needs less DOFs as compared with other discretization techniques since the PDE is solved analytically. This small number of unknowns is a typical feature of TMM. One observes a significant reduction of the number of DOFs with a ratio in the range $[20, 100]$ with respect to the FEM.

4.3 Numerical applications

In this section, several non-linear problems in one-dimension and three-dimension are considered to validate the robustness and convergence of the proposed technique which combines Taylor meshless method with algorithmic differentiation. The proposed method for non-linear PDEs includes two main steps: firstly, one linearizes the non-linear PDE by Newton method and gets a linear PDE at each iteration as in Eq. (4.17); secondly, this linear PDE is solved by combining Taylor series inside the domain and least-square collocation at boundaries and interfaces. The TMM procedure for one-dimensional linearized ODEs is recalled in Appendix B and for three-dimensional cases it has been presented in section 4.2.1, see also [116] for more details.

4.3.1 One-dimensional non-linear problems

4.3.1.1 An one-dimensional example

Here, we consider a non-linear ODE as following:

$$\begin{cases} -u''(x) + u^3(x) - 1 = 0, \quad x \in [-1, 1] \\ u(\pm 1) = 0. \end{cases} \tag{4.23}$$

One should note that only 2 DOFs and 2 points are necessary in this considered problem. The maximum residual of the ODE is considered as an

indicator for the measurement of the accuracy of the approximation. The results will be discussed under two aspects: first the convergence of the Newton iterations for a fixed degree of polynomials p; second the convergence with the refinement of the discretization (p increases).

The convergence with the number of iterations and the degree of polynomials is presented in Table 4.3. For each fixed degree of polynomials p, the maximum residual decreases rapidly with the number of iterations until a maximum accuracy is reached and then it remains constant. Only 3 iterations are necessary to get the maximum accuracy for $p = 10$ and 20, and 4 iterations are sufficient for $p = 30$ and 40.

Hence with a sufficient number of Newton iterations, the proposed technique converges exponentially with the degree of polynomials p and up to a very high accuracy.

Table 4.3: The convergence with the number of iterations and the degree of polynomials for problem Eq.(4.23).

Degree of polynomial	Iteration	$p = 10$	$p = 20$	$p = 30$	$p = 40$
log_{10}(maximum residual)	1	-0.9031	-0.9031	-0.9031	-0.9031
	2	-2.153	-2.7844	-2.7844	-2.7844
	3	-1.9376	-4.9503	-6.6041	-6.6041
	4	-1.9376	-4.9543	-8.4665	-12.7815
	8	-1.9376	-4.9543	-8.4665	-12.7817

4.3.1.2 Bratu equation

The classical one-dimensional Bratu's problem is considered as following:

$$\begin{cases} u''(x) + \lambda e^{u(x)} = 0, & x \in [0,1] \\ u(0) = u(1) = 0. \end{cases} \tag{4.24}$$

where λ is a constant. The analytical solution of Bratu's problem is in the following form:

$$u(x) = -2ln\left[\frac{cosh((x-\frac{1}{2})\frac{\theta}{2})}{cosh(\frac{\theta}{4})}\right] \tag{4.25}$$

where θ is the solution of $\theta = \sqrt{2\lambda}cosh(\frac{\theta}{4})$.

The accuracy of the discrete problem depends on two parameters: the number of iterations and the degree of polynomial shape functions p. Here we focus on the influence of the degree of polynomials on the accuracy.

Three cases are considered here with different values of λ: $\lambda = 1$, 1.5 and 2 respectively. The numerical solutions of these cases obtained by TMM with $p = 20$ are shown in Fig. 4.4. It appears that the approximated solutions agree well with the analytical solutions.

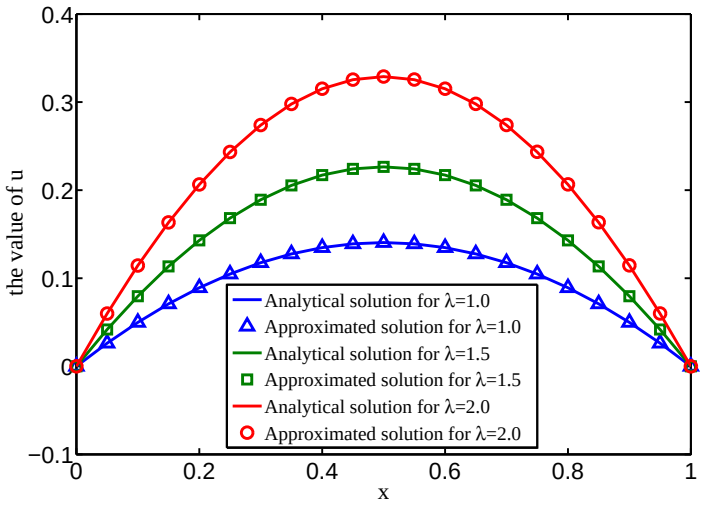


Figure 4.4: The values of u for Bratu's equation (4.24) obtained by TMM with $p = 20$.

The convergence with the degree of polynomials ($p \in [10, 40]$) is summarized in Table 4.4. In all these cases, the results show that the maximum relative error decreases rapidly when increasing the degree of polynomials, as in the linear cases.

Table 4.4: The p-convergence for Bratu's problem (4.24) with different λ.

Degree of polynomial		$p = 10$	$p = 20$	$p = 30$	$p = 40$
log_{10}(maximum error)	$\lambda = 1.0$	-3.1302	-6.9192	-9.7169	-12.3561
	$\lambda = 1.5$	-2.0113	-4.5520	-7.0990	-9.5040
	$\lambda = 2.0$	-1.3025	-3.1504	-5.0475	-7.1640

The relationship between the maximum residual and the maximum relative error of the approximated solution has been illustrated in Fig. 4.5. It can be seen that the maximum relative error decreases with the maximum residual when increasing the degree of polynomials ($p \in [10, 40]$). This means that the maximum residual can be a good indicator of accuracy of the approximated solution when there is no reference solution of the non-linear problem.

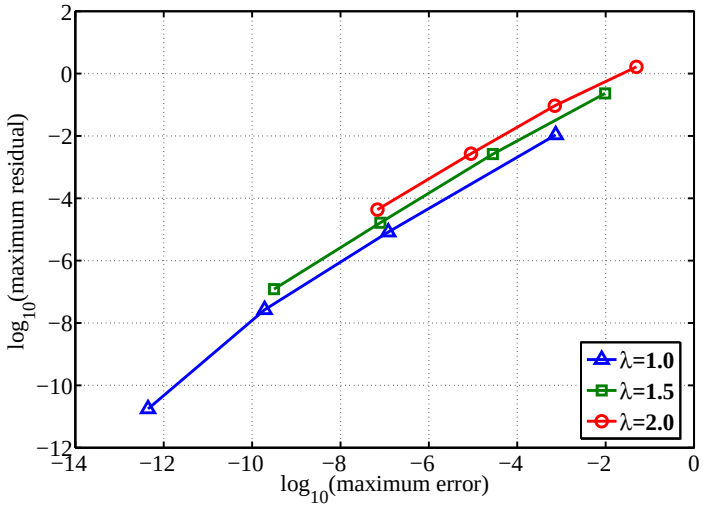


Figure 4.5: The residual of Bratu's equation (4.24) versus the error for different degrees ($p \in [10, 40]$).

4.3.2 Three-dimensional non-linear problems

In the previous section, we have tested the efficiency and robustness of the proposed technique for non-linear ODEs. Since our final goal is to apply the TMM associated with Newton method for large-scale problems, several three-dimensional non-linear PDEs are discussed in this section.

4.3.2.1 Problem one

We consider the following non-linear problem in a unit sphere:

$$\begin{cases} -\Delta u(x,y,z) + u^3(x,y,z) = f(x,y,z) & in \quad \Omega, \\ u(x,y,z) = \sin(x)\sin(y)e^{-z} & on \quad \partial\Omega. \end{cases} \tag{4.26}$$

where the right-hand function $f(x, y, z)$ denotes:

$$f(x,y,z) = \sin(x)\sin(y)e^{-z} + \sin^3(x)\sin^3(y)e^{-3z}$$

The exact solution is $u_{ex} = \sin(x)\sin(y)e^{-z}$.

The accuracy of the solution depends on three factors: the number of collocation points on the boundary; the degree of polynomials and the number of Newton iterations, among which the influence of the number of collocation points for three-dimensional linear PDEs has been well investigated in Yang et al. [116]. This has been also briefly recalled in section 4.2.4. The maximum accuracy is obtained for about $2(p+1)^2$ collocation points and it remains constant beyond this threshold. Here $2(p+2)^2$ collocation points have been

chosen to ensure the convergence.
The convergence with Newton iterations (within 10 iterations) and the discretization of TMM ($p \in [5, 30]$) are presented in Fig. 4.6. For each degree p, Newton iterations converge rapidly until obtaining a maximum accuracy and then it remains constant beyond this threshold. The number of iterations necessary for the convergence is rather small. For instance, only 3 Newton iterations are sufficient to get the maximum accuracy for $p = 20$ and 30.

The convergence with the discretization can be also seen in Fig. 4.6 and detailed information is presented in Table 4.5. One should note that the number of DOFs is $(p + 1)^2$. The maximum relative error and the maximum residual decrease exponentially with the degree of polynomials. From a degree $p = 20$, the accuracy is no longer improved because one is closed to the unit round-off error within the double precision floating-point format.

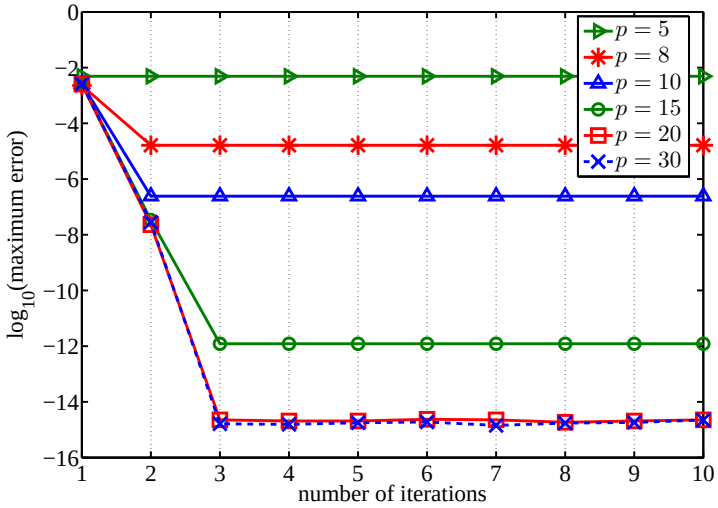


Figure 4.6: Problem Eq. (4.26). Convergence with Newton iterations and with the discretization (p-convergence).

Table 4.5: The maximal relative error of each iteration for the 3-D problem Eq.(4.26).

Degree of polynomial	$p = 5$	$p = 8$	$p = 10$	$p = 15$	$p = 20$	$p = 30$
Number of DOFs	36	81	121	256	441	961
Number of iterations	1	2	2	3	3	3
log_{10}(max error)	-2.3103	-4.7874	-6.6143	-11.9125	-14.6874	-14.8070
log_{10}(max residual)	-1.0902	-3.0926	-4.6054	-9.1402	-14.6186	-15.5758

4.3.2.2 Problem two

We consider the second non-linear problem in a unit sphere ($\Omega = \{x, y, z | x^2 + y^2 + z^2 \leqslant 1\}$):

$$\begin{cases} -\Delta u(x,y,z) + u^3(x,y,z) = \dfrac{1}{r^3} & in \quad \Omega, \\ u(x,y,z) = \dfrac{1}{r} & on \quad \partial\Omega. \end{cases} \tag{4.27}$$

where $r = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2}$. The exact solution is as follows:

$$u_{ex}(x,y,z) = \frac{1}{\sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2}} \tag{4.28}$$

This solution has a singularity at $X_0 = [x_0, y_0, z_0]$. The critical number of collocation points on the boundary for the 3D case with p order polynomial shape functions is about $2(p+1)^2$ and the accuracy remains constant beyond this threshold. Here $2(p+2)^2$ collocation points have been chosen to ensure the convergence.

The p-convergence has been reported in Table 4.6 for the problem Eq. (4.27) with the nearest singularity point $X_0 = [1.0, 1.0, 1.0]$. The maximum relative error and maximum residual decrease exponentially with the degree in several iterations.

Table 4.6: p-convergence of problem Eq.(4.27) with singularity at $X_0 = [1, 1, 1]$.

Degree of polynomial	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$
Number of iterations	2	3	3	4	4
log_{10}(max error)	-2.6448	-5.2524	-7.4341	-10.0566	-12.2454
log_{10}(max residual)	-2.1183	-5.9682	-6.6592	-10.6087	-12.4773
CPU-time (s)	0.09	1.15	10.85	94.76	434.47

The influence of the singular point on the rate of the convergence has been shown in Fig. 4.7. Two different singular points $X_0 = [1, 1, 1]$ and $X_0 = [2, 2, 2]$ have been chosen. The results show that the quality of convergence depends on the considered problem, especially on the position of the singular point X_0. The rate of convergence is better if the singularity is far from the domain which means that TMM is easier to solve problem with a flat solution. In the case with singularity at $X_0 = [2, 2, 2]$, from a degree $p = 30$, the accuracy is no longer improved because one is closed to the unit round-off error within the double precision floating-point format.

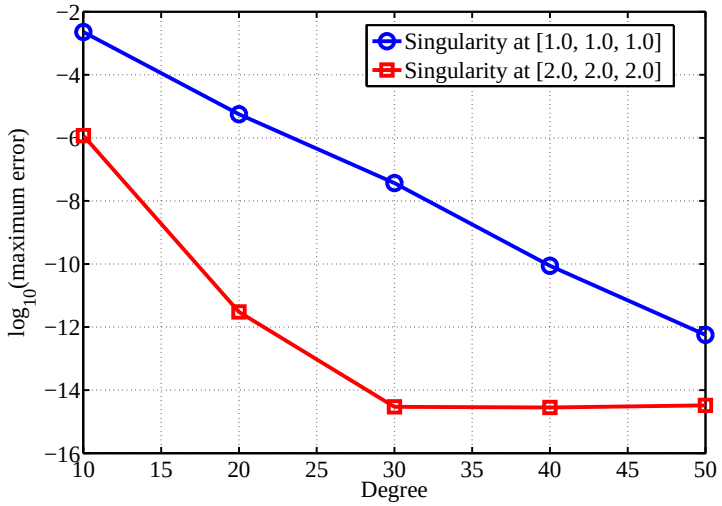


Figure 4.7: p-convergence for two non-linear problems Eq. (4.27) that differ by the singularity X_0.

The CPU-time presented in Table 4.6 increases rapidly with the degree of polynomials p, the CPU-time is 0.09s for $p = 10$ while 434.47s for $p = 50$. As analyzed in a previous paper [116], the computation of shape functions is the most time-consuming part in TMM and it relates to the number of unknowns $(p+1)^2$ and the number of collocation points $2(p+2)^2$ as shown in Eq. (5.9). Hence the number of quantities to be computed $P_\alpha(\boldsymbol{x}_j)$ is of the order of p^4. Nevertheless it is not necessary to choose a so large degree and in practice the optimal degree seems in the range $p \in [10, 20]$.

4.3.2.3 Problem three

We consider the third non-linear problem in a spherical domain ($\Omega = \{x, y, z | x^2 + y^2 + z^2 \leqslant 1\}$):

$$\begin{cases} -\Delta u(x,y,z) + u^3(x,y,z) = 1 & in \quad \Omega, \\ u(x,y,z) = 0 & on \quad \partial\Omega. \end{cases} \tag{4.29}$$

The number of DOFs of TMM with any degree of polynomials p is $(p+1)^2$ and the critical number of collocation points is about $2(p+1)^2$. In the following $2(p+2)^2$ collocation points will be chosen to ensure the convergence. The maximum residual of the PDE is considered here as an indicator for the measurement of the accuracy of the approximation. The convergence with the refinement of discretization is presented in Table 4.7. Only 2 iterations are necessary to get the maximum accuracy for $p = 10$ and 3 iterations for

Table 4.7: p-convergence of problem Eq.(4.29).

Degree of polynomial	p=5	p=8	p=10	p=20	p=30
Number of DOFs	36	81	121	441	961
Number of iterations	1	2	2	3	3
log_{10}(max residual)	-2.3345	-3.9346	-4.4316	-9.5295	-15.8210

$p = 20$ and 30. With a sufficient number of Newton iterations, the proposed technique converges exponentially with the degree of polynomials p and up to a very high accuracy.

4.3.2.4 Problem four

We consider the non-linear convection equation in a spherical domain ($\Omega = \{x, y, z | x^2 + y^2 + z^2 \leqslant 1\}$):

$$\begin{cases} -\Delta u + (\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} + \frac{\partial u}{\partial z})u = f(x, y, z) & in \quad \Omega, \\ u(x, y, z) = e^x \sin(\pi y/2) \sin(\pi z/2) & on \quad \partial\Omega. \end{cases} \tag{4.30}$$

where the right-hand side denotes:

$$f(x, y, z) = (1 + \pi)e^{2x} \sin^2(\pi y/2) \sin^2(\pi z/2) + (\pi^2/2 - 1)e^x \sin(\pi y/2) \sin(\pi z/2) \tag{4.31}$$

The exact solution is $u(x, y, z) = e^x \sin(\pi y/2) \sin(\pi z/2)$.

Table 4.8: p-convergence of problem Eq.(4.30).

Degree of polynomial	$p = 5$	$p = 10$	$p = 15$	$p = 20$	$p = 30$
Number of iterations	2	3	4	4	4
log_{10}(max error)	-1.3483	-4.7346	-8.6229	-13.4719	-14.1924
log_{10}(max residual)	-0.5556	-3.5428	-7.7981	-11.1863	-13.5416

The critical number of collocation points on the boundary is about $2(p + 1)^2$. Here $2(p + 2)^2$ collocation points have been chosen to ensure the convergence. Then the accuracy of the approximated solution depends on two factors: the discretization of TMM (the degree of Taylor series p) and the number of Newton iterations. The convergence with the previous two factors has been illustrated in Table 4.8. The number of Newton iterations necessary to ensure the convergence is still small. Only 4 iterations are needed to obtain the best accuracy in the case with $p \leq 30$. With a sufficient number of Newton iterations, the maximum relative error and the maximum residual decrease exponentially with the degree of polynomial shape functions.

4.3.2.5 A large-scale test

One considers a non-linear PDE with Dirichlet boundary conditions, the linear version being discussed in section 4.2.4.2:

$$\begin{cases} -\Delta u(x,y,z) + u^3(x,y,z) = f^3(x,y,z) & \text{in} \quad \Omega \\ u(x,y,0) = 0 \\ u(x,y,d) = \sin(\pi x)\sin(\pi y) \end{cases} \tag{4.32}$$

where $f(x,y,z) = \sin(\pi x)\sin(\pi y)\sinh(\sqrt{2}\pi z)/\sinh(\sqrt{2}\pi d)$ and Ω is a rectangular hexahedron with sides $\ell_x \times \ell_y \times d = 10 \times 10 \times 1$, see Fig. 4.8. The exact solution reads $u_{ex} = \sin(\pi x)\sin(\pi y)\sinh(\sqrt{2}\pi z)/\sinh(\sqrt{2}\pi d)$.

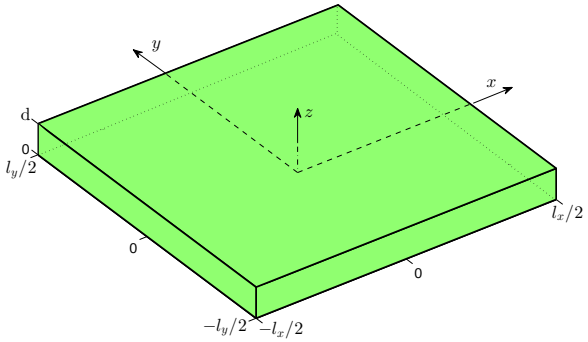


Figure 4.8: Sketch for the shape of domain, problem Eq. (4.32).

We split the domain into 100 cubes of size $1 \times 1 \times 1$ that will be called "elements" for simplicity, see Fig. 4.9. This corresponds to one element through the thickness and two elements along each period. The Dirichlet boundary

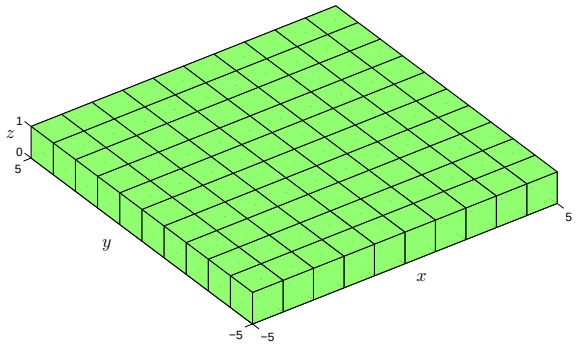


Figure 4.9: Sketch for the plate divided into 100 sub-domains, problem Eq. (4.32).

conditions and connections between sub-domains are achieved by the least-square collocation technique. On each surface of the cube "element", p^2 collocation points for the boundary conditions and $p^2/4$ collocation points for the transmission conditions are used to ensure the convergence.

The p-convergence of the proposed technique for the problem Eq. (4.32) has been shown in Table 4.9. The error decreases exponentially with the degree when taking enough Newton iterations. The number of iterations necessary for convergence is still small, only 2 Newton iterations for a degree $p \leq 10$, while 3 iterations are enough for a degree p in the range $[11, 20]$. This demonstrates the robustness of TMM associated with Newton method for solving non-linear elliptic PDEs, even with large numbers of TMM-elements and large numbers of DOFs.

Table 4.9: The p-convergence of problem Eq. (4.32) solved by TMM with 100 elements.

Degree of polynomial	p=5	p=8	p=10	p=15	p=18
Number of DOFs	3600	8100	12100	25600	36100
Number of iterations	1	2	2	3	3
log_{10}(max error)	-0.8537	-2.0655	-3.1639	-6.5548	-8.8333
log_{10}(max residual)	-1.0988	-1.787	-2.7377	-4.8785	-6.7583

The CPU-time for this case Eq. (4.32) with 100 elements, $p = 10$ and 2 Newton iterations, is 18.25s. The CPU-time for a similar linear problem Eq. (4.22) recalled in section 4.2.4.2, with the same domain and boundary conditions, is 5.08s. The CPU-time of the non-linear case for each iteration (about 9s) is a little larger than that of the linear case (5.08s). Likely this extra time is due to the calculation of the product of $h'(u_i)\delta u$ (in Eq. (4.17)) by AD. This point deserves further discussion, but in the present state, the automatization has already a moderate counterpart in CPU-time.

4.4 Conclusion

A general numerical method associating Taylor meshless method with Automatic Differentiation, called TMM, is proposed in this chapter to solve a class of non-linear elliptic PDEs. First Newton method is used to linearize the non-linear PDE, but the present method could be easily adapted to any linearization technique. Then the obtained linear problem is solved by TMM. The specificity of TMM is the numerical computation of a family of shape functions that are solutions of the PDE in the sense of Taylor series. Solving

the linearized equations needs to compute the Taylor series of the functions composed by two analytic functions. This key point is realized by using the Automatic Differentiation.

The numerical tests demonstrate that non-linear TMM keeps the same advantages as in the linear case. First the discretization technique converges exponentially with the degree and it requires much less degrees of freedom than other techniques like the finite element method. Next it is quite robust with respect to the number of collocation points and to the degree of Taylor series that can be very large. In the considered numerical example, the Newton method converges rapidly with one or two additional iterations for large degrees. TMM does not require a mesh, but merely requires a cloud of collocation points on the boundary. The computation time has been estimated in a large-scale test (12,100 DOFs for TMM, but more than 400,000 for the equivalent FEM mesh) and it remains of the same order as that required in the linear case, which demonstrates that the implementation via Automatic Differentiation does not alter the effectiveness of the method. The proposed technique also works well for a very large degree, for instance $p = 50$, but leads to high computational costs of the order $O(p^4)$. Nevertheless it's not necessary to choose such a large degree, in practice an optimal degree seems smaller and usually in the range $[10, 30]$. The numerical tests in this chapter are limited to a single non-linear elliptic PDE. Some applications to practical engineering problems in the fields of solid and fluid mechanics are in progress. Accounting for singularities is very important in solid mechanics because of cracks, corners or concentrated forces, and an ongoing research will include singular shape functions in the present computational framework.

Chapter 5

Computing singular solutions of PDEs by Taylor series

Abstract

Taylor Meshless Method (TMM) is a true meshless integration-free numerical method for solving elliptic Partial Differential Equations (PDEs). The basic idea of this method is to use high order polynomial shape functions that are approximated solutions of the PDE and are computed by the technique of Taylor series. Currently, this new method has proved robust and efficient and it has the property of exponential convergence with the degree, when solving problems with smooth solutions. This exponential convergence is no longer obtained for problems involving cracks, corners or notches. On the basis of numerical tests, this chapter establishes that the presence of a singularity leads to a worsened convergence of the Taylor series, but highly accurate solutions can be recovered by including a few singular solutions in the basis of shape functions.

Present chapter corresponds to the submitted research paper (Yang et al., Computing singular solutions of partial differential equations by Taylor series, submitted to Journal, 2017).

Keywords: Taylor series; Meshless; Singular shape functions; Angular domain.

Contents

5.1 Introduction

This chapter deals with the numerical solution of elliptic Partial Differential Equations (PDEs) by using Taylor series. The main interest of a Taylor series is the property of exponential convergence with the degree, which leads to strong reductions of the number of degrees of freedom (DOFs) as compared, for instance, with the finite element method [50, 116]. Nevertheless the radius of convergence of a Taylor series is limited by the nearest singularity and singularities are common in engineering mechanics because of corners, cracks or pointwise forces. So it is expected that such singularities will slow the convergence. In this chapter, the relation between singular solutions and convergence of Taylor series is addressed.

Polynomial solutions of some PDEs have been used since a long time to build large finite elements, for instance in plane elasticity [133] or for plate bending [134], even with relatively large degrees. When associated with a specific treatment of boundary and interface conditions, this method has been called hybrid-Trefftz method, but there are also many other ways to account for these boundary-interfaces conditions [35]. Trefftz methods reduce dramatically the number of DOFs because only the boundary and interfaces have to be discretized, but it does not work as well for non-homogeneous and non-linear problems, in which cases exact solutions are not known. To extend the method in the non-linear range, it has been proposed to compute approximate solutions in the sense of Taylor series [47] and the latter procedure is efficient for a large class of PDEs, even in non-linear cases [135], see chapter 3 and 4. When coupled with appropriate collocation-based subdomain methods [51, 116], it leads always to exponential convergence results. In this form, it has been called Taylor Meshless Method (TMM). These excellent convergence properties are illustrated in Fig. 5.1, where three problems are studied, one with a very flat response and two others with larger gradients. In the flat case, the convergence is very rapid and a small degree is sufficient to get a high precision ($\simeq 10^{-14}$) that is close to the limit for a real number in double-precision floating-point format. In the two other cases, there are two types of response. For a small degree, the response is governed by the exponential convergence of Taylor series, even for large domains. For larger degrees, the accuracy deteriorates slightly due to the propagation of round-off errors, what is favored by matrix ill-conditioning. The accuracy reversal observed in Fig. 5.1 is obtained for degrees $p = 25$ and $p = 45$, which means that TMM works well with large degrees. How to keep these excellent convergence properties when the solution is not analytic?

The treatment of singular solutions is a key issue in computational fracture mechanics. The bibliography on the topics is huge and we just refer for

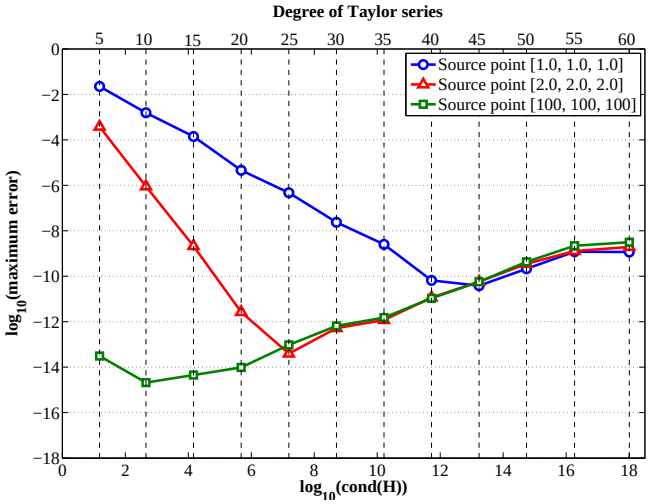


Figure 5.1: Relative error versus conditioning for a degree varying from $p = 5$ up to $p = 60$. One tries to recover fundamental solutions within 3D linear elasticity. Three cases were considered, from [116].

instance to [136] or [137]. Standard finite elements, which ignore the singular fields, could be used to analyse elastic cracked bodies, but this "should require extremely fine meshes" [136], so that various specific elements were introduced to better capture the singular behavior near crack tip. The first one [138] simply consists in including the analytic singular functions and minimizing with respect to the stress intensity factors, but there are many variants in the literature of the seventies, see for instance [139] or [140]. The modern approaches, called extended finite element method or generalized finite element method, are based on the partition of unity [141] and can include analytic singular fields and discontinuities of the displacement [142]. Note that some problems with singular solutions were solved in a meshless framework by using a full basis of singular solutions [143, 144].

The convergence properties of Taylor Meshless Method in the presence of singularities are adressed in this chapter, especially for domains with corners and for classical crack problems. As expected, it is not possible to recover exponential convergence up to a high accuracy, in the same way as in Fig. 5.1. Building on the significant achievements within fracture mechanics, we shall combine the high degree polynomials computed by Taylor series with analytically known singular solutions. Indeed the ability to solve problems with singular solutions is an important issue in the development of Taylor-based numerical methods.

5.2 Combining Taylor series and singular solution

The issue of the present chapter is the behavior of Taylor methods for elliptic systems having a singular solution. In this part 5.2, it will be shown that the convergence of Taylor meshless method, as illustrated in Fig. 5.1, deteriorates for corner-domains (section 5.2.3). Next a variant of Taylor Meshless Method is proposed simply by including few singular solutions in the basis of shape functions (section 5.2.4). For completeness, two main features of TMM are recalled: first the algorithm to compute polynomial approximate solutions of the PDE by the method of Taylor series (section 5.2.1), next a procedure to build the global problem by a boundary least-square collocation method (section 5.2.2).

5.2.1 Compute shape functions from Taylor series

The Taylor Meshless Method can be split in two parts. First the differential equation is solved in a quasi-exact manner by using the concept of Taylor series, next the boundary and interface conditions are accounted by a collocation-based technique. Let us begin by the solving inside the domain and describe the method for a simple two-dimensional linear PDE with mixed boundary conditions:

$$\begin{cases} -\Delta u(\boldsymbol{x}) + g(\boldsymbol{x})u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = u^d & \boldsymbol{x} \in \Gamma_d, \\ \mathbf{T}u(\boldsymbol{x}) = t^n & \boldsymbol{x} \in \Gamma_n. \end{cases} \tag{5.1}$$

where the function $u(\boldsymbol{x})$ is unknown while $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$ are given analytical functions, and $\mathbf{T}$ is a linear differential operator.

Firstly, one expands $u(\boldsymbol{x})$, $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$ by using truncated Taylor series with a development point $\boldsymbol{x}_c = [x_c, y_c]$. For convenience, the vector $\boldsymbol{x} - \boldsymbol{x}_c$ is denoted by $[\xi, \eta]$.

$$u(x,y) = \sum_{j=0}^{p} \sum_{i=0}^{j} \widetilde{u}(i, j-i) \xi^i \eta^{j-i} \tag{5.2}$$

$$g(x,y) = \sum_{j=0}^{p-2} \sum_{i=0}^{j} \widetilde{g}(i, j-i) \xi^i \eta^{j-i} \tag{5.3}$$

$$f(x,y) = \sum_{j=0}^{p-2} \sum_{i=0}^{j} \widetilde{f}(i, j-i) \xi^i \eta^{j-i} \tag{5.4}$$

where $\widetilde{u}(\,,\,)$, $\widetilde{g}(\,,\,)$ and $\widetilde{f}(\,,\,)$ represent the coefficients of the Taylor series of $u(\boldsymbol{x})$, $g(\boldsymbol{x})$ and $f(\boldsymbol{x})$ respectively. Note that only the coefficients $\widetilde{u}(\,,\,)$ are unknown.

The principle of TMM is to vanish the residual $R(\boldsymbol{x})$ of the PDE up to the order $(p-2)$. Substituting Eqs. (5.2-5.4) into the PDE in Eq. (5.1), one can express $R(\boldsymbol{x})$ in the form of Taylor series:

$$\begin{aligned} R(\boldsymbol{x}) =& \Delta u(\boldsymbol{x}) - g(\boldsymbol{x})u(\boldsymbol{x}) + f(\boldsymbol{x}) \\ \approx& \sum_{m=0}^{p-2}\sum_{n=0}^{m} \widetilde{R}(m,n)\xi^m\eta^n \end{aligned} \tag{5.5}$$

where $\widetilde{R}(m,n)$ represents the coefficients of the Taylor series of the residual $R(\boldsymbol{x})$. The principle is to vanish these Taylor coefficients of the residual:

$$\begin{aligned} \widetilde{R}(m,n) =& (m+2)(m+1)\widetilde{u}(m+2,n) + (n+2)(n+1)\widetilde{u}(m,n+2) \\ &- \sum_{i=0}^{m}\sum_{j=0}^{n} \widetilde{g}(i,j)\widetilde{u}(m-i,n-j) + \widetilde{f}(m,n) = 0 \end{aligned} \tag{5.6}$$

The resulting equations can be considered as recurrence formulae permitting to compute a family of polynomials that are quasi-exact solutions of the PDE. In this way, one reduces the number of unknowns from $(p+1)(p+2)/2$ to $2p+1$. The later independent unknowns are collected in a vector $\{\boldsymbol{v}\}$. Then the reduced expression u_p containing $2p+1$ unknowns writes:

$$\begin{aligned} u_p =& P_0(x,y,z) + \sum_{i=1}^{2p+1} P_i(x,y,z)v_i \\ =& P_0(x,y,z) + \langle \boldsymbol{P} \rangle \{\boldsymbol{v}\}. \end{aligned} \tag{5.7}$$

where the first polynomial $P_0(x,y,z)$ balances the right-hand side $f(\boldsymbol{x})$ of Eq. (5.1) and $\langle \boldsymbol{P} \rangle$ collects the complete family of solutions of the associated homogeneous problems, see [47, 116] for more details. As compared with Trefftz methods [35], the application to non-homogeneous and non-linear equations is straightforward.

5.2.2 Boundary least-square collocation

The polynomials u_p in Eq. (5.7) are quasi-exact solutions of the PDE, in such a way that the discretization concerns only the boundary. As proposed in [47, 48, 68], a collocation technique combined with the least-square method can be used to account for boundary conditions. Two sets of nodes $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ are

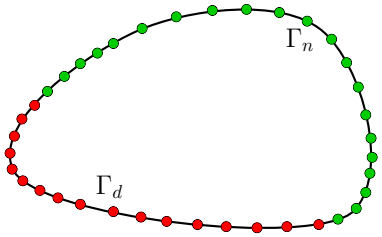


Figure 5.2: Sketch for the boundary collocation method.

chosen on Γ_d and Γ_n respectively (see Fig. 5.2). Then searching the variables $\{\boldsymbol{v}\}$ is equivalent to minimize the following function:

$$\mathscr{T}(\boldsymbol{v}) = \frac{1}{2}\sum_{\boldsymbol{x}_i\in\Gamma_d}\left|u_p(\boldsymbol{x}_i) - u^d(\boldsymbol{x}_i)\right|^2 + \omega\cdot\frac{1}{2}\sum_{\boldsymbol{x}_j\in\Gamma_n}\left|\mathbf{T}u_p(\boldsymbol{x}_j) - t^n(\boldsymbol{x}_j)\right|^2 \tag{5.8}$$

where ω permits to balance the two kinds of boundary conditions.

This minimization leads to a linear system $[\boldsymbol{K}]\{\boldsymbol{v}\} = \{\boldsymbol{b}\}$ where $[\boldsymbol{K}]$ is an invertible matrix. Solving this system gives the vector $\{\boldsymbol{v}\}$ and therefore the approximate solution of the problem Eq. (5.1). The detailed expressions of matrix $[\boldsymbol{K}]$ and the vector $\{\boldsymbol{b}\}$ write:

$$[\boldsymbol{K}]_{\alpha\beta} = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i)\cdot P_\beta(\boldsymbol{x}_i) + \omega\cdot\sum_{j=1}^{M_n}\mathbf{T}P_\alpha(\boldsymbol{x}_j)\cdot\mathbf{T}P_\beta(\boldsymbol{x}_j), \quad \alpha,\beta\in[1,2p+1] \tag{5.9}$$

$$\{\boldsymbol{b}\}_\alpha = \sum_{i=1}^{M_d} P_\alpha(\boldsymbol{x}_i)(u^d(\boldsymbol{x}_i) - P_0(\boldsymbol{x}_i)) + \omega\cdot\sum_{j=1}^{M_n}\mathbf{T}P_\alpha(\boldsymbol{x}_j)(t^n(\boldsymbol{x}_j) - \mathbf{T}P_0(\boldsymbol{x}_j)) \tag{5.10}$$

where M_d and M_n represent the number of collocation points on the Dirichlet boundary Γ_d and the Neumann boundary Γ_n respectively.

This simple least-square collocation procedure will be applied in this chapter, a single polynomial solution Eq. (5.7) being valid in the whole domain. Nevertheless the same collocation method works well also to discretize interface conditions in a multi-domain approach [116] and alternative techniques are available for the discretization of the interface [51]. There are also finite element-based alternatives within the Trefftz community [35, 133].

5.2.3 Convergence when the domain has a corner

What is the behavior of the latter method when the boundary is not smooth? Let us consider a problem of plane stress isotropic elasticity in a square domain with Dirichlet-Neumann boundary conditions as pictured in Fig. 5.3. The parameters are as follows: $L = B = 10$, $E = 1000$ (Young's modulus), $\nu = 0.3$ (Poisson's ratio) and $q = 100$. This problem has no analytical solution and we

tried to define a reference solution by the finite element method. We consider that a 500×500 mesh with $Q4$ elements leads to a sort of reference solution. Indeed with respect to a 250×250 mesh with $Q4$ elements or a 100×100 mesh with $Q8$ elements, the difference is less than $10^{-3.17}$. Throughout this chapter, the effectiveness of the proposed technique will be evaluated by the maximum relative error between the approximated solution u_p and the reference solution u_{ref} which is defined by:

$$\varepsilon = \frac{max\,|u_p(\boldsymbol{x}) - u_{ref}(\boldsymbol{x})|}{max\,|u_{ref}(\boldsymbol{x})|} \tag{5.11}$$

This elasticity problem has been solved by the Taylor method presented in the previous sections. In this case, it was not possible to get a TMM-solution as accurate as in the Fig. 5.1. The error decreases less quickly and there is a plateau of accuracy at about 1%, what is much larger than with problems having smooth solutions and larger than the difference between the two best FEM-results. More precisely the maximal error is about 4% for a degree $p = 10$, 1.5% for $p = 30$ and only 1% for $p = 50$. Note also that the accuracy is better in the center of the domain ($\simeq 0.03\%$). Clearly the worse convergence is due to the presence of the corners and this will be confirmed by forthcoming results.

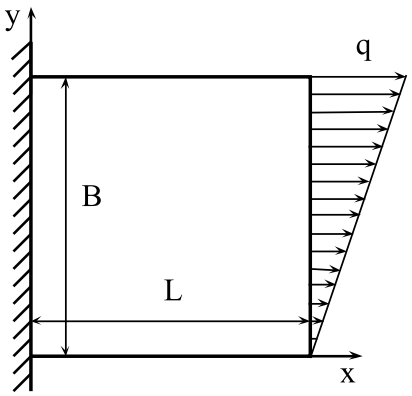


Figure 5.3: A two-dimensional elasticity problem.

5.2.4 A new TMM including singular shape functions

From the numerical test of section 5.2.3, one sees the difficulty to recover the property of exponential convergence when the boundary is not smooth. To establish undoubtedly the connection between presence of corner and poor

convergence, the Taylor series method of Sections 5.2.1 and 5.2.2 will be modified by including some singular shape functions. Indeed it has been mathematically proved a long time ago [145, 146] that the solution of an elliptic PDE in a domain with angular points can be split in two parts: a first part combining singular functions and a second part lying in a vectorial space of smoother functions. Thus it seems consistent to approximate this second part by a truncated Taylor series.

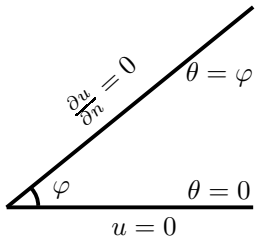


Figure 5.4: Laplace equation with Neumann-Dirichlet boundary conditions.

As it is well known, the family of singular solutions near a corner can be calculated analytically in many cases. For instance let us consider the Laplace equation $\Delta u = 0$ with Dirichlet-Neumann boundary conditions $u(r,0) = 0, \frac{\partial u(r,\varphi)}{\partial \theta} = 0$, cf Fig. 5.4. The general solution of $\Delta u = 0$ is in the form $u = Re\,[\phi(z)]$, where $z = x + iy = re^{i\theta}$ and $\phi(z)$ is a holomorphic function. Hence the singular solutions are in the form $\phi(z) = z^\alpha$ or $u(r,\theta) = r^\alpha sin(\alpha\theta)$ with account of the Dirichlet boundary condition (this solution is not smooth if α is not an integer). If one now introduces the Neumann boundary condition

$$\frac{\partial u(r,\varphi)}{\partial \theta} = \alpha r^\alpha cos(\alpha\varphi) = 0, \tag{5.12}$$

one gets a countable family of possible exponents α:

$$\alpha_j \varphi = \frac{\pi}{2} + (j-1)\pi, \quad j = 1, 2, 3, \cdots \tag{5.13}$$

and the expressions of the corresponding singular functions as follows

$$Q_j(r,\theta) = r^{\alpha_j} sin(\alpha_j \theta) \tag{5.14}$$

Such singular solutions can be built analytically for a number of PDEs and we refer to Appendix C in the case of 2D elasticity. Next the simple way to take them into account is to combine linearly the non-singular solutions in Eq. (5.7) and the singular ones in Eq. (5.14). In the case of a homogeneous problem, this leads to the following approximation:

$$u_p(\boldsymbol{x}) = \sum_{i=1}^{2p+1} v_i P_i(\boldsymbol{x}) + \sum_{j=1}^{n} v_{2p+1+j} Q_j(r,\theta) \tag{5.15}$$

Next we have to determine the $2p+1+n$ unknown coefficients v_i from the boundary conditions, what will be achieved by the same least-square collocation method described in section 5.2.2. Transmission conditions could be accounted in the same way in case of multi-domain discretization and there are alternative methods based on Lagrange multipliers [51, 116]. Note that the coefficients $\{v_j, j \geq 2p+2\}$ are often considered as quantities of interest. In fracture mechanics, the stress intensity factors belong to this set of coefficients.

5.3 Numerical applications

The modified Taylor method described in section 5.2.4 is now analyzed by considering four numerical benchmarks related to Laplace equation and 2D elasticity. Two of them have exact solutions, which permits to discuss the convergence up to a very high accuracy. These latter tests correspond to basic solutions of fracture mechanics. The two other cases are simple, but more generic, examples without exact solutions, which avoids some bias due to a too specific choice of the example. For these four simple cases a multi-domain method is not necessary: the approximation Eq. (5.15) holds in the whole domain.

5.3.1 Laplace equation with singularity

Here, we consider the Laplace equation in a domain that is a circular segment, as shown in Fig. 5.5. The left corner induces a rather weak singularity without infinite gradient at the corner ($\alpha_1 = 3/2$). The geometric parameters are as follows: $L_x = \sqrt{3}$, $L_y = 1$, $c = [\sqrt{3}, -1]$ and $\varphi = \pi/3$.

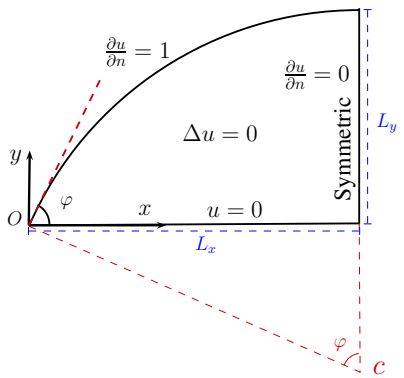
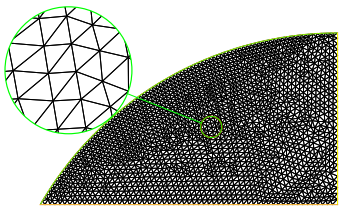
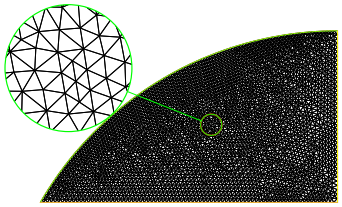


Figure 5.5: Laplace equation in a circular segment. The solution is singular at the left corner.

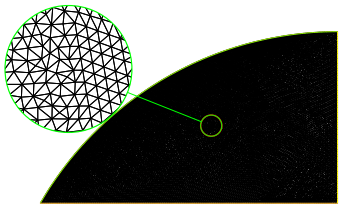
Since no analytical solution is available, a reference solution has been defined numerically by using the efficient open source code FreeFem++ [77]. Three different meshes with 6 nodes triangular elements were used as shown in Fig. 5.6. The finite element solution obtained by the most dense mesh (mesh three with 20576 P2 elements) is identified as a reference solution. Comparing mesh one (with 3810 P2 elements) and mesh two (with 6078 P2 elements) with mesh three respectively, one obtains an error equal to $10^{-4.7}$ and $10^{-4.62}$ respectively.



(a) Mesh one with 3810 elements



(b) Mesh two with 6078 elements



(c) Mesh three with 20576 elements

Figure 5.6: Three meshes of the finite element models for the Laplace problem described in Fig. 5.5.

The optimal number of collocation points has been carefully discussed in previous papers [51, 116]. Without singular functions, the number of degrees of freedom is $2p+1$ in 2D and $(p+1)^2$ in 3D and generally, the recommended number of collocation points is about the double. The influence of the number of collocation points is rediscussed here in the presence of singular shape functions and this is illustrated in Fig. 5.7. It appears that the error remains always stable from a critical number of collocation points that is a little higher than $4(p+n)$. In the following, we choose $6(p+n)$ collocation points to ensure the convergence.

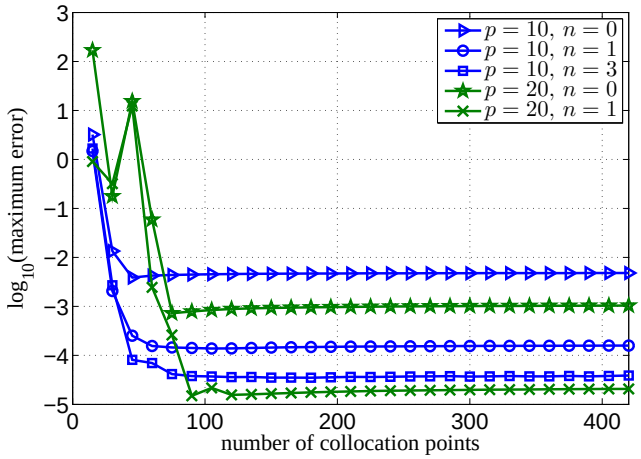


Figure 5.7: The influence of the number of collocation points for the Laplace problem in Fig. 5.5, according to the degree p and the number n of singular shape functions

When enough collocation points are adopted, the accuracy is mainly dominated by the degree of polynomial shape functions p and the number of singular shape functions n. The convergence with p and n is presented in Table 5.1. Without singular shape functions ($n = 0$), one converges slowly and reaches an accuracy of $10^{-3.8}$ for a large degree $p = 50$. Just one or two singular shape functions permit to gain two orders of magnitude: for instance an accuracy of 10^{-4} require 101 DOFs without singular functions ($n = 0, p = 50$) while only 22 are sufficient with one singular shape function ($n = 1, p = 10$). Of course such results are consistent with the state of art within fracture mechanics [136] or with alternative results in a meshfree framework [143].

Table 5.1: Convergence with the number of singular shape functions and the degree of non-singular polynomial shape functions for the Laplace problem defined in Fig. 5.5.

$log_{10}(\varepsilon)$	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$
$n = 0$	-2.3498	-3.0184	-3.4256	-3.6733	-3.8487
$n = 1$	-3.8493	-4.7543	-5.0174	-5.0227	-5.0246
$n = 3$	-4.4218	-4.6921	-5.0171	-5.0225	-5.0217
$n = 5$	-4.6823	-4.7546	-5.0174	-5.0221	-5.0196

5.3.2 Two tests from linear elastic fracture mechanics

The two next examples are basic benchmarks of fracture mechanics. The behavior near a crack tip is more singular than in the example of section

5.3.1, which means that the exponent α is smaller ($\alpha_1 = \alpha_2 = 1/2$) and the stress is infinite at the crack tip ($\sigma \sim r^{-1/2}$). The two first coefficients in the singular part, called stress intensity factors (SIFs), are very important in fracture mechanics and the accuracy with which these coefficients are obtained will be discussed. There is an ample literature about the computation of SIFs, but also of the full set of singular coefficients, see for instance [147]. In this paper, we are focused on the connection between singular shape functions and convergence of Taylor series, but the accuracy of the SIFs will be also shortly analyzed. Some informations about the analytical calculation of the singular functions are recalled in Appendix C for completeness.

In this section, two fracture mechanics problems are considered. They are designed to have an analytical solution, which permits to measure very high accuracies. The solution of the first one is nothing but the famous mode I (or crack opening mode) and it is one of the singular shape functions. As for the solution of the second problem, it comes from the problem of a cracked infinite plate under uniaxial tension. Note that the maximal error in Eq. (5.11) remains consistent for the displacement field that is continuous, but not for the stress that is not bounded at the crack tip. Nevertheless this maximal error will be applied on a cloud of 50×50 points uniformly distributed on the domain except at the crack tip, where the control point has been moved of 0.001 (typically from $x = 0.2$ to $x = 0.201$).

5.3.2.1 A test to recover the crack opening model

The physical model is two-dimensional plane strain elasticity. The famous mode I of fracture mechanics corresponds to the following stress field:

$$\begin{cases} \sigma_x = \dfrac{K_I}{\sqrt{2\pi r}} cos(\theta/2) \cdot [1 - sin(\theta/2) sin(3/2 \cdot \theta)] \\ \sigma_y = \dfrac{K_I}{\sqrt{2\pi r}} cos(\theta/2) \cdot [1 + sin(\theta/2) sin(3/2 \cdot \theta)] \\ \tau_{xy} = \dfrac{K_I}{\sqrt{2\pi r}} cos(\theta/2) \cdot sin(\theta/2) \cdot cos(3/2 \cdot \theta) \end{cases} \tag{5.16}$$

where $r = \sqrt{x^2 + y^2}$, $\theta = arctan(\frac{y}{x})$. The number K_I is the first stress intensity factor. The corresponding displacement field is:

$$\begin{cases} u = \dfrac{K_I}{2\mu} (\dfrac{r}{2\pi})^{1/2} [\kappa - cos(\theta)] \, cos(\theta/2) \\ v = \dfrac{K_I}{2\mu} (\dfrac{r}{2\pi})^{1/2} [\kappa - cos(\theta)] \, sin(\theta/2) \end{cases} \tag{5.17}$$

where $\kappa = 3 - 4\nu$, ν is the Poisson ratio and $\mu = E/[2(1 + \mu)]$. A mixed boundary value problem has been designed, see Fig. 5.8, whose solution is

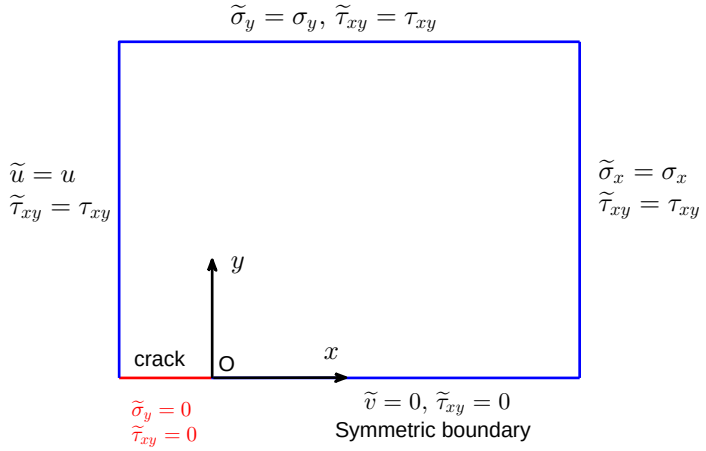


Figure 5.8: A boundary value problem to recover the crack opening mode.

exactly given by Eqs. (5.16) and (5.17). The numerical value of the Poisson ratio is $\nu = 0.3$ and the crack length is $a = 0.2$.

Table 5.2: Convergence with the degree p and the number of singular functions n for the mode I problem described in Fig. 5.8.

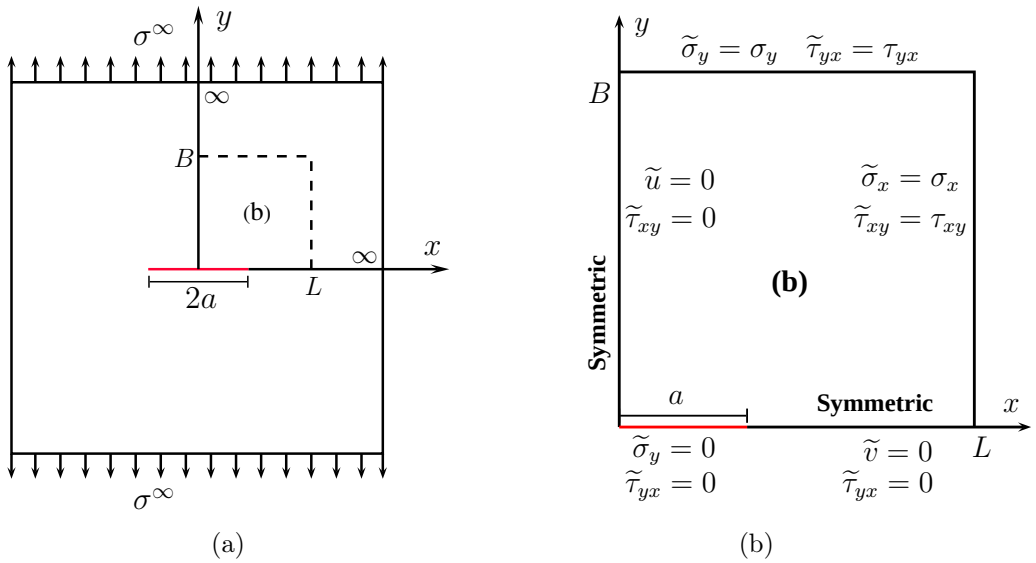
n	p	NDOFs	Error in displacement $log_{10}(\varepsilon)$	Error in stress $log_{10}(\varepsilon)$
	1	8	-15.1204	-15.2682
$n = 2$	10	44	-12.8060	-12.6139
	20	84	-11.2045	-10.1424
	1	12	-15.1804	-15.3474
$n = 6$	10	48	-12.5630	-12.2353
	20	88	-10.8113	-10.0309

The Taylor method of section 5.2.4 has been applied to this problem, without and with singular shape functions. As expected, it is not possible to get accurate solutions without singular solutions, the errors in displacement being always larger than 50%, even for large degrees $p = 30$ and $p = 50$: hence it seems difficult to use only polynomial shape functions to recover a singular solution. Some results with singular shape functions are presented in Table 5.2. When two singular shape functions ($\alpha_1 = \alpha_2 = 1/2$) and three polynomial shape functions (i.e. a degree $p = 1$) are used, the maximum relative errors on displacements and stresses are very small: $10^{-15.1}$ and $10^{-15.3}$ respectively. Such an error is very close to the unit round-off error $2^{-52} \approx 2.22 \times 10^{-16}$ for a single real number within the double-precision floating-

point format [101, 148]. If one increases the degree, the accuracy deteriorates slightly, because of the propagation of round-off errors. The same very specific scheme was also observed if one tries to get a polynomial solution by the pure Taylor method of section 5.2.1. This behavior is typical of problems where the exact solution lies in the vectorial space generated by the shape functions [116].

5.3.2.2 The infinite plate with a central crack under tensile stress

The second fracture benchmark emanates from the problem of an infinite medium with a horizontal central crack of length $2a$ and submitted to a uniaxial stress $\sigma^\infty \mathbf{e}_y \otimes \mathbf{e}_y$ at infinity, see Fig. 5.9a. The analytical solution is well known [149, 150]:



(a) (b)

Figure 5.9: (a) The infinite plate with a central crack and submitted to a unixial stress at infinity; (b) A boundary value problem in a finite domain whose solution is the same as for the infinite plate of Fig. 5.9a.

$$\begin{cases} u = \frac{\sigma^\infty}{2\mu}\left[\frac{(\kappa-1)}{2}(r_2^{1/2}cos(\theta_2/2) - \frac{1}{2}r_1 cos(\theta_1)) - y\cdot r_1 \cdot r_2^{-1/2} sin(\theta_1+\theta_2/2) - \frac{x}{2}\right] \\ v = \frac{\sigma^\infty}{2\mu}\left[\frac{-(\kappa+1)}{2}(r_2^{1/2}sin(\theta_2/2) + \frac{1}{2}r_1 sin(\theta_1)) - y\cdot r_1 \cdot r_2^{-1/2} cos(\theta_1+\theta_2/2) + y\right] \end{cases} \tag{5.18}$$

$$\begin{cases} \sigma_x = \sigma^\infty \cdot r_1 \cdot r_2^{-1/2}\left[cos(\theta_1+\theta_2/2) + a^2\cdot r_2{}^{-1}\cdot sin(\theta_1)sin(3/2\cdot\theta_2)\right] - \sigma^\infty \\ \sigma_y = \sigma^\infty \cdot r_1 \cdot r_2^{-1/2}\left[cos(\theta_1+\theta_2/2) - a^2\cdot r_2{}^{-1}\cdot sin(\theta_1)sin(3/2\cdot\theta_2)\right] \\ \tau_{xy} = \sigma^\infty \cdot a^2 \cdot r_1 \cdot r_2^{-1/2}\cdot sin(\theta_1)\cdot cos(3/2\cdot\theta_2) \end{cases} \tag{5.19}$$

where the r_1, θ_1 and r_2, θ_2 are defined as follows:

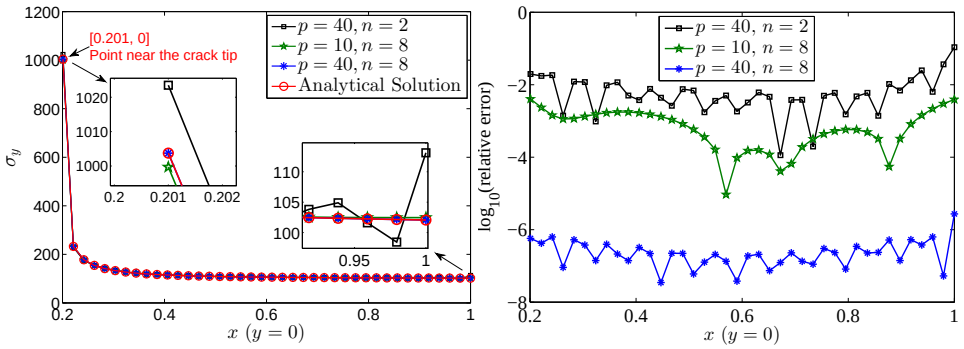
$$\begin{cases} r_1 = \sqrt{x^2+y^2}; & \theta_1 = arctan(\frac{y}{x}). \\ r_2 = \sqrt{(x^2-y^2-a^2)^2+(-2xy)^2}; & \theta_2 = arctan(\frac{-2xy}{x^2-y^2-a^2}). \end{cases} \tag{5.20}$$

The chosen parameters are as follows: $a = 0.2$, $\sigma^\infty = 100$, $\nu = 0.3$, $\kappa = 3 - 4\nu$, $E = 1000$ (Young's modulus) and $\mu = E/[2(1+\mu)]$. A square domain has been cut in the infinite domain (see Fig. 5.9a) and a corresponding boundary value problem has been posed in Fig. 5.9b so as to keep the same solution Eqs. (5.18) and (5.19).

The convergence with the degree p of the polynomial shape functions and the number n of singular shape functions is now discussed. The numerical solution is wrong without singular shape functions, the error being at least 50%. The Table 5.3 presents the obtained accuracy from two to eight singular functions. The introduction of two singular functions is sufficient to get an error of about 1%. Next the accuracy is easily improved by increasing the number of singular functions and the degree. For instance with a degree $p = 10$ (42 polynomials), one passes from an error of $10^{-1.4}$ for $n = 2$ to an error of $10^{-2.3}$ for $n = 6$ and finally to $10^{-3.3}$ for $n = 20$. There is a plateau of accuracy at a high level (10^{-5} or 10^{-6}). Moreover the error in stress and Stress Intensity Factor has about the same order of magnitude as for the displacement. The convergence with these two parameters p and n is also illustrated in Fig. 5.10, where the distribution of transverse stress σ_y is plotted: one sees clearly the strong improvement obtained by increasing the number of singular functions from $n = 2$ to $n = 8$.

Table 5.3: Convergence with the degree p and the number of singular functions n for the cracked plate problem described in Fig. 5.10.

n	p	Error in displacements $log_{10}(\varepsilon)$	Error in stresses $log_{10}(\varepsilon)$	Error in SIFs $log_{10}(\varepsilon)$
$n = 2$	10	-1.4137	-1.3121	-1.5543
	30	-2.0785	-1.6041	-1.5914
	50	-2.2081	-1.5592	-1.7093
$n = 6$	10	-2.3236	-1.6326	-2.5541
	30	-4.7309	-4.2851	-4.4160
	50	-5.8604	-5.1474	-5.0110
$n = 8$	10	-2.4972	-1.7541	-2.4210
	30	-5.0535	-4.5884	-5.8802
	40	-6.2975	-5.6968	-6.0031
	50	-5.6359	-4.8222	-5.8995



(a) (b)

Figure 5.10: The distribution of transverse stress σ_y and of the corresponding error along the horizontal line $y = 0$.

5.3.3 Application in two-dimensional elasticity

Let us come back to the 2D elasticity problem of Fig. 5.3. The accuracy of TMM without singular functions was not better than 1%. The accuracy of the vertical displacement v with two and four singular functions is reported in Table 5.4. Two types of singular functions are used: n_1 and n_2 singular functions located at upper-left and bottom-left corners respectively. At least one order of magnitude is gained simply by adding one singular function at each corner ($n_1 = n_2 = 1$), the error being reduced to 0.035% for a degree $p = 30$ and for $n_1 = n_2 = 1$. This establishes once more the connection between singularities and bad convergence of Taylor series, as well as a simple manner to get tremendous improvements of this convergence. Note that the maximal error mentioned in the Table ($10^{-3.67}$) may be due as well to FEM as to TMM.

Table 5.4: Convergence with degree in the presence of few singular functions for the square elasticity shown in Fig. 5.3.

n	p	Error of v ($log_{10}(\varepsilon)$)
$n_1 = n_2 = 1$	10	-2.4695
	30	-3.4594
	50	-3.5111
$n_1 = n_2 = 2$	10	-2.5001
	30	-3.4637
	50	-3.6689

5.4 Conclusion

A boundary collocation meshless method involving high order polynomial shape functions and singular shape functions was proposed in this chapter to solve problems with singularities. Since these shape functions are the approximated solutions of the considered problems, the discretization concerns only the boundary and least-square collocation is employed to determine the unknown coefficients. Based on previous research, the proposed technique without singular shape functions always works well when the solution of a considered problem is smooth, while it fails to recover the solution of problem with singularities. So we wonder the probability of solving singular problem effectively by introducing the singular shape functions. It was well established [145] that the solution of a problem in an angular domain can be split in a smooth part and a linear combination of singular functions: in the present method, the smooth part is discretized by Taylor series.

Few benchmark tests established that the accuracy of the approximated solution is widely improved by introducing few singular shape functions. When enough collocation points are chosen, the accuracy is dominated by two factors: the degree of polynomial shape functions and the number of singular shape functions. For a fixed number of singular shape functions, the proposed technique converges with the degree of polynomial shape functions. And for a fixed degree of polynomial shape functions, the accuracy increases with the number of singular shape functions.

Chapter 6

Conclusion and perspectives

Concluding remarks

A boundary collocation meshless method based on Taylor series, named Taylor Meshless Method (TMM), has been discussed in this thesis. The specificity of TMM is the numerical computation of a family of polynomial shape functions that are solutions of the considered PDE in the sense of Taylor series. Since the PDE has been satisfied inside the domain, the discretization concerns only the boundary. Then collocation technique associated with least-square method has been employed to determine the unknown coefficients.

In Chapter 2, the interest has been focused on the treatment of boundary and interface conditions in piecewise resolutions. Least-square collocation has been validated for boundary conditions while bridging techniques based on Lagrange multipliers have been validated for transmission conditions by Tampango et al. [51]. In this chapter, we verified the possibility of using Lagrange multiplier method to account for boundary conditions and re-visiting least-square collocation method to account for interface conditions. Based on the numerical results, least-square collocation is confirmed to be very easy, efficient and robust for accounting for both boundary conditions and interface conditions.

In chapter 3, the application of TMM has been extended to three dimensions. The computation time and ill-conditioning have been well analyzed. The results show that the accuracy of TMM is governed by two antagonistic effects: exponential convergence of Taylor series with the degree while propagation of round-off errors facilitated by ill-conditioned matrix increases the error. As for computation time, the most time-consuming part in TMM is the calculation of shape functions at all collocation points and this tedious work can be issued by parallel computing. The considered large-scale cases show the possibility of combining a rather large degree of Taylor series and many sub-domains.

In chapter 4, TMM associated with Newton method and Automatic Differentiation is proposed to solve non-linear elliptic PDEs. The numerical tests demonstrate that non-linear TMM keeps the same exponential convergence and robustness features as the linear one. Non-linear numerical applications

in one dimension and three dimensions illustrate that the maximum relative error decreases exponentially with the refinement of discretization of TMM and the number of Newton iterations.

In Chapter 5, singular shape functions are introduced in TMM to account for problems with singularities. It was well established that the solution of a problem in an angular domain can be split into a smooth part and a linear combination of singular functions. In this chapter, the polynomial shape functions deduced from Taylor series were employed to account for the smooth part solutions while singular part solutions were recovered by using a few analytically known singular solutions. Based on the numerical tests, the accuracy of the approximated solution is widely improved by introducing a few singular shape functions.

In this thesis, the applications of TMM are extended from two dimensions to three dimensions, from linear cases to non-linear cases, from one Taylor series or a few Taylor series to hundreds of sub-domains. TMM is confirmed to be efficient, fast convergent and robust.

Future perspective

The present thesis leaves a number of challenging open issues that are interesting to address in future research.

As mentioned previously, the most time-consuming part in TMM is the calculation of shape functions at all collocation points. Fortunately, since the shape functions of different collocation points are independent of each other, this time-consuming part can be realized by using parallel computing. Only small-scale parallel computing is tested in the present thesis. With the rapid development of the computing technique, larger-scale parallel computing is worthy of further study.

Although TMM has many advantages, for instance the exponential convergence, the strong reduction of the number of degrees of freedom and the advantages what meshles methods possess, it is not as versatile as the classical numerical methods, for instance finite element method. One may wonder the possibility of combining TMM with FEM and play their respective advantages. This will further extend the applications of TMM and ensure the accuracy of approximations while improving the efficiency of computing.

Although the algorithm discussed in this thesis is very simple, it is still difficult for potential users to solve practical engineering problems. If TMM is combined with graphics software and packaged into a user-oriented program software, the more value of TMM will be achieved.

Chapter 7

Appendix

Appendix A
Discretization of the Laplace operator

Few formulae are presented to define the matrices in section 3.2.

$$[L^k] = [G_x^{k+1}][G_x^{k+2}] + [G_y^{k+1}][G_y^{k+2}] + [G_z^{k+1}][G_z^{k+2}] \tag{1}$$

where

$$[G_x^k] = \begin{bmatrix} G_{xe}^k & 0 & 0 & \cdots & 0 & 0 \\ 0 & G_{xe}^{k-1} & 0 & \cdots & 0 & 0 \\ 0 & 0 & G_{xe}^{k-2} & & 0 & 0 \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & G_{xe}^1 & 0 \end{bmatrix} \tag{2}$$

$$[G_{xe}^k] = \begin{bmatrix} k & 0 & 0 & \cdots & 0 & 0 \\ 0 & k-1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & k-2 & & 0 & 0 \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & 1 & 0 \end{bmatrix} \tag{3}$$

$$[G_y^k] = \left[\begin{array}{c|cccc} & \mathbf{I}_k & 0 & 0 & \cdots & 0 \\ & 0 & 2\mathbf{I}_{k-1} & 0 & \cdots & 0 \\ 0_{p(p+1)/2} & 0 & 0 & 3\mathbf{I}_{k-2} & & 0 \\ & \vdots & \vdots & & \ddots & \\ & 0 & 0 & 0 & & k\mathbf{I}_1 \end{array}\right] \tag{4}$$

$$[G_z^k] = \begin{bmatrix} G_{ze}^k & 0 & 0 & \cdots & 0 & 0 \\ 0 & G_{ze}^{k-1} & 0 & \cdots & 0 & 0 \\ 0 & 0 & G_{ze}^{k-2} & & 0 & 0 \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & G_{ze}^1 & 0 \end{bmatrix} \tag{5}$$

$$[G_{ze}^k] = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 3 & & 0 \\ \vdots & \vdots & \vdots & & \ddots & \\ 0 & 0 & 0 & 0 & & k \end{bmatrix} \tag{6}$$

Appendix B
TMM for ODE

Here, we consider an ODE:

$$-u''(x) + g(x)u(x) = f(x) \tag{7}$$

In Eq.(7), the functions $g(x)$ and $f(x)$ are known. We expand $g(x)$ and $f(x)$ by Taylor series up to the degree $p-2$ as follows.

$$\begin{cases} g(x) = \sum_{i=0}^{p-2} g_i x^i + O(x^{p-1}) \\ f(x) = \sum_{i=0}^{p-2} f_i x^i + O(x^{p-1}) \end{cases} \tag{8}$$

where g_i and f_i are known coefficients.

The approximate solution of Eq.(7) is sought in the form of a polynomial of degree p:

$$u(x) = \sum_{i=0}^{p} u_i x^i \tag{9}$$

we have the following expression of derivative respect to $u(x)$:

$$u''(x) = \sum_{n=0}^{p-2} (n+2)(n+1) u_{n+2} x^n \tag{10}$$

The product $g(x)u(x)$ is truncated at the order p:

$$g(x)u(x) = \sum_{n=0}^{p-2} (\sum_{i=0}^{n} g_{n-i} u_i) x^n + O(x^{p-1}) \tag{11}$$

If the Eq.(7) is satisfied, one can get the following relationship:

$$-(i+2)(i+1)u_{i+2} + \sum_{j=0}^{i} g_{i-j} u_j - f_i = 0, \quad \forall i, \quad 0 \leqslant i \leqslant p-2 \tag{12}$$

There are $p-1$ equations in Eq.(12) for $p+1$ unknowns $(u_i, i = 0, 1, \cdots, p)$: u_0 and u_1 are chosen as the independent unknowns. One can find the relationship between u_n and u_0, u_1:

$$u_n = \gamma_n + \alpha_n u_0 + \beta_n u_1, \quad 0 \leqslant n \leqslant p \tag{13}$$

Substituting Eq.(13) into Eq.(12), one can get:

$$n(n-1)u_n = (-f_{n-2} + \sum_{j=0}^{n-2} g_{n-j-2}\gamma_j) + (\sum_{j=0}^{n-2} g_{n-j-2}\alpha_j)u_0 + (\sum_{j=0}^{n-2} g_{n-j-2}\beta_j)u_1 \tag{14}$$

Comparing Eq.(13) with Eq.(14), one can get the iterative relationships for γ_n, α_n and β_n.

$$\begin{cases} \alpha_0 = 1, \quad \alpha_1 = 0, \\ \alpha_i = \dfrac{1}{i(i-1)} \sum\limits_{j=0}^{i-2} g_{i-j-2}\alpha_j \quad \forall i,\ 2 \leqslant i \leqslant p; \\ \beta_0 = 0, \quad \beta_1 = 1, \\ \beta_i = \dfrac{1}{i(i-1)} \sum\limits_{j=0}^{i-2} g_{i-j-2}\beta_j \quad \forall i,\ 2 \leqslant i \leqslant p; \\ \gamma_0 = 0, \quad \gamma_1 = 0, \\ \gamma_i = \dfrac{1}{i(i-1)}(-f_{i-2} + \sum\limits_{j=0}^{i-2} g_{i-j-2}\gamma_j) \quad \forall i,\ 2 \leqslant i \leqslant p. \end{cases} \tag{15}$$

Then, $u(x)$ can be written as

$$u(x) = P_0 + \langle P \rangle \{v\} = P_0 + P_1 u_0 + P_2 u_1 \tag{16}$$

where $P_0 = \sum\limits_{i=2}^{p} \gamma_i x^i$, $P_1 = 1 + \sum\limits_{i=2}^{p} \alpha_i x^i$ and $P_2 = x + \sum\limits_{i=2}^{p} \beta_i x^i$.

So the approximate solution of the problem is completely obtained as a function of u_0 and u_1. Last the two unknowns u_0 and u_1 can be deduced from the boundary conditions.

$$[S^{vk}] = \begin{bmatrix} \mathbf{I}_{(2k+1)\times(2k+1)} \\ \mathbf{0}_{k(k-1)/2\times(2k+1)} \end{bmatrix} \tag{17}$$

$$[S^{wk}] = \begin{bmatrix} \mathbf{0}_{(2k+1)\times k(k-1)/2} \\ \mathbf{I}_{k(k-1)/2\times k(k-1)/2} \end{bmatrix} \tag{18}$$

Appendix C

Singular solutions of opening crack

In this Appendix, we compute analytically the whole set of singular functions in the case of a crack tip. This is based on the well-known complex variable method and we refer to [150] for the details. A similar approach can be done for the square domain problem for Fig. 5.3.

The Airy stress function U satisfying the biharmonic equation $\Delta^2 U = 0$ reads:

$$U = Re\left[\overline{z}\varphi(z) + \chi(z)\right] \tag{19}$$

where $z = x + iy$ and both $\varphi(z)$ and $\chi(z)$ are harmonic functions.

The complex representation of displacements denotes

$$2\mu(u + iv) = \kappa\varphi(z) - z\overline{\varphi'(z)} - \overline{\chi'(z)} \tag{20}$$

where ν and μ represent the Poisson's ratio and the shear modulus respectively. The number κ is a constant related to ν and it takes the value $3 - 4\nu$ for plane strain while it takes $(3-\nu)/(1+\nu)$ for generalized plane stress. The complex representations of stresses denote

$$\begin{cases} \sigma_x + \sigma_y = 2\left[\varphi'(z) + \overline{\varphi'(z)}\right] \\ \sigma_y - \sigma_x + 2i\tau_{xy} = 2\left[\overline{z}\varphi''(z) + \psi'(z)\right] \end{cases} \tag{21}$$

The force acting on an arc AB then can be expressed by:

$$X + iY = -i\left[\varphi(z) + z\overline{\varphi'(z)} + \overline{\psi(z)}\right]_A^B \tag{22}$$

where $\psi(z) = \chi'(z)$.

One sets the harmonic functions as follows: $\varphi(z) = c_1 z^\alpha$ and $\psi(z) = c_2 z^\alpha$, where α is a real number and c_1, c_2 are unknown complex numbers determined by the boundary conditions on the upper and lower surfaces of the crack tip. The force $(X + iY)$ vanishes along the crack tip, hence the Eq. (22) satisfies the following conditions:

$$\varphi(z) + z\overline{\varphi'(z)} + \overline{\psi(z)} = 0, \quad \theta = \pm\pi \tag{23}$$

Following relationships can be obtained from Eq. (23):

$$\begin{cases} c_1 e^{i\alpha\pi} + \overline{c_1}\alpha e^{-i\alpha\pi} + \overline{c_2}\alpha e^{-i\alpha\pi} = 0 \\ c_1 e^{-i\alpha\pi} + \overline{c_1}\alpha e^{i\alpha\pi} + \overline{c_2}\alpha e^{i\alpha\pi} = 0 \end{cases} \tag{24}$$

Considering Eq. (24), one can get

$$e^{i\cdot 4\alpha\pi} = 1 \tag{25}$$

which means $\alpha = \frac{1}{2}, 1, \frac{3}{2}, 2, \frac{5}{2}, \cdots$. To ensure the singularity of the approximation, the integer values of α should be removed. Finally $\alpha = \frac{1}{2}, \frac{3}{2}, \frac{5}{2}, \cdots$ are obtained.

Considering the value of α deduced from the boundary conditions, one can get the following relationship between c_1 and c_2 by Eq. (24):

$$c_2 = -c_1\alpha + \overline{c_1} \tag{26}$$

If c_1 is set to be 1 and i respectively, these corresponding values of c_2 are as follows:

$$\begin{cases} c_1 = 1, \quad c_2 = 1 - \alpha \\ c_1 = i, \quad c_2 = -i(1+\alpha) \end{cases} \tag{27}$$

Substituting the values of c_1 and c_2 in Eqs. (20) and (21), one can obtain the expressions of displacements and stresses as follows:

• The first set of singular solutions, $c_1 = 1$ and $c_2 = 1 - \alpha$:

$$\begin{cases} u = \frac{1}{2\mu} r^{\alpha} \left[(\kappa - 1 + \alpha) \cdot cos(\alpha\theta) - \alpha \cdot cos\left((2-\alpha)\theta\right) \right] \\ v = \frac{1}{2\mu} r^{\alpha} \left[(\kappa + 1 - \alpha) \cdot sin(\alpha\theta) - \alpha \cdot sin\left((2-\alpha)\theta\right) \right] \end{cases} \tag{28}$$

$$\begin{cases} \sigma_x = r^{\alpha-1} \cdot \alpha \cdot \left[(\alpha+1)cos((\alpha-1)\theta) - (\alpha-1)cos((\alpha-3)\theta) \right] \\ \sigma_y = r^{\alpha-1} \cdot \alpha \cdot \left[(3-\alpha)cos((\alpha-1)\theta) + (\alpha-1)cos((\alpha-3)\theta) \right] \\ \tau_{xy} = r^{\alpha-1} \cdot \alpha \cdot (\alpha-1) \left[sin((\alpha-3)\theta) - sin((\alpha-1)\theta) \right] \end{cases} \tag{29}$$

• The second set of singular solutions, $c_1 = i$ and $c_2 = -i(1+\alpha)$:

$$\begin{cases} u = \frac{-1}{2\mu} r^{\alpha} \left[(\kappa + 1 + \alpha) \cdot sin(\alpha\theta) + \alpha \cdot sin\left((2-\alpha)\theta\right) \right] \\ v = \frac{1}{2\mu} r^{\alpha} \left[(\kappa - 1 - \alpha) \cdot cos(\alpha\theta) + \alpha \cdot cos\left((2-\alpha)\theta\right) \right] \end{cases} \tag{30}$$

$$\begin{cases} \sigma_x = r^{\alpha-1} \cdot \alpha \cdot \left[(\alpha-1)sin((\alpha-3)\theta) - (\alpha+3)cos((\alpha-1)\theta) \right] \\ \sigma_y = r^{\alpha-1} \cdot \alpha \cdot (\alpha-1) \left[sin((\alpha-1)\theta) - sin((\alpha-3)\theta) \right] \\ \tau_{xy} = r^{\alpha-1} \cdot \alpha \cdot \left[(\alpha-1)cos((\alpha-3)\theta) - (\alpha+1)sin((\alpha-1)\theta) \right] \end{cases} \tag{31}$$

Selecting one value of α, there will be two sets of singular solutions. The way to deduce the non-singular shape functions from Taylor series has been introduced in section 5.2.1. The linear combination of the non-singular shape functions and singular solutions shown in Eqs. (28) and (30) leads to the approximations of this considered problem. Since the approximations satisfy the control equations, only boundary conditions need to be considered to determine the unknown coefficients. This last step is realized by using least-square collocation technique as shown in section 5.2.2.

Bibliography

[1] O. C. Zienkiewicz, R. L. Taylor, The finite element method (Fifth edition) Volume 1: The Basis, Oxford: Butterworth-Heinemann, 2000. (Cited on page 2.)

[2] K. J. Bathe, Finite element method, Wiley Online Library, 2008. (Cited on page 2.)

[3] C. A. Brebbia, The boundary element method for engineers, Pentech Pr, 1980. (Cited on page 2.)

[4] J. W. Thomas, Numerical partial differential equations: finite difference methods, Vol. 22, Springer Science & Business Media, 2013. (Cited on page 2.)

[5] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly Notices of the Royal Astronomical Society 181 (3) (1977) 375–389. (Cited on pages 4 and 5.)

[6] L. B. Lucy, A numerical approach to the testing of the fission hypothesis, Astronomical Journal 82 (82) (1977) 1013–1024. (Cited on pages 4 and 5.)

[7] B. Nayroles, G. Touzot, P. Villon, Generalizing the finite element method: Diffuse approximation and diffuse elements, Computational Mechanics 10 (5) (1992) 307–318. (Cited on pages 4 and 5.)

[8] T. Belytschko, Y. Y. Lu, L. Gu, Element-free Galerkin methods, International Journal for Numerical Methods in Engineering 37 (2) (1994) 229–256. (Cited on pages 4, 5, 15, 49, 50 and 79.)

[9] W. K. Liu, S. Jun, Y. F. Zhang, Reproducing kernel particle methods, International Journal for Numerical Methods in Fluids 20 (8-9) (1995) 1081–1106. (Cited on pages 4 and 5.)

[10] P. Lancaster, K. Salkauskas, Surfaces generated by moving least squares methods, Mathematics of Computation 37 (155) (1981) 141–141. (Cited on page 4.)

[11] D. H. Mclain, Drawing contours from arbitrary data points, The Computer Journal 17 (4) (1974) 318–324. (Cited on page 4.)

[12] W. J. Gordon, J. A. Wixom, Shepard's method of "Metric Interpolatio" to bivariate and multivariate interpolation, Mathematics of Computation 32 (141) (1978) 253–264. (Cited on page 4.)

[13] C. A. Duarte, J. T. Oden, Hp clouds: A meshless method to solve boundary-value problems, in: Technical Report, Austin: University of Texas, 1995, pp. 95–105. (Cited on pages 4 and 5.)

[14] R. L. Hardy, Multiquadric equations of topography and other irregular surfaces, Journal of Geophysical Research 76 (8) (1971) 1905–1915. (Cited on page 4.)

[15] R. Franke, Scattered data interpolation: tests of some methods, Mathematics of Computation 38 (157) (1982) 181–200. (Cited on page 4.)

[16] C. A. Micchelli, Interpolation of scattered data: distance matrices and conditionally positive definite functions, in: Approximation theory and spline functions, Springer, 1984, pp. 143–145. (Cited on page 4.)

[17] M. D. Buhmann, Multivariable interpolation using radial basis functions., Ph.D. thesis, University of Cambridge (1989). (Cited on page 4.)

[18] R. L. Hardy, Theory and applications of the multiquadric-biharmonic method 20 years of discovery, Computers & Mathematics with Applications 19 (8-9) (1990) 163–208. (Cited on page 4.)

[19] G. Liu, Y. Gu, A local radial point interpolation method (lrpim) for free vibration analyses of 2-d solids, Journal of Sound and Vibration 246 (1) (2001) 29–46. (Cited on page 4.)

[20] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, P. Krysl, Meshless methods: an overview and recent developments, Computer Methods in Applied Mechanics and Engineering 139 (1-4) (1996) 3–47. (Cited on page 5.)

[21] G. R. Liu, Meshfree methods: moving beyond the finite element method, Taylor & Francis, 2009. (Cited on page 5.)

[22] W. K. Liu, Y. Chen, Wavelet and multiple scale reproducing kernel methods, International Journal for Numerical Methods in Fluids 21 (10) (1995) 901–931. (Cited on page 5.)

[23] S. P. Shen, S. N. Atluri, The meshless local petrov-galerkin (mlpg) method: A simple & less-costly alternative to the finite element and boundary element methods, Computer Modeling in Engineering & Sciences 3 (2002) 11–51. (Cited on page 5.)

[24] N. R. Aluru, A point collocation method based on reproducing kernel approximations, International Journal for Numerical Methods in Engineering 47 (6) (2000) 1083–1121. (Cited on page 5.)

[25] S. N. Atluri, T. Zhu, A new meshless local Petrov-Galerkin (MLPG) approach, Computational Mechanics 22 (2) (1998) 117–127. (Cited on page 5.)

[26] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor, A finite point method in computational mechanics. Applications to convective transport and fluid flow, International Journal for Numerical Methods in Engineering 39 (22) (1996) 3839–3866. (Cited on page 5.)

[27] X. Zhang, X. H. Liu, K. Z. Song, M. W. Lu, Least-squares collocation meshless method, International Journal for Numerical Methods in Engineering 51 (9) (2001) 1089–1100. (Cited on pages 5 and 7.)

[28] H. Wendland, Meshless Galerkin methods using radial basis functions, Mathematics of Computation of the American Mathematical Society 68 (228) (1999) 1521–1531. (Cited on page 5.)

[29] E. J. Kansa, Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics-I surface approximations and partial derivative estimates, Computers & Mathematics with Applications 19 (8) (1990) 127–145. (Cited on pages 5, 15, 18, 49, 50 and 79.)

[30] E. J. Kansa, Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics-II solutions to parabolic, hyperbolic and elliptic partial differential equations, Computers & Mathematics with Applications 19 (8) (1990) 147–161. (Cited on pages 5, 15, 18 and 79.)

[31] W. Chen, M. Tanaka, A meshless, integration-free, and boundary-only RBF technique, Computers & Mathematics with Applications 43 (3-5) (2002) 379–391. (Cited on pages 5, 6 and 79.)

[32] T. J. Liszka, C. A. M. Duarte, W. W. Tworzydlo, hp-meshless cloud method, Computer Methods in Applied Mechanics and Engineering 139 (1-4) (1996) 263–288. (Cited on page 5.)

[33] G. R. Liu, Y. T. Gu, A local point interpolation method for stress analysis of two-dimensional solids, Structural Engineering and Mechanics 11 (2) (2001) 221–236. (Cited on page 5.)

[34] I. Herrera, Trefftz Method. In: Brebbia C.A. (eds) Basic Principles and Applications. Topics in Boundary Element Research, Vol. 1, Springer, Berlin, Heidelberg, 1984. (Cited on page 5.)

[35] E. Kita, N. Kamiya, Trefftz method: an overview, Advances in Engineering Software 24 (1-3) (1995) 3–12. (Cited on pages 5, 105, 108 and 109.)

[36] J. Jirousek, A. Zieliński, Survey of trefftz-type element formulations, Computers & structures 63 (2) (1997) 225–242. (Cited on page 5.)

[37] Q.-H. Qin, Formulation of hybrid trefftz finite element method for elastoplasticity, Applied Mathematical Modelling 29 (3) (2005) 235–252. (Cited on page 5.)

[38] Z.-C. Li, T.-T. Lu, H.-Y. Hu, A. H. Cheng, Trefftz and collocation methods, WIT press, 2008. (Cited on page 5.)

[39] J. A. Kołodziej, A. Zielinski, Boundary collocation techniques and their application in engineering, WIT Press, 2009. (Cited on page 5.)

[40] Y. X. Mukherjee, S. Mukherjee, The boundary node method for potential problems, International Journal for Numerical Methods in Engineering 40 (5) (1997) 797–815. (Cited on page 5.)

[41] K. M. Liew, Y. M. Cheng, S. Kitipornchai, Boundary element-free method (BEFM) and its application to two-dimensional elasticity problems, International Journal for Numerical Methods in Engineering 65 (8) (2006) 1310–1332. (Cited on page 5.)

[42] G. R. Liu, Y. T. Gu, Boundary meshfree methods based on the boundary point interpolation methods, Engineering Analysis with Boundary Elements 28 (5) (2004) 475–487. (Cited on page 5.)

[43] T. L. Zhu, J. D. Zhang, S. N. Atluri, A local boundary integral equation (LBIE) method in computational mechanics, and a meshless discretization approach, Computational Mechanics 21 (3) (1998) 223–235. (Cited on page 5.)

[44] V. D. Kupradze, M. A. Aleksidze, The method of functional equations for the approximate solution of certain boundary value problems, USSR Computational Mathematics and Mathematical Physics 4 (4) (1964) 82–126. (Cited on page 6.)

[45] D. L. Young, K. H. Chen, C. W. Lee, Novel meshless method for solving the potential problems with arbitrary domain, Journal of Computational Physics 209 (1) (2005) 290–321. (Cited on page 6.)

[46] W. Chen, Singular boundary method: a novel, simple, meshfree, boundary collocation numerical method, Chinese Journal of Solid Mechanics 30 (6) (2009) 592–599. (Cited on page 6.)

[47] D. S. Zézé, M. Potier-Ferry, N. Damil, A boundary meshless method with shape functions computed from the PDE, Engineering Analysis with Boundary Elements 34 (8) (2010) 747–754. (Cited on pages 6, 10, 11, 15, 16, 20, 28, 45, 49, 51, 54, 61, 79, 80, 81, 91, 105 and 108.)

[48] G. Fairweather, A. Karageorghis, The method of fundamental solutions for elliptic boundary value problems, Advances in Computational Mathematics 9 (1-2) (1998) 69–95. (Cited on pages 7, 15, 17, 49, 54, 79, 83 and 108.)

[49] Y. Tampango, M. Potier-Ferry, Y. Koutsawa, S. Belouettar, A new meshless method using Taylor series to solve elasticity problems, European Journal of Computational Mechanics 21 (3-6) (2012) 365–373. (Cited on pages 10, 15, 36, 61, 80, 81 and 91.)

[50] Y. Tampango, M. Potier-Ferry, Y. Koutsawa, S. Belouettar, Convergence analysis and detection of singularities within a boundary meshless method based on Taylor series, Engineering Analysis with Boundary Elements 36 (10) (2012) 1465–1472. (Cited on pages 10, 11, 15, 20, 36, 49, 61, 62, 67, 79, 80, 81, 91 and 105.)

[51] Y. Tampango, M. Potier-Ferry, Y. Koutsawa, S. Tiem, Coupling of polynomial approximations with application to a boundary meshless method, International Journal for Numerical Methods in Engineering 95 (13) (2013) 1094–1112. (Cited on pages 10, 13, 15, 16, 18, 19, 20, 22, 29, 30, 32, 34, 35, 43, 45, 49, 50, 51, 55, 56, 61, 64, 65, 80, 81, 84, 88, 91, 105, 109, 112, 113 and 121.)

[52] D. Zézé, Calcul de fonctions de forme de haut degré par une technique de perturbation (Computing high degree shape functions by a perturbation technique), Ph.D. thesis, Université Paul Verlaine Metz (2009). (Cited on pages 11, 15, 16, 17, 20, 50, 76 and 80.)

[53] Y. Tampango, Développement d'une méthode sans maillage utilisant les approximations de Taylor (Development of a meshless method using

Taylor approximations), Ph.D. thesis, Université de Lorraine (2012). (Cited on pages 11 and 80.)

[54] H. Akpama, Amélioration de la "Taylor meshless method" et application à des structures élastiques en grandes transformations, Master's thesis, Université de Lorraine (2013). (Cited on pages 11, 15, 17, 36, 45, 76 and 80.)

[55] I. Babuska, B. A. Szabo, I. N. Katz, The p-version of the finite element method, SIAM Journal on Numerical Analysis 18 (3) (1981) 515–545. (Cited on page 15.)

[56] C. Schwab, M. Suri, The p and hp versions of the finite element method for problems with boundary layers, Mathematics of Computation 65 (216) (1996) 1403–1429. (Cited on page 15.)

[57] B. Nayroles, G. Touzot, P. Villon, Generalizing the finite element method: diffuse approximation and diffuse elements, Computational Mechanics 10 (5) (1992) 307–318. (Cited on pages 15, 50 and 79.)

[58] V. P. Nguyen, T. Rabczuk, S. Bordas, M. Duflot, Meshless methods: a review and computer implementation aspects, Mathematics and Computers in Simulation 79 (3) (2008) 763–813. (Cited on pages 15 and 50.)

[59] M. A. Golberg, The method of fundamental solutions for Poisson's equation, Engineering Analysis with Boundary Elements 16 (3) (1995) 205–213. (Cited on page 15.)

[60] M. A. Golberg, C. S. Chen, The method of fundamental solutions for potential, helmholtz and diffusion problems, Boundary Integral Methods−Numerical and Mathematical Aspects (1998) 103–176. (Cited on page 15.)

[61] A. Frangi, M. Guiggiani, A direct approach for boundary integral equations with high-order singularities, International Journal for Numerical Methods in Engineering 49 (7) (2000) 871–898. (Cited on page 15.)

[62] A. Frangi, G. Novati, R. Springhetti, M. Rovizzi, 3d fracture analysis by the symmetric Galerkin BEM, Computational Mechanics 28 (3-4) (2002) 220–232. (Cited on page 15.)

[63] E. J. Kansa, Y. C. Hon, Circumventing the ill-conditioning problem with multiquadric radial basis functions: applications to elliptic partial differential equations, Computers & Mathematics with Applications 39 (7) (2000) 123–137. (Cited on pages 15 and 50.)

[64] P. Rentrop, A Taylor series method for the numerical solution of two-point boundary value problems, Numerische Mathematik 31 (4) (1978) 359–375. (Cited on page 15.)

[65] G. Corliss, Y. Chang, Solving ordinary differential equations using Taylor series, ACM Transactions on Mathematical Software 8 (2) (1982) 114–144. (Cited on page 15.)

[66] J. L. López, N. M. Temme, Two-point Taylor expansions of analytic functions, Studies in Applied Mathematics 109 (4) (2002) 297–311. (Cited on page 15.)

[67] J. L. López, E. P. Sinusía, N. M. Temme, Multi-point Taylor approximations in one-dimensional linear boundary value problems, Applied Mathematics and Computation 207 (2) (2009) 519–527. (Cited on page 15.)

[68] X. Zhang, X. H. Liu, K. Z. Song, M. W. Lu, Least-squares collocation meshless method, International Journal for Numerical Methods in Engineering 51 (9) (2001) 1089–1100. (Cited on pages 15, 17, 49, 54, 79, 83 and 108.)

[69] C. Farhat, F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, International Journal for Numerical Methods in Engineering 32 (6) (1991) 1205–1227. (Cited on page 15.)

[70] P. Gosselet, C. Rey, Non-overlapping domain decomposition methods in structural mechanics, Archives of Computational Methods in Engineering 13 (4) (2006) 515–572. (Cited on page 15.)

[71] D. Stefanica, Parallel FETI algorithms for mortars, Applied Numerical Mathematics 54 (2) (2005) 266–279. (Cited on page 15.)

[72] C. Bernardi, Y. Maday, A. T. Patera, A new nonconforming approach to domain decomposition: the mortar element method, in: Nonlinear "Partial Differential Equations and Their Applications", H. Brezis & J.-L. Lions (ed.). Pitman: Harlow, 1994, pp. 13–51. (Cited on page 15.)

[73] H. Ben Dhia, Problèmes mécaniques multi-échelles: la méthode Arlequin, Comptes Rendus de l'Académie des Sciences-Series IIB-Mechanics-Physics-Astronomy 326 (12) (1998) 899–904. (Cited on pages 15 and 56.)

[74] H. Ben Dhia, G. Rateau, The Arlequin method as a flexible engineering design tool, International Journal for Numerical Methods in Engineering 62 (11) (2005) 1442–1462. (Cited on page 15.)

[75] X. Chen, J. W. Hutchinson, Herringbone buckling patterns of compressed thin films on compliant substrates, Journal of Applied Mechanics 71 (5) (2004) 597–603. (Cited on pages 42, 59, 87 and 88.)

[76] F. Xu, M. Potier-Ferry, S. Belouettar, Y. Cong, 3D finite element modeling for instabilities in thin films on soft substrates, International Journal of Solids & Structures 51 (21-22) (2014) 3619–3632. (Cited on pages 42, 59, 87 and 88.)

[77] F. Hecht, New development in FreeFem++, Journal of Numerical Mathematics 20 (3) (2012) 251–265. (Cited on pages 44, 67, 72, 90 and 113.)

[78] T. A. Davis, Algorithm 832: UMFPACK V4.3 – An unsymmetric-pattern multifrontal method, ACM Transactions on Mathematical Software (2004) 196–199. (Cited on pages 44, 73 and 90.)

[79] J. Yang, H. Hu, M. Potier-Ferry, Least-square collocation and Lagrange multipliers for Taylor meshless method, in: 11th World Congress on Computational Mechanics (WCCM), Barcelona, Spain, 2014. (Cited on pages 49, 50, 51 and 55.)

[80] D. Li, F. Bai, Y. Cheng, K. M. Liew, A novel complex variable element-free Galerkin method for two-dimensional large deformation problems, Computer Methods in Applied Mechanics and Engineering 233–236 (3) (2012) 1–10. (Cited on page 49.)

[81] L. W. Zhang, Y. J. Deng, K. M. Liew, Y. M. Cheng, The improved complex variable element-free galerkin method for two-dimensional schrödinger equation, Computers & Mathematics with Applications 68 (10) (2014) 1093–1106. (Cited on page 49.)

[82] W. Chen, Y. C. Hon, Numerical convergence of boundary knot method in the analysis of helmholtz, modified helmholtz, and convection-diffusion problems, Computer Methods in Applied Mechanics and Engineering 192 (2003) 1859–1875. (Cited on page 49.)

[83] A. J. M. Ferreira, A formulation of the multiquadric radial basis function method for the analysis of laminated composite plates, Composite Structures 59 (3) (2003) 385–392. (Cited on page 49.)

[84] A. Bogomolny, Fundamental solutions method for elliptic boundary value problems, SIAM Journal on Numerical Analysis 22 (4) (1985) 644–669. (Cited on page 50.)

[85] A. J. M. Ferreira, C. M. C. Roque, R. M. N. Jorge, Free vibration analysis of symmetric laminated composite plates by FSDT and radial basis functions, Computer Methods in Applied Mechanics and Engineering 194 (39–41) (2005) 4265–4278. (Cited on page 50.)

[86] A. J. M. Ferreira, G. E. Fasshauer, Computation of natural frequencies of shear deformable beams and plates by an RBF-pseudospectral method, Computer Methods in Applied Mechanics and Engineering 196 (1–3) (2006) 134–146. (Cited on page 50.)

[87] T. A. Driscoll, B. Fornberg, Interpolation in the limit of increasingly flat radial basis functions, Computers & Mathematics with Applications 43 (3) (2002) 413–422. (Cited on page 50.)

[88] A. H.-D. Cheng, M. A. Golberg, E. J. Kansa, G. Zammito, Exponential convergence and H-c multiquadric collocation method for partial differential equations, Numerical Methods for Partial Differential Equations 19 (5) (2003) 571–594. (Cited on pages 50 and 61.)

[89] R. Schaback, Error estimates and condition numbers for radial basis function interpolation, Advances in Computational Mathematics 3 (3) (1995) 251–264. (Cited on pages 50 and 64.)

[90] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Advances in computational Mathematics 4 (1) (1995) 389–396. (Cited on page 50.)

[91] M. R. Dubal, Domain decomposition and local refinement for multiquadric approximations. I: second-order equations in one dimension, Journal of Applied Science and Computation 1 (1) (1994) 146–171. (Cited on page 50.)

[92] A. S. M. Wong, Y. C. Hon, T. S. Li, S. L. Chung, E. J. Kansa, Multizone decomposition for simulation of time-dependent problems using the multiquadric scheme, Computers & Mathematics with Applications 37 (8) (1999) 23–43. (Cited on page 50.)

[93] F. Chu, L. Wang, Z. Zhong, Finite subdomain radial basis collocation method, Computational Mechanics 54 (2) (2014) 235–254. (Cited on page 50.)

[94] K. M. Liew, L. X. Peng, S. Kitipornchai, Nonlinear analysis of corrugated plates using a FSDT and a meshfree method, Computer Methods in Applied Mechanics and Engineering 196 (21–24) (2007) 2358–2376. (Cited on page 50.)

[95] D. Li, F. Bai, Y. Cheng, K. M. Liew, A novel complex variable element-free Galerkin method for two-dimensional large deformation problems, Computer Methods in Applied Mechanics and Engineering s 233–236 (3) (2012) 1–10. (Cited on page 50.)

[96] Y. Tampango, Développement d'une méthode sans maillage utilisant les approximations de Taylor (Development of a meshless method using Taylor approximations), Ph.D. thesis, Université de Lorraine (2012). (Cited on pages 50 and 76.)

[97] H. K. Akpama, Y. Koutsawa, M. Potier-Ferry, A Taylor meshless method for hyperelasticity, in: 11th World Congress on Computational Mechanics (WCCM), Barcelona, Spain, 2014. (Cited on page 50.)

[98] A. Griewank, J. Utke, A. Walther, Evaluating higher derivative tensors by forward propagation of univariate Taylor series, Mathematics of Computation 69 (231) (2000) 1117–1130. (Cited on pages 50, 80, 81 and 86.)

[99] H. Hu, N. Damil, M. Potier-Ferry, A bridging technique to analyze the influence of boundary conditions on instability patterns, Journal of Computational Physics 230 (10) (2011) 3753–3764. (Cited on pages 56 and 84.)

[100] A. Poullikkas, A. Karageorghis, G. Georgiou, The method of fundamental solutions for three-dimensional elastostatics problems, Computers & Structures 80 (3) (2002) 365–370. (Cited on page 59.)

[101] J.-M. Muller, N. Brisebarre, F. de Dinechin, C.-P. Jeannerod, V. Lefèvre, G. Melquiond, N. Revol, D. Stehlé, S. Torres, Handbook of Floating-Point Arithmetic, Birkhäuser Basel, 2010. (Cited on pages 61 and 117.)

[102] B. A. Szabo, I. Babuška, Finite element analysis, John Wiley & sons, New York, 1991. (Cited on page 73.)

[103] T. Liszka, J. Orkisz, The finite difference method at arbitrary irregular grids and its application in applied mechanics, Computers & Structures 11 (1-2) (1980) 83–95. (Cited on page 79.)

[104] H. F. Chan, C. M. Fan, C. W. Kuo, Generalized finite difference method for solving two-dimensional non-linear obstacle problems, Engineering Analysis with Boundary Elements 37 (9) (2013) 1189–1196. (Cited on page 79.)

[105] C. W. Shu, High-order finite difference and finite volume WENO schemes and discontinuous galerkin methods for CFD, International Journal of Computational Fluid Dynamics 17 (2) (2003) 107–118. (Cited on page 79.)

[106] F. Boyer, F. Hubert, Finite volume method for 2D linear and nonlinear elliptic problems with discontinuities, SIAM Journal on Numerical Analysis 46 (6) (2008) 3032–3070. (Cited on page 79.)

[107] E. Tadmor, Convergence of spectral methods for nonlinear conservation laws, SIAM Journal on Numerical Analysis 26 (1) (1989) 30–44. (Cited on page 79.)

[108] E. Tadmor, A review of numerical methods for nonlinear partial differential equations, Bulletin of the American Mathematical Society 49 (4) (2012) 507–554. (Cited on page 79.)

[109] M. Y. Hussaini, T. A. Zang, Spectral methods in fluid dynamics, Advances in Applied Mechanics 57 (196) (1987) 285–331. (Cited on page 79.)

[110] K. Balakrishnan, P. A. Ramachandran, Osculatory interpolation in the method of fundamental solution for nonlinear Poisson problems, Journal of Computational Physics 172 (1) (2001) 1–18. (Cited on page 79.)

[111] A. Tri, H. Zahrouni, M. Potier-Ferry, Perturbation technique and method of fundamental solution to solve nonlinear Poisson problems, Engineering Analysis with Boundary Elements 35 (3) (2011) 273–278. (Cited on page 79.)

[112] A. Tri, H. Zahrouni, M. Potier-Ferry, High order continuation algorithm and meshless procedures to solve nonlinear Poisson problems, Engineering Analysis with Boundary Elements 36 (11) (2012) 1705–1714. (Cited on page 79.)

[113] Y. Gu, W. Chen, C. Z. Zhang, Singular boundary method for solving plane strain elastostatic problems, International Journal of Solids and Structures 48 (18) (2011) 2549–2556. (Cited on page 79.)

[114] G. R. Liu, Meshfree methods: moving beyond the finite element method, CRC Press, USA: Boca Raton, 2002. (Cited on page 79.)

[115] A. Timesli, B. Braikat, H. Lahmam, H. Zahrouni, A new algorithm based on moving least square method to simulate material mixing in friction stir welding, Engineering Analysis with Boundary Elements 50 (2015) 372–380. (Cited on page 79.)

[116] J. Yang, H. Hu, M. Potier-Ferry, Solving large-scale problems by taylor meshless method, International Journal for Numerical Methods in Engineering 112 (2) (2017) 103–124. (Cited on pages 79, 80, 88, 91, 94, 97, 105, 106, 108, 109, 112, 113 and 117.)

[117] J. Yang, H. Hu, M. Potier-Ferry, Least-square collocation and Lagrange multipliers for Taylor meshless method, submitted for publication, 2017. (Cited on pages 80 and 84.)

[118] N. Damil, M. Potier-Ferry, A new method to compute perturbed bifurcations: Application to the buckling of imperfect elastic structures, International Journal of Engineering Science 28 (9) (1990) 943–957. (Cited on page 80.)

[119] J. Yang, Q. Huang, H. Hu, G. Giunta, S. Belouettar, M. Potier-Ferry, A new family of finite elements for wrinkling analysis of thin films on compliant substrates, Composite Structures 119 (2015) 568–577. (Cited on page 80.)

[120] C. S. Chen, Y. F. Rashed, Evaluation of thin plate spline based particular solutions for Helmholtz-type operators for the DRM, Mechanics Research Communications 25 (2) (1998) 195–201. (Cited on page 80.)

[121] M. A. Golberg, C. S. Chen, An efficient mesh-free method for nonlinear reaction-diffusion equations, CMES−Computer Modeling in Engineering 2 (1) (2001) 87–95. (Cited on page 80.)

[122] K. L. Johnson, One hundred years of hertz contact, Proceedings of the Institution of Mechanical Engineers 196 (1982) 363–378. (Cited on page 80.)

[123] C. S. Liu, Solving singular convection-diffusion equation by exponentially-fitted trial functions and adjoint Trefftz test functions, Journal of King Saud University − Science (2016) in press, doi:http://doi.org/10.1016/j.jksus.2016.09.011. (Cited on page 80.)

[124] A. Griewank, A. Walther, Evaluating derivatives: principles and techniques of algorithmic differentiation, Society for Industrial and Applied Mathematics, Philadelphia, 2000. (Cited on page 86.)

[125] C. Bischof, G. Corliss, A. Griewank, Structured second-and higher-order derivatives through univariate Taylor series, Optimization Methods and Software 2 (3-4) (1993) 211–232. (Cited on page 86.)

[126] R. D. Neidinger, Directions for computing truncated multivariate Taylor series, Mathematics of Computation 74 (249) (2005) 321–340. (Cited on pages 86 and 87.)

[127] R. D. Neidinger, Efficient recurrence relations for univariate and multivariate Taylor series coefficients, American Institute of Mathematical Sciences (2013) 587–596. (Cited on page 86.)

[128] R. D. Neidinger, Directions for computing truncated multivariate Taylor series: MATLAB implementation, http://academics.davidson.edu/math/neidinger/publicat.html. (Cited on page 87.)

[129] D. Kaltchev, Building truncated Taylor maps with mathematica and applications to FFAG, in: Proc. 9th European Particle Accelerator Conference, Lucerne, Switzerland, 2004, pp. 1822–1824. (Cited on page 87.)

[130] D. Kaltchev, Implementation of TPSA in the mathematica code LieMath, in: Proc. 10th European Particle Accelerator Conference, Edinburgh, UK, 2006, pp. 2077–2079. (Cited on page 87.)

[131] D. Kaltchev, Mathematica program for extracting one-turn lie generator map. Application of TPSA, Physics Procedia 1 (1) (2008) 333–338. (Cited on page 87.)

[132] K. Yu, H. Hu, H. Tang, G. Giunta, M. Potier-Ferry, S. Belouettar, A novel two-dimensional finite element to study the instability phenomena of sandwich plates, Computer Methods in Applied Mechanics and Engineering 283 (2015) 1117–1137. (Cited on pages 87 and 88.)

[133] J. Jirousek, P. Teodorescu, Large finite elements method for the solution of problems in the theory of elasticity, Computers & Structures 15 (1982) 575–587. (Cited on pages 105 and 109.)

[134] J. Jirousek, P. Teodorescu, Hybrid-Trefftz plate bending elements with p-method capabilities, International Journal for Numerical Methods in Engineering 24 (1987) 1367–1393. (Cited on page 105.)

[135] J. Yang, H. Hu, Y. Koutsawa, M. Potier-Ferry, Taylor meshless method for solving non-linear partial differential equations, Journal of Computational Physics 348 (2017) 385–400. (Cited on page 105.)

[136] H. Liebowitz, E. T. Moyer, Finite element methods in fracture mechanics, Computers & Structures 31 (1989) 1–9. (Cited on pages 106 and 114.)

[137] T. Belytschko, R. Gracie, G. Ventura, A review of extended/generalized finite element methods for material modeling, Modelling and Simulation in Materials Science and Engineering 17 (2009) 043001. (Cited on page 106.)

[138] W. K. Wilson, Combined mode fracture mechanics, Ph.D. thesis, University of Pittsburgh (1969). (Cited on page 106.)

[139] D. M. Tracey, Finite elements for determination of crack tip elastic stress intensity factors, Engineering Fracture Mechanics 8 (1971) 255–265. (Cited on page 106.)

[140] S. E. Benzley, Representation of singularities with isoparametric finite elements, International Journal for Numerical Methods in Engineering 8 (1974) 131–150. (Cited on page 106.)

[141] J. M. Melenk, I. Babuška, The partition of unity finite element method: basic theory and applications, Computer Methods in Applied Mechanics and Engineering 139 (1996) 289–314. (Cited on page 106.)

[142] N. Moës, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, International Journal for Numerical Methods in Engineering 46 (1999) 131–150. (Cited on page 106.)

[143] Z. C. Li, Combined methods for elliptic equations with singularities, interfaces and infinities, Kluwer Academic Publishers, Dordrecht, 1998. (Cited on pages 106 and 114.)

[144] Z. C. Li, T. T. Lu, H. T. Huang, A. H. D. Cheng, Trefftz, collocation, and other boundary methods: a comparison, Numerical Methods for Partial Differential Equations 23 (2007) 93–144. (Cited on page 106.)

[145] P. Grisvard, Elliptic problems in nonsmooth domains, Pitman, Marshfield, 1985. (Cited on pages 111 and 120.)

[146] M. Dauge, Elliptic boundary value problems on corner domains: smoothness and asymptotics of solutions, Springer, Berlin, Heidelberg, 1988. (Cited on page 111.)

[147] M. Costabel, M. Dauge, Z. Yosibash, A quasi-dual function method for extracting edge stress intensity functions, SIAM Journal on Mathematical Analysis 35 (2004) 1177–1202. (Cited on page 115.)

[148] E. J. Kansa, P. Holoborodko, On the ill-conditioned nature of C^{∞} RBF strong collocation, Engineering Analysis with Boundary Elements 78 (2017) 26–30. (Cited on page 117.)

[149] G. C. Sih, On the Westergaard method of crack analysis, International Journal of Fracture 2 (4) (1966) 628–631. (Cited on page 117.)

[150] N. I. Muskhelishvili, Some basic problems of the mathematical theory of elasticity, Springer Netherlands, Berlin, 1977. (Cited on pages 117 and 127.)