

MESHLESS ADAPTIVE SOLUTION PROCEDURE FOR EFFICIENT SOLVING OF PARTIAL DIFFERENTIAL EQUATIONS

Mitja Jančič

Doctoral Dissertation
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Supervisor: Dr. Gregor Kosec, Jožef Stefan Institute, Parallel and Distributed Systems,
Ljubljana, Slovenia

Evaluation Board:

Prof. Dr. Juš Kocijan, Chair, Jožef Stefan Institute, Department of Systems and Control,
Ljubljana, Slovenia

Doc. Dr. Boštjan Mavrič, Member, University of Ljubljana, Faculty of Mechanical Engineering,
Ljubljana, Slovenia

Prof. Dr. Varun Shankar, Member, Kahlert School of Computing, University of Utah, Salt
Lake City, USA

MEDNARODNA PODIPLOMSKA ŠOLA JOŽEFA STEFANA
JOŽEF STEFAN INTERNATIONAL POSTGRADUATE SCHOOL



Mitja Jančič

MESHLESS ADAPTIVE SOLUTION PROCEDURE FOR
EFFICIENT SOLVING OF PARTIAL DIFFERENTIAL EQUA-
TIONS

Doctoral Dissertation

PRILAGODLJIVE BREZMREŽNE METODE ZA
UČINKOVITO REŠEVANJE PARČIALNIH
DIFERENCIALNIH ENAČB

Doktorska disertacija

Supervisor: Dr. Gregor Kosec

Ljubljana, Slovenia, January 2024

Acknowledgments

First, I wish to express my gratitude to Dr. Gregor Kosec, my supervisor, for his guidance during my doctoral studies over the past years. With his help, I gained a better understanding of the science behind numerical simulations and improved my writing, communication and time management skills. His help also extended beyond the confines of our professional interaction, for which I wish to thank him specifically.

Then I would like to thank my colleagues at the Department of Communication Systems at the Jožef Stefan Institute for all the fruitful discussions that inspired and motivated me to pursue my PhD. In particular, I would like to thank the Parallel and Distributed Systems Lab. Thank you for the wonderful years we spent together.

I would also like to thank the members of the evaluation board for their constructive criticism, which helped me to further improve the thesis and my understanding of the topic.

I wish to thank ARRS for providing financial support through research core funding No. P2-0095 and the World Federation of Scientists for granting me a 12-month scholarship.

Finally, I would like to thank my family and friends for putting up with my absence and long working hours. Thank you!

Abstract

Meshless methods are becoming increasingly popular in computational mechanics and engineering. Their main feature is the ability to manage complex geometries while avoiding the often tedious process of mesh generation required by the traditional methods. Various meshless approximations of linear differential operators appearing in the governing problem have been proposed over the years. However, even with the state-of-the-art methods and immense computational power at our disposal, efficient solution procedures for solving systems of partial differential equations (PDEs) are being actively studied, as the complexity of the problems under consideration continues to increase.

In this thesis, we study different approaches towards efficient PDE-solving solution procedures employing meshless approximation methods. Our study is limited to three commonly used approximation methods: the Radial Basis Function-generated Finite Differences (RBF-FD), the Diffuse Approximation Method (DAM), referred to as Weighted Least Squares (WLS) in this thesis, and the simplest collocation method, referred to as MON throughout this thesis.

The first part of the thesis is devoted to a discussion of the fundamentals of meshless approximation. The effects of monomial augmentation on the accuracy of the numerical solution, as well as on the stability and the computational complexity of the solution procedure are investigated. Our understanding of the meshless approximation is then enhanced with a study of the stencil size impact on the accuracy of the numerical solution. We then compare the performance of the RBF-FD and the WLS approximation methods in terms of the accuracy of the obtained solution and the stability of the solution procedure.

After demonstrating the stability of the high-order RBF-FD approximations, we focus on the development and implementation of the hp -adaptive solution procedure in the second part of the dissertation. We first employ p -refinement to demonstrate the effects of spatially-variable approximation order on the efficiency of the solution procedure. We then develop an original error indicator, which subsequently enables the implementation of a fully adaptive strong-form meshless method employing both h - and p -refinement procedures. This method simultaneously adjusts the spatial discretization resolution (h -adaptivity) and the approximation order (p -adaptivity) to allocate the available computational resources in the domain regions where they are most needed.

The last part of the thesis deals with yet another attempt to improve the efficiency of PDE-solving solution procedures, namely by spatially varying the approximation method. Specifically, the advantages of local approximation and properties of different approximation methods are exploited to propose a hybrid WLS–RBF-FD approximation method. A step further is done by introducing spatially variable node regularity in conjunction with spatially variable approximation method, employing RBF-FD on scattered nodes and MON on uniform nodes. The performance of both spatially-varying approximation methods is demonstrated on a set of fluid-flow and linear elasticity problems to illustrate the computational efficiency gains offered by such solution procedures.

Povzetek

Vse pogosteje se na področju inženirstva in numeričnih simulacij uporabljajo brez mrežne metode. Njihova glavna prednost je, da v nasprotju s tradicionalnimi numeričnimi metodami zapleteno grajenje mreže za diskretizacijo računske domene ni potrebno. Do danes so znanstveniki razvili več metod za brez mrežno aproksimacijo linearnih diferencialnih operatorjev, ki se pojavijo v matematičnem opisu problemov. Kljub modernim in učinkovitim metodam ter visoki zmogljivosti današnjih računalnikov pa je razvoj učinkovitih metod za reševanje sistemov parcialnih diferencialnih enačb (PDE) še vedno aktivno področje raziskav – predvsem zaradi naraščajoče kompleksnosti obravnavanih problemov, ki zahtevajo vse več računske moči.

V disertaciji se osredotočimo na različne pristope k učinkovitejšim postopkom reševanja PDE z brez mrežnimi metodami. Pri tem se omejimo na tri pogosto uporabljene aproksimacijske metode: končne difference, generirane z radialnimi baznimi funkcijami (RBF-FD), metodo difuzivne aproksimacije (DAM), ki jo v tem delu poenostavimo imenujemo kar metoda uteženih najmanjših kvadratov (WLS), in najpreprostejšo metodo kolokacije MON.

Prvi del disertacije je posvečen diskusiji osnovnih značilnosti aproksimacij z brez mrežnimi metodami. Najprej preučimo vpliv razširitve baznih funkcij aproksimacije z monomskimi členi na natančnost numerične rešitve ter na stabilnost in računsko kompleksnost rešitvenega postopka. Razumevanje brez mrežnih aproksimacij nato poglobimo s študijo vpliva velikosti podpornih domen na natančnost numerične rešitve. Nazadnje, primerjamo še RBF-FD in WLS metodi ter ju ovrednotimo z vidika natančnosti numerične rešitve in stabilnosti rešitvenega postopka.

Po demonstraciji stabilnosti RBF-FD aproksimacij višjega reda, se v drugem delu disertacije poglobimo v razvoj in implementacijo *hp*-prilagodljivega postopka reševanja sistemov PDE. Diskusijo pričnemo z implementacijo *p*-prilagodljivosti in pokažemo vpliv prostorsko spremenljivega reda aproksimacije na učinkovitost rešitvenega postopka. V nadaljevanju razvijemo izvirni indikator napake numerične rešitve, ki nam omogoči razvoj popolnoma prilagodljive brez mrežne metode z uporabo *h*- in *p*-prilagodljivosti. Tako razvit *hp*-rešitveni postopek na podlagi vrednosti indikatorja napake iterativno hkrati prilagaja prostorsko ločljivost diskretizacije (*h*-prilagodljivost) in red aproksimacije (*p*-prilagodljivost) ter s tem prerazporedi razpoložljive računske vire v dele domene, kjer so le-ti najbolj potrebni.

V zadnjem delu disertacije obravnavamo možnost izboljšanja učinkovitosti postopkov reševanja PDE preko prostorsko spremenljive metode aproksimacije. Natančneje, uporabimo osnovne lastnosti lokalne aproksimacije in značilnosti različnih aproksimacijskih metod ter razvijemo hibridno WLS–RBF-FD aproksimacijo diferencialnih operatorjev. V naslednjem koraku uvedemo prostorsko spremenljivo regularnost računskih točk, ki jo spremlja prostorsko spremenljiva metoda aproksimacije. Pri tem uporabimo stabilno RBF-FD metodo na razpršenih točkah in računsko učinkovito metodo MON na uniformnih točkah. Vpliv obeh prostorsko spremenljivih aproksimacijskih metod na učinkovitost rešitvenega postopka je demonstriran na več problemih s področja elastomehanike in dinamike tekočin.

Contents

1	Introduction	1
1.1	Numerical Treatment of PDEs	1
1.2	Historical Overview of Meshless Methods	2
1.3	Adaptive Solution Procedures	4
1.4	Purpose of the Dissertation	5
1.5	Contributions	7
2	The Fundamental Properties of Meshless Approximation Methods	9
2.1	Monomial Augmentation Guidelines for RBF-FD from Accuracy Versus Computational Time Perspective	10
2.2	Oscillatory Behaviour of the RBF-FD Approximation Accuracy Under Increasing Stencil Size	30
2.3	Stability Analysis of RBF-FD and WLS-Based Local Strong-Form Meshless Methods on Scattered Nodes	39
3	<i>hp</i>-Adaptive Solution Procedure	47
3.1	p -refined RBF-FD Solution of a Poisson Problem	47
3.2	The IMEX (IMplicit-EXplicit) Error Indicator	55
3.3	Strong-Form Mesh-Free hp -Adaptive Solution of Linear Elasticity Problem	61
4	Spatially-Adaptive Approximation Methods	83
4.1	A Hybrid RBF-FD and WLS Mesh-Free Strong-Form Approximation Method	83
4.2	Spatially-Varying Meshless Approximation Method Based on Node Regularity	91
5	Conclusions and Future Work	107
5.1	Summary of Conclusions	107
5.2	Future Work Opportunities	110
	References	113
	Bibliography	123
	Biography	125

Chapter 1

Introduction

Human progress has always been driven by the desire to understand natural phenomena. With the increasing computational power at our disposal, a commonly used approach to expand our understanding of nature is computational modelling, where various phenomena are described with mathematical models that are then numerically solved by computers.

A common approach to describe dynamic processes that evolve in space and time are Partial Differential Equations (PDEs). In their essence, PDEs bridge the gap between theoretical concepts and real-world observations of various problems, e.g. heat transfer, fluid flow, wave propagation, quantum mechanics, etc. They serve as the basis for both analytical studies and computer simulations and thus play an important role in various scientific disciplines and technology.

1.1 Numerical Treatment of PDEs

Due to the complexity of realistic models, analytic solutions to systems of PDEs are rare [1]. In some cases, analytic solution can be obtained by using advanced mathematical procedures and/or a series of simplifications, which makes the obtained solutions applicable only in certain scenarios. As an alternative, numerical treatment of PDEs has been proposed [2]. In the process of numerical treatment, the considered system of PDEs is transformed into a system of algebraic equations [2], [3]. The solution to the algebraic system then provides us with an approximate, i.e. *numerical*, solution to the governing PDE system.

To date, various methods for numerical treatment of PDEs have been proposed. Ranging from the weak form Finite Element Method (FEM) [4] to relatively simple strong form Finite Difference Method (FDM) [3]. The FDM is often presented as the most intuitive, easy to implement, and computationally effective method, but at a cost of limited usability on irregularly shaped domains. Modern numerical analysis is dominated by FEM [5], as it offers a mature and versatile solution approach that includes all types of adaptive solution procedures [6] and well understood error indicators [7]. The matured FEM, has been accepted as the standard numerical method for solving PDEs in weak form by the community. Despite its widespread acceptance, the meshing of realistic three-dimensional domains¹ is still a problem that often requires user assistance or development of domain-specific meshing algorithms [8]. It is widely accepted that mesh generation is one of the most cumbersome stages of the FEM analysis. In response to the tedious meshing of realistic domains, required by FEM, and the geometric limitations of FDM, a new class of *meshless* or *mesh-free methods* [9] emerged in the 1970s.

¹Meshing the computational domain is a crucial part of FEM analysis where nodes are structured into polyhedrons covering the entire domain of interest.

The main feature of the meshless methods is that a topological relationship between computational nodes is not required and they can, thus, operate on scattered nodes. This greatly simplifies the discretization of the domain [10], regardless of its dimensionality or shape [11], [12]. Although positioning of nodes is considered to be a much simpler task than meshing, it is still not trivial. In the early years, many authors used available mesh generators to generate the discretization nodes and discarded the internodal connectivity after the mesh had been generated [13]. Such procedure is conceptually flawed, since the idea of meshless methods is to avoid meshing. It also generates nodal distributions of insufficient quality [14]. Consequently, several algorithms dedicated to positioning of nodes for meshless approximation emerged, ranging from sphere packing-based algorithms [15], to front advancing [16] and iterative methods [17]. In 2018, a purely meshless algorithm based on Poisson disk sampling [18] was introduced. In the same year, the first dimension-independent node generation algorithm that supports distributions with spatially variable density [19] was also proposed. Today, a parallelized version of the same node positioning algorithm is available [20]. Just recently, meshless methods have also been promoted to Computer Aided Design (CAD) geometry-aware numerical analysis [21].

Researchers have also reported that large deformations are handled more robustly [22] and argued that the implementation of adaptive procedures [23] and stabilisations [24], [25] is simpler when meshless methods are used. That is largely due to the elegant formulation of meshless methods resulting in straightforward implementation of different refinement procedures [26], [27], considering different approximations of differential operators in terms of the shape and size of the *stencil* [28], [29] and the local approximation order [30]–[32]. The less attractive property of meshless methods is its computational cost, which is generally higher than that of mesh-based methods due to their generality² and larger stencil requirements for stable approximations [30]–[32].

1.2 Historical Overview of Meshless Methods

The local meshless methods can be understood as a generalisation of FDM to scattered nodes. One of the first attempts to generalise FDM dates back to 1972 when Jense considered Taylor series expansions interpolated to stencils with six nodes [34]. This idea was later developed into a Generalised Finite Differences Method (GFDM) using Weighted Least Squares (WLS) approximation [35].

Many consider the Smoothed Particle Hydrodynamics (SPH), developed by Lucy [36] and Gingold and Monaghan [37] in 1977, to be the start of meshless methods. SPH was originally used to solve astrophysical problems. Later, applications to fluid dynamics [38]–[40] and solid mechanics [41] were also reported. SPH is based on the idea of representing the domain of interest as a set of particles. This approach has the advantage of being very flexible, as the particles can be easily positioned or moved to represent or follow complex geometrical shapes.

The first meshless methods for boundary value problems were developed in the early 1990s as generalizations of the FEM to a meshless setting. The first such method was the Diffuse Element Method (DEM) [42], proposed by Nayroles et al. in 1992. In 1994, Belytschko et al. [43] extended DEM to the Element Free Galerkin (EFG) method. EFG is similar to FEM in that the problem is solved in its weak form using test and trial functions. However, EFG uses Moving Least Squares (MLS)-based shape functions and

²Generation of shape functions in weak form, or stencil weights computation in the strong form context, is transferred from pre-process to the solution procedure [23], [33].

requires a background grid for integration³.

In addition to DEM and EFG, a number of other meshless methods were developed in the early 1990s. In 1995, Oñate et al. introduced one of the most notable strong-form collocation methods, the Finite Point method (FPM) [45]. In general, collocation methods can be viewed as a special case of weighted residual methods, where the test functions are chosen to be the Dirac delta distributions [46]. FPM approximates a partial differential operator using either least squares, weighted least squares, or moving least squares methods. The method has been well-researched and has been used with a wide variety of problems [47]. Around the same time, a similar method appeared in conjunction with the flow of porous media, called the Diffuse Approximate Method (DAM) [48]. Other methods include methods based on reproducing kernels [49] and Partition of Unity Methods (PUM) [50], [51]. A more thorough overview of these methods is given by Belytschko in his review paper [23] from 1996.

A few years after the EFG, Atluri and Zhu [52], [53] proposed the Meshless Local Petrov Galerkin (MLPG) method. MLPG solves the problem locally in its weak form by enforcing the governing equation for selected subdomains of the whole computational domain. Concurrently, the Boundary Element Method (BEM) was generalized to a meshless setting, resulting in application of global weak forms known as the Boundary Node Method (BNM) [54] and local weak forms referred to as the Local Boundary Integral Method (LBIM) [55]. Around that period, the Moving Point Method [56], [57] was also proposed.

In the 1990s, Kansa proposed a global method using Radial Basis Functions (RBFs) to solve PDEs in the strong-form collocation formula [58]. His proposal was theoretically justified in 1998 [59] and showed high-order convergence and spectral accuracy but has high computational complexity and possible ill-conditioning in solving the global system due to its global formulation. The shortcomings of the global approach encouraged the development of local variant of the method, resulting in methods such as Local RBF Collocation Method (LRBFCM) [60], Radial Point Collocation Method (RPCM) [61] and Radial Basis Function-generated Finite Differences (RBF-FD) [62], where the methods mainly differ in the selection of basis functions and augmenting monomials.

Among all local collocation methods, RBF-FD is likely the most popular meshless method at the moment. RBF-FD uses RBFs, e.g. Polyharmonic Splines (PHS), augmented with monomials [30], [31] to avoid stagnation errors and allow control over the convergence rate, respectively. Recently, RBF-FD has been used in the h -adaptive solution of elliptic problems [63] and linear elasticity situations [26], where the authors also presented special balanced stencils, effectively reducing the computational cost. In their latest paper, Davydov et al. continued the development of advanced stencil selection [64] for RBF-FD in three-dimensional domains. Furthermore, the ease of order regularisation in RBF-FD naturally led to several publications discussing high-order solutions [32], [65], [66].

The historical overview on the development of meshless methods is by no means complete. Further details can be found in textbooks by Fasshauser [67] or more recent reviews by Chen and Belytschko [68] and Patel [69].

It is perhaps also worth mentioning that the development of meshless methods is still active today. In recent years, a theoretical study on the error analysis of the FPM has been presented [70], the solution of PDEs defined on evolving manifolds with Lagrangian GFDM [71], high-order GFDM with a strategy for avoiding ill-conditioned stencils [72], high-order WLS Radial Basis Functions (RBF) approach with improved stability in the

³The use of a background grid in DEM and EFG led some authors to classify these methods as not *truly meshless*. However, the precise definition of a *truly meshless* approximation method is not unique. Some researchers argue that any method using a background grid or cells of any kind is no longer truly meshless. Others argue that the use of a background grid is acceptable, as long as it is not used to enforce boundary conditions [44].

presence of Neumann boundaries [73], to name but a few.

1.3 Adaptive Solution Procedures

Despite the immense computing power and a number of different numerical methods at our disposal, the development of efficient numerical approaches is still crucial to solve ever more complex problems. Relying solely on brute force computing often leads to unnecessarily long execution times – not to mention the wasted energy.

In problems where the spatial variation in solution’s accuracy is significant, adaptive solution procedures play a crucial role and have recently become the focus of extensive research. Note that the regions requiring higher accuracy are often not known *a priori*. Thus, the use of *a posteriori* error indicators is essential in adaptive solution procedures [8]. One of the most well-known error indicators, often referred to as the ZZ-type error indicator, was introduced by Zienkiewicz and Zhu [74] in 1987 in the context of FEM, and it continues to be an active area of research [75]. The ZZ-type error indicator relates the error of the numerical solution to the difference between the numerical solution and a locally recovered solution. This indicator has been applied in the context of meshless solutions for elasticity problems using the meshless Finite Volume Method (FVM) [76], both in weak and strong forms using the FPM [77], and also served as inspiration in the context of RBF-FD solutions to Laplace equation [63]. Additionally, a class of error indicators based on residuals has been demonstrated on elasticity problems using a Discrete Least Squares Meshless Method [78]. Nevertheless, the simplest and most intuitive indicators are based on the physical interpretation of the solution, usually evaluating the first derivative of the field under consideration [28] or calculating the variance of the field values within the stencil [26].

After the domain regions in need of higher accuracy are identified, conceptually different techniques to locally improve the solution’s accuracy have been proposed: p -adaptivity and h -, r -adaptivities. In p -adaptivity, the local accuracy of the numerical solution is adjusted by controlling the order of approximation. On the other hand, in h - and r -adaptivities, the spatial discretization resolution is modified to achieve the same objective. Specifically, in the h -adaptive approach, nodes are added or removed from the domain as needed, while in the r -adaptive approach, the total number of nodes remains constant, but their positions are optimized to improve the local accuracy of the numerical solution. The h - and p -adaptive strategies can be combined to form the so-called hp -adaptivity [79]–[81]. This approach enables control over the solution’s accuracy by simultaneously adjusting the order of the approximation method and the spatial discretization resolution.

The concept of hp -adaptive numerical analysis originated with FEM in the 1980s [82]. In hp -FEM, one can choose to either split a given finite element into smaller elements or increase its approximation order (or both). This choice is often considered as the primary challenge in implementing hp -adaptive solution procedures and has been studied by Babuška [82] as early as 1986. Since then, various decision-making strategies, also known as *marking strategies*, have been proposed [6], [83]. The early works employed a simple Texas Three Step algorithm, initially proposed for Boundary Element Method (BEM) [84], where refinement is based on the maximum value of the error indicator. The first true hp -strategy was presented by Ainsworth [85] in 1997. Since then, many other strategies have been proposed – with condensed descriptions and performance analyses provided by Mitchell in [6] and [83]. Generally, p -refinement in FEM is more efficient when the solution is smooth. Based on this observation, most authors nowadays use the local Sobolev regularity estimate to choose between h - and p -refinements [86]–[88] for a given finite element. As an alternative, in [89], local boundary values are solved, while the

authors of [90], [91] employ global interpolation error minimization methods.

For meshless methods, h -adaptivity comes naturally due to their ability to work with scattered nodes, and as such, it has been extensively studied in the context of several meshless methods. The h -adaptive LRBFCM has been tackled in solution of 2D Burgers' equation in [92], Benito et al. demonstrated the h -adaptive GFDM [93] in solving Laplace equation, in [94], Liu et al. discussed the h -adaptive RPCM solution of a cantilever beam where they used a Voronoi diagram to assign the positions of new computational nodes. Recently, an h -adaptive RBF-FD solution of the elasticity problem has been demonstrated in [26], [95], an h -adaptive generalised moving least squares solution in fluid-structure interactions in [96] and h -adaptive solutions to elliptic problems [63], [97]. Moreover, h -adaptivity has been used with the RBF-PUM collocation refinement schemes [98]. Some researchers have also reported on combining the h - and r -adaptivity, forming a so-called hr -adaptive solution procedure [99].

On the other hand, the p -adaptive method remains relatively unexplored in the meshless community. The authors of [100] approach the p -adaptive RBF-FD method in solving Poisson's equation by varying the order of the augmenting monomials to maintain the global order of convergence over the domain, regardless of potential variations in spatial discretization distances. It should also be noted that some authors reported p -adaptive methods by locally increasing the number of shape functions, changing the interpolation basis functions, or simply increasing the stencil size [72], [101], [102]. While these approaches are to some extent p -adaptive, they may not fully capture its true essence. The closest attempt at a true hp -adaptive solution procedure has been presented by Duarte et al. with the h - p adaptive clouds [103], where the authors use grid-like h -enrichment to improve the local field description.

Yet another approach for efficient PDE solving is based on spatially-variable approximation method. Similar approaches have already been introduced, such as the hybrid FEM-meshless method [104], [105] addressing issues with unstable Neumann boundary conditions in a meshless approximation. Additionally, hybrid methods that combine FDM on a conventional cartesian grid with meshless approximation on scattered nodes have been proposed [106], [107]. While these hybrid approaches are computationally efficient, they often require additional implementation-related efforts at the transition from cartesian to scattered nodes (or similarly at the transition from one approximation method to the other) [108].

1.4 Purpose of the Dissertation

The goal of this dissertation is to study, implement and discuss some approaches towards efficient solving of PDEs with meshless methods. While development of an hp -adaptive solution procedure is considered to be the main scientific contribution of this thesis, the following set of hypotheses have been investigated in the process:

- H1:** There exists an optimal setup of monomial augmentation in terms of accuracy with respect to the computational complexity.
- H2:** The RBF-FD approximation method is superior to the WLS approximation, particularly in terms of stability of high-order approximations.
- H3:** The RBF-FD and WLS strong-form numerical methods can be tailored for hp -adaptive numerical analysis.
- H4:** The a posteriori error indicator can be constructed solely on assessment of local high-order operator approximation.

- H5:** Spatially variable approximation order can be used to improve accuracy of the solution with relatively small effect on computational time.
- H6:** The local strong form *hp*-refined solution procedure can be used to efficiently solve elliptic PDE.
- H7:** Spatially varying approximation methods can be used to reduce computational complexity while maintaining the accuracy of the solution.

In the first chapter (see Chapter 2), we give a brief fundamentals overview of the two commonly used meshless approximation methods, namely the RBF-FD and DAM — also referred to as WLS in this thesis — approximations. In Section 2.1, we focus on understanding the impact of high-order approximations on the accuracy of the numerical solution as well as on the computational complexity and on the stability of the solution procedure. We provide an estimate on the optimal setup of augmenting monomials with respect to the computational complexity, essentially confirming hypothesis **H1**. Due to the demonstrated stability of the high-order RBF-FD approximation, this section also partially confirms hypothesis **H3**. Section 2.2 addresses hypothesis **H7** by discussing the effect of stencil size on the accuracy of the numerical solution. We show that the stencil size (which varies with the chosen approximation method) affects not only the computational complexity of the approximation, but also the accuracy of the numerical solution. Finally, in Section 2.3, we perform a brief stability analysis of the RBF-FD and WLS approximations with respect to computational node positions, which allows us to confirm hypotheses **H2** and **H3** and conclude that the RBF-FD is more stable in case of high-order approximations and therefore more suitable for solution procedures with spatially-varying approximation orders presented in Chapter 3.

In Chapter 3, we present all the building blocks needed for a meshless *hp*-adaptive solution procedure. We start with an implementation of the *p*-refined numerical solution and demonstrate the advantages of spatially-variable approximation orders in Section 3.1. This section presents a practical demonstration of hypothesis **H3** and confirms hypotheses **H1** and **H5** on a synthetic example. Hypothesis **H4** is confirmed in Section 3.2, where we propose an original *a posteriori* IMplicit-EXplicit (IMEX) error indicator – required to eliminate the need for human intervention in a fully adaptive solution procedure. We then follow the well-established *solve-estimate-mark-refine* paradigm and present our attempt at the *hp*-adaptive solution procedure in detail in Section 3.3, which confirms hypotheses **H1** and **H3** – **H6**. The performance of the proposed *hp*-adaptive solution procedure is demonstrated on a set of linear elasticity problems from two- and three-dimensional domains, where improved computational efficiency of the solution procedure is observed. Perhaps more importantly, we demonstrate that the proposed *hp*-adaptive solution procedure is able to obtain numerical solutions to a set of contact problems that are normally difficult or impossible to solve without a supercomputer.

Another potential strategy to improve the computational efficiency of the solution procedure is explored in Chapter 4. Our idea is to exploit the distinctive characteristics of different approximation methods. For example, the RBF-FD approximation method is more resistant to non-optimal node positions [26], [109], [110], while WLS is computationally less demanding [111]. Combining both characteristics can lead to improved computational efficiency without notably compromising the accuracy of the numerical solution. Therefore, in Section 4.1, we propose a hybrid WLS–RBF-FD method where the computational nodes are divided into two sets: one set using the WLS approach to approximate differential operators where stability is not critical, and the other set of computational nodes using the RBF-FD approach in domain segments that are possibly detrimental from a stability point of view. We confirm hypotheses **H7** and **H2** and show that such spatially-variable

approximation method (i) reduces the computational complexity compared to the pure RBF-FD approximation, (ii) improves the stability of the solution procedure compared to the pure WLS approximation, and (iii) minimally affects the accuracy of the numerical solution.

Hypothesis **H7** is reaffirmed in Section 4.2, where we discuss an approach enhancing the computational efficiency by spatially-varying the regularity of computational nodes. Although meshless methods are formulated without any restrictions regarding the node layouts, it is generally accepted that quasi-uniformly distributed nodes improve the stability of meshless methods. However, even with quasi-uniform nodes generated with state-of-the-art algorithms, a stable approximation still requires a sufficiently large stencil size. For instance, for a stable second-order RBF-FD approximation in a two-dimensional domain, a stencil with $n = 12$ nodes is recommended [30]. On the other hand, the approximation of same order on uniform nodes can be performed with a much smaller stencil, such as $n = 5$ in a two-dimensional domain, using only a monomial basis [17]. Considering that a dense linear system must be solved for the approximation of each linear differential operator, leading to a time complexity of $\mathcal{O}(n^3)$ and an additional $\mathcal{O}(n)$ for the evaluation, smaller stencil sizes are clearly preferred in terms of shorter wall-clock times. Thus, to enhance the overall computational efficiency while considering the discretization-related errors, we propose a hybrid discretization that includes uniform and scattered nodes. Uniform nodes, which use the computationally cheap MON approximation, are placed far from geometric irregularities in the domain, while scattered nodes, which use the stable but computationally expensive RBF-FD approximation, are used in their vicinity. In contrast to previous hybrid discretization or spatially-varying approximation method attempts, both approaches presented in this dissertation rely solely on the framework of meshless methods and avoid additional treatment on the transitions.

Finally, some conclusions and future work opportunities are given in Chapter 5.

1.5 Contributions

The dissertation is composed of several published scientific papers. All scientific contributions of the dissertation are summarized as follows:

- C1:** There exists a favoured monomial degree m , which should be used in the RBF-FD approximation basis to minimize the execution times and achieve the desired accuracy of the numerical solution.
- C2:** We demonstrate that a minor adjustment to the stencil size can result in a notable change of accuracy of the numerical solution. We also report on observing interesting oscillatory behaviour of the solution error.
- C3:** The high-order RBF-FD approximations are generally more stable and more accurate than the WLS approximations. This makes the RBF-FD a better candidate for the development of *hp*-adaptive solution procedure where stability of the high-order approximation is crucial.
- C4:** A well-thought-out spatial distribution of approximation orders can be beneficial in terms of stability and computational complexity of the solution procedure, and accuracy of the numerical solution.
- C5:** We develop an original error indicator easily implemented in the context of meshless methods.

- C6:** Verification and performance analyses of the IMEX error indicator, when applied to two- and three-dimensional linear elasticity problems, demonstrating its versatile use.
- C7:** We provide a detailed description of all building blocks required to employ a *hp*-adaptive meshless solution procedure based on the RBF-FD approximations.
- C8:** Spatially-varying the approximation method has desirable effects on the stability of the solution procedure.
- C9:** Spatially-varying node regularity accompanied with spatially-variable approximation method can improve the computational efficiency with a negligible cost to the accuracy of the numerical solution.

Chapter 2

The Fundamental Properties of Meshless Approximation Methods

In the context of meshless methods, linear differential operator $\mathcal{L}$ at a specific point $\mathbf{x}_c$ is approximated over n stencil nodes $\mathbf{x}_i$ using the expression:

$$\mathcal{L}u(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i), \quad (2.1)$$

where u is an arbitrary smooth function and weights w_i are the unknowns. The equality of the above equation is enforced for a given set of basis functions, yielding a dense linear system with weights w_i as the only unknowns [13], [31], [66]. In general, the computation of weights requires $\mathcal{O}(n^3)$ operations and an additional $\mathcal{O}(n)$ for evaluation of the approximated differential operator.

Throughout the history, various meshless methods for numerical treatment of systems of PDEs have been introduced. In this dissertation, the focus is shifted towards three frequently employed approximation methods:

- the Radial Basis Function-generated Finite Differences (RBF-FD) [112],
- the Diffuse Approximation Method (DAM) [48] also referred to as the Weighted Least Squares (WLS) approach, and
- the simplest version of collocation methods operating on uniform nodes [113] referred to as MON throughout this thesis.

All three approximation methods fall under the category of strong-form meshless methods used for solving boundary value problems. RBF-FD is recognized for its stability on scattered nodes [66], but at the cost of high computational complexity due to larger stencil size requirements for stable approximations [30], [31] and due to large approximation basis¹. In comparison, WLS is less susceptible to non-optimal domain discretizations [111] but computationally more efficient due to its smaller approximation basis². The MON approximation is the cheapest among the three methods (requiring a stencil size of $n = 5$ compared to the recommended $n = 12$ for RBF-FD in a two-dimensional domain and a second-order approximation). However, it is also considered to be the least stable, as it can only operate on uniform nodes.

¹The RBF-FD approximation basis in this thesis is limited to consist of Polyharmonic Splines (PHS) augmented with monomials.

²In this thesis, the WLS approximation basis consists solely of monomials up to degree m .

The objective of this chapter is to examine the high-order approximation methods, with a specific focus on understanding the fundamental characteristics of commonly used RBF-FD and WLS approximation methods. By analysing both, we aim to shed some light on their capabilities, limitations, and potentials for increasing the computational efficiency, particularly for application in *hp*-adaptive solution procedures presented in Chapter 3 and spatially-varying approximation methods presented in Chapter 4.

2.1 Monomial Augmentation Guidelines for RBF-FD from Accuracy Versus Computational Time Perspective

A common approach to increase the accuracy of the numerical solution is to increase the approximation order [30], [32], [66], [100]. In meshless methods, the control over the approximation order is enabled through approximation basis, as the highest monomial from the approximation basis also determines the approximation order [30], [31]. However increasing the approximation order also increases the computational complexity, as more monomials are added to the approximation basis. This is particularly evident in case of the RBF-FD approximation, as the recommended stencil size n for a stable approximation depends on the highest monomial order m in a d -dimensional basis following the expression

$$n = 2 \binom{m+d}{d}, \quad (2.2)$$

as discussed by Bayona [30].

Therefore, for improved computational efficiency, employing the lowest monomials is theoretically preferred. However, after an in-depth investigation of the approximation order and its impact on the numerical solution and solution procedure, it becomes evident that higher monomial degrees can lead to a faster attainment of the numerical solution of the desired accuracy for a given domain discretization.

Contributions.

This section represents contribution **C1**. The publication below demonstrates that there exists a favoured monomial degree m , which should be used in the RBF-FD approximation basis to minimize the execution times and achieve the target accuracy of the numerical solution. We provide a rule of thumb to estimate the required monomial degree value m , which yields the desired accuracy of the solution in a time-efficient manner.

Additionally, by providing a solution to a four-dimensional Poisson problem, we also demonstrate that the meshless methods are indeed dimension-agnostic.

Addressed hypotheses.

Our analyses reveal two findings. Firstly, we establish that the approximation order can be effectively controlled with the highest monomial degree m utilized in the approximation basis. Secondly, we emphasize that choosing the appropriate approximation order becomes non-trivial when both the computational efficiency of the solution procedure and accuracy of the numerical solution are considered. This observation confirms hypothesis **H1**.

The following publication also supports hypothesis **H3**, indicating that the RBF-FD approximation is stable and accurate even for high-order approximations.

Publications included in this section:

- M. Jančič, J. Slak, and G. Kosec, “Monomial augmentation guidelines for RBF-FD from accuracy versus computational time perspective,” *Journal of Scientific Computing*, vol. 87, no. 1, pp. 1–18, 2021

Regarding my contribution: I made a literature overview of the topic, participated in planning, performed the experiments and jointly prepared the manuscript with co-authors.

Note: During the proof-reading process of the publication below, a typo was overlooked. In the publication process, all binomial symbols were replaced by fractional expressions. Therefore, while reading the following publication, keep in mind that the number of monomials is $s = \binom{m+d}{d}$ and not $s = \frac{m+d}{d}$ as denoted in the publicly available version.



Monomial Augmentation Guidelines for RBF-FD from Accuracy Versus Computational Time Perspective

Mitja Jančič^{1,2} · Jure Slak^{2,3} · Gregor Kosec²

Received: 21 January 2020 / Revised: 30 October 2020 / Accepted: 26 December 2020 /

Published online: 17 February 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC part of Springer Nature 2021

Abstract

Local meshless methods using RBFs augmented with monomials have become increasingly popular, due to the fact that they can be used to solve PDEs on scattered node sets in a dimension-independent way, with the ability to easily control the order of the method, but at a greater cost to execution time. We analyze this ability on a Poisson problem with mixed boundary conditions in 1D, 2D and 3D, and reproduce theoretical convergence orders practically, also in a dimension-independent manner, as demonstrated with a solution of Poisson's equation in an irregular 4D domain. The results are further combined with theoretical complexity analyses and with conforming execution time measurements, into a study of accuracy versus execution time trade-off for each dimension. Optimal regimes of order for given target accuracy ranges are extracted and presented, along with guidelines for generalization.

Keywords Meshless methods · RBF-FD · Poisson's equation · n -dimensional · Convergence rates · Optimal order

1 Introduction

The Radial Basis Function-generated Finite Differences (RBF-FD), a local strong form mesh-free method for solving partial differential equations (PDEs) that generalizes the traditional Finite Difference Method (FDM), was first mentioned by Tolstykh [32]. Since then, the method has become increasingly popular [9], with recent uses in linear elasticity [31], con-

✉ Mitja Jančič
Mitja.Jancic@ijs.si

Jure Slak
Jure.Slak@ijs.si

Gregor Kosec
Gregor.Kosec@ijs.si

¹ “Jožef Stefan” International Postgraduate School, Jamova cesta 39, 1000 Ljubljana, Slovenia

² “Jožef Stefan” Institute, E6, Parallel and Distributed Systems Laboratory, Jamova cesta 39, 1000 Ljubljana, Slovenia

³ Faculty of Mathematics and Physics, University of Ljubljana, Jadranska 19, 1000 Ljubljana, Slovenia

tact problems [28], geosciences [8], fluid mechanics [14], dynamic thermal rating of power lines [20], advection-dominated problems [21,25], financial sector [22], etc.

RBF-FD, similarly to other mesh-free methods, relies on approximation of differential operators on scattered nodes, which is an important advantage over mesh-based methods, as node generation is considered much easier than the mesh generation. In fact, mesh generation is often the most cumbersome part of the solution procedure in traditional methods, which, especially in 3D geometries, often requires significant assistance from the user. When meshless methods were first developed, many solutions used available mesh generators for generating discretization nodes and discarding the connectivity information after the mesh had been generated [16]. Such approach is computationally wasteful, does not generalize to higher dimensions, and some authors even reported that it failed to generate distributions of sufficient quality [26]. Since then, various node positioning algorithms have been proposed. Popular algorithms use iterative approaches [12,17], advancing front methods [7,18] or sphere packing methods [5]. In 2018, a pure meshless algorithm based on Poisson disk sampling [4] was introduced. Later that year, the first dimension-independent node generation algorithm that supported distributions with spatially variable density appeared [30], where the authors also demonstrated the stability of RBF-FD on scattered nodes, even for complex non-linear problems in 3D without any special treatment of stencil selection as proposed in [23]. Instead, a cluster of nearest neighboring nodes proved to be a satisfactory stencil that can also be efficiently implemented in dimension-independent code, using specialized data structures, such as k -d tree [35].

A common drawback of often used RBFs, such as Gaussians or Hardy's multiquadrics, is that they include a shape parameter that crucially affects accuracy and stability of the approximation [33]. If the shape parameter is kept constant, the method converges, but stability issues arise when computing in the standard basis, due to high condition numbers of the collocation matrices. To fix the stability issue, more sophisticated algorithms can be used, such as RBF-CP, RBF-QA, RBF-GA and others [34], but such methods sometimes add significant additional costs. A simpler solution for the stability issue is to scale the shape parameter so that the product of the shape parameter and the nodal spacing is constant. However, this can lead to local approximations that are not convergent - this phenomenon has been called lack of convergence due to stagnation errors [6]. Stagnation can be fixed by adding monomial terms that ensure consistency up to a certain order. This technique has been used together with Polyharmonic splines (PHS) as RBFs, which have an additional advantage of not having a shape parameter [3]. In addition, the order of added monomials directly effects the order of the RBF-FD approximation, effectively enabling control over the convergence rate of the RBF-FD [2]. Various successful applications of RBF-FD with PHS have since been demonstrated, both in 2D and in 3D [3,26,30]. The dimensional independence has already been noted by, e.g., Ahmad et al. [1], but the high order RBF-FD has not yet been thoroughly analyzed with computational efficiency in mind, as the authors were more focused on solving the time-dependent part of the PDE of interest.

Although the RBF-FD formulation is dimension-independent, in the sense that the same formulation can be used in 1D, 2D, 3D and higher, translating this elegant mathematical formulation and algorithms into actual efficient computer code is far from trivial. In this paper, we present a dimension-independent PDE solution procedure based on our in-house dimension-agnostic implementation [29] of RBF-FD. By dimension-agnostic implementation we refer to the fact that exactly the same code can be used to solve problems in one, two, three or more dimensional spaces, while values of parameters are optimised for each dimension separately. The paper describes all solution procedure elements in detail and presents a thorough analysis of accuracy and execution time in one, two and three dimensions, on a

Poisson problem on scattered nodes with mixed boundary conditions. To fully illustrate the dimension independence, a solution of a 4-dimensional problem on an irregular domain is presented. A C++ implementation of all discussed solution elements is freely available for download [13].

The rest of the paper is organized as follows: In Sect. 2, the RBF-FD solution procedure is presented, in Sect. 3, the model problem is investigated, in Sect. 4, an additional example is shown, and in Sect. 5, the conclusions are presented.

2 RBF-FD Solution Procedure

In this section, the main steps of the RBF-FD solution procedure are described. First, the domain is populated with scattered nodes. Once the nodes are positioned, in each discretization node the approximation of the partial differential operator is performed, resulting in stencil weights. Finally, in the PDE discretization phase, the PDE is transformed into a system of linear equations, whose solution stands for a numerical solution of the considered PDE.

2.1 Positioning of Nodes

In the node generation algorithm, candidate nodes are generated on a d -sphere in a d -dimensional space. This effectively means that the node positioning algorithm remains the same for every number of dimensions. However, some parameters, e.g. the number of candidates, can be optimized for various numbers of dimensions.

The node positioning algorithm takes as an input a domain $\Omega \subset \mathbb{R}^d$ with a spacing function $h: \Omega \rightarrow (0, \infty)$ and optionally a list of arbitrary starting “seed nodes” $X \subset \Omega$, often distributed along the boundary. It returns a set of nodes that are suitable for strong-form discretizations and distributed over Ω with mutual spacing around a point p approximately $h(p)$.

The algorithm used in this paper processes nodes in the input list in order. For each node p , a number of expansion candidates distributed uniformly on a sphere centered at p , of radius $h(p)$, are examined. If a candidate is inside the domain and sufficiently away from the already processed nodes, it is accepted and added to the list X . During the course of the algorithm, the list X is implicitly partitioned into already processed nodes, the current node, and future queued nodes. Figure 1 shows this partition at a selected iteration in 2D and 3D, along with the generated candidates from the current node, and flags the accepted ones.

Once all the elements of the list X have been processed, X is returned as the resulting set of discretization nodes. Further details and analyses of the algorithm are available in [30]. The stand-alone implementation of the algorithm is available online [27] and also included as a part of our in-house implementation of RBF-FD, the *Medusa* library [29].

2.2 Approximation of Partial Differential Operators

Consider a partial differential operator $\mathcal{L}$ at a point $\mathbf{x}_c$. Approximation of $\mathcal{L}$ at a point $\mathbf{x}_c$ is sought using an ansatz

$$(\mathcal{L}u)(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i), \quad (1)$$

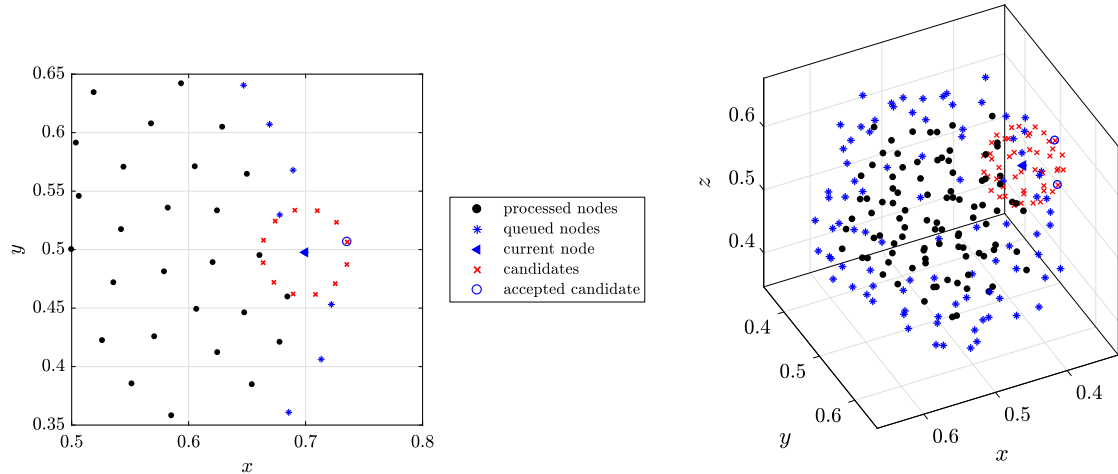


Fig. 1 Node positioning algorithm during candidate generation phase

where $\mathbf{x}_i$ are the neighboring nodes of $\mathbf{x}_c$ which constitute its *stencil*, w_i are called *stencil weights*, n is the *stencil size* and u is an arbitrary function.

This form of approximation is desirable, since operator $\mathcal{L}$ at point $\mathbf{x}_c$ is approximated by a linear functional $\mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)^\top$, assembled of weights w_i ,

$$\mathcal{L}|_{\mathbf{x}_c} \approx \mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)^\top \quad (2)$$

and the approximation is obtained using just a dot product with the function values in neighboring nodes. The dependence of $\mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)^\top$ on $\mathcal{L}$ and $\mathbf{x}_c$ is often omitted, with $\mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)^\top$ written simply as $\mathbf{w}$.

To determine the unknown weights $\mathbf{w}$, equality of (1) is enforced for a given set of basis functions. A natural choice are monomials, which are also used in FDM, resulting in the Finite Point Method [24]. However, using monomial basis suffers from potential ill conditioning [19]. An alternative approach is using an RBF basis.

In the RBF-FD discretization, the equality is satisfied for radial basis functions ϕ_j . These are RBFs, generated by a function $\phi: [0, \infty) \rightarrow \mathbb{R}$, centered at neighboring nodes of $\mathbf{x}_c$, given by

$$\phi_j(\mathbf{x}) = \phi(\|\mathbf{x} - \mathbf{x}_j\|). \quad (3)$$

Each ϕ_j , for $j = 1, \dots, n$, corresponds to one linear equation

$$\sum_{i=1}^n w_i \phi_j(\mathbf{x}_i) = (\mathcal{L}\phi_j)(\mathbf{x}_c) \quad (4)$$

for unknowns w_i . Assembling these n equations into matrix form, we obtain the following linear system:

$$\begin{bmatrix} \phi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_1\|) \\ \vdots & \ddots & \vdots \\ \phi(\|\mathbf{x}_1 - \mathbf{x}_n\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_n\|) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} (\mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_1\|))|_{\mathbf{x}=\mathbf{x}_c} \\ \vdots \\ (\mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_n\|))|_{\mathbf{x}=\mathbf{x}_c} \end{bmatrix}, \quad (5)$$

where ϕ_j have been expanded for clarity.

The above system can be written more compactly as

$$\mathbf{A}\mathbf{w} = \boldsymbol{\ell}_\phi. \quad (6)$$

The matrix $\mathbf{A}$ is symmetric, and for some basis functions ϕ even positive definite [33].

Many commonly used RBFs, such as Hardy's multiquadrics or Gaussians, depend on a shape parameter, which governs their shape and consequently affects the accuracy and stability of the approximation. In this work, we use polyharmonic splines (PHS), defined as

$$\phi(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \quad (7)$$

to eliminate the need for a shape parameter tuning where r denotes the Euclidean distance between two nodes. Without monomial augmentation, local approximations using only PHS are not convergent, nor do we have any guarantees of solvability. However, if the approximation given by (5) is augmented with polynomials, we obtain convergence and conditional positive definiteness, provided that the stencil nodes form a polynomially unisolvent set [33]. Augmentation is performed as follows: Let $p_1, \dots, p_s$ be polynomials forming the basis of the space of d -dimensional multivariate polynomials up to and including total degree m , with $s = \frac{m+d}{d}$. In addition to the RBF part of the approximation, an exactness constraint

$$\sum_{i=1}^s w_i p_j(\mathbf{x}_i) = (\mathcal{L}p_j)(\mathbf{x}_c) \quad (8)$$

for monomials, is enforced. These additional constraints make the approximation overdetermined, which is treated as a constrained optimization problem [6]:

$$\min_{\mathbf{w}} \left(\frac{1}{2} \mathbf{w}^\top \mathbf{A} \mathbf{w} - \mathbf{w}^\top \ell_\phi \right), \text{ subject to } \mathbf{P}^\top \mathbf{w} = \ell_p. \quad (9)$$

For practical computation, the optimal solution can be expressed as a solution of a linear system

$$\begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \ell_\phi \\ \ell_p \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} p_1(\mathbf{x}_1) & \cdots & p_s(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_n) & \cdots & p_s(\mathbf{x}_n) \end{bmatrix}, \quad \ell_p = \begin{bmatrix} (\mathcal{L}p_1)|_{\mathbf{x}=\mathbf{x}_c} \\ \vdots \\ (\mathcal{L}p_s)|_{\mathbf{x}=\mathbf{x}_c} \end{bmatrix}, \quad (10)$$

where $\mathbf{P}$ is a $n \times s$ matrix of polynomials evaluated at stencil nodes, ℓ_p is the vector of values assembled by applying the considered operator $\mathcal{L}$ to the polynomials at $\mathbf{x}_c$, and $\boldsymbol{\lambda}$ are Lagrange multipliers. Weights obtained by solving (10) are taken as approximations of $\mathcal{L}$ at $\mathbf{x}_c$, while values $\boldsymbol{\lambda}$ are discarded. The system (10) is solvable if the stencil nodes form a polynomially unisolvent set. This could potentially be problematic near the boundary, where it might happen that all stencil nodes would be e.g. colinear or coplanar, but experience shows that this happens only with stencil sizes which are too small to be a feasible approximation. With large enough stencil sizes, stencils near the boundary always include at least some internal nodes. We did not use any special techniques to ensure unisolvency, and did not run into any unisolvency-related issues.

The exactness of (8) ensures convergence behavior and control over the convergence rate, since the local approximation has the same order as the polynomial basis used [3], while the RBF part of the approximation (5) takes care of potential ill-conditioning in purely polynomial approximation [6].

2.3 PDE Discretization

Consider the boundary value problem

$$\mathcal{L}u = f \text{ in } \Omega, \quad (11)$$

$$u = g_d \text{ on } \Gamma_d, \quad (12)$$

$$\mathbf{n} \cdot \nabla u = g_n \text{ on } \Gamma_n, \quad (13)$$

with $\partial\Omega = \Gamma_d \cup \Gamma_n$, where the union is disjoint. The domain Ω is discretized by placing N scattered nodes $\mathbf{x}_i$ with quasi-uniform internodal spacing h , of which N_i are in the interior, N_d on the Dirichlet and N_n on the Neumann boundary. Additionally, N_g *ghost* or *fictitious* nodes are added outside the domain on both Neumann and Dirichlet boundary, by translating the N_d and the N_n nodes on $\partial\Omega$ for distance h in the normal direction.

In the next step, stencils $\mathcal{N}(\mathbf{x}_i)$ consisting of neighboring nodes are selected for each node $\mathbf{x}_i$. The most common approach is to compute stencils automatically, by taking n closest nodes for each node (including the node itself) as its stencil.

Next, partial differential operators appearing in the problem, such as $\mathcal{L}$ and ∂_i , are approximated at nodes $\mathbf{x}_i$, using the procedure described in Sect. 2.2. The computed stencils $\mathbf{w}_{\mathcal{L}}$ and $\mathbf{w}_{\partial_i}$ are stored for later use.

For each interior node $\mathbf{x}_i$, the equation $(\mathcal{L}u)(\mathbf{x}_i) = f(\mathbf{x}_i)$ is approximated by a linear equation

$$\mathbf{w}_{\mathcal{L}}(\mathbf{x}_i)^{\top} \mathbf{u} = f, \quad (14)$$

where vectors $\mathbf{f}$ and $\mathbf{u}$ represent values of function f and unknowns u in stencil nodes of $\mathbf{x}_i$. For each Dirichlet boundary node $\mathbf{x}_i$, we have the equation

$$u_i = g_d(\mathbf{x}_i). \quad (15)$$

For Neumann boundary nodes $\mathbf{x}_i$, the linear equation

$$\sum_{j=1}^d n_j \mathbf{w}_{\partial_j}(\mathbf{x}_i)^{\top} \mathbf{u} = g_d \quad (16)$$

approximates the boundary condition, where similarly to before, vectors $\mathbf{g}_d$ and $\mathbf{u}$ represent values of function g_d and unknowns u in stencil nodes of $\mathbf{x}_i$. Another set of N_g equations is needed to determine the unknowns introduced by ghost nodes. Additionally to (15) and (16), we also enforce (14) to hold for boundary nodes.

All $N_i + N_d + N_n + N_g$ equations are assembled into a sparse system with $n(N_i + N_n + N_g) + N_d$ non-zero elements in general. The solution u_h of this system is a numerical approximation of u , excluding the values obtained in ghost nodes.

2.4 Note on Implementation

We implemented the solution procedure described in this section in C++ using object oriented approach and C++'s strong template system to achieve modularity and consequent dimension independence. The strongest advantage of the presented method is that all building blocks, namely *node positioning*, *stencil selection*, *differential operator approximation* and *PDE discretization*, are independent and can therefore be elegantly coded as abstract modules, not knowing about each other in the core of their implementation. To ease the implementation of the solution procedure, additional abstractions, such as *operators*, *basis functions*, *domain*

shapes and *approximations*, are introduced, acting as interfaces between the main blocks. For example, to construct a RBF-FD approximation, one combines the RBF basis class with an augmented RBF-FD class, computes stencil weights and supplies the computed weights into the “operators” class that enables the user to explicitly transform governing equations into the C++ code, as demonstrated in the listing 1.

```
// Define differential operator approximation.
Monomials<vec> mon(m);
Polyharmonic<double, k> ph;
RBFFD<decltype(ph), vec, ScaleToFarthest> appr(ph, mon);

// Compute stencil weights (shapes) with RBF-FD.
auto storage = domain.computeShapes<sh::lap|sh::d1>(appr);
Eigen::SparseMatrix<double, Eigen::RowMajor> M(N, N);
M.reserve(storage.supportSizes());
Eigen::VectorXd rhs(N); rhs.setZero();

// Prepare "operators" abstraction.
auto op = storage.implicitOperators(M, rhs);

// PDE discretization.
// Interior.
for (int i : interior) {
    op.lap(i) = f_lap(domain.pos(i));
}
// Dirichlet boundary.
for (int i : dir) {
    op.value(i) = f(domain.pos(i));
    op.lap(i, gh[i]) = f_lap(domain.pos(i));
}
// Neumann boundary.
for (int i : neu) {
    op.neumann(i, domain.normal(i)) = f_grad(domain.pos(i));
    op.lap(i, gh[i]) = f_lap(domain.pos(i));
}
```

Listing 1: A part of dimension-independent source code showing definition and sparse system assembly.

Vector and scalar fields are implemented as plain arrays, using a well developed linear algebra library [11] that also implements or otherwise supports various direct and iterative linear solvers. Please, refer to our open source Medusa library [29] for more examples and features.

3 Numerical Example

The behavior of the proposed solution procedure and its implementation are studied on a Poisson problem with mixed boundary conditions. The aim is to analyze accuracy and convergence properties in one, two and three dimensions. Furthermore, theoretical computational complexity is discussed and supported by experimental measurements of execution time, which allows us to quantify the accuracy versus execution time trade-off.

The problem is solved on an irregular domain Ω , defined as $\Omega = (B_0 \cup B_1) \setminus (B_2 \cup B_3)$, where

$$B_0 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \frac{\mathbf{1}}{2} \right\| < \frac{1}{2} \right\}, \quad (17)$$

$$B_1 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \frac{\mathbf{1}}{5} \right\| \leq \frac{1}{4} \right\}, \quad (18)$$

$$B_2 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \frac{\mathbf{1}}{2} \right\| \leq \frac{1}{10} \right\} \text{ and} \quad (19)$$

$$B_3 = \left\{ \mathbf{x} \in \mathbb{R}^4, \|\mathbf{x} - \mathbf{1}\| \leq \frac{1}{2} \right\} \quad (20)$$

are balls in $\mathbb{R}^d$. For later use, the boundary $\partial\Omega$ is divided into Γ_d and Γ_n , the left and the right half of the boundary, respectively

$$\Gamma_d = \left\{ \mathbf{x} \in \partial\Omega, x_1 < \frac{1}{2} \right\}, \quad (21)$$

$$\Gamma_n = \left\{ \mathbf{x} \in \partial\Omega, x_1 \geq \frac{1}{2} \right\}. \quad (22)$$

3.1 Governing Equation

Numerical solution u_h of Poisson's equation with both Dirichlet and Neumann boundary conditions is studied:

$$\nabla^2 u(\mathbf{x}) = f_{lap}(\mathbf{x}) \quad \text{in } \Omega, \quad (23)$$

$$u(\mathbf{x}) = f(\mathbf{x}) \quad \text{on } \Gamma_d, \quad (24)$$

$$\nabla u(\mathbf{x}) = \mathbf{f}_{grad}(\mathbf{x}) \quad \text{on } \Gamma_n. \quad (25)$$

Here, the right hand side was chosen as

$$f(\mathbf{x}) = \frac{E(\mathbf{x})}{g(\mathbf{x})}, \quad (26)$$

where

$$E(\mathbf{x}) = \exp\left(\sum_{i=1}^d x_i^{a_i}\right), \quad g(\mathbf{x}) = 1 + \mathbf{x}^\top \mathbf{H} \mathbf{x}, \quad a_i = 2 + i, \quad (27)$$

$\mathbf{H}$ is a Hilbert matrix of size d , and $\hat{\mathbf{e}}_i$ is the i -th unit vector. The values of Laplacian and the gradient are computed from f as

$$\begin{aligned} f_{lap}(\mathbf{x}) &= \frac{8E(\mathbf{x})}{g(\mathbf{x})^3} (\mathbf{H}\mathbf{x})^\top (\mathbf{H}\mathbf{x}) - \frac{2E(\mathbf{x})}{g(\mathbf{x})^2} \left[2(\mathbf{H}\mathbf{x})^\top \left(\sum_{i=1}^d a_i x_i^{a_i-1} \hat{\mathbf{e}}_i \right) + \text{Tr}(\mathbf{H}) \right] \\ &\quad + \frac{E(\mathbf{x})}{g(\mathbf{x})} \left[\sum_{i=1}^d a_i (a_i - 1) x_i^{a_i-2} + \left(\sum_{i=1}^d a_i x_i^{a_i-1} \hat{\mathbf{e}}_i \right)^\top \left(\sum_{i=1}^d a_i x_i^{a_i-1} \hat{\mathbf{e}}_i \right) \right], \end{aligned} \quad (28)$$

$$\mathbf{f}_{grad}(\mathbf{x}) = \frac{E(\mathbf{x})}{g(\mathbf{x})} \left[\sum_{i=1}^d a_i x_i^{a_i-1} \hat{\mathbf{e}}_i - \frac{2}{g(\mathbf{x})} (\mathbf{H}\mathbf{x})^\top \right]. \quad (29)$$

The closed-form solution f of the above problem is a rational non-easily separable function allowing us to validate the numerically obtained solution u_h . The computed u_h is only known at discretization points $\mathbf{x}_i$. The errors between u_h and u are measured in three different norms:

$$e_1 = \frac{\|u_h - u\|_1}{\|u\|_1}, \quad \|u\|_1 = \frac{1}{N} \sum_{i=1}^N |u_i|, \quad (30)$$

$$e_2 = \frac{\|u_h - u\|_2}{\|u\|_2}, \quad \|u\|_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N |u_i|^2}, \quad (31)$$

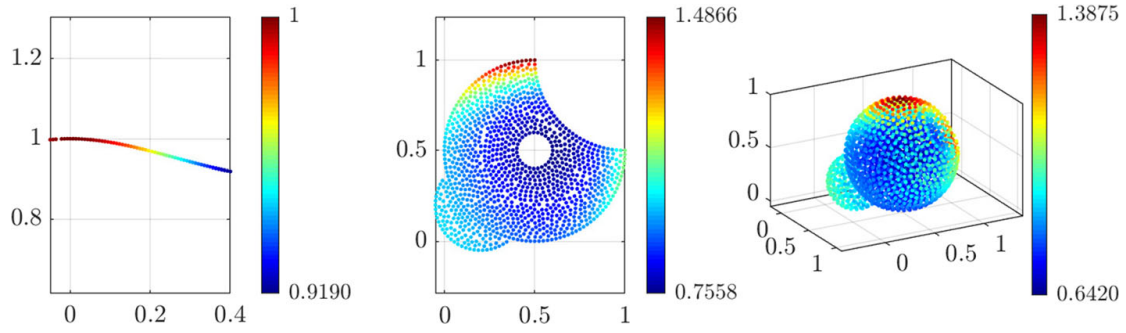


Fig. 2 Computed numerical solution u_h for $d = 1, 2, 3$, from left to right. Chosen highest polynomial degree m and node count N are as follows: $N = 64$ and $m = 4$ for $d = 1$, $N = 1286$ and $m = 2$ for $d = 2$ and $N = 3850$ and $m = 4$ for $d = 3$

Table 1 Support sizes in different dimensions for various augmentation orders

m	$d = 1$	$d = 2$	$d = 3$
-1	3	5	7
0	3	5	7
2	6	12	20
4	10	30	70
6	14	56	168
8	18	90	330

$$e_\infty = \frac{\|u_h - u\|_\infty}{\|u\|_\infty}, \quad \|u\|_\infty = \max_{i=1,\dots,N} |u_i|. \quad (32)$$

The problem (23–25) is studied in $d \in \{1, 2, 3\}$ dimensions. Scattered computational nodes are generated using a dimension-agnostic node positioning algorithm described in Sect. 2.1. Ghost nodes were added to both Dirichlet and Neumann boundaries, and are excluded from any post-processing. An example of node distribution is shown in Fig. 2.

Numerical results are computed using RBF-FD with PHS radial basis function $\phi(r) = r^3$ and monomial augmentation, as described in Sect. 2. Radial function was kept same for all cases; however, various orders of monomial augmentation were tested. For each dimension d , solution to the problem is obtained using monomials up to and including degree m , for $m \in \{-1, 0, 2, 4, 6, 8\}$, where $m = -1$ represents a pure RBF case with no monomials added. Only even orders of m were used, because the same order of convergence is observed with odd powers, but at a higher computational cost [6].

Stencils for each node were selected by taking the closest n nodes, where n was equal to two times the number of augmenting monomials, as recommended by Bayona [3], or at least a FDM minimum of $2d + 1$, i.e.

$$n = \max \left\{ 2 \frac{m+d}{d}, 2d+1 \right\}. \quad (33)$$

Specific values for m , n and d are presented in Table 1.

BiCGSTAB with ILUT preconditioner was used to solve the sparse system. Global tolerance was set to 10^{-15} with a maximum number of 500 iterations, while the drop tolerance and fill-factor were dimension dependent: 10^{-4} and 20 for $d = 1$, 10^{-4} and 30 for $d = 2$, and 10^{-5} and 50 for $d = 3$, respectively.

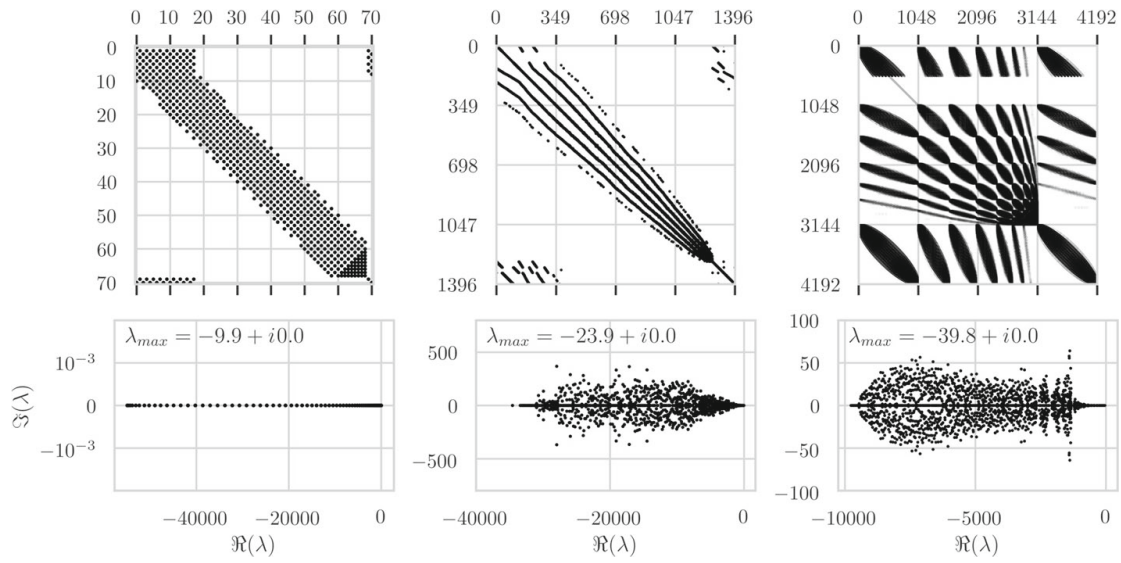


Fig. 3 Plots of global sparse matrices (top row) and spectra of the Laplacian differentiation matrices (bottom row), corresponding to the solutions in Fig. 2

Figure 2 shows three examples of computed numerical solution u_h for each domain dimension d . The solutions are shown for various values of m and for small enough values of N to also show nodal distributions.

In the top row of Fig. 3 global sparse matrices are shown. Additionally, spectra of the Laplacian differentiation matrices for cases shown in Fig. 2 are shown in the bottom row of Fig. 3, to better assess the approximation quality. For all three cases, the eigenvalues have negative real parts with relatively small spread around the imaginary axis.

3.2 Convergence Rate

When using RBF-FD augmented with monomials, consistency is ensured up to order m , which makes the expected convergence rate of at least $O(h^m)$. Here, h denotes the nodal spacing, which is inversely proportional to $\sqrt[d]{N}$.

Figure 4 shows e_1 , e_2 and e_∞ errors for various augmentation orders in two dimensions. The three errors have very similar values and similar convergence rates. Convergence rates were estimated by computing the slope of a least-squares linear trend line over the appropriate subset of the data. Divergence is observed in the $m = 0$ and the $m = -1$ case, which is consistent with properties of PHS RBFs. These two cases are excluded from any further analyses in this paper.

In the rest of the discussion, only e_∞ is used for convergence analysis, since it measures the lowest convergence rates and does not involve averaging, contrary to e_1 and e_2 .

Figure 5 shows the e_∞ error for $d = 1$, $d = 2$, and $d = 3$ dimensions. The span of the horizontal axis was chosen in such a way that the total number of nodes in the largest case was around $N = 10^5$ in all dimensions. The observed convergence rates are independent of domain dimension and match the predicted order $O(h^m)$.

All of the plots in the $d = 1$ case eventually diverge, due to the errors in finite precision arithmetic, as previously noted for interpolation by Flyer et al. [6]. The dotted line in the $d = 1$ case shows the ε/h^2 line, where $\varepsilon \approx 2.22 \cdot 10^{-16}$. The numerically obtained solution

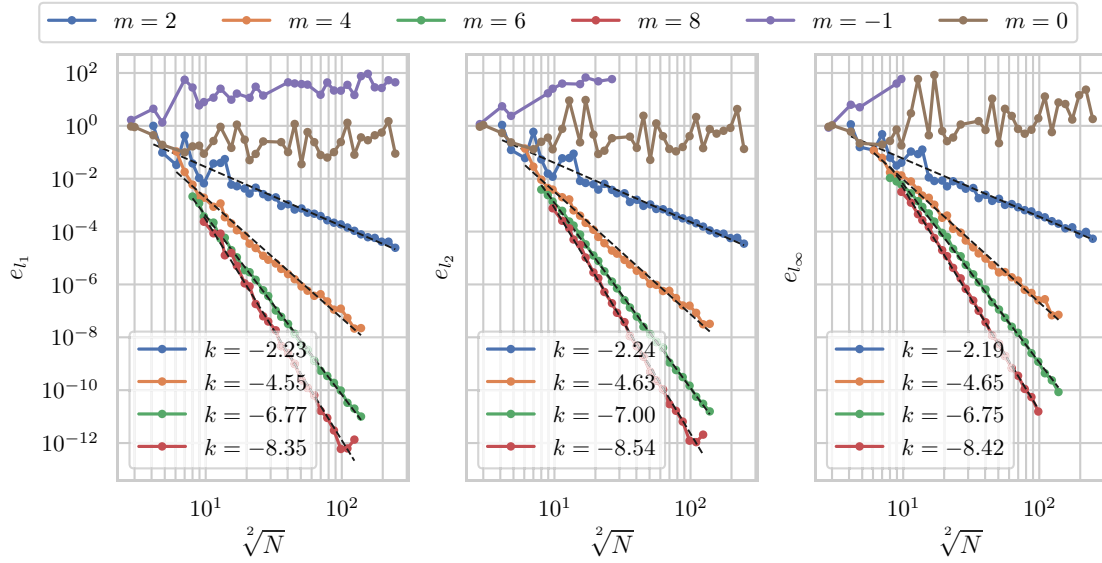


Fig. 4 Errors between analytical solution u and numerically obtained u_h , measured in three different norms. Computed are e_1 , e_2 and e_∞ , from left to right, respectively, for the $d = 2$ dimensional case

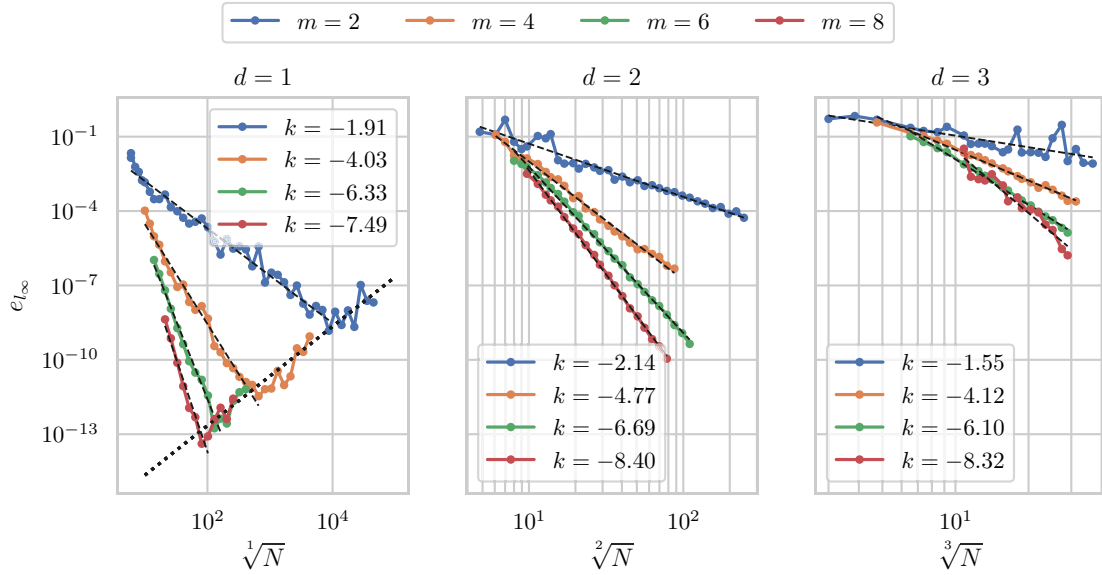


Fig. 5 Convergence rate of e_∞ for all domain dimensions $d = 1, 2, 3$, from left to right, respectively

for the $d = 3$ and $m = 8$ case is unstable for smaller N . For higher node counts N , the expected convergence behavior is obtained, as seen from the fitted dashed line.

3.3 Computational Efficiency

The importance of several different stages of u_h computation is studied. The computational procedure is divided into

- *node positioning*, where quasi-uniform placing of nodes in the domain Ω and the domain boundary $\partial\Omega$, including positioning of N_g ghost nodes, takes place. Node positioning time also includes finding the stencils for each node in the domain,

- *stencil weights computation*, where basis functions are defined and shapes for the Laplace operator and first derivatives are stored,
- *system assembly*, where computed weights are assembled in a sparse matrix and its right-hand side is computed and
- *system solution*, where the sparse system is solved.

3.3.1 Computational Complexity

The theoretical computational complexity is analyzed in this section. The total number of nodes will be denoted as $N_t = N + N_g$; however, as N_g nodes are distributed only along the boundary, it holds that $N_g = O(N^{\frac{d-1}{d}})$ and thus $N_t = O(N)$.

The node positioning algorithm has complexity $O(N_t \log N_t)$ [30]. Finding stencils of n closest nodes takes $O(nN_t \log N_t)$ time, using a fast spatial search structure, such as a k -d tree. The computation of stencil weights performs N_t solutions of linear systems of size $(n + s) \times (n + s)$, where $s = \frac{m + d}{d}$ is the number of monomials used for augmentation. Since n was chosen to be at least $2s$, it holds that $s = O(n)$. Using LU decomposition or any other standard solution procedure for dense linear systems takes $O((n + s)^3) = O(n^3)$ time. The total cost of weight computation is therefore $O(n^3 N_t)$.

With appropriate pre-allocation of storage for the sparse matrix, system assembly takes linear time in number of stencil nodes for each node, and right hand-side computation taken $O(1)$ per node. The total cost of system assembly is thus $O(nN_t)$.

The solution of the sparse system uses iterative BiCGSTAB with ILUT preconditioner, whose speed of convergence depends on the matrix properties.

The time complexity of the complete procedure is

$$O(nN_t \log N_t + n^3 N_t) + T,$$

where T is the complexity of the sparse solver.

3.3.2 Execution Time

In this section, we measure execution time spent on different parts of the solution procedure. All computations were performed on a single core of a computer with Intel (R) Xeon (R) CPU E5-2620 v3 @ 2.40GHz processor and 64 GB of DDR4 memory. Code was compiled using g++ (GCC) 8.1.0 for Linux with `-O3 -DNDEBUG` flags.

Total execution times are shown in Fig. 6 and correspond to accuracy results in Fig. 5. The computational time grows with N and with m , as expected from theoretical predictions in Sect. 3.3.1.

Absolute times of different computation stages and their proportions to the total time are shown in Fig. 7, on the left and the right side, respectively. The observed growth rates match the theoretical complexities predicted for node positioning, weight computation and system assembly.

Relative execution times provide additional insight into the execution of the solution procedure and into optimization and parallelization opportunities. The majority of the computational time is usually spent on either computing the stencil weights (for smaller N) or on system solution (for large N). Similar behavior was observed for other m and in other dimensions, with different percentage of total time spent on node positioning, weight computation and system solution [15].

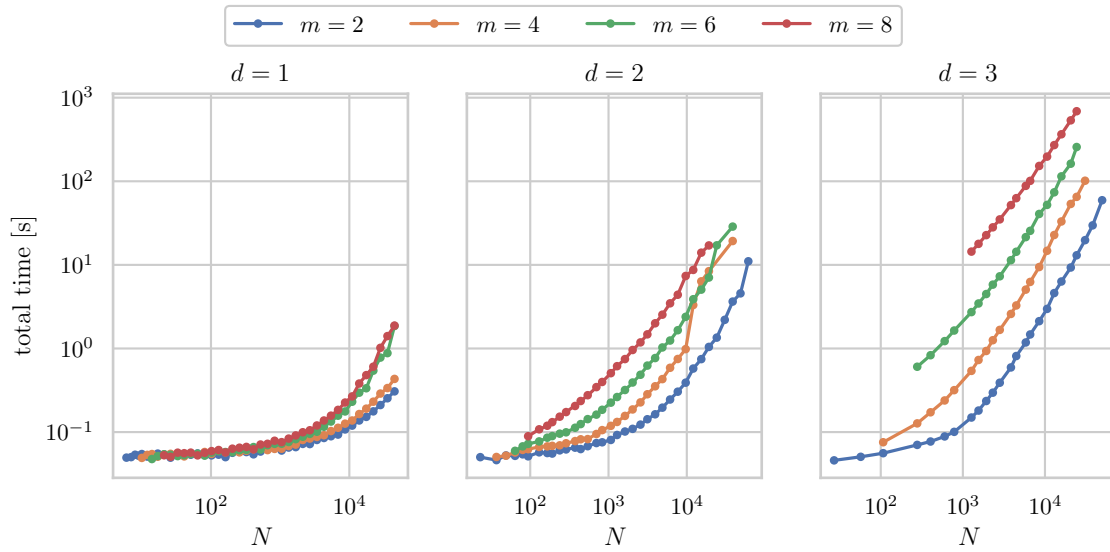


Fig. 6 Median of 10 total execution times of u_h computation for various setups

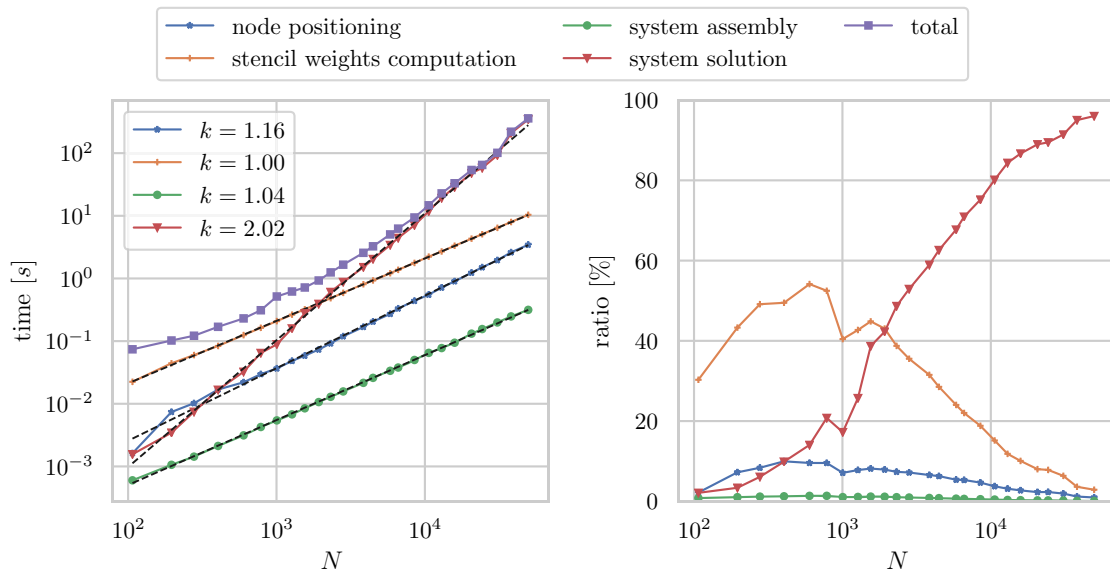


Fig. 7 Absolute and relative times of different parts of the solution procedure for $d = 3$ and $m = 4$

3.4 Accuracy Versus Execution Time

In the previous sections, we have shown that using higher orders, both accuracy and execution time increase. In this section, we analyze the accuracy versus execution time trade-off. Figure 8 shows e_∞ error plotted with respect to the total computational time needed to achieve it.

Significant differences can be observed between different orders of monomial augmentation. For prototyping or any other sort of quick scanning of how or if the computed solution u_h converges, using polynomials of a lower degree is undeniably very beneficial – the computation of u_h takes little time, but at a cost of limited accuracy. When higher accuracy is required, using polynomials of a higher degree can lead to a several orders faster computation time. In some cases, using higher orders might even be a necessity, e.g. for $d = 2$, where

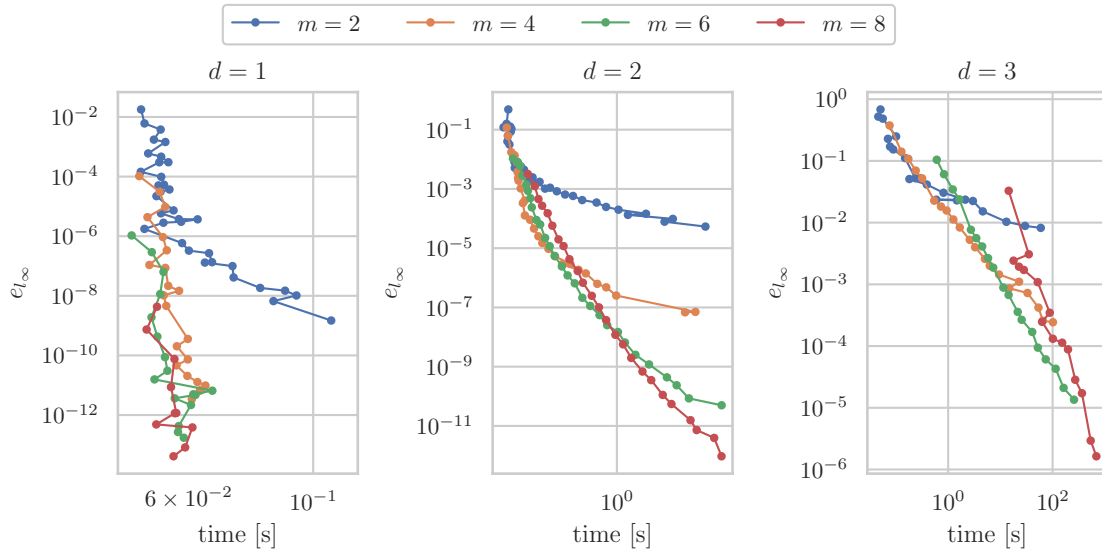


Fig. 8 Accuracy versus execution time trade-off for different orders of monomial augmentation

Table 2 Optimal setups for various desired target accuracy ranges in 1, 2 and 3 dimensions

$d = 1$		$d = 2$		$d = 3$	
Target accuracy e_∞	Optimal m	Target accuracy e_∞	Optimal m	Target accuracy e_∞	Optimal m
10^0 to 10^{-4}	2	10^0 to 10^{-2}	2	10^0 to 10^{-1}	2
10^{-4} to 10^{-6}	4	10^{-2} to 10^{-5}	4	10^{-1} to 10^{-3}	4
10^{-6} to 10^{-8}	6	10^{-5} to 10^{-8}	6	10^{-3} to 10^{-5}	6
10^{-8} to 10^{-13}	8	10^{-8} to 10^{-12}	8	10^{-5} to 10^{-7}	8

accuracy of $e_\infty \approx 10^{-10}$ is reached the fastest by $m = 8$, while solution for $m = 2$ would require N out of reasonable computing capabilities. The findings are summarized in Table 2.

Using the data in the table, we can extract a rough general recommendation. As a rule of thumb, for the desired accuracy $e_\infty = 10^{-k}$ and dimension d , the recommended order of augmentation is

$$m = \frac{5}{4}k + \frac{4}{5}d - 2, \quad (34)$$

rounded to the nearest positive even integer. Even though the data points in the table are close to being planar, the formula (34) does not necessarily generalize well. A more general rule is that the order of monomials should be increased with every two to three orders of increase in accuracy, and that higher order augmentation should be more aggressively used in higher dimensions.

4 Additional Example

In addition to already solved cases, we now demonstrate a solution of a 4-dimensional Poisson problem (23–25). The irregular domain Ω is now defined as $\Omega = B_0 \setminus (B_1 \cup B_2 \cup B_3)$, where

$$B_0 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \frac{\mathbf{1}}{2} \right\| < \frac{1}{2} \right\}, \quad (35)$$

$$B_1 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \left(\frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2} \right) \right\| \leq \frac{1}{4} \right\}, \quad (36)$$

$$B_2 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \mathbf{0} \right\| \leq \frac{13}{16} \right\} \text{ and} \quad (37)$$

$$B_3 = \left\{ \mathbf{x} \in \mathbb{R}^4, \left\| \mathbf{x} - \left(\frac{1}{2}, \frac{1}{2}, \frac{3}{4}, \frac{1}{2} \right) \right\| \leq \frac{1}{8} \right\} \quad (38)$$

are balls in $\mathbb{R}^4$.

Dirichlet and Neumann boundary conditions are defined similarly to before, i.e., Γ_d is the left half and Γ_n is the right half of $\partial\Omega$. Additionally, the boundary of the smallest ball ∂B_3 is added to the Dirichlet boundary:

$$\Gamma_d = \left\{ \mathbf{x} \in \partial\Omega, x_1 < \frac{1}{2} \right\} \cup \partial B_3, \quad (39)$$

$$\Gamma_n = \left\{ \mathbf{x} \in \partial\Omega, x_1 \geq \frac{1}{2} \right\} \setminus \partial B_3. \quad (40)$$

Scattered computational nodes were positioned using the same dimension-agnostic node positioning algorithm as before. A numerical solution u_h was obtained using RBF-FD with PHS $\phi(r) = r^3$ augmented with polynomials of degree $m = 4$, according to our rule of thumb (34) for the desired accuracy $e_\infty = 10^{-2}$.

Approximately $N = 85000$ nodes were positioned in Ω and closest $n = 950$ nodes were selected as stencils for each node from the domain. Ghost nodes were, as in the previous case, added to both Dirichlet and Neumann boundaries, and excluded from any post-processing. The final system was solved using a direct sparse solver.

Figure 9 shows the numerically obtained solutions. Four three-dimensional slices are shown, defined by setting one coordinate to $x_i = 1/2$. Modified Sheppard's interpolation algorithm [10] was used to interpolate the solution to an intermediate grid, used for plotting the slices.

The solution is well-behaved even in 4 dimensions; however, a relatively large support size is needed to obtain a desirable numerical stability. The errors equal to $e_1 = 6.83 \cdot 10^{-4}$, $e_2 = 2.11 \cdot 10^{-3}$ and $e_\infty = 1.72 \cdot 10^{-2}$. The total computational time spent was approximately 15 hours.

5 Conclusions

The message of this paper is twofold. First, we demonstrated that it is possible to design an appropriately abstract implementation, which encompasses most of the meshless mathematical elegance, allowing the user to construct a high order dimension-independent solution procedure. To fully demonstrate the dimensional independence, we also presented a solu-

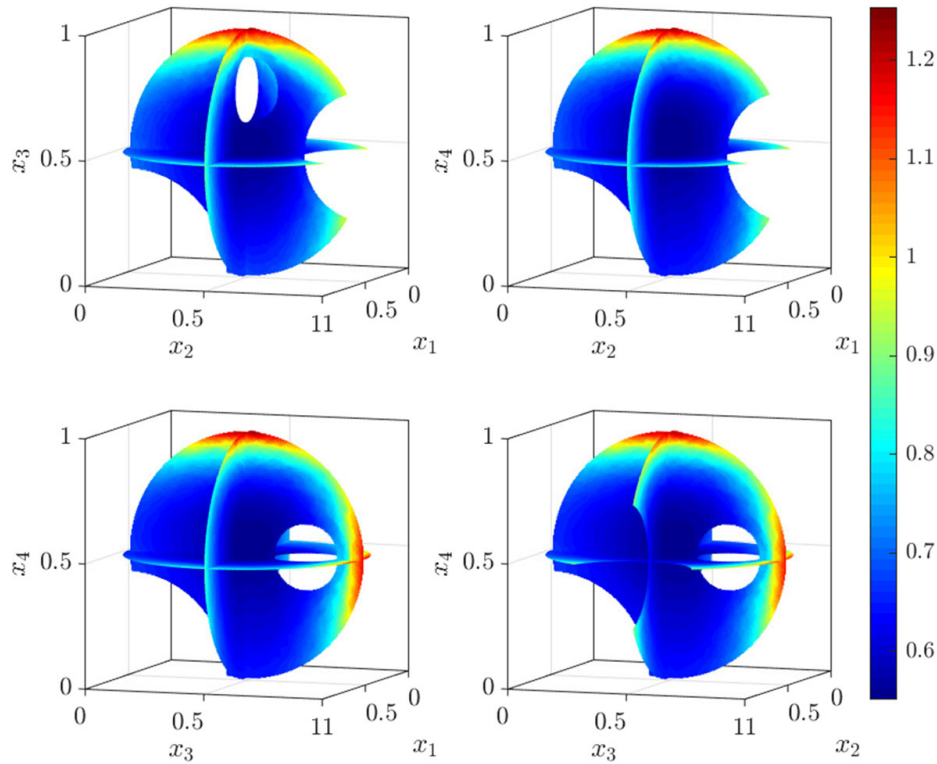


Fig. 9 3-dimensional cross sections of a solution to a 4-dimensional Poisson problem

tion of a 4-dimensional Poisson's problem on an irregular domain with both Neumann and Dirichlet boundary conditions.

Second, we used the devised implementation to analyze the increasing execution time that comes tied with high order augmentation, to determine the conditions of optimal computation efficiency for a desired target accuracy.

The analyses are performed on the solution of a Poisson problem with mixed boundary conditions in one, two and three dimensions. To avoid shape parameter dependency, we used PHS augmented with monomials as RBFs. Scattered nodes were positioned with a dedicated dimension-agnostic node generation algorithm. The theoretical findings on how the highest order of the augmenting polynomial directly controls the approximation rate of the RBF-FD independently of the domain dimension are verified. A detailed breakdown of the computational complexity and the execution time of different computational stages is also provided, to ensure that the implementation agrees with the theoretical predictions. Finally, the high order versus execution time trade-off is analyzed and the findings are summarized in Fig. 8 and Table 2. While the analyses were done only for this particular problem, the results can be generalized in the sense that for a high target accuracy, a high order method is a better choice, and vice versa.

Another interesting point are the increasing stencil sizes required for high order methods, as shown in Table 1. Especially in higher dimensions, this cost quickly becomes unmanageable. Therefore, our future work will be focused primarily on better understanding of the impact of the stencil size on the approximation quality.

Acknowledgements The authors would like to acknowledge the financial support of the Slovenian Research Agency (ARRS) research core Funding No. P2-0095 and the Young Researcher Program PR-08346.

References

1. Ahmad, I., Islam, SuI, Khaliq, A.Q.: Local RBF method for multi-dimensional partial differential equations. *Comput. Math. Appl.* **74**(2), 292–324 (2017). <https://doi.org/10.1016/j.camwa.2017.04.026>
2. Bayona, V.: An insight into RBF-FD approximations augmented with polynomials. *Comput. Math. Appl.* **77**(9), 2337–2353 (2019). <https://doi.org/10.1016/j.camwa.2018.12.029>
3. Bayona, V., Flyer, N., Fornberg, B., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs. *J. Comput. Phys.* **332**, 257–273 (2017). <https://doi.org/10.1016/j.jcp.2016.12.008>
4. Bridson, R.: Fast Poisson disk sampling in arbitrary dimensions. In: SIGGRAPH sketches, p. 22 (2007). <https://doi.org/10.1145/1278780.1278807>
5. Choi, Y., Kim, S.: Node generation scheme for meshfree method by Voronoi diagram and weighted bubble packing. In: Fifth us National Congress on Computational Mechanics, Boulder, CO (1999)
6. Flyer, N., Fornberg, B., Bayona, V., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. *J. Comput. Phys.* **321**, 21–38 (2016). <https://doi.org/10.1016/j.jcp.2016.05.026>
7. Fornberg, B., Flyer, N.: Fast generation of 2-D node distributions for mesh-free PDE discretizations. *Comput. Math. Appl.* **69**(7), 531–544 (2015)
8. Fornberg, B., Flyer, N.: A primer on radial basis functions with applications to the geosciences, CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 87. SIAM (2015). <https://doi.org/10.1137/1.9781611974041>
9. Fornberg, B., Flyer, N.: Solving PDEs with radial basis functions. *Acta Numer.* **24**, 215–258 (2015). <https://doi.org/10.1017/S0962492914000130>
10. Franke, R., Nielson, G.: Smooth interpolation of large sets of scattered data. *Int. J. Numer. Methods Eng.* **15**(11), 1691–1704 (1980). <https://doi.org/10.1002/nme.1620151110>
11. Guennebaud, G., Jacob, B., et al.: Eigen v3. <http://eigen.tuxfamily.org> (2010)
12. Hardin, D., Saff, E.: Discretizing manifolds via minimum energy points. *Not. AMS* **51**(10), 1186–1194 (2004)
13. Jančič, M., Slak, J., Kosec, G.: Standalone implementation of solution to the Poisson’s equation. <http://e6.ijs.si/medusa/static/DimensionIndependentPoisson.zip> (2019)
14. Kosec, G.: A local numerical solution of a fluid-flow problem on an irregular domain. *Adv. Eng. Softw.* **120**, 36–44 (2018). <https://doi.org/10.1016/j.advengsoft.2016.05.010>
15. Kosec, G., Slak, J.: Parallel RBF-FD solution of the Boussinesq’s problem. In: P. Iványi, B.H.V. Topping (eds.) *Proceedings of the Sixth International Conference on Parallel, Distributed, GPU and Cloud Computing for Engineering*, June 5–6, 2019, Pécs, Hungary, Civil-comp proceedings. Stirlingshire: Civil-Comp Press (2019)
16. Liu, G.R.: *Mesh Free Methods: Moving Beyond the Finite Element Method*, 1st edn. CRC Press, Boca Raton (2002). <https://doi.org/10.1201/9781420040586>
17. Liu, Y., Nie, Y., Zhang, W., Wang, L.: Node placement method by bubble simulation and its application. *Comput. Model. Eng. Sci. (CMES)* **55**(1), 89 (2010)
18. Löhner, R., Oñate, E.: A general advancing front technique for filling space with arbitrary objects. *Int. J. Numer. Methods Eng.* **61**(12), 1977–1991 (2004)
19. Mairhuber, J.C.: On Haar’s theorem concerning Chebychev approximation problems having unique solutions. *Proc. Am. Math. Soc.* **7**(4), 609–615 (1956). <https://doi.org/10.2307/2033359>
20. Maksić, M., Djurica, V., Souvent, A., Slak, J., Depolli, M., Kosec, G.: Cooling of overhead power lines due to the natural convection. *Int. J. Electr. Power Energy Syst.* **113**, 333–343 (2019). <https://doi.org/10.1016/j.ijepes.2019.05.005>
21. Mavrič, B., Šarler, B.: Equivalent-PDE based stabilization of strong-form meshless methods applied to advection-dominated problems. *Eng. Anal. Bound. Elem.* **113**, 315–327 (2020). <https://doi.org/10.1016/j.enganabound.2020.01.014>
22. Milovanović, S., von Sydow, L.: Radial basis function generated finite differences for option pricing problems. *Comput. Math. Appl.* **75**(4), 1462–1481 (2018). <https://doi.org/10.1016/j.camwa.2017.11.015>
23. Oanh, D.T., Davydov, O., Phu, H.X.: Adaptive RBF-FD method for elliptic problems with point singularities in 2D. *Appl. Math. Comput.* **313**, 474–497 (2017). <https://doi.org/10.1016/j.amc.2017.06.006>
24. Onate, E., Idelsohn, S., Zienkiewicz, O.C., Taylor, R.L.: A finite point method in computational mechanics: applications to convective transport and fluid flow. *Int. J. Numer. Methods Eng.* **39**(22), 3839–3866 (1996)
25. Shankar, V., Fogelson, A.L.: Hyperviscosity-based stabilization for radial basis function-finite difference (RBF-FD) discretizations of advection–diffusion equations. *J. Comput. Phys.* **372**, 616–639 (2018). <https://doi.org/10.1016/j.jcp.2018.06.036>

26. Shankar, V., Kirby, R.M., Fogelson, A.L.: Robust node generation for meshfree discretizations on irregular domains and surfaces. *SIAM J. Sci. Comput.* **40**(4), 2584–2608 (2018). <https://doi.org/10.1137/17m114090x>
27. Slak, J., Kosec, G.: Standalone implementation of the proposed node placing algorithm (2018). <http://e6.ijs.si/medusa/static/PNP.zip>
28. Slak, J., Kosec, G.: Adaptive radial basis function-generated finite differences method for contact problems. *Int. J. Numer. Methods Eng.* **119**(7), 661–686 (2019). <https://doi.org/10.1002/nme.6067>
29. Slak, J., Kosec, G.: Medusa: A C++ library for solving pdes using strong form mesh-free methods (2019). <http://e6.ijs.si/medusa/>
30. Slak, J., Kosec, G.: On generation of node distributions for meshless PDE discretizations. *SIAM J. Sci. Comput.* **41**(5), A3202–A3229 (2019). <https://doi.org/10.1137/18M1231456>
31. Slak, J., Kosec, G.: Refined meshless local strong form solution of Cauchy–Navier equation on an irregular domain. *Eng. Anal. Bound. Elem.* **100**, 3–13 (2019). <https://doi.org/10.1016/j.enganabound.2018.01.001>
32. Tolstykh, A.I., Shirobokov, D.A.: On using radial basis functions in a “finite difference mode” with applications to elasticity problems. *Comput. Mech.* **33**(1), 68–79 (2003). <https://doi.org/10.1007/s00466-003-0501-9>
33. Wendland, H.: Scattered data approximation, Cambridge Monographs on Applied and Computational Mathematics, vol. 17. Cambridge university press (2004). <https://doi.org/10.1017/cbo9780511617539>
34. Wright, G.B., Fornberg, B.: Stable computations with flat radial basis functions using vector-valued rational approximations. *J. Comput. Phys.* **331**, 137–156 (2017). <https://doi.org/10.1016/j.jcp.2016.11.030>
35. Yianilos, P.N.: Data structures and algorithms for nearest neighbor search in general metric spaces. In: Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '93, pp. 311–321. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (1993). <http://dl.acm.org/citation.cfm?id=313559.313789>

2.2 Oscillatory Behaviour of the RBF-FD Approximation Accuracy Under Increasing Stencil Size

This section extends RBF-FD analyses in terms of the stencil size impact on the accuracy of the numerical solution. While some researchers proposed dedicated stencil selection algorithms [28], [64], a common approach is to select n nearest neighbors based on the Euclidean distance. For the latter approach, a stencil size of $n = 2\binom{m+d}{d}$ for stable RBF-FD of order m in a d -dimensional domain is recommended [30].

Our analyses show that this recommendation may not be optimal particularly in terms of solution's accuracy.

Contributions.

This section represents contribution **C2**. We show that a slight but non-trivial adjustment to the stencil size can result in a significant improvement of solution's accuracy, with comparable or even shorter execution times, sometimes yielding up to an order of magnitude more accurate numerical solutions. We also report on observing interesting oscillatory behaviour of the solution error. The oscillatory behaviour is yet to be fully understood, nevertheless, we believe this observation could be beneficial towards the definition of an *optimal* stencil size.

Addressed hypotheses.

Since the three approximation methods subject to research in this dissertation come with different stencil size recommendations, which clearly affects the computational complexity, the publication below partially confirms hypothesis **H7**.

Publications included in this section:

- A. Kolar-Požun, M. Jančíč, M. Rot, *et al.*, “Oscillatory behaviour of the rbf-fd approximation accuracy under increasing stencil size,” in *Computational Science – ICCS 2023*, J. Míkyška, C. de Mulatier, M. Paszynski, *et al.*, Eds., Cham: Springer Nature Switzerland, 2023, pp. 515–522, ISBN: 978-3-031-36027-5

Regarding my contribution: I made a literature overview of the topic, participated in planning, prepared the preliminary analyses and jointly prepared the manuscript with co-authors.



Oscillatory Behaviour of the RBF-FD Approximation Accuracy Under Increasing Stencil Size

Andrej Kolar-Požun^{1,2}(✉) , Mitja Jančič^{1,3} , Miha Rot^{1,3} ,
and Gregor Kosec¹

¹ Parallel and Distributed Systems Laboratory, Jožef Stefan Institute, Ljubljana, Slovenia

{andrej.pozun,mitja.jancic,miha.rot,gregor.kosec}@ijs.si

² Faculty of Mathematics and Physics, University of Ljubljana, Ljubljana, Slovenia

³ Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Abstract. When solving partial differential equations on scattered nodes using the Radial Basis Function generated Finite Difference (RBF-FD) method, one of the parameters that must be chosen is the stencil size. Focusing on Polyharmonic Spline RBFs with monomial augmentation, we observe that it affects the approximation accuracy in a particularly interesting way - the solution error oscillates under increasing stencil size. We find that we can connect this behaviour with the spatial dependence of the signed approximation error. Based on this observation we are then able to introduce a numerical quantity that indicates whether a given stencil size is locally optimal.

Keywords: Meshless · Stencil · RBF-FD · PHS

1 Introduction

Radial Basis Function generated Finite Differences (RBF-FD) is a method for solving Partial Differential Equations (PDEs) on scattered nodes that has recently been increasing in popularity. It uses Radial Basis Functions (RBFs) to locally approximate a linear differential operator in a chosen neighbourhood, generalising the well known finite difference methods. This neighbourhood used for the approximation is referred to as the stencil of a given point and is commonly chosen to simply consist of its n closest neighbours.

Among the different possible choices of a RBF used, the Polyharmonic Splines (PHS) with appropriate polynomial augmentation stand out due to the fact that they possess no shape parameter, eliminating all the hassle that comes with having to find its optimal value. PHS RBF-FD has been studied extensively and proved to work well in several different contexts [1, 2, 6–8]. Unlike in the case of RBFs with a shape parameter, where the approximation order is determined by the stencil size [3], in PHS RBF-FD it is determined by the degree of the monomials included in the augmentation [1]. Despite that, the choice of an appropriate stencil size can have a substantial impact on the accuracy. More

516 A. Kolar-Požun et al.

precisely, the accuracy of the method displays an oscillatory behaviour under increasing stencil size.

In the remainder of the paper, we present this observation and our findings. Ideally we would like to be able to predict the stencil sizes that correspond to the accuracy minima or at least provide some indicator on whether a given stencil size is near the minimum.

The following section describes our problem setup along with the numerical solution procedure and in Sect. 3 our results are discussed.

2 Problem Setup

Our analyses are performed on the case of the Poisson equation

$$\nabla^2 u(\mathbf{x}) = f(\mathbf{x}), \quad (1)$$

where the domain is a disc $\Omega = \{\mathbf{x} \in \mathbb{R}^2 : \|x - (0.5, 0.5)\| \leq 0.5\}$. We choose the function $f(\mathbf{x})$ such, that the problem given by Eq. (1) has a known analytic solution. Concretely, we choose

$$u(x, y) = \sin(\pi x) \sin(\pi y), \quad (2)$$

$$f(x, y) = -2\pi^2 \sin(\pi x) \sin(\pi y) \quad (3)$$

with the Dirichlet boundary conditions given by a restriction of $u(\mathbf{x})$ to the boundary $\partial\Omega$.

We discretise the domain with the discretisation distance $h = 0.01$, first discretising the boundary and then the interior using the algorithm proposed in [4]. The Laplacian is then discretised using the RBF-FD algorithm as described in [5], where we choose the radial cubics as our PHS ($\phi(r) = r^3$) augmented with monomials up to degree $m = 3$, inclusive. This requires us to associate to each discretisation point $\mathbf{x}_i$ its stencil, which we take to consist of its n nearest neighbours. We can now convert the PDE (1) into a sparse linear system, which we then solve to obtain an approximate solution $\hat{u}(\mathbf{x})$. The source code is readily available in our git repository¹.

The chosen analytical solution $u(\mathbf{x})$ is displayed in Fig. 1, which additionally serves as a visual representation of how the domain is discretised.

Having both the analytical and approximate solutions, we will be interested in the approximation error. It will turn out to be useful to consider the signed pointwise errors of both the solution and the Laplacian approximation:

$$e_{\text{poiss}}^{\pm}(\mathbf{x}_i) = \hat{u}_i - u_i, \quad (4)$$

$$e_{\text{lap}}^{\pm}(\mathbf{x}_i) = \tilde{\nabla}^2 u_i - f_i, \quad (5)$$

where $\tilde{\nabla}^2$ is the discrete approximation of the Laplacian and we have introduced the notation $u_i = u(\mathbf{x}_i)$. The “poiss” and “lap” subscripts may be omitted in the text when referring to both errors at once.

¹ https://gitlab.com/e62Lab/public/2023_cp_iccs_stencil_size_effect.

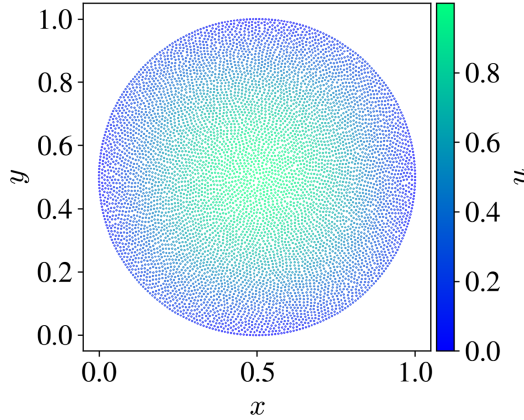


Fig. 1. The analytical solution to the considered Poisson problem.

As a quantitative measure of the approximation quality, we will also look at the average/max absolute value error:

$$e_{\text{poiss}}^{\max} = \max_{\mathbf{x}_i \in \hat{\Omega}} |e_{\text{poiss}}^{\pm}(\mathbf{x}_i)|, \quad (6)$$

$$e_{\text{poiss}}^{\text{avg}} = \frac{1}{N_{\text{int}}} \sum_{\mathbf{x}_i \in \hat{\Omega}} |e_{\text{poiss}}^{\pm}(\mathbf{x}_i)| \quad (7)$$

and analogously for $e_{\text{lap}}^{\max}$ and $e_{\text{lap}}^{\text{avg}}$. N_{int} is the number of discretisation points inside the domain interior $\hat{\Omega}$.

In the next section we will calculate the approximation error for various stencil sizes n and further investigate its (non-trivial) behaviour.

It is worth noting that the setup considered is almost as simple as it can be. The fact that we have decided not to consider a more complicated problem is intentional - there is no need to complicate the analysis by considering a more complex problem if the investigated phenomena already appears in a simpler one. This reinforces the idea that such behaviour arises from the properties of the methods used and not from the complexity of the problem itself.

3 Results

In Fig. 2 we see that $e_{\text{poiss}}^{\max}(n)$ oscillates with several local minima (at stencil sizes $n = 28, 46$) and maxima (stencil sizes $n = 17, 46$). The dependence $e_{\text{poiss}}^{\max}(n)$ seems to mostly resemble a smooth function. This is even more evident in $e_{\text{poiss}}^{\text{avg}}(n)$. The errors of the Laplacian are also plotted and we can observe that $e_{\text{lap}}^{\text{avg}}(n)$ has local minima and maxima at same stencil sizes. Such regularity implies that the existence of the minima in the error is not merely a coincidence, but a consequence of a certain mechanism that could be explained. Further understanding of this mechanism would be beneficial, as it could potentially allow us to predict the location of these local minima a priori. Considering that

the error difference between the neighbouring local maxima and minima can be over an order of magnitude apart this could greatly increase the accuracy of the method without having to increase the order of the augmentation or the discretisation density. Note that the behaviour of $e_{\text{lap}}^{\text{max}}$ stands out as it is much more irregular. This implies that in order to explain the observed oscillations, we have to consider the collective behaviour of multiple points. This will be confirmed later on, when we consider the error's spatial dependence.

An immediate idea is that the choice of a sparse solver employed at the end of the solution procedure is responsible for the observed behaviour. We have eliminated this possibility by repeating the analysis with both the SparseLU and BiCGSTAB solvers, where no difference has been observed. The next idea we explore is the possibility of the discretisation being too coarse. Figure 3 shows that under discretisation refinement $e_{\text{poiss}}^{\text{max}}(n)$ maintains the same shape and is just shifted vertically towards a lower error. The latter shift is expected, as we are dealing with a convergent method, for which the solution error behaves as $e \propto h^p$ as $h \rightarrow 0$, where p is the order of the method. We also show $e_{\text{poiss}}^{\text{max}}(h)$ in a log-log scale for some different stencil sizes. It can be seen that the slopes and therefore the orders p generally do not change with the stencil size and that the observed oscillations mainly affect the proportionality constant in $e \propto h^p$. The stencil dependence of the error proportionality constant has already been observed in similar methods [3, 9].

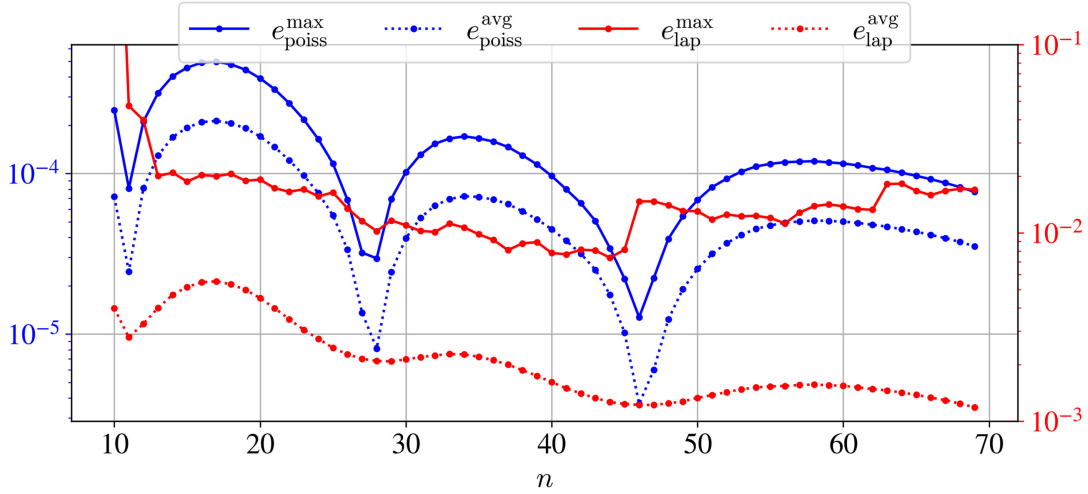


Fig. 2. Dependence of the approximation errors on the stencil size n .

Next we check if boundary stencils are responsible for the observed behaviour as it is known that they can be problematic due to their one-sidedness [2]. In Fig. 4 we have split our domain into two regions - the nodes near the boundary $\{\mathbf{x}_i \in \Omega : \|\mathbf{x}_i - (0.5, 0.5)\| > 0.4\}$ are coloured red, while the nodes far from the boundary $\{\mathbf{x}_i \in \Omega : \|\mathbf{x}_i - (0.5, 0.5)\| \leq 0.4\}$ are black. We can see that the dependence of $e_{\text{poiss}}^{\text{max}}(n)$ marginally changes if we keep the stencil size near the

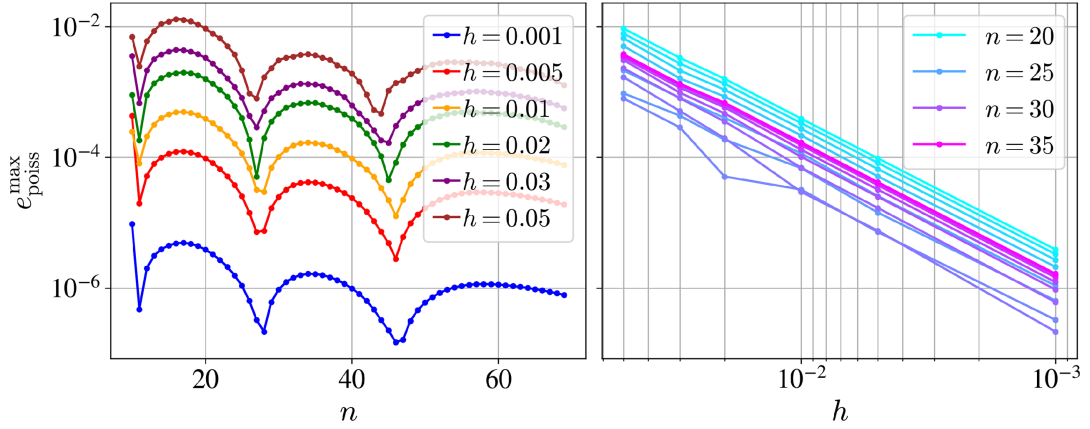


Fig. 3. Behaviour of the approximation errors under a refinement of the discretisation.

boundary fixed at $n = 28$ (corresponding to one of the previously mentioned minima), while only changing the stencil sizes of the remaining nodes. This shows that the observed phenomena is not a consequence of the particularly problematic boundary stencils.

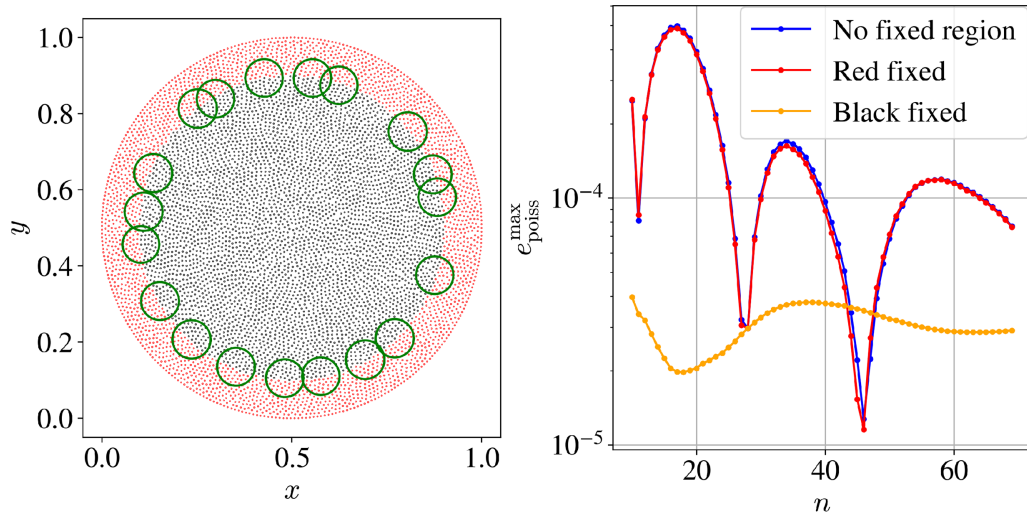


Fig. 4. The separation of the domain into two regions is seen on the left, where the green circles show the radii of the biggest stencils considered ($n = 69$). The right graph shows the error dependence when either of the regions is at a fixed stencil size $n = 28$. The previous result with no fixed stencil size regions is also shown.

Figure 5 provides some more insight into the mechanism behind the oscillating error. Here we have plotted the spatial dependence of the signed error $e_{\text{poiss}}^{\pm}$ for those stencils that correspond to the marked local extrema. We can observe that in the maxima, the error has the same sign throughout the whole domain.

520 A. Kolar-Požun et al.

On the other hand, near the values of n that correspond to the local minima there are parts of the domain that have differing error signs. Concretely, the sign of $e_{\text{poiss}}^{\pm}$ is negative for stencil sizes between 17 and 27 inclusive. In the minima at $n = 28$ both error signs are present, while for bigger stencil sizes (between 29 and 45 inclusive) the error again has constant sign only this time positive. The story turns out to repeat on the next minimum.

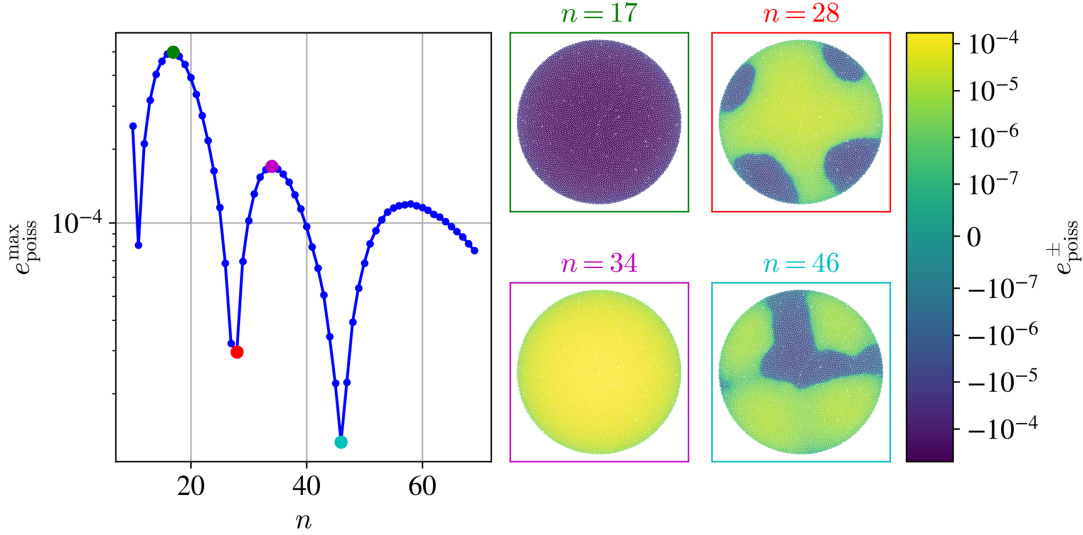


Fig. 5. Spatial dependence of $e_{\text{poiss}}^{\pm}$ in some local extrema. The colour scale is the same for all drawn plots.

This connection between the sign of $e_{\text{poiss}}^{\pm}$ and the minima in $e_{\text{poiss}}^{\text{max}}(n)$ motivates us to define a new quantity:

$$\delta N_{\text{poiss}}^{\pm} = \frac{1}{N_{\text{int}}} \left(|\{\mathbf{x}_i \in \hat{\Omega} : e_{\text{poiss}}^{\pm}(\mathbf{x}_i) > 0\}| - |\{\mathbf{x}_i \in \hat{\Omega} : e_{\text{poiss}}^{\pm}(\mathbf{x}_i) < 0\}| \right) \quad (8)$$

and analogously for $\delta N_{\text{lap}}^{\pm}$. Simply put, the quantity $\delta N_{\text{poiss}}^{\pm}$ is proportional to the difference between the number of nodes with positively and negatively signed error. Assigning values of ± 1 to positive/negative errors respectively, this quantity can be roughly interpreted as the average sign of the error. It should hold that $|\delta N_{\text{poiss}}^{\pm}|$ is approximately equal to 1 near the maxima and lowers in magnitude as we approach the minima. Figure 6 confirms this intuition - $\delta N_{\text{poiss}}^{\pm}(n)$ changes its values between ± 1 very abruptly only near the n that correspond to the minima of $e_{\text{poiss}}^{\text{max}}(n)$. A similar conclusion can be made for $\delta N_{\text{lap}}^{\pm}$, which acts as a sort of “smoothed out” version of $\delta N_{\text{poiss}}^{\pm}(n)$.

At a first glance, $N_{\text{lap}}^{\pm}$ seems like a good candidate for an error indicator - it has a well-behaved dependence on n , approaches ± 1 as we get closer to the error maxima and has a root near the error minima. The major downside that completely eliminates its applicability in the current state is the fact that we need access to the analytical solution to be able to compute it.

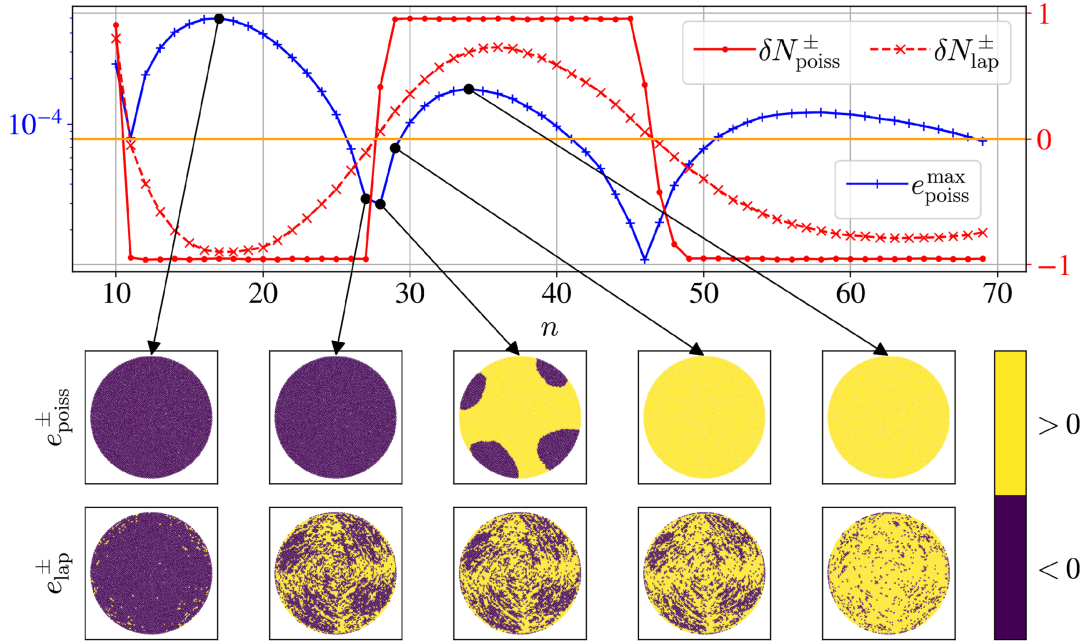


Fig. 6. The quantities $\delta N^{\pm}(n)$ along with the spatial profiles of the signs of $e^{\pm}$ for some chosen stencil sizes. For convenience, $\delta N^{\pm} = 0$ is marked with an orange line.

4 Conclusions

Our study started with a simple observation - when solving a Poisson problem with PHS RBF-FD, the approximation accuracy depends on the stencil size n in a non-trivial manner. In particular, there exist certain stencil sizes where the method is especially accurate. A priori knowledge of these stencil sizes could decrease the solution error without any additional effort and is therefore strongly desirable. We have made a small step towards understanding this phenomena, eliminating various common numerical issues as the cause. Looking at the spatial dependence of the signed solution error, we have noticed that in the stencil sizes corresponding to the local error minima, the signed solution error is not strictly positive or negative. This is unlike the generic stencil sizes, where the error has the same sign throughout the domain. Motivated by this observation, we have introduced a quantity that is roughly the average sign of the pointwise Laplace operator errors and appears to have a root only near the stencils corresponding to the local error minima.

The research presented is a step towards defining a more practically useful indicator, which would reveal the most accurate stencil sizes even without having access to the analytical solution and is a part of our ongoing research. Additional future work includes more rigorous theoretical explanations for the observations presented. Further experimental investigations should also be made, particularly to what extent our observations carry over to different problem setups - other differential equations and domain shapes.

522 A. Kolar-Požun et al.

Acknowledgements. The authors would like to acknowledge the financial support of Slovenian Research Agency (ARRS) in the framework of the research core funding No. P2-0095 and the Young Researcher programs PR-10468 and PR-12347.

References

1. Bayona, V.: An insight into RBF-FD approximations augmented with polynomials. *Comput. Math. Appl.* **77**(9), 2337–2353 (2019)
2. Bayona, V., Flyer, N., Fornberg, B., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: II. numerical solution of elliptic PDEs. *J. Computat. Phys.* **332**, 257–273 (2017)
3. Bayona, V., Moscoso, M., Carretero, M., Kindelan, M.: RBF-FD formulas and convergence properties. *J. Comput. Phys.* **229**(22), 8281–8295 (2010)
4. Depolli, M., Slak, J., Kosec, G.: Parallel domain discretization algorithm for RBF-FD and other meshless numerical methods for solving PDEs. *Comput. Struct.* **264**, 106773 (2022)
5. Le Borne, S., Leinen, W.: Guidelines for RBF-FD discretization: numerical experiments on the interplay of a multitude of parameter choices. *J. Sci. Comput.* **95**(1), 8 (2023)
6. Oruç, Ö.: A radial basis function finite difference (RBF-FD) method for numerical simulation of interaction of high and low frequency waves: Zakharov-Rubenchik equations. *Appl. Math. Comput.* **394**, 125787 (2021)
7. Shankar, V., Fogelson, A.L.: Hyperviscosity-based stabilization for radial basis function-finite difference (RBF-FD) discretizations of advection-diffusion equations. *J. Comput. Phys.* **372**, 616–639 (2018)
8. Strniša, F., Jančič, M., Kosec, G.: A meshless solution of a small-strain plasticity problem. In: 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), pp. 257–262. IEEE (2022)
9. Tominec, I., Larsson, E., Heryudono, A.: A least squares radial basis function finite difference method with improved stability properties. *SIAM J. Sci. Comput.* **43**(2), A1441–A1471 (2021)

2.3 Stability Analysis of RBF-FD and WLS-Based Local Strong-Form Meshless Methods on Scattered Nodes

To further improve our understanding of RBF-FD and WLS approximations, we study their stability on scattered nodes. Such study (i) provides insights into the computational effectiveness of the two widely used approximation methods with the ability to operate on scattered nodes, encompassing aspects ranging from the accuracy of the numerical solution to the stability of the solution procedure and, thus, (ii) allows us to conclude which of the two methods is more appropriately used in the hp -adaptive solution procedure.

Contributions.

This section represents contribution **C3**. We demonstrate that the high-order RBF-FD approximations are generally more stable and more accurate. This observation makes the RBF-FD approximation method a better candidate for the development of hp -adaptive solution procedure where stability of the high-order approximation is crucial.

Addressed hypotheses.

On a two-dimensional Poisson problem, we demonstrate the superior stability of RBF-FD approximation method, thus, confirming hypothesis **H2**. Additionally, as both methods demonstrate stable high-order approximations, we also confirm hypothesis **H3**.

Publications included in this section:

- M. Jančič and G. Kosec, “Stability analysis of RBF-FD and WLS based local strong form meshless methods on scattered nodes,” in *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 275–280. DOI: 10.23919/MIPRO55190.2022.9803334

Regarding my contribution: I made a literature overview of the topic, planned and performed the experiments and jointly prepared the manuscript with co-authors.

MIPRO 2022, May 23 - 27, 2022, Opatija, Croatia

Stability analysis of RBF-FD and WLS based local strong form meshless methods on scattered nodes

Mitja Jančič^{1,2}, Gregor Kosec¹

¹ “Jožef Stefan” Institute, Parallel and Distributed Systems Laboratory, Ljubljana, Slovenia

² “Jožef Stefan” International Postgraduate School, Ljubljana, Slovenia

mitja.jancic@ijs.si, gregor.kosec@ijs.si

Abstract – The popularity of local meshless methods in the field of numerical simulations has increased greatly in recent years. This is mainly due to the fact that they can operate on scattered nodes and that they allow a direct control over the approximation order and basis functions. In this paper we analyse two popular variants of local strong form meshless methods, namely the radial basis function-generated finite differences (RBF-FD) using polyharmonic splines (PHS) augmented with monomials, and the weighted least squares (WLS) approach using only monomials. Our analysis focuses on the accuracy and stability of the numerical solution computed on scattered nodes in a two- and three-dimensional domain. We show that while the WLS variant is a better choice when lower order approximations are sufficient, the RBF-FD variant exhibits a more stable behavior and a higher accuracy of the numerical solution for higher order approximations, but at the cost of higher computational complexity.

Keywords – meshless; WLS; RBF-FD; stability; scattered nodes

I. INTRODUCTION

Computational science has become an important aspect of technological advancement in the fields of science and engineering. Thanks to the unprecedented computing power at our disposal, many real-life problems are being numerically treated to deepen our understanding of a phenomenon under consideration.

In the field of numerical simulations, meshless methods are becoming increasingly popular with recent uses in the fields of fluid mechanics [1], linear elasticity [2], contact problems [3], advection-dominated problems [4] and even in financial sector [5]. Historically, mesh-free methods were introduced in the 1970s with the smoothed particle hydrodynamics (SPH) [6, 7] and then followed by several generalizations of the Finite Difference Method (FDM), e.g. the Finite Point Method [8], the Generalized Finite Difference Method [9] and the Radial Basis Function-Generated Finite Differences (RBF-FD) [10]. Nowadays, a lot of research is also devoted to reducing the computational time by employing the advantages of modern computer architecture [11, 12].

The ability of meshless methods to operate on scattered nodes makes them very attractive in many real-life

cases, where the domain shapes are often non-trivial. This is mainly because node positioning is easier than mesh generation (required by the mesh-based methods). Several algorithms for node positioning have been proposed to the meshless community. Some even support variable node density distributions [13] and employing parallelization [14] to reduce computational time. Another attractive feature of meshless methods is that the linear differential operator approximation allows a direct control over the order of the approximation method, as demonstrated in [15, 16], which can effectively be used to increase stability, as will be shown in this work.

With many proposed meshless variants it is often not clear which is the most optimal in terms of numerical stability. Therefore, the aim of this paper is to compare the two commonly used variants, namely the WLS with monomials, also known as diffuse approximation method, and the RBF-FD variant with polyharmonic splines (PHS) augmented with monomials. The stability of the two methods is evaluated by solving a Poisson problem in two- and three-dimensional domain for lower and higher order approximations.

The rest of the paper is organized as follows: In Section II both WLS and RBF-FD approximation methods are presented, in Section III our case study is presented, in Section IV the results are shown and commented. Finally, in Section V conclusions and our findings are given.

II. MESHLESS METHODS

In meshless methods, a linear differential operator $\mathcal{L}$ at each node $\mathbf{x}_c$ from the domain space Ω is approximated over a set of nearby nodes

$$\widehat{\mathcal{L}}u(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i) = \mathbf{w}_c \mathbf{u} \quad (1)$$

for any function u , n nearby nodes also known as *stencil nodes* and weights w_i , that are obtained by enforcing the equality of the equation (1) for a given set of s basis functions $\{p_j\}_{j=1}^s$. The most common choice is to declare the nearest n nodes as stencil, but some authors reported special stencil selection algorithms that increase the overall stability of the approximation [17, 18].

The approximation (1) is general in the sense that it holds for any linear differential operator $\mathcal{L}$ for any support

size n and any type or number s of chosen basis functions p . As long as the number of basis functions is equal to the number of support nodes ($s = n$), the formulation of (1) yields a quadratic system of equations

$$\underbrace{\begin{bmatrix} p_1(\mathbf{x}_1) & \cdots & p_1(\mathbf{x}_n) \\ \vdots & & \vdots \\ p_s(\mathbf{x}_1) & \cdots & p_s(\mathbf{x}_n) \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}}_{\mathbf{w}} = \underbrace{\begin{bmatrix} (\mathcal{L}p_1)(\mathbf{x}_c) \\ \vdots \\ (\mathcal{L}p_s)(\mathbf{x}_c) \end{bmatrix}}_{\mathbf{l}} \quad (2)$$

as is the case with the so-called local collocation methods [19]. However, larger support sizes are often used, resulting in an overdetermined system of equations. In such cases, the linear system is usually treated as a minimization of the weighted least squares (WLS) norm [20]

$$\begin{aligned} \|e\|_{2,w} &= \sqrt{\sum_{i=1}^n w^2 (\hat{u}(\mathbf{x}_i) - u_i)^2} = \\ &= \sqrt{\sum_{i=1}^2 (we_i)^2}. \end{aligned} \quad (3)$$

Common basis functions include: Multiquadrics, Gaussians, Radial Basis Functions (RBFs) and Monomials. In this paper we focus on two different types of basis functions, i.e., monomials and polyharmonic splines (PHS) augmented with monomials, resulting in two variants of meshless methods also known as the WLS approach and the RBF-FD variant respectively. While WLS approximation using a set $\{p_j\}_{j=1}^s$ monomials up to and including degree m as basis functions is fully defined above, the RBF-FD approximation is defined in the following section.

A. Radial Basis Function-Generated Finite Differences

Let us take RBFs φ , such that $\varphi : [0, \infty) \rightarrow \mathbb{R}$ is centered at the stencil nodes of a central node $\mathbf{x}_c$. The matrix Φ from the linear system (2) is then obtained by evaluating basis functions

$$\Phi_{ij} = \varphi(\|\mathbf{x}_i - \mathbf{x}_j\|) \quad (4)$$

and the vector $\mathbf{l}$ is assembled by applying the considered operator $\mathcal{L}$ to the basis functions evaluated at $\mathbf{x}_c$, i.e.,

$$l_\varphi^i = (\mathcal{L}\varphi(\|\mathbf{x} - \mathbf{x}_i\|))|_{\mathbf{x}=\mathbf{x}_c}. \quad (5)$$

We can choose from different types of RBFs. Until recently, Hardy's multiquadrics or Gaussians were commonly used, but both depend on a shape parameter which effectively governs the accuracy and stability of the approximation [21]. To avoid the shape parameter, we use PHS, defined as

$$\varphi(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \quad (6)$$

where r denotes the Euclidean distance between two nodes.

However, the use of pure RBFs as basis functions may lead to stagnation errors [22]. Therefore, in addition to the RBFs, augmentation with monomials up to and including degree m is added. This essentially means that we take a set of polynomials $\{p_j\}_{j=1}^s$ with up to and including degree m with $s = \binom{m+d}{d}$ and in addition to the RBF part of the approximation, the following exactness constraint for monomials is enforced

$$\sum_{i=1}^s w_i p_j(\mathbf{x}_i) = (\mathcal{L}p_j)(\mathbf{x}_c). \quad (7)$$

The additional constraints make the approximation overdetermined, which is treated as a constrained optimization problem [22]. For practical computation, the optimal solution can be expressed as a solution of a linear system

$$\begin{bmatrix} \Phi & \mathbf{P} \\ \mathbf{P}^\top & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\ell}_\varphi \\ \boldsymbol{\ell}_p \end{bmatrix}, \quad (8)$$

where $\mathbf{P}$ is an $n \times s$ matrix of polynomials evaluated at stencil nodes as is defined in a purely monomial approximation (2), and $\boldsymbol{\ell}_p$ is the vector of values assembled by applying the considered operator $\mathcal{L}$ to the polynomials at $\mathbf{x}_c$

$$l_p^i = (\mathcal{L}p_i(\mathbf{x}))|_{\mathbf{x}=\mathbf{x}_c} \quad (9)$$

and $\boldsymbol{\lambda}$ are Lagrange multipliers.

Finally, the system (8) is solved to obtain weights and the approximate operator $\mathcal{L}$ at $\mathbf{x}_c$. Lagrange multipliers are discarded.

Note that the exactness of (7) ensures the convergence behavior and also provides direct control over the convergence rate of the RBF-FD variant, since the local approximation has the same order as the polynomial basis used [21]. Also notice that the linear system (8) is larger when RBFs are augmented with monomials compared to when only monomials are used, making the RBF-FD approximation method computationally more expensive than the WLS approach.

Implementation note

All elements of the solution procedure using meshless methods used in this paper are implemented in C++ using an object-oriented approach and a template system to achieve dimensionality independence. The numerical library used in this work and developed in-house is the *Medusa library* [23].

III. CASE SETUP

Any conclusions drawn from the analysis should not be specific to any particular domain shape or problem setup. We therefore choose a simple Poisson problem with Dirichlet boundary conditions on a d -dimensional sphere. All observations we make on this simple example should be understood as fundamental properties of the approximation method employed and therefore apply to all more complex domain shapes and problem setups.

In the form of a PDE system, Poisson problem can be

written as

$$\nabla^2 u(\mathbf{x}) = f(\mathbf{x}) \quad \text{in } \Omega, \quad (10)$$

$$u(\mathbf{x}) = \prod_{i=1}^d \sin(\pi x_i) \quad \text{on } \partial\Omega \quad (11)$$

where the right hand side was chosen to be

$$f(\mathbf{x}) = -d\pi^2 \prod_{i=1}^d \sin(\pi x_i) \quad (12)$$

and, as previously noted, the domain Ω is a d -dimensional sphere

$$\Omega = \{\mathbf{r} \in \mathbb{R}^d, \|\mathbf{r}\| \leq 1\}. \quad (13)$$

Example solution to equations (10)–(12) is shown in Figure 1.

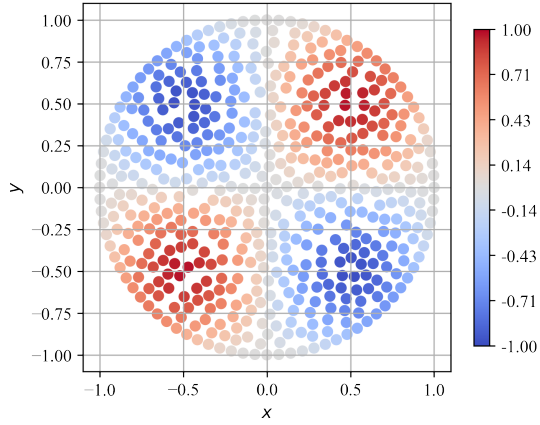


Figure 1. Example solution to 2D Poisson problem with Dirichlet boundary conditions on $N = 523$ scattered nodes.

The closed form solution of the above problem is $u(\mathbf{x}) = \prod_{i=1}^d \sin(\pi x_i)$. Having the closed form solution allows us to evaluate the accuracy of the numerical solution $\hat{u}$ by computing the infinity norm error

$$e_\infty = \frac{\|\hat{u} - u\|_\infty}{\|u\|_\infty}, \quad \|u\|_\infty = \max_{i=1, \dots, N} |u_i|. \quad (14)$$

The infinity norm was chosen because the authors have shown in [15] that it measures the lowest convergence rates and does not involve averaging, unlike the commonly chosen 2-norm error.

Numerical results are obtained using the two meshless methods described in Section II, for a given domain discretization. Firstly, the RBF-FD using PHS radial basis function $\varphi(r) = r^3$ and monomial augmentation up to and including order $m \in \{2, 4, 6\}$ is used, and secondly, the WLS approach using only monomials up to and including the same order m is used. After both variants have been employed, the domain discretization is discarded and discretized again so that the number of nodes N remains approximately the same, but the positions of

the discretization nodes are different. The same process is repeated $N_{runs} = 100$ times. Note that after every domain discretization, the shapes must also be recomputed. This potentially leads to a different accuracy of the numerical solution and allows us to observe the stability of the numerical methods.

The aim of this research is to determine which of the two approximation methods is more prone to a non-optimal discretization of the domain. For that, we introduce a normalized infinity norm error

$$\frac{e_\infty^{max} - e_\infty^{min}}{e_\infty^{median}} \quad (15)$$

effectively describing the largest norm difference identified within the $N_{runs} = 100$ runs, divided by the median value.

IV. RESULTS

In this section results are presented. All computations were performed with parallel execution on a computer with AMD Threadripper 3990X processor and 8x32GB DDR4 memory. The code¹ was compiled using g++ (GCC) 9.3.0 for Linux with `-O3 -DNDEBUG -fopenmp` flags.

In all of the following figures, the blue colour is used to indicate the order of the approximation method $m = 2$, red for $m = 4$ and green for $m = 6$. In the following subsections, the results for each dimension are presented separately, while our findings are summarized and presented as part of our conclusions in Section V. Only $d = \{2, 3\}$ dimensional domains are studied, since scattered nodes do not make sense in $d = 1$ dimensional problems and higher dimensional spaces $d > 3$ are beyond the scope of this paper.

A. The effect of stencil size

This section presents the effect of stencil size on accuracy of the numerical solution and on the stability of the meshless variant. A scan over a range of stencil sizes n is made and shown in Figure 2. Numerical solution has been obtained $N_{runs} = 100$ times at any given stencil size n , and with new domain discretization after every run keeping the total node count $N \approx 40600$. We can clearly observe that the error can be significantly higher if the support sizes are not sufficiently large, independent of the meshless variant. However, beyond that point, the dependency of the accuracy and the stability of the numerical solution on the stencil size is practically negligible. The tipping point for both approximation methods is in the neighborhood of recommendations made by Bayona [21] for the RBF-FD, that is

$$n = 2 \binom{m+d}{d}. \quad (16)$$

¹The source code is available at: https://gitlab.com/e62Lab/public/cp-2022-mipro-engine_stability under tag v1.2.

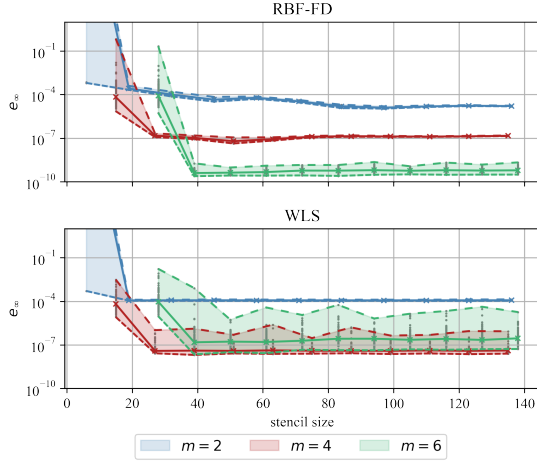


Figure 2. Support size scan in two-dimensional domain with approximately $N \approx 40600$ nodes.

Although all of the above conclusions were made on a 2-dimensional case, similar observations can be made in a three-dimensional domain, shown in Figure 3. However, in a three-dimensional case, a smaller number of discretization points $N \approx 28100$ has been used due to the fact that the support sizes are generally larger (see equation (16)), making the computational times longer.

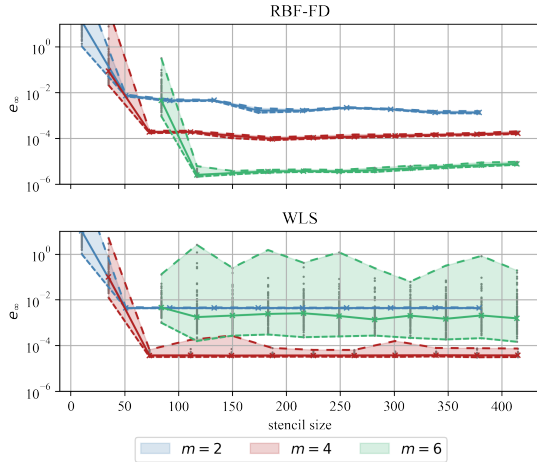


Figure 3. Support size scan in three-dimensional domain with approximately $N \approx 28100$ nodes.

B. Convergence rates

Considering the observations from the previous subsection, we continue our analysis by limiting ourselves to a single support size as defined in equation (16).

Convergence rates in the case of a two-dimensional domain are shown in Figure 4. The fact that the errors eventually diverge is a consequence of the errors in finite precision arithmetic, as previously observed by Flyer [22]. As expected, the order of magnitude of the infinity norm error is the same for both approximation methods, but small differences can be observed. First, the accuracy

achieved with the higher order ($m = 6$) approximation is significantly better with the RBF-FD than with the WLS approach, and second, the spread around a median error value is significantly smaller for the RBF-FD.

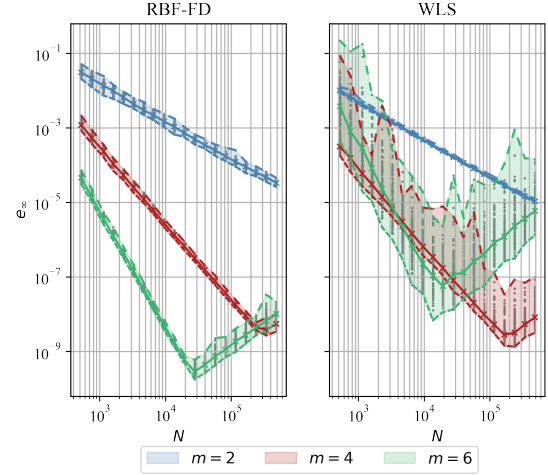


Figure 4. Convergence rates in two-dimensional domain.

The spread observed after successive $N_{runs} = 100$ runs is further examined in Figure 5, where the normalized spread computed as defined in (15) is evaluated and shown. We can see that the normalized spread is on average approximately constant, i.e., independent of the number of discretization points and approximately equal to 1 for a RBF-FD variant. We also find that the spread is about two orders of magnitude larger and unpredictable for the WLS approximation - with one exception, that is the low order WLS approximation ($m = 2$), which clearly outperforms the RBF-FD variant in terms of stability and precision. In general, in two-dimensional domains, the RBF-FD variant is more stable, achieving better accuracy for higher order approximations, while lower order approximations are more stable and computationally cheaper to obtain with the WLS variant.

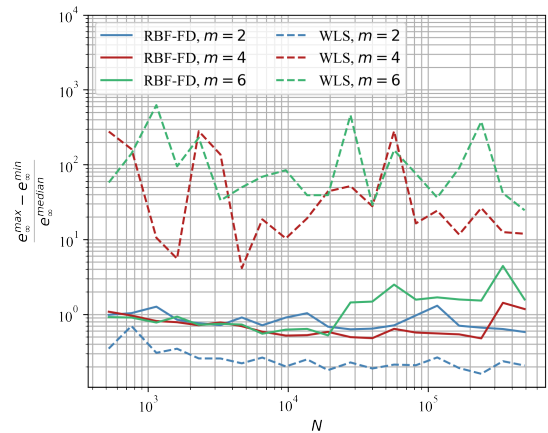


Figure 5. Normalized spread size around median value in two-dimensional domain.

Furthermore, convergence rates for a three-dimensional case are shown in Figure 6. At first glance, we observe some similarities with the two-dimensional case in Figure 4: Firstly, the errors are of the same order of magnitude for both meshless variants, and secondly, the spread size after successive $N_{runs} = 100$ runs is again in favour of the RBF-FD approximation method. More importantly, for the high order WLS approximation ($m = 6$) and smaller number of discretization nodes $N \approx 10^3$, the infinity error norm is of the order of 10^1 . This essentially means that the WLS variant did not converge well. This is an important observation when studying stability, as we do not observe such a case in the results obtained with the RBF-FD variant.

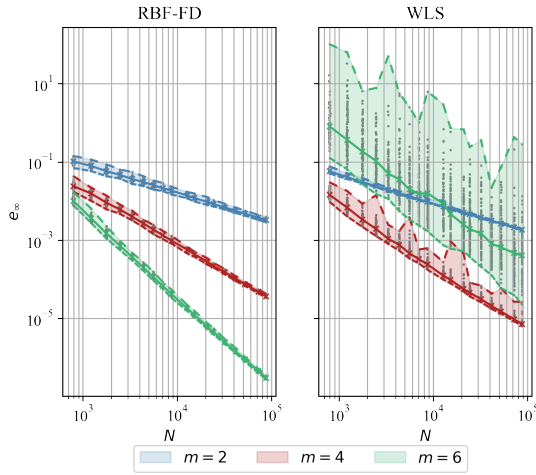


Figure 6. Convergence rates in three-dimensional domain.

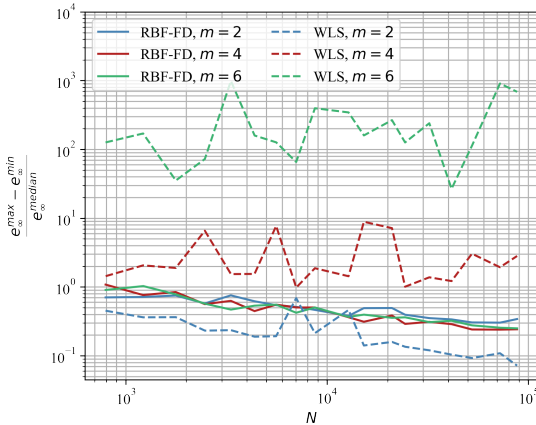


Figure 7. Normalized spread size around median value in three-dimensional domain.

Stability in a three-dimensional space is further studied in Figure 7. Similarly to the two-dimensional case in Figure 5, we find that the RBF-FD variant is substantially more stable than the WLS variant, especially for higher order approximations ($m > 2$). We also find that the amount of spread is slowly decreasing with the number

of discretization points N which was not observed in the two-dimensional case. This phenomenon is clearly seen for the RBF-FD approximations and for lower order WLS approximations ($m = 2$). The low order approximations ($m = 2$) again appear to be in favour of the WLS meshless variant in terms of stability and computational complexity, although in some cases the normalized spread can be larger than that obtained by the RBF-FD (note the two peaks in the normalized spread for the low order WLS approximation in Figure 7). However, the higher order approximations are significantly more stable when obtained with the RBF-FD variant.

V. CONCLUSIONS

In this paper, we compare the stability of two variants of meshless methods. We study solutions obtained with RBF-FD using polyharmonic splines augmented with monomials and with WLS approximation using only monomials as basis functions. Stability is assessed by solving a two- and three-dimensional Poisson problem with a tractable solution that allows us to evaluate the numerical solution in terms of the infinity norm error.

We observe the effect of large enough stencil sizes has a negligible effect on the accuracy of the numerical solution and stability of the meshless variant. Additionally we show that in terms of stability, the RBF-FD variant can be several orders of magnitude better making it more prone to a non optimal domain discretization. This is particularly evident for the higher order approximations ($m > 2$), while the lower order approximations ($m = 2$) are better and computationally cheaper to obtain using the WLS variant of meshless methods. We also find that the accuracy of the numerical solution obtained with a higher order approximation method can be significantly better when using RBF-FD than when using WLS.

Further research is required to provide more accurate and descriptive guidelines as to which approximation method is most appropriate in particular cases. Although we provide some comments on the effect of stencil selection, we believe this aspect is in need of a more detailed study to make the methods not only stable but also computationally effective.

According to our observations, RBF-FD variant is best used for problems that require a higher order approximation, while lower order approximations return better and faster results with WLS variant.

ACKNOWLEDGMENTS

The authors would like to acknowledge the financial support of the ARRS research core funding No. P2-0095, ARRS project funding No. J2-3048 and the World Federation of Scientists.

REFERENCES

- [1] Miha Rot and Gregor Kosce. Refined rbf-fd analysis of non-newtonian natural convection, 2022.
- [2] Jure Slak and Gregor Kosce. Refined meshless local strong form solution of cauchy-navier equation on an irregular

- domain. *Engineering analysis with boundary elements*, 100:3–13, 2019.
- [3] Jure Slak and Gregor Kosec. Adaptive radial basis function-generated finite differences method for contact problems. *International Journal for Numerical Methods in Engineering*, 119(7):661–686, 2019.
- [4] Boštjan Mavrič and Božidar Šarler. Equivalent-pde based stabilization of strong-form meshless methods applied to advection-dominated problems. *Engineering Analysis with Boundary Elements*, 113:315–327, 2020.
- [5] Slobodan Milovanović and Lina von Sydow. Radial basis function generated finite differences for option pricing problems. *Computers & Mathematics with Applications*, 75(4):1462–1481, 2018.
- [6] Leon B Lucy. A numerical approach to the testing of the fission hypothesis. *The astronomical journal*, 82:1013–1024, 1977.
- [7] Robert A Gingold and Joseph J Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977.
- [8] Eugenio Oñate, Franco Perazzo, and J Miquel. A finite point method for elasticity problems. *Computers & Structures*, 79(22-25):2151–2163, 2001.
- [9] L Gavete, ML Gavete, and JJ1046 Benito. Improvements of generalized finite difference method and comparison with other meshless method. *Applied Mathematical Modelling*, 27(10):831–847, 2003.
- [10] AI Tolstykh and DA Shirobokov. On using radial basis functions in a “finite difference mode” with applications to elasticity problems. *Computational Mechanics*, 33(1):68–79, 2003.
- [11] Matjaž Depolli and Roman Trobec. Computational efficiency of linear system construction for MLPG method on a multicore computer. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 200–205, 2019.
- [12] Roman Trobec and Matjaž Depolli. A k-d tree based partitioning of computational domains for efficient parallel computing. In *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, pages 284–290, 2021.
- [13] Jure Slak and Gregor Kosec. On generation of node distributions for meshless pde discretizations. *SIAM journal on scientific computing*, 41(5):A3202–A3229, 2019.
- [14] Matjaž Depolli, Jure Slak, and Gregor Kosec. Parallel domain discretization algorithm for rbf-fd and other meshless numerical methods for solving pdes, 2022.
- [15] Mitja Jančič, Jure Slak, and Gregor Kosec. Monomial augmentation guidelines for rbf-fd from accuracy versus computational time perspective. *Journal of Scientific Computing*, 87(1):1–18, 2021.
- [16] Mitja Jančič, Jure Slak, and Gregor Kosec. p-refined rbf-fd solution of a poisson problem. In *2021 6th International Conference on Smart and Sustainable Technologies (SpliTech)*, pages 01–06. IEEE, 2021.
- [17] Thibault Jacquemin and Stéphane PA Bordas. A unified algorithm for the selection of collocation stencils for convex, concave, and singular problems. *International Journal for Numerical Methods in Engineering*, 122(16):4292–4312, 2021.
- [18] Oleg Davydov and Dang Thi Oanh. Adaptive meshless centres and rbf stencils for poisson equation. *Journal of Computational Physics*, 230(2):287–304, 2011.
- [19] Gregor Kosec and B Sarler. Local rbf collocation method for darcy flow. *Computer Modeling in Engineering and Sciences*, 25(3):197, 2008.
- [20] Simon Sirca and Martin Horvat. *Computational methods for physicists: compendium for students*. Springer, 2012.
- [21] Victor Bayona, Natasha Flyer, Bengt Fornberg, and Gregory A Barnett. On the role of polynomials in rbf-fd approximations: II. numerical solution of elliptic pdes. *Journal of Computational Physics*, 332:257–273, 2017.
- [22] Natasha Flyer, Bengt Fornberg, Victor Bayona, and Gregory A Barnett. On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy. *Journal of Computational Physics*, 321:21–38, 2016.
- [23] Jure Slak and Gregor Kosec. Medusa: A c++ library for solving pdes using strong form mesh-free methods. *ACM Transactions on Mathematical Software (TOMS)*, 47(3):1–25, 2021.

Chapter 3

hp-Adaptive Solution Procedure

Consider a problem where the solution is known to vary smoothly in some regions of the domain, but rapidly in others. A traditional numerical method would require a fine discretization throughout the entire domain, even in the regions where the solution is smoother. This would result in an unnecessarily large number of computational nodes and correspondingly large computational complexity. An *adaptive* solution procedure, on the other hand, starts with a coarse discretization and then, based on error indication or error estimation, iteratively refines the discretization in regions where the accuracy of the numerical solution is insufficient and coarsens it in regions where the accuracy is high. This allows the numerical method to achieve the desired accuracy while using a smaller number of discretization nodes – focusing the available computational resources on the parts of the domain that require the most accuracy.

The *adaptive technique* controlling the spatial discretization described in this simplified example is referred to as the *h*-adaptivity, while *p*-adaptivity spatially controls the approximation order for the same purpose. The two can be combined to form a so-called *hp*-adaptive solution procedure.

This chapter presents the key components of a strong-form meshless *hp*-adaptive solution procedure, which iteratively adjust both the local discretization and the approximation order in regions with large error indicator values. We discuss the two remaining components needed for a complete *hp*-adaptive solution procedure (presented in Section 3.3): the benefits of spatially-varying the approximation order in Section 3.1, and the development of an original error indicator that eliminates the need for human intervention in the solution procedure in Section 3.2.

3.1 *p*-refined RBF-FD Solution of a Poisson Problem

The traditional approach to numerical treatment of PDEs involves the use of a fixed approximation order throughout the entire computational domain. However, the underlying phenomena may exhibit spatial variations in its complexity or require different levels of accuracy in different domain regions. In such cases, a constant approximation order across the domain can lead to a computationally ineffective solution procedure, as it results in overuse of computational resources in simpler regions or a lack of precision in more complex domain regions.

To address this issue and achieve a higher level of efficient PDE-solving procedure, some researchers have turned their attention towards spatially-varying approximation order [82]. This technique enables the adaptation of approximation orders based on the spatial characteristics of the problem – employing high-order approximations where needed and low-order approximations where the accuracy of the numerical solution is satisfying.

In the following publication, a simple Poisson problem is used to demonstrate the advantages of p -refined solutions. For demonstration purposes, the spatial distribution of the approximation order is defined a priori, i.e. without the use of error indicators or error estimators.

Contributions.

This section represents contribution **C4**. We demonstrate that a well-thought-out spatial distribution of approximation orders can be beneficial in terms of stability and computational complexity of the solution procedure and accuracy of the numerical solution.

Addressed hypotheses.

In the following publication, we demonstrate that the RBF-FD approximation method can be successfully employed with spatially-variable approximation orders, thus, finally confirming hypothesis **H3**. Moreover, on a simple two-dimensional peak problem we demonstrate that a non-trivial spatial distribution of approximation orders can improve the accuracy of the numerical solution with negligible increase in the total computational complexity of the solution procedure due to larger stencil size requirement of high-order approximations. The latter observation confirms hypotheses **H1** and **H5**.

Publications included in this section:

- M. Jančič, J. Slak, and G. Kosec, “ p -refined RBF-FD solution of a poisson problem,” in *2021 6th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2021, pp. 01–06. DOI: 10.23919/SpliTech52315.2021.9566401

Regarding my contribution: I made a literature overview of the topic, planned and performed the experiments and jointly prepared the manuscript with co-authors.

p -refined RBF-FD solution of a Poisson problem

Mitja Jančič

“Jožef Stefan” International Postgraduate School
Jamova cesta 39, 1000 Ljubljana, Slovenia
and

“Jožef Stefan” Institute
Parallel and Distributed Systems Laboratory,
Jamova cesta 39, 1000 Ljubljana, Slovenia
Email: mitja.jancic@ijs.si

Gregor Kosec

“Jožef Stefan” Institute
Parallel and Distributed Systems Laboratory,
Jamova cesta 39, 1000 Ljubljana, Slovenia
Email: gregor.kosec@ijs.si

Jure Slak

“Jožef Stefan” Institute
Parallel and Distributed Systems Laboratory,
Jamova cesta 39, 1000 Ljubljana, Slovenia
Email: jure.slak@ijs.si

Abstract—Local meshless methods obtain higher convergence rates when RBF approximations are augmented with monomials up to a given order. If the order of the approximation method is spatially variable, the numerical solution is said to be p -refined. In this work, we employ RBF-FD approximation method with polyharmonic splines augmented with monomials and study the numerical properties of p -refined solutions, such as convergence orders and execution time. To fully exploit the refinement advantages, the numerical performance is studied on a Poisson problem with a strong source within the domain.

Index Terms—Meshless methods, p -refinement, RBF-FD, high order method

I. INTRODUCTION

Meshless methods are becoming a strong alternative to mesh based methods, when numerical treatment of partial differential equations is required. A strong advantage of meshless methods is that they can operate on scattered nodes, contrary to mesh-based methods, that need a computationally expensive mesh to operate. Many different meshless methods have been proposed so far, e.g. meshless Element Free Galerkin [1], the Local Petrov-Galerkin [2], h-p cloud method [3] and others. In this paper we use a method that generalizes the traditional Finite Difference Method, called radial-basis-function-generated finite differences (RBF-FD). From a historical point of view, RBF-FD was first mentioned by Tolstykh [4] in 2003 and has since been successfully used in a vast range of problems, e.g. convection-diffusion problems [5], fluid flow problems [6], contact problems [7], scattering [8], dynamic thermal rating of power lines [9], etc.

The RBF-FD use RBFs to approximate the linear differential operators [10]. Most of the RBFs, like Hardy’s multiquadrics or Gaussians, include shape parameter [11] that directly affects the stability of the approximation and accuracy

of the solution [12]. To avoid shape parameter problems altogether, Polyharmonic splines (PHS) have been proposed [13], however, PHS alone do not guarantee convergent behavior. Therefore, RBFs are augmented with monomials up to given order [13]. The RBF part of the approximation takes care of the potential ill-conditioning [14], while the polynomial part not only ensures convergent behavior but also allows direct control over the convergence rate.

It has already been proven that having the control over the convergence rate is beneficial, when a compromise between the accuracy of the solution and computational time is needed [15]. However, in this paper, we exploit the ability to control the order of the approximation method to employ spatial variability of the approximating method order. Such solution procedure refinement is also known as p -refinement [16] and is a well known refinement procedure in the scope of finite element methods [17], where it also forms the basis of the highly successful hp -adaptive methods. In this paper, convergence rates and computational times of p -refined solutions are studied. It is shown that spatially variable order of the approximation method can notably reduce the computational time and improve the convergence rate at the same time.

The rest of the paper is organized as follows: In section II the main steps of solution procedure are described, in section III, the numerical example used to test the numerical performance of p -refinement is presented, in section IV, results are presented and finally, in section V conclusions are given and future work is proposed.

II. NUMERICAL APPROXIMATION

The solution procedure can be roughly divided into three steps. Using a dedicated node positioning algorithm the domain is first discretized. Afterwards, the differential operators are approximated in each node, resulting in stencil weights. Finally, the system of PDEs is discretized and, therefore, transformed to a system of linear equations. The system is

The authors would like to acknowledge the financial support of the ARRS research core funding No. P2-0095 and the The World Federation of Scientists program.

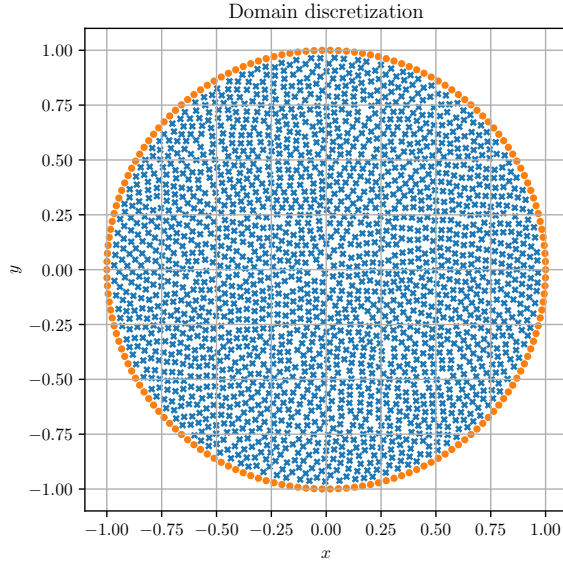


Fig. 1. An example of circular domain populated with scattered nodes. Circles represent the nodes in the interior of the domain while the crosses are on its boundary.

solved and its solution stands for a numerical solution u_h of the considered system of PDEs.

A. Domain discretization

In the first applications of meshless methods, researchers used existing mesh generators and simply discarded the connectivity information to obtain the nodes [18]. However, such procedure is computationally wasteful and conceptually wrong. Additionally, it can also be problematic, since some authors reported that it failed to produce nodal distributions of sufficient quality [19].

Researchers therefore soon started proposing dedicated node positioning algorithms. In this paper, a dimension-independent node generation algorithm [20] is used to populate the domain with scattered nodes. The algorithm ensures a quasi-uniform internodal distance h as seen in figure 1.

B. Approximation of partial differential operators

The behavior of a numerical method for solving systems of PDEs is defined by the approximation of partial differential operators. In the scope of meshless methods, the approximation is done as follows: Consider an operator $\mathcal{L}$ at a point $\mathbf{x}_c$. The approximation of $\mathcal{L}$ is sought using the ansatz

$$(\mathcal{L}u)(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i), \quad (1)$$

where w_i are the *stencil weights*, n is the *stencil size* or *support size*, and u is the unknown function. To simplify the writing, the weights and function values are assembled into vectors $\mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)$ and $\mathbf{u}$ respectively. This notation allows us rewrite

the operator approximation (1) in the form of a dot product between the two vectors

$$(\mathcal{L}u)(\mathbf{x}_c) \approx \mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)^T \mathbf{u}. \quad (2)$$

While the field values $\mathbf{u}$ from equation (2) are considered as unknown, the weights $\mathbf{w}_{\mathcal{L}}(\mathbf{x}_c)$ need to be determined. To determine the weights, equality in equation (2) is enforced for a given set of basis functions. In this paper we use RBFs, denoted as ϕ_j . The radially symmetric RBFs, centered at stencil nodes $\mathbf{x}_i$, can be written in the form

$$\phi_j(\mathbf{x}) = \phi(\|\mathbf{x} - \mathbf{x}_j\|), \quad (3)$$

for a radial function ϕ . Each RBF corresponds to one linear equation

$$\sum_{i=1}^n \mathbf{w}_i \phi_j(\mathbf{x}_i) = (\mathcal{L}\phi_j)(\mathbf{x}_c) \quad (4)$$

with unknown weights $\mathbf{w}_i$ and index i running over all support nodes. Assembling these n equations into matrix form, we obtain a system of linear equations

$$\begin{bmatrix} \phi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_1\|) \\ \vdots & \ddots & \vdots \\ \phi(\|\mathbf{x}_1 - \mathbf{x}_n\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_n\|) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} l_{\phi,1} \\ \vdots \\ l_{\phi,n} \end{bmatrix} \quad (5)$$

for

$$l_{\phi,j} = (\mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_j\|))|_{\mathbf{x}=\mathbf{x}_c}. \quad (6)$$

The system (5) is often compactly written as

$$\mathbf{A}\mathbf{w} = \mathbf{l}_{\phi}, \quad (7)$$

where $\mathbf{A}$ is symmetric and for some basis functions positive definite [21].

There are many different choices for RBFs. However, commonly used Hardy's multiquadrics or Gaussians both depend on the shape parameter that directly affects the stability and accuracy of the approximation [12]. To avoid shape parameters entirely, Polyharmonic splines (PHS) are used, defined as

$$\phi(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \quad (8)$$

where r denotes the Euclidean distance.

Note that using the PHS alone does not guarantee the convergence of local approximations from equation (5). Therefore, the approximation is additionally augmented with monomials to omit the problems [13], which is done as follows. Let $p_1, \dots, p_s$ be polynomials for which exactness of ansatz (2) is again enforced. Monomials are often chosen up to a certain order m , resulting in $s = \binom{m+d}{m}$ monomials for a d -dimensional space.

The additional enforcement is introduced by extending the system (5) with the new conditions

$$\begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{l}_{\phi} \\ \mathbf{l}_p \end{bmatrix}. \quad (9)$$

Here

$$P = \begin{bmatrix} p_1(\mathbf{x}_1) & \cdots & p_s(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_n) & \cdots & p_s(\mathbf{x}_n) \end{bmatrix} \quad (10)$$

is a $n \times s$ matrix of polynomials evaluated at stencil nodes and

$$\ell_p = \begin{bmatrix} (\mathcal{L}p_1)|_{\mathbf{x}=\mathbf{x}_c} \\ \vdots \\ (\mathcal{L}p_s)|_{\mathbf{x}=\mathbf{x}_c} \end{bmatrix} \quad (11)$$

is the vector of values assembled by applying the considered operator $\mathcal{L}$ to the polynomials at $\mathbf{x}_c$. Weights obtained by solving (9) are taken as approximations of $\mathcal{L}$ at $\mathbf{x}_c$ while additional unknowns λ , the Lagrange multipliers, are discarded.

The augmentation with monomials not only helps with convergence but also provides direct control over the convergence rate, since the local approximation of the linear operator has the same order as the basis used [13], while the RBF part handles the potential ill-conditioning in purely polynomial approximations [14].

C. PDE discretization

Consider the boundary value problem with dirichlet boundary condition

$$\mathcal{L}u = f \text{ in } \Omega, \quad (12)$$

$$u = g \text{ on } \partial\Omega. \quad (13)$$

The domain Ω is discretized with N scattered nodes $\mathbf{x}_i$ with quasi-uniform inter-nodal spacing h . Out of N nodes, N_i are in the interior and N_d on the boundary $\partial\Omega$.

The stencils $\mathcal{N}(\mathbf{x}_i)$ for each node $\mathbf{x}_i$ are then selected. Commonly a single stencil constitutes of n closest points, including the node itself. Choosing the right stencil size n is far from trivial, however it has been recommended by Bayona [13] to take at least $n = 2\binom{m+d}{d}$ nodes.

In the next step, linear operator $\mathcal{L}$ is approximated at nodes $\mathbf{x}_i$, using the procedure described in section II-B. For each interior node $\mathbf{x}_i$, the equation $(\mathcal{L}u)(\mathbf{x}_i) = f(\mathbf{x}_i)$ is approximated by a linear equation

$$\mathbf{w}_{\mathcal{L}}(\mathbf{x}_i)^T \mathbf{u} = \mathbf{f}, \quad (14)$$

where vectors $\mathbf{f}$ and $\mathbf{u}$ represent values of function f and unknowns u in stencil nodes of $\mathbf{x}_i$. Similarly, for each Dirichlet boundary node $\mathbf{x}_i$, we obtain the equation

$$u_i = g(\mathbf{x}_i). \quad (15)$$

All $N_i + N_d$ equations are assembled into a linear sparse system, with approximately Nn nonzero elements. The solution u_h of the system is a desired numerical approximation of u . Note that using the spatially variable order of the approximation method can lead to a very different number of nonzero elements in the linear sparse system.

III. NUMERICAL EXAMPLE

The behavior of the described solution procedure and its implementation is studied on a Poisson problem with Dirichlet boundary condition. We aim to demonstrate and analyze the p -refined solution procedure, where the order of the approximation method varies throughout the computational nodes in the domain.

Governing equations are

$$\nabla^2 u(\mathbf{x}) = f_{\text{lap}}(\mathbf{x}) \quad \text{in } \Omega, \quad (16)$$

$$u(\mathbf{x}) = f(\mathbf{x}) \quad \text{on } \partial\Omega \quad (17)$$

where the domain Ω is a $d = 2$ dimensional circle

$$\Omega = \left\{ \mathbf{x} \in \mathbb{R}^2, \|\mathbf{x}\| \leq \frac{3}{2} \right\}. \quad (18)$$

To fully exploit the advantages of p -refinement, the right hand side $f(\mathbf{x})$ was chosen to have a relatively strong source within the domain at $\mathbf{x}_s = \frac{1}{2}$, i.e.

$$f(\mathbf{x}) = \frac{1}{25 \|4\mathbf{x} - \mathbf{2}\|^2 + 1}. \quad (19)$$

The Laplacian from f can also be computed as

$$f_{\text{lap}}(\mathbf{x}) = 3200 \frac{25 \|4\mathbf{x} - \mathbf{2}\|^2}{f(\mathbf{x})^3} - 800 \frac{d}{f(\mathbf{x})^2}. \quad (20)$$

The domain was filled with N scattered nodes ranging from $N = 1093$ to $N = 978013$. The problem was solved using RBF-FD with PHS augmented with monomials of order $m \in \{2, 4, 6\}$. Stencils for each node were selected by taking n closest nodes where n was determined as recommended by Bayona [13]

$$n = 2 \binom{m+d}{d}. \quad (21)$$

To demonstrate the effect of p -refinement any combination of approximation orders m can be used. Naturally, to increase the overall convergence rate of the numerical solutions, the highest approximation order is used where the numerical solution is expected to have the biggest error, e.g. in the neighborhood of the strong source at $\mathbf{x}_s$. We define two radii r_6 and r_4 around source center $\mathbf{x}_s$. All computational nodes within the radius r_6 , i.e. $\{\mathbf{x}_i, \|\mathbf{x}_i - \mathbf{x}_s\| \leq r_6\}$, are assigned with approximation augmented with monomials of degree $m = 6$, nodes within the annulus $\{\mathbf{x}_i, r_6 < \|\mathbf{x}_i - \mathbf{x}_s\| \leq r_4\}$ are assigned approximation augmented with monomials of degree $m = 4$, while the remaining nodes are assigned approximation augmented with monomials of degree $m = 2$. So the order of the approximation method is spatially variable and can be compactly written as

$$m = \begin{cases} 6, & \|\mathbf{x}_i - \mathbf{x}_s\| \leq r_6 \\ 4, & r_6 < \|\mathbf{x}_i - \mathbf{x}_s\| \leq r_4 \\ 2, & \text{otherwise.} \end{cases} \quad (22)$$

Additionally, three different combinations c_i of radii r_6 and r_4 have been used in this paper

$$c_1 = \left\{ r_6 = 0, r_4 = \frac{1}{10} \right\}, \quad (23)$$

$$c_2 = \left\{ r_6 = \frac{1}{10}, r_4 = \frac{1}{5} \right\} \text{ and} \quad (24)$$

$$c_3 = \left\{ r_6 = \frac{1}{5}, r_4 = \frac{2}{5} \right\}. \quad (25)$$

All three cases of spatially variable order of the approximation method are also shown in figure 2.

A. Error evaluation

Closed form solution u allows us to compute the accuracy of numerical solution u_h . In this paper, the error is evaluated in computational nodes with the infinity norm

$$e_\infty = \frac{\|u_h - u\|_\infty}{\|u\|_\infty}, \quad \|u\|_\infty = \max_{i=1,\dots,N} (26)$$

The infinity norm is chosen as it is the strictest, but the authors also observed the same behavior using 2-norm or 1-norm.

IV. RESULTS

We compare the convergence rates of unrefined and p -refined numerical solution u_h to the problem from section III. Finally, we also study the effect of p -refinement on computational times.

A. Convergence rates

Convergence rates were estimated by computing the slope of the appropriate data subset. Selected convergence rates are shown in figure 3. We observe that the numerical solutions converge for all chosen augmentation orders $m \in \{2, 4, 6\}$. The expected convergence rate of $O(h^m)$ is, however, not reached, but that is to be expected due to the strong source within the computational domain. The convergence curve of a p -refined solution for combination c_2 is also added to the figure 3. It is clear that the refined solution convergences at a significantly better convergence rate compared to the convergence rate at $m = 2$, regardless of the fact that the majority of the stencils were still computed using monomials of order $m = 2$. This confirms our beliefs that the biggest contribution to the error comes from the strong source and that the error can be, to some extent, mitigated by locally using a higher order method, i.e. p -refinement.

The effect of p -refinement is furthermore studied in figure 4, where convergence rates of refined solutions for all combinations c of radius values are shown. We observe how the number of nodes used for higher order approximation affects the convergence rates. The convergence rate for combination c_3 with the most higher order node stencils, is practically the same as the convergence rate of unrefined solution with the highest augmented monomial $m = 6$, even though $m = 6$ augmentation has only been applied to roughly 2 % of all computational nodes and $m = 4$ to roughly 5 %.

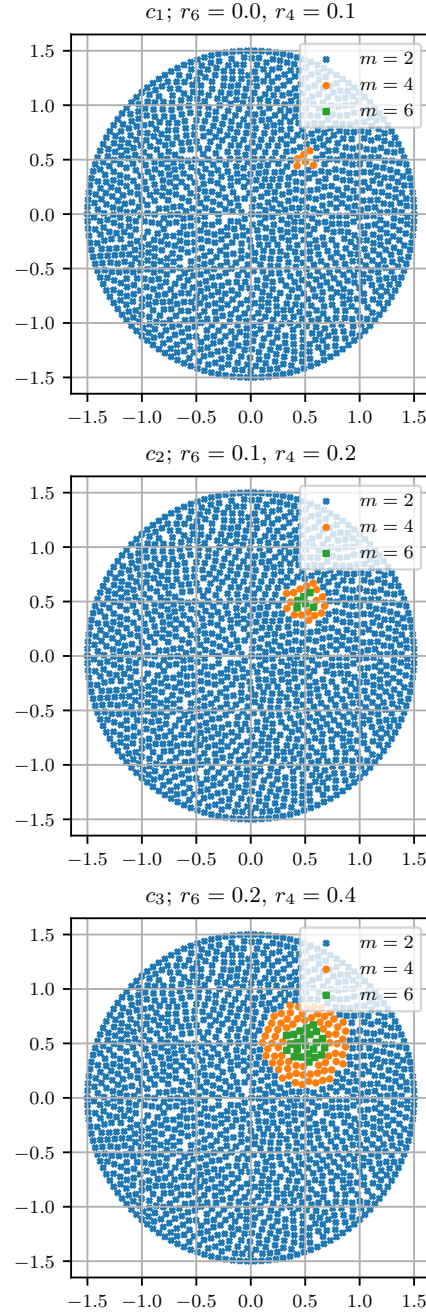


Fig. 2. Three different stages of p -refinement used. Squares are used to mark the nodes where approximation is augmented with monomials of order $m = 6$, circles for $m = 4$ and crosses for $m = 2$.

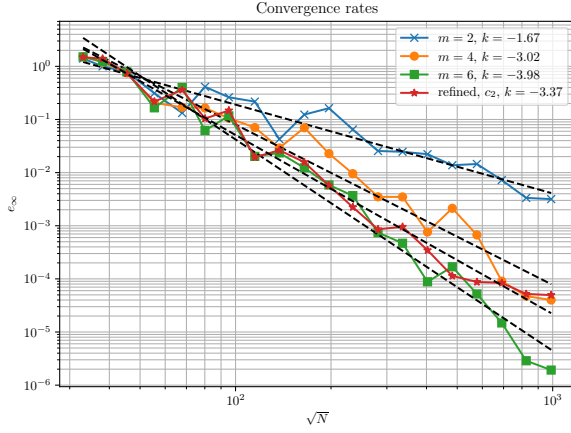


Fig. 3. Convergence rates for different augmentation orders m with respect to the number of nodes N .

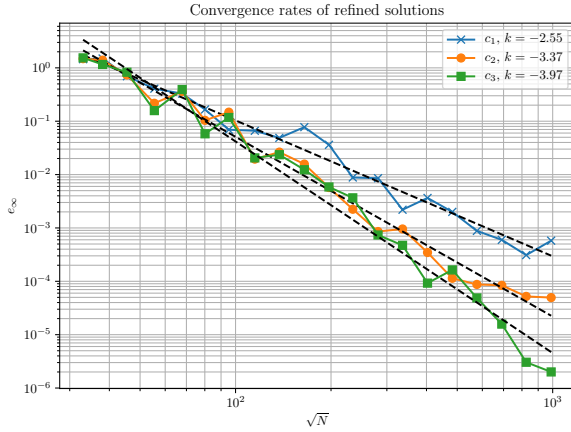


Fig. 4. Refined convergence rates for different radius values combinations.

B. Computation times

In this section an overview of the total computational times is provided. All computations were performed on a single core of a computer with AMD EPYC 7702 64-Core processor and 512GB of DDR4 memory. Code was compiled using g++ (GCC) 9.3.0 for Linux with `-O3 -DNDEBUG` flags. The sparse system is solved using the Pardiso solver on a single thread.

The total computational times are shown in figure 5, where the best run out of 5 is selected. The total computational times of unrefined solutions (dashed lines) increases with the monomial order m . This is expected, since the higher the order the larger the required stencil size and consequently longer computational times. The computational times for all refined solutions are similar to the unrefined solutions augmented with monomial order $m = 2$, which is also expected since the majority ($\approx 93\%$) of the nodes are assigned with

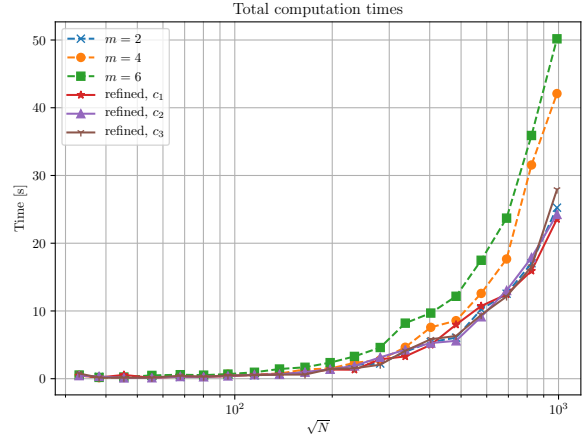


Fig. 5. Total computational times with respect to the domain size N .

augmentation using monomials of degree $m = 2$, however, results show that all refined solutions exhibit much better convergence rates (see figures 3 and 4). This ultimately means that employing the p -refinement enabled us to obtain significantly better convergence behavior with little to no additional costs to execution time. Furthermore, refined solution for c_3 combination with the largest radius values, measures the same convergence rate as unrefined with $m = 6$ ($k = -3.97$ vs. $k = -3.98$ respectively), but the former solution was obtained approximately two times faster.

V. CONCLUSIONS

A p -refinement procedure the RBF-FD meshless method is presented, where the order of the local approximation is spatially variable. We employed RBF-FD using PHS augmented with monomials of different degrees to solve a Poisson problem with a strong source within the computational domain. It is shown that p -refinement can improve the convergence rates at a very small cost to execution time, and much faster, than using a method with a higher global order of convergence.

However, observations show that the p -refinement has its limitations. In some cases, specially with local strong sources, the local description of the considered field is just not sufficient to provide good local approximations of linear differential operators. Therefore a plan is to combine the benefits of p -refinement with spatially variable nodal distributions, to provide better approximations around critical areas within the domain. This is also known as hp -refinement, and presents a major step towards hp -adaptivity.

REFERENCES

- [1] T. Belytschko, Y. Y. Lu, and L. Gu, "Element-free galerkin methods," *International journal for numerical methods in engineering*, vol. 37, no. 2, pp. 229–256, 1994.
- [2] S. N. Atluri and T. Zhu, "A new meshless local petrov-galerkin (mlpg) approach in computational mechanics," *Computational mechanics*, vol. 22, no. 2, pp. 117–127, 1998.

- [3] C. A. Duarte and J. T. Oden, “H-p clouds—an h-p meshless method,” *Numerical Methods for Partial Differential Equations: An International Journal*, vol. 12, no. 6, pp. 673–705, 1996.
- [4] A. Tolstykh and D. Shirobokov, “On using radial basis functions in a “finite difference mode” with applications to elasticity problems,” *Computational Mechanics*, vol. 33, no. 1, pp. 68–79, 2003.
- [5] G. Chandhini and Y. Sanyasiraju, “Local rbf-fd solutions for steady convection–diffusion problems,” *International Journal for Numerical Methods in Engineering*, vol. 72, no. 3, pp. 352–378, 2007.
- [6] G. Kosec, “A local numerical solution of a fluid-flow problem on an irregular domain,” *Advances in engineering software*, vol. 120, pp. 36–44, 2018.
- [7] J. Slak and G. Kosec, “Adaptive radial basis function–generated finite differences method for contact problems,” *International Journal for Numerical Methods in Engineering*, vol. 119, no. 7, pp. 661–686, 2019.
- [8] J. Slak, B. Stojanović, and G. Kosec, “High order rbf-fd approximations with application to a scattering problem,” in *2019 4th International Conference on Smart and Sustainable Technologies (SpliTech)*. IEEE, 2019, pp. 1–5.
- [9] M. Maksić, V. Djurica, A. Souvent, J. Slak, M. Depolli, and G. Kosec, “Cooling of overhead power lines due to the natural convection,” *International Journal of Electrical Power & Energy Systems*, vol. 113, pp. 333–343, 2019.
- [10] V. Bayona, M. Moscoso, M. Carretero, and M. Kindelan, “Rbf-fd formulas and convergence properties,” *Journal of Computational Physics*, vol. 229, no. 22, pp. 8281–8295, 2010.
- [11] H. Wendland, *Scattered data approximation*. Cambridge university press, 2004, vol. 17.
- [12] R. Schaback, “Error estimates and condition numbers for radial basis function interpolation,” *Advances in Computational Mathematics*, vol. 3, no. 3, pp. 251–264, 1995.
- [13] V. Bayona, N. Flyer, B. Fornberg, and G. A. Barnett, “On the role of polynomials in rbf-fd approximations: II. numerical solution of elliptic pdes,” *Journal of Computational Physics*, vol. 332, pp. 257–273, 2017.
- [14] N. Flyer, B. Fornberg, V. Bayona, and G. A. Barnett, “On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy,” *Journal of Computational Physics*, vol. 321, pp. 21–38, 2016.
- [15] M. Jančič, J. Slak, and G. Kosec, “Monomial augmentation guidelines for rbf-fd from accuracy versus computational time perspective,” *Journal of Scientific Computing*, vol. 87, no. 1, pp. 1–18, 2021.
- [16] F. Barros, S. Proença, and C. de Barcellos, “On error estimator and p-adaptivity in the generalized finite element method,” *International Journal for Numerical Methods in Engineering*, vol. 60, no. 14, pp. 2373–2398, 2004.
- [17] I. Babuska and M. Suri, “The p and hp versions of the finite element method: Basic principles and properties,” MARYLAND UNIV COLLEGE PARK INST FOR PHYSICAL SCIENCE AND TECHNOLOGY, Tech. Rep., 1992.
- [18] G.-R. Liu and D. Karamanlidis, “Mesh free methods: moving beyond the finite element method,” *Appl. Mech. Rev.*, vol. 56, no. 2, pp. B17–B18, 2003.
- [19] V. Shankar, R. M. Kirby, and A. L. Fogelson, “Robust node generation for mesh-free discretizations on irregular domains and surfaces,” *SIAM Journal on Scientific Computing*, vol. 40, no. 4, pp. A2584–A2608, 2018.
- [20] J. Slak and G. Kosec, “On generation of node distributions for meshless pde discretizations,” *SIAM Journal on Scientific Computing*, vol. 41, no. 5, pp. A3202–A3229, 2019.
- [21] M. D. Buhmann, *Radial basis functions: theory and implementations*. Cambridge university press, 2003, vol. 12.

3.2 The IMEX (IMplicit-EXplicit) Error Indicator

Error indicators are essential for adaptive solution procedures, because they provide a way to objectively indicate domain regions with inaccurate numerical solution [74]. Error indicators, combined with an adaptive procedure, allow us to achieve higher accuracy of the numerical solution through a non-trivial distribution of discretization resolution and/or spatial distribution of approximation order, in a fully automated manner.

Contributions.

This section represents contribution **C5**. We develop an original error indicator easily implemented in the context of meshless methods. The proposed IMEX error indicator makes use of the implicitly obtained numerical solution and explicit operators (approximated by a higher order approximation basis) to reconstruct the right-hand side of the governing PDE. The deviation of the explicit high order reconstruction ($f_{\text{RHS}}^{\text{ex}}$) from the exact right-hand side (f_{RHS}) corresponds to the error of the implicit solution (u^{im}).

The idea is similar to the one behind the ZZ-type indicators, where the deviation of the recovered high-order solution from the computed solution characterises the error. As long as the error in u^{im} is high, the explicit re-evaluation will not correctly solve the governing PDE. However, as the error in u^{im} decreases, the difference between f_{RHS} and $f_{\text{RHS}}^{\text{ex}}$ will also decrease, assuming that the error is dominated by the inaccuracy of u_{im} and not by the differential operator approximation.

The performance of the proposed error indicator is demonstrated on a simple two-dimensional Poisson problem with an exponentially strong peak positioned inside the domain to simulate a rapid change in the numerical solution and consequently its accuracy.

Addressed hypotheses.

In the following publication, we demonstrate that a posteriori error indicator can be constructed solely on assessment of local high-order operator approximation, thus, confirming hypothesis **H4**.

Publications included in this section:

- M. Jančič, F. Strniša, and G. Kosec, “Implicit-explicit error indicator based on approximation order,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 01–04. DOI: 10.23919/SpliTech55088.2022.9854342

Regarding my contribution: I made a literature overview of the topic, participated in planning, executed preliminary analyses and jointly prepared the manuscript with co-authors.

Implicit-Explicit Error Indicator based on Approximation Order

Mitja Jančič

*Parallel and Distributed Systems Laboratory
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia
mitja.jancic@ijs.si*

Filip Strniša

*Parallel and Distributed Systems Laboratory
Jožef Stefan Institute
Ljubljana, Slovenia
filip.strnisa@ijs.si*

Gregor Kosec

*Parallel and Distributed Systems Laboratory
Jožef Stefan Institute
Ljubljana, Slovenia
gregor.kosec@ijs.si*

Abstract—With the immense computing power at our disposal, the numerical solution of partial differential equations (PDEs) is becoming a day-to-day task for modern computational scientists. However, the complexity of real-life problems is such that tractable solutions do not exist. This makes it difficult to validate the numerically obtained solution, so good error estimation is crucial in such cases. It allows the user to identify problematic areas in the computational domain that may affect the stability and accuracy of the numerical method. Such areas can then be remedied by either *h*- or *p*-adaptive procedures. In this paper, we propose to estimate the error of the numerical solution by solving the same governing problem implicitly and explicitly, using a different approximation order in each case. We demonstrate the newly proposed error indicator on the solution of a synthetic two-dimensional Poisson problem with tractable solution for easier validation. We show that the proposed error indicator has good potential for locating areas of high error.

Index Terms—implicit; explicit; error indicator; meshless; RBF-FD; Poisson equation

I. INTRODUCTION

In physical modelling, systems of partial differential equations (PDEs) are used to describe the dynamical properties of many natural phenomena. Moreover, the solution of such systems is often of interest to engineers and scientists. However, due to their complexity, they almost never have analytical solutions, and need to be treated numerically, leading to a numerical solution. In general, PDE problems are often solved using one of the following three methods: the finite volume method (FVM), the finite element method (FEM) and the finite difference method (FDM). Recently, however, a generalised formulation of FDM, the radial basis function-generated finite differences (RBF-FD) [1] [2], has become increasingly popular. This is mainly because RBF-FD is a variant of the mesh-free methods [3], i.e. the method can

operate on scattered nodes, unlike the previously mentioned mesh-based methods.

In the context of RBF-FD, linear differential operators are approximated over a set of RBFs augmented with monomials. Augmentation is necessary to ensure convergent and stable behaviour of the method [4] [5]. Additionally, it also enables a direct control over the order of the approximation method, as it corresponds to the highest order used in the approximation basis.

Nevertheless, after the numerical solution is obtained, scientists are often confronted with the difficulty of validating it. For that reason, researchers proposed error indicators [6] [7] to identify problematic areas with a high error of the numerical solution. In practise, different adaptive numerical methods are then applied to these areas [8] ensuring a finer local field description (*h*-adaptivity) or higher polynomial degree approximations (*p*-adaptivity), effectively improving the accuracy of numerical solution.

In this paper, we present an *a posteriori* error indicator that measures the error of an implicit solution. The error indicator is applied through the meshless RBF-FD method as found in the *Medusa library* [9]. In general, the idea is to apply higher order explicit differential operators approximations to the implicitly obtained solution and thus indicate the areas with high error of the numerical solution. In the continuation of this work, the proposed error indicator will be named IMEX (*implicit-explicit*) error indicator.

II. IMEX ERROR INDICATOR

Let there be a partial differential equation of type:

$$\mathcal{L}u = a, \quad (1)$$

where $\mathcal{L}$ is an arbitrary partial differential operator applied to u , and equaling the constant a . Such a problem is first solved implicitly, using a lower-order approximation of $\mathcal{L}$, $\mathcal{L}^{(lo)}$, obtaining the solution $u^{(im)}$ in the process. The $u^{(im)}$ is

The authors would like to acknowledge the financial support of Slovenian Research Agency (ARRS) in the framework of the research core funding No. P2-0095, and the research project J2-3048.

then used to reconstruct a explicitly with the help of higher-order approximation of $\mathcal{L}$, $\mathcal{L}^{(hi)}$, giving $a^{(ex)}$. Finally, $a^{(ex)}$ is then tested against the analytical a to indicate the error. These steps can be summarized as follows:

- 1) compute approximations $\mathcal{L}^{(lo)}$ and $\mathcal{L}^{(hi)}$;
- 2) solve $\mathcal{L}^{(lo)}u = a$ implicitly, obtain $u^{(im)}$;
- 3) compute $a^{(ex)} = \mathcal{L}^{(hi)}u^{(im)}$;
- 4) compare $a^{(ex)}$ and a to indicate high error areas.

III. RBF-FD APPROXIMATION OF DIFFERENTIAL OPERATORS

Since the introduction of meshless methods in the 1970s, many variants have been proposed. The first mention of RBF-FD dates from 2000 with the introduction from Tolstykh [1]. Since then, the method has been thoroughly studied and applied to many real-world problems with recent applications to fluid flow [10] and plasticity [11] problems.

In the framework of RBF-FD, a linear differential operator $\mathcal{L}$ in the node $\mathbf{x}_c$ is approximated over a set of n neighbouring (often called *stencil*) nodes

$$\widehat{\mathcal{L}}u(\mathbf{x}_c) = \sum_{i=1}^n w_i u(\mathbf{x}_i) \quad (2)$$

for an arbitrary function u and weights w yet to be determined. The weights w are obtained by constructing a localised RBF approximation with a given set of radial basis functions (RBFs) θ centred at the stencil nodes of a central node $\mathbf{x}_c$

$$\theta(\mathbf{x}) = \theta(\|\mathbf{x} - \mathbf{x}_c\|). \quad (3)$$

The localized interpolation (2) can be written in a linear system

$$\underbrace{\begin{bmatrix} \theta(\mathbf{x}_1) & \cdots & \theta(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \theta(\mathbf{x}_n) & \cdots & \theta(\mathbf{x}_n) \end{bmatrix}}_{\Theta} \underbrace{\begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}}_w = \underbrace{\begin{bmatrix} (\mathcal{L}\theta_1(\mathbf{x}))|_{\mathbf{x}=\mathbf{x}_c} \\ \vdots \\ (\mathcal{L}\theta_n(\mathbf{x}))|_{\mathbf{x}=\mathbf{x}_c} \end{bmatrix}}_{\ell_\theta}. \quad (4)$$

However, as previously observed by Bayona et al. [4], RBFs alone do not guarantee convergent behaviour or solvability of the system. To mitigate these problems, the approximation basis is extended by a set of $s = \binom{m+d}{d}$ monomials with up to and including degree m in a d -dimensional domain.

With the additional constraints, the RBF-FD approximation can be written compactly as

$$\begin{bmatrix} \Theta & P \\ P^\top & 0 \end{bmatrix} \begin{bmatrix} w \\ \lambda \end{bmatrix} = \begin{bmatrix} \ell_\theta \\ \ell_p \end{bmatrix}, \quad (5)$$

where P is a $n \times s$ matrix of monomials evaluated at stencil points, ℓ_p is the vector of values composed by applying the operator under consideration $\mathcal{L}$ to the polynomials at $\mathbf{x}_c$, i.e. $\ell_p^i = (\mathcal{L}p_i(\mathbf{x}))|_{\mathbf{x}=\mathbf{x}_c}$ and λ are Lagrangian multipliers (which we discard after the solution had been obtained).

IV. EXAMPLE

The IMEX error indicator's performance is demonstrated on a synthetic example, which is commonly used when testing adaptive algorithms in mesh-based methods [12].

The example is the Poisson equation, which is solved in a 2D circular domain Ω with its center at $(0, 0)$, and radius 1:

$$\begin{aligned} \nabla^2 u &= f_{lap}(\mathbf{x}) & \text{in } \Omega, \\ \frac{du}{dn} &= f_{neu}(\mathbf{x}) & \text{on } \partial\Omega, \ x \leq 0, \\ u &= f_{dir}(\mathbf{x}) & \text{on } \partial\Omega, \ x > 0. \end{aligned} \quad (6)$$

The Neumann, and Dirichlet boundary conditions are defined through f_{neu} , and f_{dir} , respectively:

$$f_{neu}(\mathbf{x}) = -2\alpha [\exp(-\alpha\|\mathbf{x} - \mathbf{x}_s\|^2)] \mathbf{x}, \quad (7)$$

$$f_{dir}(\mathbf{x}) = \exp(-\alpha\|\mathbf{x} - \mathbf{x}_s\|^2). \quad (8)$$

From these one can derive the analytical solution of the Laplacian f_{lap} at point $\mathbf{x} = (x, y)$:

$$f_{lap}(\mathbf{x}) = 4(\alpha^2\|\mathbf{x} - \mathbf{x}_s\|^2 - \alpha) \exp(-\alpha\|\mathbf{x} - \mathbf{x}_s\|^2). \quad (9)$$

The source is positioned at $\mathbf{x}_s$ while α controls the source strength. $\mathbf{n}$ is the boundary normal at $\mathbf{x}$ on $\partial\Omega$. $\mathbf{x}_s$ is positioned at $(0.5, 0.5)$, and α is set to 1000.

The example was solved on a laptop with Intel Core i7-8750H CPU, and 16 GB RAM. Results were computed, and written into a file in about 2 s¹.

V. RESULTS AND DISCUSSION

The computational domain is discretized and filled with scattered nodes using Medusa's built-in algorithms [9] [13]. This procedure results in a domain discretized with 24882 points. An example solution is shown in Fig. 1. Support sizes for $\mathcal{L}^{(lo)}$, and $\mathcal{L}^{(hi)}$ are set to $2\binom{m+d}{d}$ (following the recommendations by Bayona et al. [4]), m being the monomial degree, and d the number of dimensions of the domain. The system in Eq. (6) is first solved implicitly, with lower order approximation of differential operators $\nabla^{2(lo)}$, and $\frac{d}{dn}^{(lo)}$, which were obtained with 2nd degree monomials. The solution for the scalar field $u^{(im)}$ is obtained with Eigen's BICGSTAB solver [14]. To compute the RHS explicitly, a higher order approximation of the operator $\nabla^{2(hi)}$, obtained with 4th degree monomials, is applied to $u^{(im)}$. The results are then compared to produce the IMEX error indicator ϵ_{IMEX} :

$$\epsilon_{IMEX} = \left| \nabla^{2(hi)} u^{(im)}(\mathbf{x}) - f_{lap}(\mathbf{x}) \right|. \quad (10)$$

For validation purposes, the error of $u^{(im)}$, ϵ_{an} , is also computed by comparing the implicit to the analytical solution. The latter is obtained with Eq. (8), and ϵ_{an} is:

$$\epsilon_{an} = \left| u^{(im)}(\mathbf{x}) - f_{dir}(\mathbf{x}) \right|. \quad (11)$$

¹The source code for the example can be found at: https://gitlab.com/e62Lab/2022_CP_splitech_IMEX_error_indicator_poisson_eg

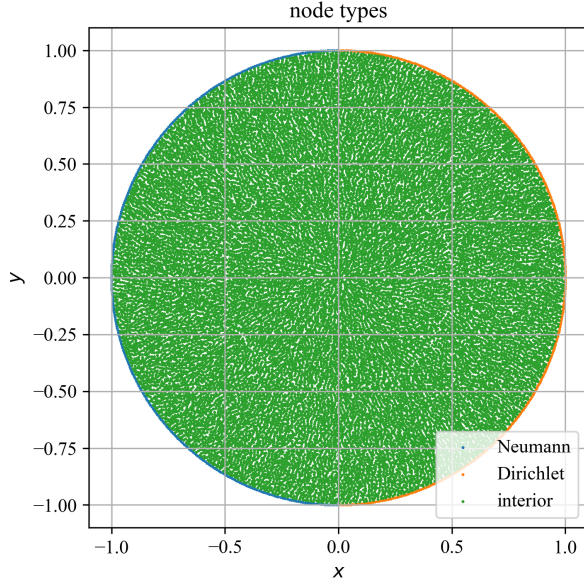


Fig. 1. Domain discretization displaying positions of Neumann, and Dirichlet boundaries, as well as interior nodes.

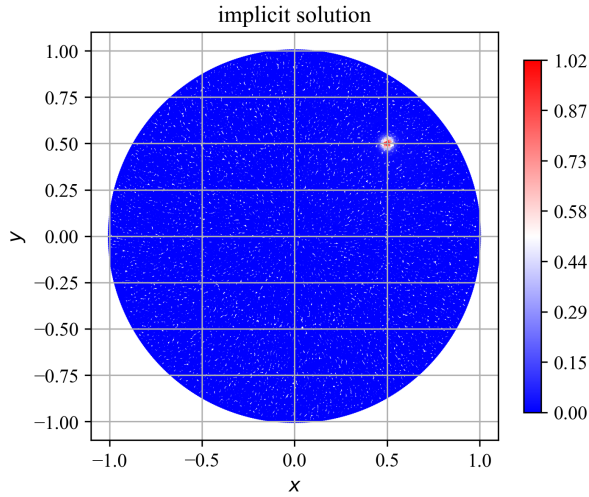


Fig. 2. An example of the implicit solution.

Fig. 2 is displaying the implicit solution $u^{(im)}$ of Eq. (1), while ϵ_{an} and ϵ_{IMEX} are plotted in Fig. 3. For better clarity the implicit solution, ϵ_{an} , and ϵ_{IMEX} are plotted in Fig. 4 along the line $y = x$; $x, y \in \Omega$. As the solution was obtained on scattered nodes, the source for the aforementioned line is obtained by Shepard interpolation (Python, `ShepardIDWInterpolator` from `photutils.utils` [15]), sampling each plot line point from 9 nearest neighbors. Additionally, the same case is solved with 6th degree monomials used to produce $\mathcal{L}^{(hi)}$ for IMEX,

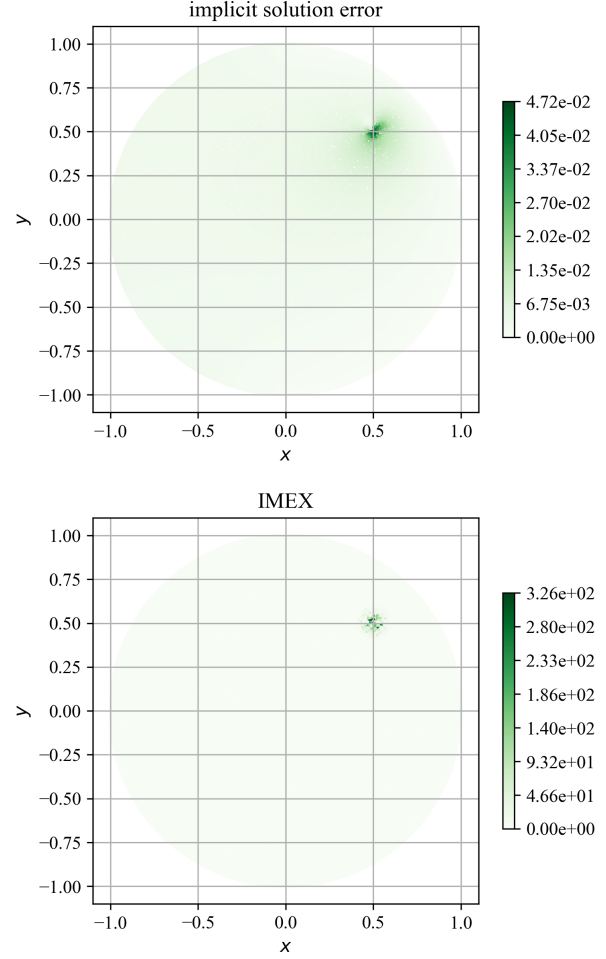


Fig. 3. Comparison plots of ϵ_{an} (above), and ϵ_{IMEX} (below).

with results plotted in Fig. 5.

Comparing Figs. 2, and 3 it is noticeable that the solution's error is the biggest around the source at point $\mathbf{x}_s = (0.5, 0.5)$. The IMEX error indicator also predicts the biggest error to be around the same point, as can be seen in Fig. 3. This is further supported by the graph in Fig. 4. Although the IMEX error indicator does not follow the actual error, it successfully identifies the area of the biggest error. Increasing the monomial degree to compute $\mathcal{L}^{(hi)}$ does not noticeably impact IMEX's performance, as can be seen by comparing Fig. 4, and 5. However, increasing the monomial degree results in a significant compute performance hit in this particular case (total computation time increased to 4 s, compared to previous 2 s).

VI. CONCLUSIONS

A synthetic example of the Poisson equation was solved and the IMEX error indicator was tested on it. The error indicator correctly indicated the area of increased error, which

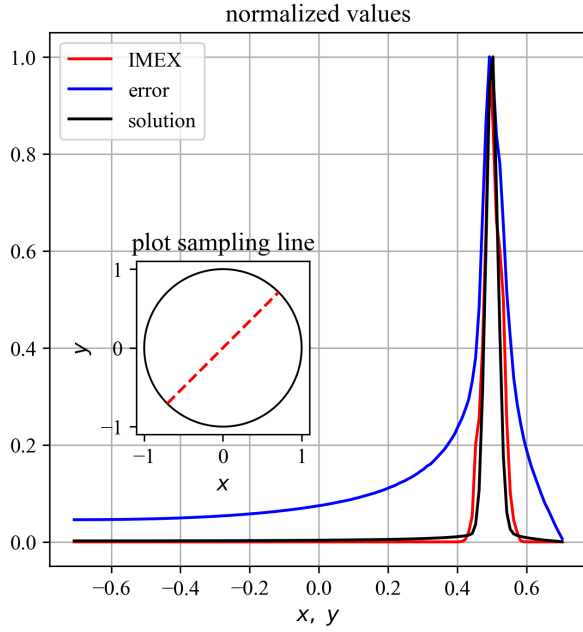


Fig. 4. Implicit solution (solution), ϵ_{an} (error), and ϵ_{IMEX} (IMEX) normalized to their respective maximal values, plotted along the line $y = x$; $x, y \in \Omega$, $\mathcal{L}^{(hi)}$ are computed with 4th degree monomials.

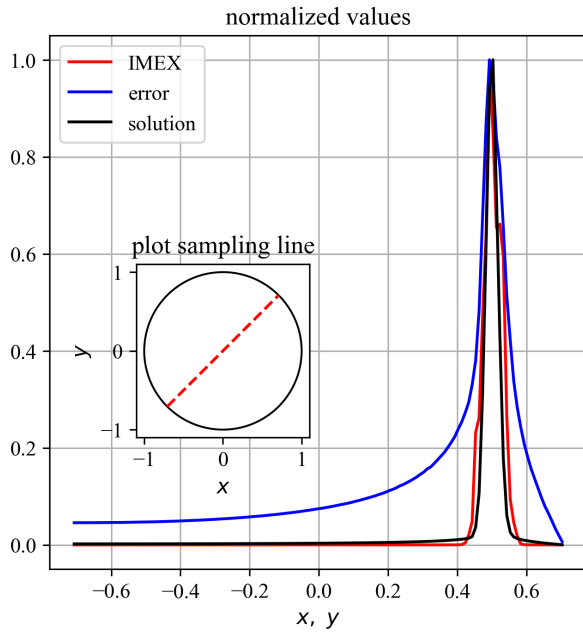


Fig. 5. Implicit solution (solution), ϵ_{an} (error), and ϵ_{IMEX} (IMEX) normalized to their respective maximal values, plotted along the line $y = x$; $x, y \in \Omega$, $\mathcal{L}^{(hi)}$ are computed with 6th degree monomials.

also coincided with the source in the Poisson equation. Results were produced by increasing the monomial degree of the explicit approximations by 2 compared to the implicit counterparts. Further increasing the monomial degree did not prove beneficial in this specific example.

We show that the proposed error indicator successfully identifies the areas with high error of the numerical solution. In the continuation, these findings could be used to adaptively refine the critical areas and improve the precision of the numerical solution.

REFERENCES

- [1] A. I. Tolstykh, "On using rbf-based differencing formulas for unstructured and mixed structured-unstructured grid calculations," in *Proceedings of the 16th IMACS world congress*, vol. 228. Lausanne, 2000, pp. 4606–4624.
- [2] A. Tolstykh and D. Shirobokov, "On using radial basis functions in a "finite difference mode" with applications to elasticity problems," *Computational Mechanics*, vol. 33, no. 1, pp. 68–79, 2003.
- [3] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl, "Meshless methods: an overview and recent developments," *Computer methods in applied mechanics and engineering*, vol. 139, no. 1-4, pp. 3–47, 1996.
- [4] V. Bayona, N. Flyer, B. Fornberg, and G. A. Barnett, "On the role of polynomials in rbf-fd approximations: II. numerical solution of elliptic pdes," *Journal of Computational Physics*, vol. 332, pp. 257–273, 2017.
- [5] N. Flyer, B. Fornberg, V. Bayona, and G. A. Barnett, "On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy," *Journal of Computational Physics*, vol. 321, pp. 21–38, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0021999116301632>
- [6] J. Slak, "Partition-of-unity based error indicator for local collocation meshless methods," in *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2021, pp. 254–258.
- [7] C. Carstensen, R. Lazarov, and S. Tomov, "Explicit and averaging a posteriori error estimates for adaptive finite volume methods," *SIAM Journal on Numerical Analysis*, vol. 42, no. 6, pp. 2496–2521, 2005.
- [8] K. Segeth, "A review of some a posteriori error estimates for adaptive finite element methods," *Mathematics and Computers in Simulation*, vol. 80, no. 8, pp. 1589–1600, 2010, eSCO 2008 Conference. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378475408004230>
- [9] J. Slak and G. Kosec, "Medusa: A C++ Library for solving PDEs using Strong Form Mesh-Free methods," *ACM Transactions on Mathematical Software*, 2021.
- [10] M. Rot and G. Kosec, "Refined rbf-fd analysis of non-newtonian natural convection," 2022. [Online]. Available: <https://arxiv.org/abs/2202.08095>
- [11] F. Strniša, M. Jančič, and G. Kosec, "A meshless solution of a small-strain plasticity problem," 2022. [Online]. Available: <https://arxiv.org/abs/2203.08462>
- [12] W. F. Mitchell, "A collection of 2d elliptic problems for testing adaptive grid refinement algorithms," *Appl. Math. Comput.*, vol. 220, p. 350–364, sep 2013. [Online]. Available: <https://doi.org/10.1016/j.amc.2013.05.068>
- [13] J. Slak and G. Kosec, "On generation of node distributions for meshless pde discretizations," *SIAM Journal on Scientific Computing*, vol. 41, no. 5, pp. A3202–A3229, 2019.
- [14] G. Guennebaud, B. Jacob *et al.*, "Eigen v3," <http://eigen.tuxfamily.org>, 2010.
- [15] L. Bradley, B. Sipińcz, T. Robitaille, E. Tollerud, Z. Vinicius, C. Deil, K. Barbary, T. J. Wilson, I. Busko, H. M. Günther, M. Cara, S. Conseil, A. Bostroem, M. Droettboom, E. M. Bray, L. A. Bratholm, P. L. Lim, G. Barentsen, M. Craig, S. Pascual, G. Perren, J. Greco, A. Donath, M. de Val-Borro, W. Kerzendorf, Y. P. Bach, B. A. Weaver, F. D'Eugenio, H. Souchereau, and L. Ferreira, "astropy/photutils: 1.0.0," Sep. 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.4044744>

On the Theoretical Background of IMEX Error Indicator

The presented paper introducing the IMEX error indicator lacks theoretical background, which is, to a certain extent, mitigated in the following discussion based on the recent work on the adaptivity of local kernel based methods [116].

The derivation of the error estimation, assumes the local interpolant s_m of function f and given order m is constructed over n stencil nodes as

$$s_m[f](\mathbf{x}) = \sum_{j=1}^n \lambda_j[f] \phi_j(\mathbf{x}) + \sum_{l=1}^{M_m} \gamma_l[f] \pi_l(\mathbf{x}), \quad (3.1)$$

using a linear combination of (conditionally-) positive definite kernel functions $\phi_j(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_j\|)$ for $j = 1, 2, \dots, n$, and $M_m = \binom{m+d}{m}$ multivariate polynomial terms π up to and including degree m in a d -dimensional domain. Coefficient vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\gamma}$ are chosen such that the interpolation conditions $s_m[f](\mathbf{x}_j) = f(\mathbf{x}_j)$ are satisfied¹ for all j .

Based on the Taylor expansion of $f(\mathbf{x})$ about the point $\mathbf{x}_0$, with f having continuous mixed partial derivatives up to order $m + \mu + 1$ — where μ is a positive integer parameter the meaning of which will be clear later — Reeger [116] provides a mathematical proof that the point-wise error of interpolant $s_m[f]$ can be expressed as

$$\begin{aligned} s_m[f](\mathbf{x}) - f(\mathbf{x}) &= \\ &= \sum_{l=1}^{M_{m+\mu}-M_m} \frac{1}{\boldsymbol{\alpha}_{M_{m+l}}!} \partial^{\boldsymbol{\alpha}_{M_{m+l}}} f(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_0} E_{m,M_{m+l}}(\mathbf{x}) + \\ &+ \left(s_m[R_{m+\mu}[f]](\mathbf{x}) - R_{m+\mu}[f](\mathbf{x}) \right) \end{aligned} \quad (3.2)$$

with $E_{m,M_{m+l}}(\mathbf{x}) = s_m[\pi_{m,M_{m+l}}](\mathbf{x}) - \pi_{m,M_{m+l}}(\mathbf{x})$ defined as the error in approximating the polynomial term $\pi_{m,M_{m+l}}$ and remainder terms R behaving as $\mathcal{O}(h^{m+\mu+1})$ as $h \rightarrow 0$ for typical spacing h within between the stencil nodes used for the construction of the interpolant $s_m[f]$. Here, the multi-index notation $\boldsymbol{\alpha}_l$ has been used (see [116] and [117] for clarity concerning the multi-index notation).

In adaptive solution procedures, the goal is to reduce the error of

$$\|\mathcal{L}s_m[f] - \mathcal{L}f\| = \|\mathcal{L}(s_m[f] - f)\|, \quad (3.3)$$

where

$$\|\mathcal{L}s_m[f] - \mathcal{L}s_{m+\mu}[f]\| = \|\mathcal{L}(s_m[f] - s_{m+\mu}[f])\| \quad (3.4)$$

is used as the estimate of the error in the approximation of $\mathcal{L}f$ and $\mu \geq 1$. In the process, an assumption that approximation order of $m + \mu$ provides a better description of the function than an approximation order of m is made. This estimate is understood as an approximation to the dominant term from expression (3.2).

Following [118], [119], we know that the approximation error of the interpolant s_m is upper-bounded

$$\left| \partial^{\boldsymbol{\alpha}_l} f(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_0} - \partial^{\boldsymbol{\alpha}_l} s_m[f](\mathbf{x})|_{\mathbf{x}=\mathbf{x}_{k,0}} \right| \leq \mathcal{O}(h^{m+\mu+1-|\boldsymbol{\alpha}_l|}) \quad \text{as } h \rightarrow 0, \quad (3.5)$$

¹This requirement alone would yield an underdetermined system of linear equations that needs to be solved to obtain the coefficient vectors. For a well defined problem, additional constraints, i.e., $\sum_{j=1}^n \lambda_j[f] \pi_l(\mathbf{x}_j) = 0$, are added.

which allowed Reeger to conclude that the terms $s_m[\pi_{m,l}(\mathbf{x})] - \pi_{m,l}(\mathbf{x})$ appearing in the approximation error $E(\mathbf{x})$ are then also upper-bounded by $\mathcal{O}(h^{|\alpha_l|})$. Therefore, the difference between the error estimate and the dominant term in the error of kernel-based interpolant with polynomial terms up to degree m is at most $\mathcal{O}(h^{m+\mu+1})$ as $h \rightarrow 0$. For $\mu > 0$ this is at least one order higher than the size of the dominant term appearing in the interpolant. The estimation thus provides a reasonable approximation of the interpolation error.

Naturally, the derivation of such pointwise error estimation only applies to smooth enough functions, which is a clear limitation of the IMEX error indicator.

3.3 Strong-Form Mesh-Free hp -Adaptive Solution of Linear Elasticity Problem

With all the building blocks required for the development of a strong-form meshless hp -adaptive solution procedure, in this section, we finally provide the algorithm and its implementation. To the best of our knowledge, our attempt is the first of its kind – strong-form meshless hp -adaptive solution procedure simultaneously controlling local spatial discretization resolution and local approximation order.

The following publication also discusses the performance of the IMEX error indicator on a set of linear elasticity problems.

Contributions.

This section represents the following contributions:

- C6:** We provide additional verification and performance analyses of the IMEX error indicator, when applied to two- and three-dimensional linear elasticity problems, demonstrating its versatile use.
- C7:** By following the well-established *solve-estimate-mark-refine* paradigm, we provide a detailed description of all building blocks required to employ a hp -adaptive meshless solution procedure based on the RBF-FD approximation.

Addressed hypotheses.

The following publication more or less encompasses all previous findings and, thus, further confirms hypotheses **H1** and **H3–H5**. Additionally, we demonstrate that an hp -adaptive solution procedure can be used to effectively solve elliptic PDEs, thus confirming hypothesis **H6**.

Publications included in this section:

- M. Jančič and G. Kosec, “Strong form mesh-free hp -adaptive solution of linear elasticity problem,” *Engineering with Computers*, May 2023, ISSN: 1435-5663. DOI: 10.1007/s00366-023-01843-6. [Online]. Available: <https://doi.org/10.1007/s00366-023-01843-6>

Regarding my contribution: I made a literature overview of the topic, participated in planning and development of the algorithm, implemented the solution procedure, executed the analyses and jointly prepared the manuscript with co-authors.



Strong form mesh-free *hp*-adaptive solution of linear elasticity problem

Mitja Jančič^{1,2} · Gregor Kosec¹Received: 25 October 2022 / Accepted: 4 May 2023
© The Author(s) 2023

Abstract

We present an algorithm for *hp*-adaptive collocation-based mesh-free numerical analysis of partial differential equations. Our solution procedure follows a well-established iterative solve–estimate–mark–refine paradigm. The solve phase relies on the Radial Basis Function-generated Finite Differences (RBF-FD) using point clouds generated by advancing front node positioning algorithm that supports variable node density. In the estimate phase, we introduce an Implicit-Explicit (IMEX) error indicator, which assumes that the error relates to the difference between the implicitly obtained solution (from the solve phase) and a local explicit re-evaluation of the PDE at hand using a higher order approximation. Based on the IMEX error indicator, the modified Texas Three Step marking strategy is used to mark the computational nodes for *h*-, *p*- or *hp*-(de-)refinement. Finally, in the refine phase, nodes are repositioned and the order of the method is locally redefined using the variable order of the augmenting monomials according to the instructions from the mark phase. The performance of the introduced *hp*-adaptive method is first investigated on a two-dimensional Peak problem and further applied to two- and three-dimensional contact problems. We show that the proposed IMEX error indicator adequately captures the global behaviour of the error in all cases considered and that the proposed *hp*-adaptive solution procedure significantly outperforms the non-adaptive approach. The proposed *hp*-adaptive method stands for another important step towards a fully autonomous numerical method capable of solving complex problems in realistic geometries without the need for user intervention.

Keywords RBF-FD · *hp*-adaptivity · Mesh-free · Linear elasticity · Error indicator

1 Introduction

Many natural and technological phenomena are modelled through Partial Differential Equations (PDEs), which can rarely be solved analytically—either because of geometric complexity or because of the complexity of the model at hand. Instead, realistic simulations are performed numerically. There are well-developed numerical methods that can be implemented in a more or less effective numerical solution procedure and executed on modern computers to perform virtual experiments or simulate the evolution of

various natural or technological phenomena. Nonetheless, despite the immense computing power at our disposal, which allows us to solve ever more complex problems numerically, the development of efficient numerical approaches is still crucial. Relying solely on brute force computing often leads to unnecessarily long computations—not to mention wasted energy.

Most numerical solutions are obtained using mesh-based methods such as the Finite Volume Method (FVM), the Finite Difference Method (FDM), the Boundary Element Method (BEM) or the Finite Element Method (FEM). Modern numerical analysis is dominated by FEM [1] as it offers a mature and versatile solution approach that includes all types of adaptive solution procedures [2] and well understood error indicators [3]. Despite the widespread acceptance of FEM, the meshing of realistic 3D domains, a crucial part of FEM analysis where nodes are structured into polyhedrons covering the entire domain of interest, is still a problem that often requires user assistance or development of domain-specific algorithms [4].

✉ Mitja Jančič
mitja.jancic@ijs.siGregor Kosec
gregor.kosec@ijs.si¹ Parallel and Distributed Systems Laboratory, Institute Jožef Stefan, Jamova Cesta 39, Ljubljana 1000, Slovenia² Jožef Stefan International Postgraduate School, Jamova Cesta 39, Ljubljana 1000, Slovenia

In response to the tedious meshing of realistic 3D domains, required by FEM, and the geometric limitations of FDM and FVM, a new class of mesh-free methods [5] emerged in the 1970s. Mesh-free methods do not require a topological relationship between computational nodes and can therefore operate on scattered nodes, which greatly simplifies the discretisation of the domain [6], regardless of its dimensionality or shape [7, 8]. Just recently, they have also been promoted to Computer Aided Design (CAD) geometry aware numerical analysis [9]. Moreover, the formulation of mesh-free methods is extremely convenient for implementing *h*-refinement [10], considering different approximations of partial differential operators in terms of the shape and size of the *stencil* [11, 12] and the local approximation order [13]. However, they tend to be more computationally intensive as they require larger stencils for stable computations [13, 14] and have limited preprocessing capabilities [15]. This may make them less attractive from a computational point of view, but the ability to work with scattered nodes and easily control the approximation order makes them good candidates for many applications in science and industry [16, 17].

Adaptive solution procedures are essential in problems where the accuracy of the numerical solution varies spatially and are currently subject of intensive studies. Two conceptually different adaptive approaches have been proposed, namely *p*-adaptivity or *h*-, *r*-adaptivity. In *p*-adaptivity, the accuracy of the numerical solution is varied by changing the order of approximation, while in *h*- and *r*-adaptivity, the resolution of the spatial discretisation is adjusted for the same purpose. In the *h*-adaptive approach, nodes are added or removed from the domain as needed, while in the *r*-adaptive approach the total number of nodes remains constant – the nodes are only repositioned with respect to the desired accuracy. Ultimately, *h*- and *p*-adaptivities can be combined to form the so-called *hp*-adaptivity [18–20], where the accuracy of the solution is controlled with the order of the method and the resolution of the spatial discretisation.

Since the regions where higher accuracy is required are often not known *a priori*, and to eliminate the need for human intervention in the solution procedure, a measure of the quality of the numerical solution, commonly called a posterior error indicator, is a necessary additional step in an adaptive solution procedure [4]. The most famous error indicator, commonly referred to as the ZZ-type error indicator, was introduced in 1987 by Zienkiewicz and Zhu [21] in the context of FEM and it is still an active research topic [22]. The ZZ-type error indicator assumes that the error of the numerical solution is related to the difference between the numerical solution and a locally recovered solution. The ZZ-type error indicator has also been employed in the context of mesh-free solutions of elasticity problems using the mesh-free Finite Volume Method [23] in both weak and strong form using the Finite Point Method [24]. Furthermore, it also served as an inspiration in the context of Radial Basis

Function-Generated Finite Difference (RBF-FD) solution to Laplace equation [25]. Moreover, a residual-based class of error indicators [26] has been demonstrated in the elasticity problems using a Discrete Least Squares mesh-free method [27]. Nevertheless, the most intuitive error indicators are based on the physical interpretation of the solution, usually evaluating the first derivative of the field under consideration [11] or calculating the variance of the field values within the support domain [10].

The advent of *hp*-adaptive numerical analysis began with FEM in the 1980s [28]. In *hp*-FEM, for example, one has the option of splitting an element into a set of smaller elements or increasing its approximation order. This decision is often considered to be the main difficulty in implementing the *hp*-adaptive solution procedure and was already studied by Babuška [28] in 1986. Since then, various decision-making strategies, commonly referred to as marking strategies, have been proposed [2, 29]. The early works use a simple Texas Three Step algorithm, originally proposed in the context of BEM [30], where the refinement is based on the maximum value of the error indicator. The first true *hp*-strategy was presented by Ainsworth [31] in 1997, since then many others have been proposed [2, 29]. In general, *p*- in FEM is more efficient when the solution is smooth. Based on this observation, most authors nowadays use the local Sobolev regularity estimate to choose between the *h*- and the *p*-refinement [32–34] for a given finite element. Moreover, in [35] local boundary values are solved, while the authors of [36, 37] use minimisation of the global interpolation error methods.

For mesh-free methods, *h*-adaptivity comes naturally with the ability to work with scattered nodes, and as such has been thoroughly studied in the context of several mesh-free methods [38–40]. Only recently, the popular Radial Basis Function-generated Finite Differences (RBF-FD) [41] have been used in the *h*-adaptive solution of elliptic problems [25, 42] and linear elasticity problems [10, 43]. Researchers have also reported the combination of *h*- and *r*-adaptivity, which form a so-called *hr*-adaptive solution procedure [44]. The *p*-adaptive method, on the other hand, is still quite unexplored in the mesh-free community. However, the authors of [45] approach the *p*-adaptive RBF-FD method in solving Poisson's equation with the idea of varying the order of the augmenting monomials to maintain the global order of convergence over the domain regardless of the potential variations in the spatial discretisation distance. It should also be noted that some authors reported *p*-adaptive methods by locally increasing the number of shape functions, changing the interpolation basis functions, or simply increasing the stencil size [46–48]. These approaches are all to some extent *p*-adaptive, but not in their true essence. The authors of [49] have introduced a *p*-refinement with spatially variable local approximation order and come closest to a true *p*-adaptive solution procedure on scattered nodes. However, this work lacks an automated marking and refinement strategy for the local approximation order,

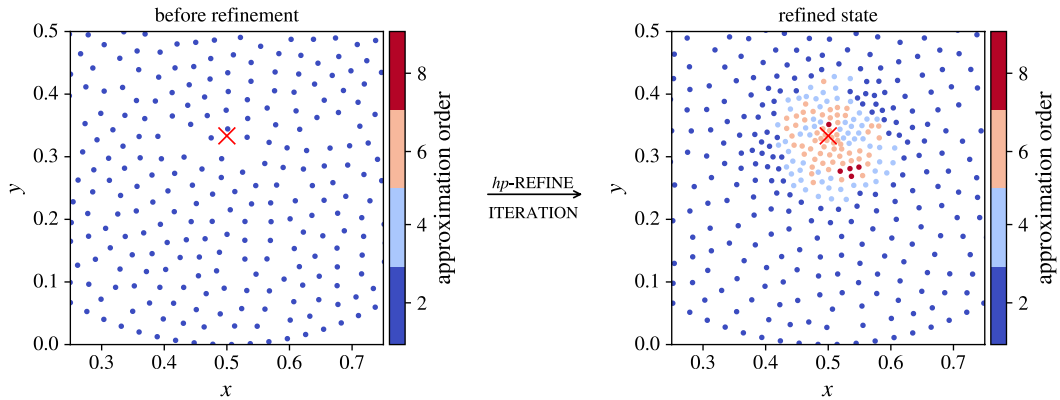


Fig. 1 A sketch of a single *hp*-refinement iteration for a two-dimensional problem. Note that the exponentially strong source (marked with red cross) is set at $p = \left(\frac{1}{2}, \frac{1}{3}\right)$. The refined state has been

obtained by employing *h*- and *p*-refinement strategies, thus the number of nodes and the local approximation orders in the neighbourhood of the strong source have been modified. Closed form solution has been used to indicate the error in the estimate module

e.g. based on an error indicator. The automated marking and refinement strategies were used with the weak form *h*-*p* adaptive clouds [50], where the authors use grid-like *h*-enrichment to improve the local field description.

In this paper, we present our attempt to implement the *hp*-adaptive strong form mesh-free solution procedure using the mesh-free RBF-FD approximation on scattered nodes. Our solution procedure follows a well-established paradigm based on an iterative loop. To estimate the accuracy of the numerical solution, we employ original IMEX error indicator. The marking strategy used in this work is based on the Texas Three Step algorithm [34], where the basic idea is to estimate the smoothness or analyticity of the numerical solution. Our refinement strategy is based on the recommendations of [10], where the authors were able to obtain satisfactory results using a purely *h*-adaptive solution procedure for elasticity problems. Although the chosen refinement and marking strategies are not optimal [36], the obtained results clearly outperform the non-adaptive approach.

2 *hp*-adaptive solution procedure

In the present work, we focus on the implementation of mesh-free *hp*-adaptive refinement, which combines the advantages of *h*- and *p*-refinement procedures. The proposed *hp*-adaptive solution procedure follows the well-established paradigm based on an iterative loop, where each iteration step consists of four modules:

1. **Solve** – A numerical solution $\hat{u}$ is obtained.
2. **Estimate** – An error indication of the obtained numerical solution.

3. **Mark** – Marking of nodes for refinement/de-refinement.
4. **Refine** – Refinement/de-refinement of the spatial discretisation and local approximation order of the numerical method.

The workings of each module are further explained in the following subsections, while a full *hp*-adaptive solution procedure algorithm is given in Algorithm 1. For clarity, Fig. 1 also graphically sketches the ultimate goal of a single refinement iteration.

Algorithm 1 *hp*-adaptive solution procedure

Input: The problem, computational domain Ω , initial nodal density function $h : \Omega \rightarrow \mathbb{R}$, initial approximation order distribution $m : \Omega \rightarrow \mathbb{N}$, the maximal number of iterations $I_{\max}$ and adaptivity parameters $\alpha_{h,p}, \beta_{h,p}, \lambda_{h,p}, \vartheta_{h,p}$.
Output: The *hp*-refined numerical solution of the problem.

```

1: function ADAPTIVE_SOLVE(problem,  $\Omega$ ,  $h$ ,  $m$ ,  $I_{\max}$ ,  $\alpha_{h,p}, \beta_{h,p}, \lambda_{h,p}, \vartheta_{h,p}$ )
2:   for  $i \leftarrow 0$  to  $I_{\max}$  do
3:      $\Omega^* \leftarrow \text{DISCRETISE}(\Omega, h)$   $\triangleright$  Discretises domain using nodal density function  $h$ .
4:     solution  $\leftarrow \text{SOLVE}(\text{problem}, \Omega^*, m)$   $\triangleright$  Obtains a numerical solution to the problem.
5:     indicator  $\leftarrow \text{IMEX}(\text{problem}, \text{solution}, \Omega^*, m)$   $\triangleright$  Error indicator computation.
6:     if STOPPING_CRITERIA then
7:       return solution
8:     end if
9:      $h, m \leftarrow \text{ADAPT}(\text{indicator}, h, m, \Omega^*, \alpha_{h,p}, \beta_{h,p}, \lambda_{h,p}, \vartheta_{h,p})$   $\triangleright$  Refine the nodes and approximation orders.
10:  end for
11:  return solution
12: end function

```

2.1 The SOLVE module

First, a numerical solution $\hat{u}$ to the governing problem must be obtained. In general, the numerical treatment of a system of PDEs is done in several steps. First, the domain is

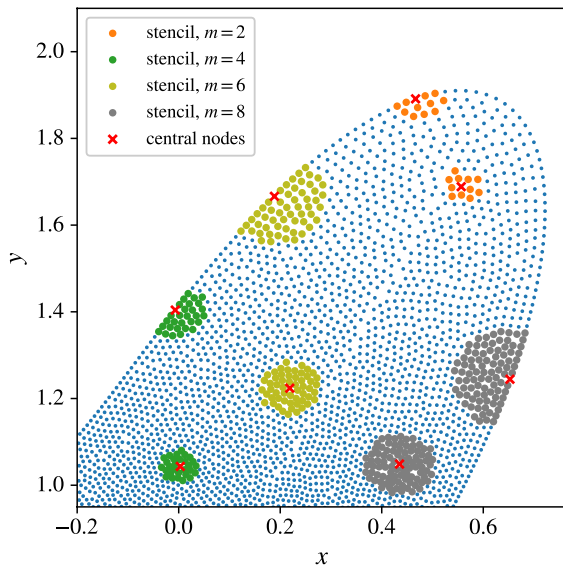


Fig. 2 An example of domain discretisation with scattered nodes and variable node density. Example stencils are also shown for different approximation orders m on the domain boundary and its interior

discretised by positioning the nodes, then the linear differential operators in each computational node are approximated, and finally the system of PDEs is discretised and assembled into a sparse linear system. To obtain a numerical solution $\hat{u}$, the sparse system is solved.

2.1.1 Domain discretisation

While traditional mesh-based methods discretise the domain by building a mesh, mesh-free methods simplify this step to the positioning of nodes, as no information about internal connectivity is required. With the mathematical formulation of the mesh-free methods being dimension-independent, we accordingly choose a dimension-independent algorithm for node generation based on Poisson disc sampling [51]. Conveniently, the algorithm also supports spatially variable nodal densities required by the h -adaptive refinement methods. An example of a variable node density discretisation can be found in Fig. 2.

Interested readers are further referred to the original paper [51] for more details on the node generation algorithm, its stand-alone C++ implementation in the *Medusa library* [52], and follow-up research focusing on its parallel implementation [53] and parametric surface discretisations [54].

2.1.2 Approximation of linear differential operators

Having discretised the domain, we proceed to the approximation of linear differential operators. In this step, a linear

differential operator $\mathcal{L}$ is approximated over a set of neighbouring nodes, commonly referred to as *stencil nodes*.

To derive the approximation, we assume a central point $\mathbf{x}_c \in \Omega$ and its stencil nodes $\{\mathbf{x}_i\}_{i=1}^n = \mathcal{N}$ for stencil size n . A linear differential operator in $\mathbf{x}_c$ is then approximated over its stencil with the following expression

$$(\mathcal{L}u)(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i), \quad (1)$$

for an arbitrary function u and yet to be determined weights w which are computed by enforcing the equality of approximation (1) for a chosen set of basis functions.

In this work, we use Radial Basis Functions (RBFs) augmented with monomials. To eliminate the dependency on a shape parameter, we choose Polyharmonic Splines (PHS) [14] defined as

$$f(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \quad (2)$$

for Euclidean distance r . The chosen approximation basis effectively results in what is commonly called the RBF-FD approximation method [41].

Furthermore, it is necessary that the stencil nodes form a so-called polynomial unisolvent set [55]. In this work, we follow the recommendations of Bayona [14] and define the stencil size as twice the number of augmenting monomials, i.e.

$$n = 2 \binom{m+d}{m} \quad (3)$$

for monomial order m and domain dimensionality d . This, in practice, results in large enough stencil sizes to satisfy the requirement, so that no special treatment was needed to assure unisolvency. While special stencil selection strategies showed promising results [11, 56], a common choice for selecting a set of stencil nodes $\mathcal{N}$ is to simply select the nearest n nodes. The latter approach was also used in this work. Figure 2 shows example stencils for different approximation orders m on domain boundary and its interior.

It is important to note that the augmenting monomials allow us to directly control the order of the local approximation method. The approximation order corresponds to the highest augmenting monomial order m in the approximation basis. However, the greater the approximation order the greater the computational complexity due to larger stencil sizes [13]. Nevertheless, the ability to control the local order of the approximation method sets the foundation for the p -adaptive refinement.

To conclude the solve module, the PDEs of the governing problem are discretised and assembled into a global

sparse system. The solution of the assembled system stands for the numerical solution $\hat{u}$.

2.2 The ESTIMATE module (Implicit-Explicit error indicator)

In the estimation step, critical areas with high error of the numerical solution are identified. Identifying such areas is not a trivial task. In rare cases where a closed form solution to the governing problem exists, we can directly determine the accuracy of the numerical solution. Therefore, other objective metrics, commonly referred to as *error indicators*, are needed to indicate areas with high error of the numerical solution.

2.2.1 IMPLICIT-EXPLICIT (IMEX) error indicator

In this work we will use an error indicator based on the implicit-explicit [57] evaluation of the considered field. IMEX makes use of the implicitly obtained numerical solution and explicit operators (approximated by a higher order basis) to reconstruct the right-hand side of the governing problem. To explain the basic idea of IMEX, let us define a PDE of type

$$\mathcal{L}u = f_{RHS}, \quad (4)$$

where $\mathcal{L}$ is a differential operator applied to the scalar field u and f_{RHS} is a scalar function. To obtain an error indicator field η , the problem (4) is first solved implicitly using a lower order approximation $\mathcal{L}^{im}$ of operators $\mathcal{L}$, obtaining the solution u^{im} in the process. The explicit high order operators $\mathcal{L}^{ex}$ are then used over the implicitly computed field u^{im} to reconstruct the right-hand side of the problem (4) obtaining f_{RHS}^{ex} in the process. The error indication is then calculated as $\eta = |f_{RHS} - f_{RHS}^{ex}|$. The calculation steps of the IMEX error indicator are also shown in Algorithm 2.

Algorithm 2 IMEX error indicator

Input: The problem, domain Ω , differential operators $\mathcal{L}$, low-order approximation basis ξ , high order approximation basis ζ .

Output: Error indicator field η .

```

1: function INDICATE_ERROR(problem,  $\Omega$ ,  $\mathcal{L}$ ,  $\xi$ ,  $\zeta$ )
2:    $\mathcal{L}^{im} \leftarrow \text{APPROXIMATE}(\Omega, \xi)$       ▷ Obtain low-order approximation of
   differential operators  $\mathcal{L}$ .
3:    $u^{im}, f_{RHS} \leftarrow \text{SOLVE}(\text{problem}, \Omega, \mathcal{L}^{im})$   ▷ Obtain a numerical solution
   to the problem.
4:    $\mathcal{L}^{ex} \leftarrow \text{APPROXIMATE}(\Omega, \zeta)$       ▷ Obtain high order approximation of
   differential operators  $\mathcal{L}$ .
5:    $f_{RHS}^{ex} \leftarrow \text{EVALUATE}(\text{problem}, \Omega, \mathcal{L}^{ex}, u^{im})$   ▷ Explicit re-evaluation.
6:    $\eta \leftarrow \text{COMPUTE}(f_{RHS}, f_{RHS}^{ex})$       ▷ Obtain error indicator field.
7:   return  $\eta$ 
8: end function

```

The assumption that the deviation of the explicit high order evaluation $\mathcal{L}^{ex}u^{im}$ from the exact f_{RHS} corresponds

to the error of the solution u^{im} is similar to the reasoning behind the ZZ-type indicators, where the deviation of the recovered high order solution from the computed solution characterises the error. As long as the error in u^{im} is high, the explicit re-evaluation will not correctly solve the Equation (4). However, as the error in u^{im} decreases, the difference between f_{RHS} and f_{RHS}^{ex} will also decrease, assuming that the error is dominated by the inaccuracy of u^{im} and not by the differential operator approximation.

It is worth noting that the definition of IMEX is general in the sense that computing the error indication η does not distinguish between the interior and boundary nodes. In the boundary nodes, the error indicator η is calculated in the same way as in the interior nodes. In the case of Dirichlet boundary conditions, the error indicator is trivial because the solution fields are exactly imposed, i.e. the error indicator results in $\eta = 0$. However, in case of boundary conditions involving the evaluation of derivatives (Robin and Neumann), $\eta \neq 0$.

2.3 The MARK module

After the error indicator η has been obtained for each computational point in domain Ω , a marking strategy is applied. The main goal of this module is to mark the nodes with too high or too low values of the error indicator to achieve a uniformly distributed accuracy of the numerical solution and to reduce the computational cost of the solution procedure – by avoiding fine local field descriptions and high order approximations where this is not required. Moreover, the marking strategy not only decides whether or not (de-)refinement should take place at a particular computational node, but also defines the type of refinement procedure if there are several to choose from. In this work, we use a modified Texas Three Step marking strategy [30, 58], originally restricted to refinement (no de-refinement) with the *h*- and *p*-refinement types. This chosen strategy was also considered in one of the recent papers by Eibner [34], who showed that, although extremely simple to understand and implement, it can provide results good enough to demonstrate the advantages of mesh-based *hp*-adaptive solution procedures.

In each iteration of the adaptive procedure, the marking strategy starts by checking the error indicator values η_i for all computational nodes in the domain. Unlike the originally proposed marking strategy [34] that used only refinement, we additionally introduce de-refinement. Therefore, if η_i is greater than $\alpha\eta_{max}$ for the maximum indicator value η_{max} and a free model parameter $\alpha \in (0, 1)$, the node is marked for refinement. If η_i is less than $\beta\eta_{max}$ for a free model parameter $\beta \in (0, 1) \wedge \beta \leq \alpha$, the node is marked for de-refinement.

Fig. 3 A visual representation of h - and p -(de)refinement marking strategy

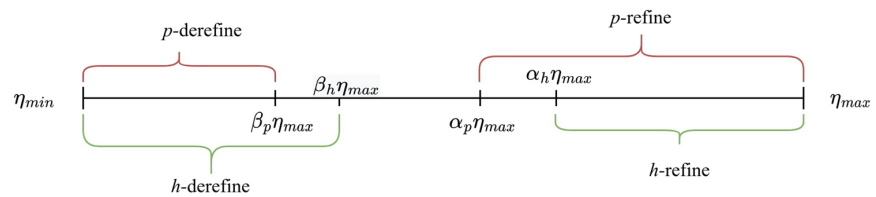
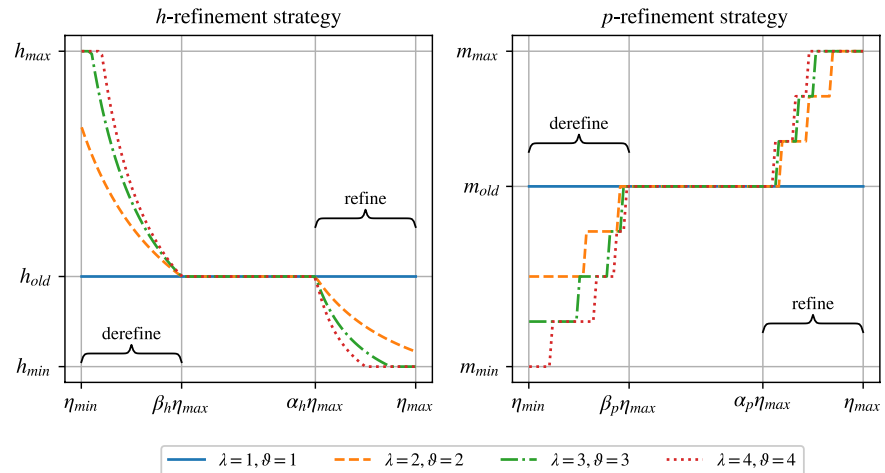


Fig. 4 A visual representation of the (de-)refinement strategies for different values of refinement aggressiveness λ and de-refinement aggressiveness ϑ . Notice that both refinement types also have lower and upper limits



Otherwise, the node remains unmarked, which means that no (de-)refinement should take place. The marking strategy can be summarised with a single equation

$$\begin{cases} \eta_i > \alpha \eta_{\max}, & \text{refine} \\ \beta \eta_{\max} \leq \eta_i \leq \alpha \eta_{\max}, & \text{do nothing} \\ \eta_i < \beta \eta_{\max}, & \text{de-refine} \end{cases} \quad (5)$$

In the context of mesh-based methods, it has already been observed, that such marking strategy, although easy to implement, is far from optimal [2, 34]. Additionally, it has also been demonstrated that in case of smooth solutions p -refinement is preferred while h -refinement is preferred in volatile fields, e.g. in vicinity of a singularity in the solution [2, 36], which cannot be achieved with the chosen marking strategy. Additional discussion on this issue can be found in Sect. 4, where problems with singularity in the solution are discussed, and in Sect. 2.6.3 where we discuss some guidelines for possible work on improved marking strategies.

Since our work is focused on the implementation of hp -adaptive solution procedure rather than discussing the optimal marking strategy, we decided to secure full control over the marking strategy by treating h - and p -methods separately – but at the cost of higher number of free parameters. Therefore, the marking strategy is modified by introducing parameters $\{\alpha_h, \beta_h\}$ and $\{\alpha_p, \beta_p\}$ for separate treatment of h - and

p -refinements, respectively (see Fig. 3 for clarification). Note that the proposed modified marking strategy can mark a particular node for h -, p - or hp -(de-)refinement if required, otherwise the computational node is left unchanged.

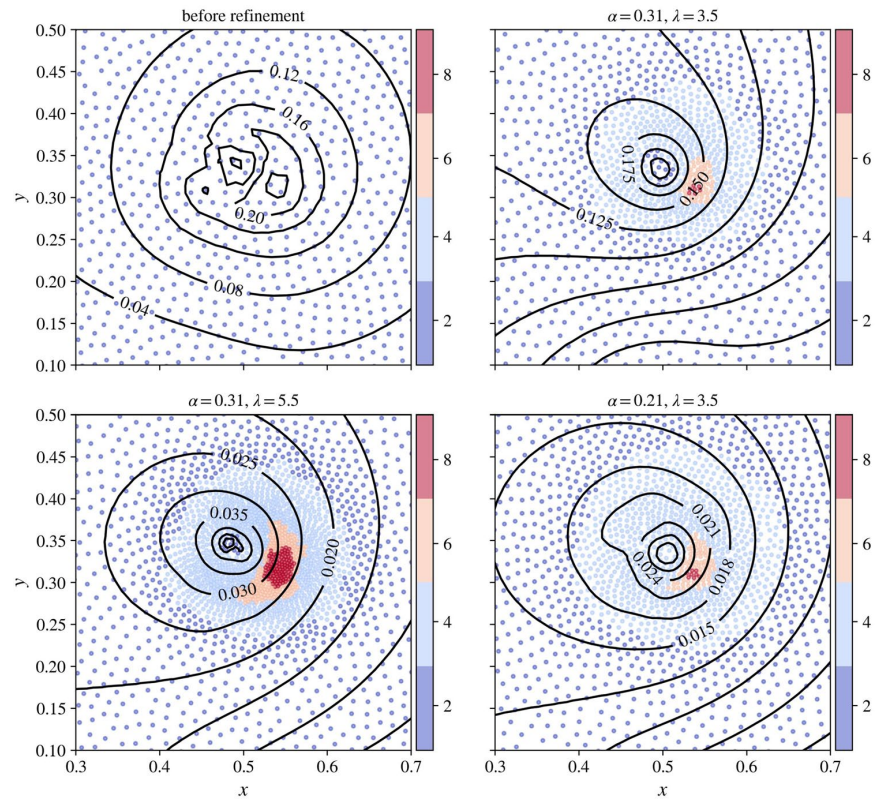
2.4 The REFIN module

After obtaining the list of nodes marked for modification, the refinement module is initialised. In this module, the local field description and local approximation order are left unchanged for the unmarked nodes, while the remaining nodes are further processed to determine other refinement-type-specific details – such as the amount of the (de-)refinement. Our h -refinement strategy is inspired by the recent h -adaptive mesh-free solution of elasticity problem [10], where the following h -refinement rule was introduced

$$h_i^{\text{new}}(\mathbf{p}) = \frac{h_i^{\text{old}}}{\frac{\eta_i - \alpha \eta_{\max}}{\eta_{\max} - \alpha \eta_{\max}} (\lambda - 1) + 1} \quad (6)$$

for the dimensionless parameter $\lambda \in [1, \infty)$ allowing us to control the aggressiveness of the refinement – the larger the value, the greater the change in nodal density, as shown in Fig. 4 on the left. This refinement rule also conveniently refines the areas with higher error indicator values more than

Fig. 5 Demonstration of *hp*-refinement for selected values of refinement parameters. The top left figure shows the numerical solution before its refinement, while the rest show its refined state for different values of refinement parameters. Contour lines are used to show the absolute error of the numerical solution. To denote the *p*-refinement, the nodes are coloured according to the local approximation order. For clarity, all figures are zoomed to show only the neighbourhood of an exponentially strong source $e^{-\alpha\|x-x_s\|^2}$ positioned at $x_s = \left(\frac{1}{2}, \frac{1}{3}\right)$



those closer to the upper refinement threshold $\alpha_h \eta_{\max}$. Similarly, a de-refinement rule is proposed

$$h_i^{\text{new}}(\mathbf{p}) = \frac{h_i^{\text{old}}}{\frac{\beta \eta_{\max} - \eta_i}{\beta \eta_{\max} - \eta_{\min}} \left(\frac{1}{\vartheta} - 1 \right) + 1}, \quad (7)$$

where parameter $\vartheta \in [1, \infty)$ allows us to control the aggressiveness of de-refinement.

The same refinement (6) and de-refinement (7) rules are applied to control the order of local approximation (*p*-refinement), except that this time the value is rounded to the nearest integer, as shown in Fig. 4 on the right. Similarly, and for the same reasons as for the marking strategy (see Sect. 2.3), we consider a separate treatment of *h*- and *p*-adaptive procedures by introducing (de-)refinement aggressiveness parameters $\{\lambda_h, \vartheta_h\}$ and $\{\lambda_p, \vartheta_p\}$ for *h*- and *p*-refinement types respectively.

2.5 Finalization step

Before the 4 modules can be iteratively repeated, the domain is re-discretised taking into account the newly computed local internodal distances $h_i^{\text{new}}(\mathbf{p})$ and the local

approximation orders $m_i^{\text{new}}(\mathbf{p})$. However, both are only known in the computational nodes, while global functions $\hat{h}^{\text{new}}(\mathbf{p})$ and $\hat{m}^{\text{new}}(\mathbf{p})$ over our entire domain space Ω are required.

We use Shepard's inverse distance weighting interpolation using the closest n_s^h neighbours to construct $\hat{h}^{\text{new}}(\mathbf{p})$ and the closest n_s^m neighbours to construct $\hat{m}^{\text{new}}(\mathbf{p})$. In general, the proposed refinement strategy can introduce aggressive and undesirable local jumps in node density, which ultimately leads to a potential violation of the quasi-uniform internodal spacing requirement within the stencil. To mitigate this effect, we use relatively large $n_s^h = 30$ to smoothen such potential local jumps. The $\hat{m}^{\text{new}}(\mathbf{p})$ is much less sensitive in this respect and therefore a minimum $n_s^m = 3$ is used.

Figure 5 schematically demonstrates 3 examples of *hp*-refinements. For demonstration purposes, the refinement parameters for *h*- and *p*-adaptivity are the same, i.e. $\{\alpha, \beta, \lambda, \vartheta\} = \{\alpha_h, \beta_h, \lambda_h, \vartheta_h\} = \{\alpha_p, \beta_p, \lambda_p, \vartheta_p\}$. Additionally, the de-refinement aggressiveness ϑ and the lower threshold β are kept constant, so that effectively only the upper limit of refinement α and the refinement aggressiveness λ are altered. We observe that the effect of the

refinement parameters is somewhat intuitive. The greater the aggressiveness λ , the better the local field description and the greater the number of nodes with high approximation order. A similar effect is observed when manipulating the upper refinement threshold α , except that the effect comes at a smoother manner. Note also that all refined states were able to increase the accuracy of the numerical solution from the initial state.

2.6 Note on marking and refinement strategies

With the chosen marking and refinement strategies, a separate treatment of h - and p -refinement types turned out to be a necessary complication for a better overall performance of the solution procedure. Nevertheless, we have tried to simplify the solution procedure as much as possible. In the process, important observations have been made – some of which we believe should be highlighted. This section therefore opens a discussion on important remarks related to the proposed marking and refinement modules.

2.6.1 The error indicators

Since the h - and p -refinements are conceptually different, our first attempt was to employ two different error indicators – one for each type of refinement. We employed the previously proposed variance of field values [10] for marking the h -refinement and the approximation order based IMEX for the p -refinement. Unfortunately, no notable advantages of such solution procedure has been observed and was therefore discarded due to the increased implementation complexity. However, other combinations that might show more promising results should be considered in future work.

2.6.2 Free parameters

In the proposed solution procedure, each adaptivity type comes with 4 free parameters that need to be defined, i.e. $\{\alpha_{h,p}, \beta_{h,p}, \lambda_{h,p}, \vartheta_{h,p}\}$. This gives a total of 8 free parameters that can be fine-tuned to a particular problem. While we have tried to avoid any kind of fine-tuning, we have nevertheless observed that these parameters can have a crucial impact on the overall performance of the hp -adaptive solution procedure in terms of (i) the achieved accuracy of the numerical solution, (ii) the spatial variability of the error of the numerical solution, (iii) the computational complexity, and (iv) the stability of the solution procedure.

We observed that if the refinement aggressiveness λ_h is too high, the number of nodes can either diverge into unreasonably large domain discretisations or ultimately violate the quasi-uniform internodal spacing requirement, making the solution procedure unstable. Note that here we refer to the stability of the solution of the discretised PDEs, which

ultimately governs the stability of the whole solution procedure. Furthermore, a large number of nodes combined with high approximation orders can lead to unreasonably high computational complexity in a matter of few iterations. However, when refinement aggressiveness λ_h and λ_p is set too low, the number of required iterations can increase to such an extent that the entire solution procedure becomes inefficient. On top of that, the lower and upper threshold multipliers α and β also play a crucial role. If α is too low, almost the entire domain is refined. Moreover, if α is too large, almost no refinement takes place and if it does, it is extremely local, which again has no beneficial consequences as it often leads to a violation of the quasi-uniform nodal distribution requirement.

In our tests, based on extensive experimental parameter testing, we have selected a reasonable combination of all 8 parameters that lead to a stable solution procedure while demonstrating the advantages of the proposed hp -adaptive approach. A thorough analysis of these parameters and their correlation would most likely lead to better results, as there is no guarantee that the selected parameters are optimal. However, such an analysis is beyond the scope of this paper, whose aim is to present an hp -adaptive solution procedure in the context of mesh-free methods and not to discuss the optimal marking and refinement strategies. Nevertheless, we have tried to reduce the number of free parameters using the same values for h - and p -adaptivity (see Fig. 5). While this approach also yielded satisfactory results that outperformed the numerical solutions obtained with uniform nodal and approximation order distributions in terms of accuracy, the full 8-parameter formulation easily yielded significantly better results.

2.6.3 A step beyond the artificial refinement strategies

As discussed in Sect. 2.3 and later in Sect. 4, the Texas Three Step based marking strategy cannot assure the optimal balance of h - and p -refinements due to missing local data regularity estimation [2]. In FEM, local Sobolev regularity estimate is commonly used to choose between the h - and the p -refinement [32–34]. Using an estimate for upper error bound [59, 60] one could generalise this approach to meshless methods, essentially upgrading the strategy with an information on the minimal internodal spacing required for local approximation of the partial differential operator of a certain order.

The refinement strategy could also be based on a specific knowledge about convergence rates and computational complexity in terms of internodal distance $h(\mathbf{p})$ and local approximation orders $m(\mathbf{p})$.

It has already been shown by Bayona [61] that the approximation error of mesh-free interpolant F is bounded by

$$\|F(\mathbf{p}) - u(\mathbf{p})\|_\infty \leq Ch^{m+1} \max_{p \in \Omega} |\mathcal{L}^{(m+1)}(u(\mathbf{p}))|. \quad (8)$$

Note that the constant C present in Equation (8) depends on the stencil and on the approximation order, both of which are modified by the *hp*-adaptive solution procedure. Nevertheless, for the purpose of illustrating how a better marking strategy could be constructed, we decide to simplify the Equation (8) to saying that the error e is proportional to $h(\mathbf{p})^{m(\mathbf{p})}$. Knowing the target error e_t , we write the ratio of e_t/e_0 as

$$\frac{e_t}{e_0} \propto \frac{h^{m_t}}{h^{m_0}} = h^{m_t - m_0}, \quad (9)$$

where m_t is used to denote the target approximation order and m_0 is the current order of the approximation used to compute current error e_0 .

From Equation (9) a smarter guess for target local approximation order can be obtained

$$m_t = m_0 + \ln \frac{e_t}{e_0}. \quad (10)$$

Such strategy would conveniently leave the approximation order unchanged when $e_t = e_0$, increase it when $e_t < e_0$ and decrease it when $e_t > e_0$.

A step even further could be to additionally consider the change in computational complexity, similar to what the authors of [35] and [45] have already shown. Therefore, we believe that future work should consider the minimum local computational complexity criteria. A rough computational complexity can be obtained with the help of

$$\chi \propto \frac{\left(\frac{m_t + d}{d}\right)^3 \left(\frac{1}{h_t}\right)^d}{\left(\frac{m_0 + d}{d}\right)^3 \left(\frac{1}{h_0}\right)^d}, \quad (11)$$

for domain dimensionality d and target and current inter-nodal distances h_t and h_0 respectively.

2.7 Implementation note

The entire *hp*-adaptive solution procedure from Algorithm 1 is implemented in C++. All meshless methods and approaches used in this work are included in our in-house developed *Medusa library* [52]. The code¹ was compiled using g++ (GCC) 9.3.0 for Linux with -O3 -DNDEBUG -fopenmp flags. Post-processing was done using Python 3.10

and Jupyter notebooks, also available in the provided git repository.

3 Demonstration on exponential peak problem

The proposed *hp*-adaptive solution procedure is first demonstrated on a synthetic example. We chose a 2-dimensional Poisson problem with exponentially strong source positioned at $\mathbf{x}_s = \left(\frac{1}{2}, \frac{1}{3}\right)$. This example is categorized as a difficult problem and is commonly used to test the performance of adaptive solution procedures [2, 29, 42, 62]. The problem has a tractable solution $u(\mathbf{x}) = e^{-a\|\mathbf{x}-\mathbf{x}_s\|^2}$, which allows us to evaluate the precision of the numerical solution $\hat{u}$, e.g. in terms of the infinity norm

$$e_\infty = \frac{\|\hat{u} - u\|_\infty}{\|u\|_\infty}, \quad \|u\|_\infty = \max_{i=1,\dots,N} |u_i|. \quad (12)$$

Governing equations are

$$\nabla^2 u(\mathbf{x}) = 2ae^{-a\|\mathbf{x}-\mathbf{x}_s\|^2} (2a\|\mathbf{x}-\mathbf{x}_s\| - d) \quad \text{in } \Omega, \quad (13)$$

$$u(\mathbf{x}) = e^{-a\|\mathbf{x}-\mathbf{x}_s\|^2} \quad \text{on } \Gamma_d, \quad (14)$$

$$\nabla u(\mathbf{x}) = -2a(\mathbf{x} - \mathbf{x}_s)e^{-a\|\mathbf{x}-\mathbf{x}_s\|^2} \quad \text{on } \Gamma_n, \quad (15)$$

for a d -dimensional domain Ω and strength $a = 10^3$ of the exponential source. The domain boundary is split into two sets: Neumann $\Gamma_n = \left\{\mathbf{x}, x \leq \frac{1}{2}\right\}$ and Dirichlet $\Gamma_d = \left\{\mathbf{x}, x > \frac{1}{2}\right\}$ boundaries. An example *hp*-refined numerical solution is shown in Fig. 6.

In the continuation of this paper, the numerical solution of the final linear system is obtained by employing BiCG-STAB solver with a ILUT preconditioner from the Eigen C++ library [63]. Global tolerance was set to 10^{-15} with a maximum number of 800 iterations and drop-tolerance and fill-factor set to 10^{-5} and 50 respectively. While the initial adaptivity solution was obtained without the guess, all other iterations used the previous numerical solution $\widehat{u}_{i-1}$ as the guess for new numerical solution $\hat{u}_i$, effectively reducing the number of iterations required by the BiCGSTAB solver.

3.1 Convergence analysis of unrefined solution

The problem is first solved on a two-dimensional unit disc without employing any refinement procedures, i.e. with uniform nodal and approximation order distributions. The shapes approximating the linear differential operators are

¹ The source code is available at: https://gitlab.com/e62Lab/public/2022_p_hp-adaptivity under tag v1.2.

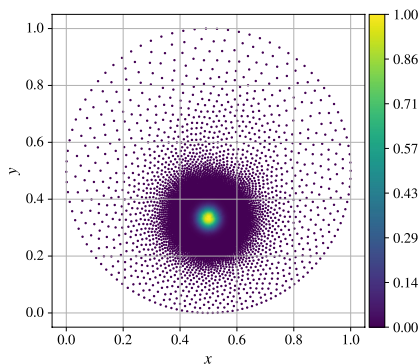


Fig. 6 Example hp -refined solution to exponential peak problem

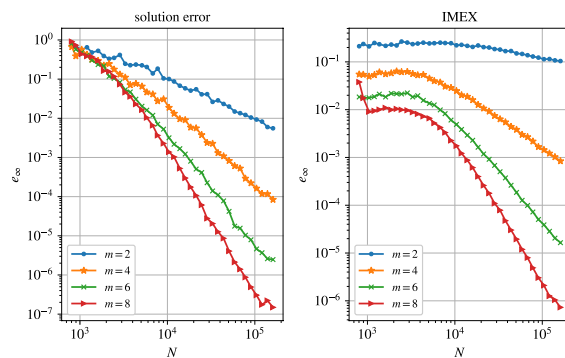


Fig. 7 Convergence of unrefined numerical solution (left) and IMEX error indicator (right). Figure only shows a median value after 50 runs with slightly different domain discretisations. Note that, the approximation order m in the right figure denotes the approximation order used to obtain the numerical solution, while the explicit operators employed by the IMEX error indicator are approximated with orders $m + 2$

computed using the RBF-FD with PHS order $k = 3$ and monomial augmentation $m \in \{2, 4, 6, 8\}$.

Figure 7 shows the results. Each plotted point is an average obtained after 50 consecutive runs with slightly different domain discretisations (a random seed for generating expansion candidates was changed, see [51] for more details). In this way, we can not only study the convergence behaviour, but also evaluate how prone the numerical method is to non-optimal domain discretisations. The convergence of the numerical solution for selected monomial augmentations is shown on the left. We observe that due to the strong source, the convergence rates no longer follow the

theoretical prediction of being proportional to h^m . Instead, the convergence rates for a small number of computational nodes ($N \lesssim 2000$) are significantly lower than that obtained for larger domain discretisations ($N \gtrsim 3000$) for all approximation orders $m > 2$. Furthermore, the accuracy gain using higher order approximations with small domain discretisations is practically negligible. However, when the local field description is sufficient, both the numerical solution and the IMEX error indicator (Fig. 7 on the right) give reliable results. While we could have forced at least one node in the neighbourhood of the source, we do not use any special techniques in this work. Instead, further research is simply limited to sufficiently large domains so that this observation does not represent an issue.

Moreover, the behaviour of the IMEX error indicator is studied on the right side of Fig. 7. Here, the approximation order m means that the implicit numerical solution u^{im} was obtained with approximation order m , while the explicit operators $\mathcal{L}^{ex}$ from IMEX were approximated using monomials up to and including order $m + 2$. The observations show that the maximum value of the error indicator also converges with the number of computational nodes. Moreover, we can also observe the aforementioned change in the convergence rate of the numerical solution, since the maximum value of the error indicator for domain sizes $N \lesssim 3000$ is approximately constant.

3.2 Analysis of hp -refined solution

The same problem is now solved by employing the hp -adaptivity. Free parameters are adjusted to each refinement type, as can be seen in Table 1. Adaptivity iteration loop is stopped after a maximum of N_{iter} iterations. For practical use, other stopping criteria could also be used, e.g. based on the maximum error indicator reduction

$$\frac{\eta_{max}^j}{\eta_{max}^0} \leq \gamma, \quad (16)$$

for the iteration index j . The shapes are computed with RBF-FD using the PHS with order $k = 3$ and local monomial augmentation restricted to choose between approximation orders $m \in \{2, 4, 6, 8\}$. Note that the IMEX error indicator increases the local approximation order by 2, effectively using monomial orders $m_{IMEX} \in \{4, 6, 8, 10\}$. Furthermore, to avoid unreasonably large number of computational nodes, the maximum number of allowed nodes N_{max} is defined. Once this number is reached, further h -refinement is prevented and

Table 1 Adaptivity parameters used to obtain solution to the peak problem

β_h	α_h	λ_h	ϑ_h	β_p	α_p	λ_p	ϑ_p	h_{max}	N_{max}	N_{iter}
0.175	0.225	2.625	1.01	10^{-4}	0.05	5	1.258	0.1	$2.5 \cdot 10^5$	70

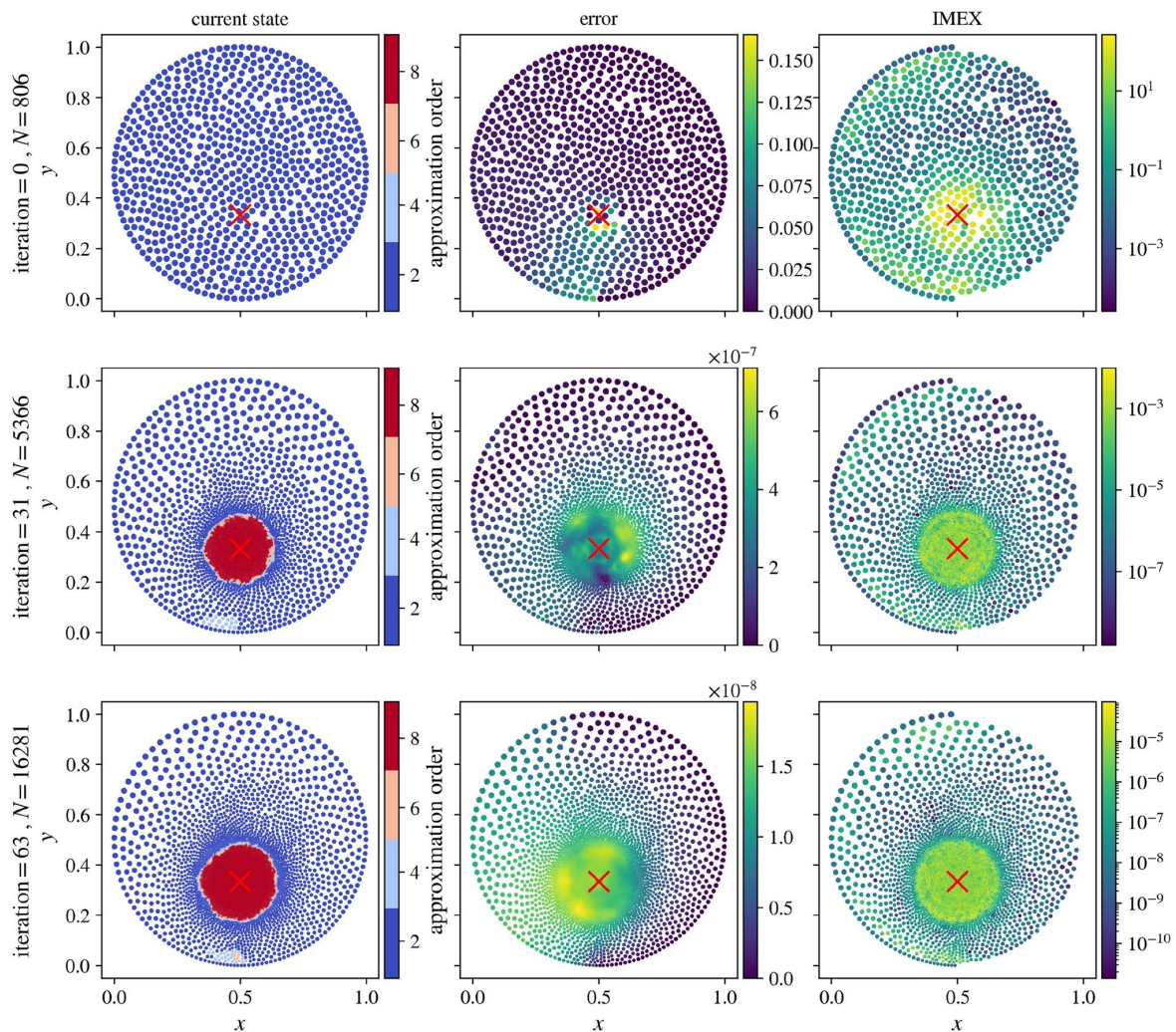


Fig. 8 Refinement demonstration. Initial iteration (top row), intermediate iteration (middle row) and best-performing iteration (bottom row) accompanied with solution error (middle column) and IMEX

error indicator values (right column). The IMEX values for Dirichlet boundary nodes are not shown. A red cross is used to mark the location of the strong peak

only de-refinement is allowed, while the p -adaptive method retains its full functionality. To avoid insufficient local field description, the local nodal density is limited by an upper bound, i.e. $h(\mathbf{p}) \leq h_{\max}$. The order of the PHS is left constant.

3.2.1 A brief analysis of IMEX error indicator

Figure 8 shows example indicator fields for the initial iteration, the intermediate iteration, and the iteration that achieved the best numerical solution accuracy – hereafter also referred to as the best-performing iteration or simply the best iteration. The third column shows the IMEX error

indicator. We can see that the IMEX has successfully located the position of the strong source at $\mathbf{x}_s = \left(\frac{1}{2}, \frac{1}{3}\right)$ as the highest indicator values are seen in its vicinity. Furthermore, the second column shows that both the accuracy of the numerical solution and the uniformity of the error distribution were significantly improved by the hp -adaptive solution procedure, further proving that IMEX can be successfully used as a reliable error indicator.

The behaviour of IMEX over 70 adaptivity iterations is also studied in Fig. 9. We are pleased to find that the convergence limit of the indicator around iteration $N_{\text{iter}} = 60$ agrees well with the convergence limit of the numerical

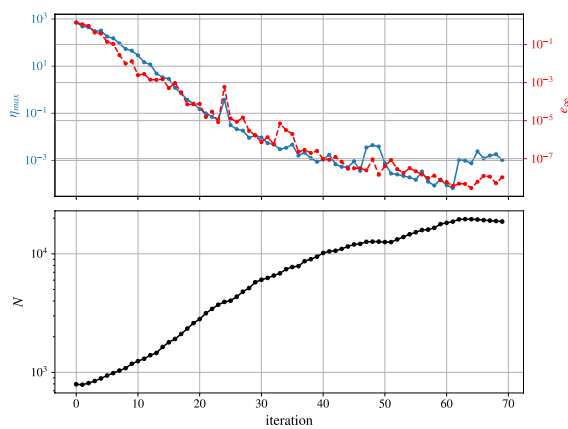


Fig. 9 In the top row convergence of IMEX error indicator (blue) and convergence of numerical solution (red) within 70 iterations is shown, while the total number of computational nodes is shown below

solution. This observation also makes the IMEX error indicator suitable for stopping criteria. Note that, in the process, the maximum error of the numerical solution has been reduced by about 9 orders of magnitude, while the maximum value of the error indicator has been reduced by about 7 orders of magnitude. In addition, Fig. 9 also shows the number of computational nodes with respect to the adaptivity iterations.

3.2.2 Approximation order distribution

The iterative adaptive procedure starts by obtaining the numerical solution of the unrefined problem setup. In this step, the approximation with the lowest approximation order, i.e. $m = 2$, is assigned to all computational nodes. Later, the approximation orders are changed according to the marking and refinement strategies. Figure 8 shows the approximation order distributions for 3 selected adaptivity iterations. We can observe that the highest approximation orders are all near the exponentially strong source. Moreover, due to h -adaptivity, the node density in the neighbourhood of the strong source is also significantly increased, i.e. $h_{\max}/h_{\min} \approx 52$ in the best-performing iteration.

After applying the p -refinement strategy in the refinement step, the approximation order in two neighbouring nodes may differ by more than one. While numerical experiments with FEM have shown that heterogeneity of polynomial order in FEM leads to undesired oscillations of the approximated solution [64], no similar behaviour was observed in our analyses with our setup using mesh-free methods. Thus, in contrast to p -FEM, where additional smoothing of the approximation order takes place within the refinement

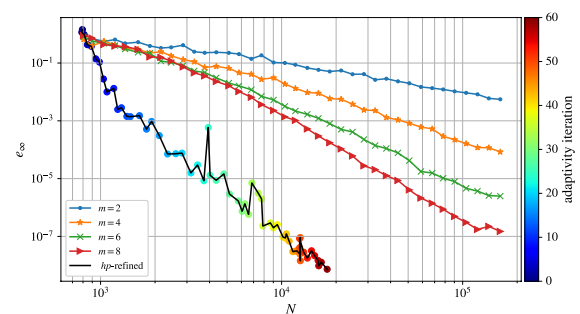


Fig. 10 Convergence of the hp -refined solution compared to the convergence of the unrefined solutions

module, we have completely avoided such manipulations and allow the approximation order in two neighbouring nodes to differ by more than one.

3.2.3 Convergence rates of hp -adaptive solution procedure

Finally, the convergence behaviour of the proposed hp -adaptive solution procedure is studied. In addition to the convergence of a single hp -adaptive run, Fig. 10 shows the convergences obtained without the use of refinement procedures, i.e. solutions obtained with uniform internodal spacing and approximation orders over the entire domain. The figure clearly shows that a hp -adaptive solution procedure was able to significantly improve the numerical solution in terms of accuracy and computational points required.

As previously discussed by Eibner [34] and Demkowicz [36], we believe that a more complex marking and refinement strategies would further improve the convergence behaviour, but already the proposed hp -adaptive solution procedure significantly outperforms the unrefined solutions. Specifically, the refined solution is almost 4 orders of magnitude more accurate than the unrefined solution (for the highest approximation order $m = 8$ used) at about 10^4 computational nodes.

4 Application to linear elasticity problems

In this section we address two problems from linear elasticity that are conceptually different from the exponential peak problem discussed in Sect. 3. While the solution of exponential peak problem is infinitely smooth, these two problems both have a singularity in the solution.

In areas of smooth solution, the hp -strategy should favour p -refinement (assuming that the local discretization is sufficient, as briefly discussed in Sect. 3.1), while near the singularity, h -refinement should be preferred [2, 36]. However, the Texas Three Step based marking strategy used in this paper

cannot trivially achieve this, since the strategy has no knowledge of the smoothness of the solution field. In addition, the strategy also cannot perform pure *h*- or pure *p*-refinement [34] (see Fig. 3), which would be ideal in the limiting situations. Instead, the strategy used enforces an increase in the approximation order by its design – even if the solution is not smooth and even if low-regularity data is being used to construct the approximation. Nevertheless, in our experiments we observed an increase of the approximation order near the singularity only in the first few iterations, while the following iterations were focused on improving the local field description with *h*-refinement. This observation is also in agreement with reports from the literature [2, 34], where authors justify the use of the Texas Three Step marking strategy also for problems with singularity in the solution.

4.1 Fretting fatigue contact

The application of the proposed *hp*-adaptive solution procedure is further expanded to study a linear elasticity problem. Specifically, we obtain a *hp*-refined solution to fretting fatigue contact problem [65] for which no closed form solution is known.

The problem dynamics is governed by the Cauchy-Navier equations

$$(\lambda + \mu)\nabla(\nabla \cdot \mathbf{u}) + \mu\nabla^2 \mathbf{u} = \mathbf{f} \quad (17)$$

with unknown displacement vector $\mathbf{u}$, external body force $\mathbf{f}$ and Lamé parameters μ and λ . The domain of interest is a thin rectangle of width W , length L and thickness D . Axial traction σ_{ax} is applied to the right side of the rectangle, while a compression force is applied to the centre of the rectangle to simulate contact. The contact is simulated by a compressing force F generated by two oscillating cylindrical pads of radius R , causing a tangential force Q . The tractions introduced by the two pads are predicted using an extension of Hertzian contact theory, which splits the contact area into the stick and slip zones depending on the friction coefficient μ and the combined elasticity modulus $E^{*-1} = \left(\frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2}\right)$, where E_i and ν_i are the Young's modulus and the Poisson's ratios of the sample and the pad, respectively. The problem is shown schematically in Fig. 11 together with the boundary conditions.

Theoretical predictions from [10] are used to obtain the contact half-width

$$a = 2\sqrt{\frac{FR}{t\pi E^*}}, \quad (18)$$

with normal traction

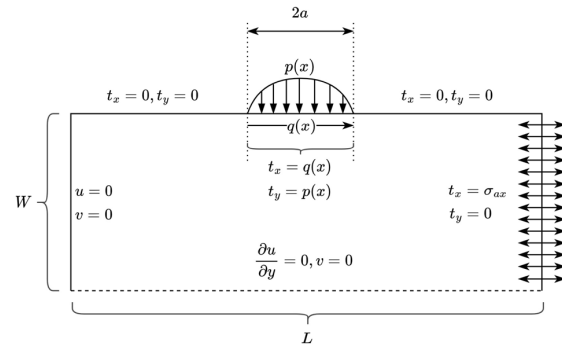


Fig. 11 Fretting fatigue contact problem scheme and boundary conditions

$$p(x) = \begin{cases} p_0 \sqrt{1 - \frac{x^2}{a^2}}, & |x| < a \\ 0, & \text{else} \end{cases}, \quad p_0 = \sqrt{\frac{FE^*}{t\pi R}}, \quad (19)$$

and tangential traction

$$q(x) = \begin{cases} -\mu p_0 \left(\sqrt{1 - \frac{x^2}{a^2}} - \frac{c}{a} \sqrt{1 - \frac{(x-e)^2}{c^2}} \right), & |x-e| < c \\ -\mu p_0 \sqrt{1 - \frac{x^2}{a^2}}, & c \leq |x-e| \text{ and } |x| \leq a \\ 0, & \text{else} \end{cases} \quad (20)$$

for $c = a\sqrt{1 - \frac{Q}{\mu F}}$ defined as the half-width of the slip zone and $e = \text{sgn}(Q)\frac{a\sigma_{ax}}{4\mu p_0}$ is the eccentricity due to axial loading. Note that the inequalities $Q \leq \mu F$ and $\sigma_{ax} \leq 4\left(1 - \sqrt{1 - \frac{Q}{\mu F}}\right)$ must hold for these expressions to be valid.

Plane strain approximation is used to reduce the problem from three to two dimensions and symmetry along the horizontal axis is used to further halve the problem size. Finally, $\Omega = [-L/2, L/2] \times [-W/2, 0]$ is taken as the domain.

We take $E_1 = E_2 = 72.1 \text{ GPa}$, $\nu_1 = \nu_2 = 0.33$, $L = 40 \text{ mm}$, $W = 10 \text{ mm}$, $t = 4 \text{ mm}$, $F = 543 \text{ N}$, $Q = 155 \text{ N}$, $\sigma_{ax} = 100 \text{ MPa}$, $R = 10 \text{ mm}$ and $\mu = 0.3$ for the model parameters. With this setup, the half-contact width a is equal to 0.2067 mm , which is about 200 times smaller than the domain width W . For stability reasons, the 4 corner nodes were removed after the domain was discretised.

The linear differential operators are approximated with RBF-FD using the PHS with order $k = 3$ and local monomial augmentation limited to choose between approximation orders $m \in \{2, 4, 6, 8\}$. The PHS order was left constant during the adaptive refinement. The *hp*-refinement parameters used to obtain the numerical solution are given in Table 2.

Table 2 Adaptivity parameters used to obtain solution to fretting fatigue contact problem

β_h	α_h	λ_h	ϑ_h	β_p	α_p	λ_p	ϑ_p	h_{max}	N_{max}	N_{iter}
$5 \cdot 10^{-5}$	10^{-4}	5	1.05	10^{-3}	0.1	4	1.05	$2.5 \cdot 10^{-4}$	$5 \cdot 10^5$	19

Figure 12 shows an example of a *hp*-refined solution to fretting fatigue problem in the last adaptivity iteration with $N = 46\,626$ computational nodes. We see that the solution procedure has successfully located the two critical points, i.e. the fixed upper left corner with a stress singularity and the area in the middle of the upper edge where contact is simulated. Note that the highest stress values (about 2 times higher) were calculated in the singularity in the upper left corner, but these nodes are not shown as our focus is shifted towards the area under the contact.

4.1.1 Surface traction under the contact

For a detailed analysis, we consider the surface traction σ_{xx} , as it is often used to determine the location of crack initiation. The surface traction is shown in Fig. 13 for 6 selected adaptivity iterations. The mesh-free nodes are coloured according to the local approximation order enforced by the *hp*-adaptive solution procedure. The message of this figure is twofold. First, it is clear that the proposed IMEX error indicator can be successfully used in linear elasticity problems, and second, we find that the *hp*-adaptive solution procedure has successfully approximated the surface traction near the contact. In doing so, the local field description under the contact has been significantly improved and the local approximation orders have taken a non-trivial distribution.

The surface traction in Fig. 13 is additionally accompanied with the FEM results on a much denser mesh with more than 100,000 DOFs obtained with the commercial solver Abaqus® [65]. To calculate the absolute difference between the two methods, the mesh-free solution was interpolated to Abaqus's computational points using Sheppard's inverse distance weighting interpolation with 2 nearest neighbours. We see that the absolute difference under the contact decreases with the number of adaptivity iterations and eventually settles at approximately 2 % of the maximum difference from the initial iteration. As expected, the highest absolute difference is at the edges of the contact, i.e. around $x = a$ and $x = -a$, while the difference is even smaller in the rest of the contact area. The absolute difference between the two methods is further studied in Fig. 14, where the mean of $|\sigma_{xx}^{FEM} - \sigma_{xx}^{mesh-free}|$ under the contact area, i.e. $-a \leq x \leq a$, is shown. We observe that the mesh-free *hp*-refined solution converges towards the reference FEM solution with respect to the adaptivity iterations. Moreover, Fig. 14 also shows the

number of computational nodes with respect to the adaptivity iteration.

4.2 The three-dimensional Boussinesq's problem

As a final benchmark problem we solve the three-dimensional Boussinesq's problem, where a concentrated normal traction acts on an isotropic half-space [66].

The problem has a closed form solution given in cylindrical coordinates r, θ and z as

$$\begin{aligned}
 u_r &= \frac{Pr}{4\pi\mu} \left(\frac{z}{R^3} - \frac{1-2\nu}{R(z+R)} \right), & u_\theta &= 0, \\
 u_z &= \frac{P}{4\pi\mu} \left(\frac{2(1-\nu)}{R} + \frac{z^2}{R^3} \right), \\
 \sigma_{rr} &= \frac{P}{2\pi} \left(\frac{1-2\nu}{R(z+R)} - \frac{3r^2z}{R^5} \right), \\
 \sigma_{\theta\theta} &= \frac{P(1-2\nu)}{2\pi} \left(\frac{z}{R^3} - \frac{1}{R(z+R)} \right), \\
 \sigma_{zz} &= -\frac{3Pz^3}{2\pi R^5}, & \sigma_{rz} &= -\frac{3Prz^2}{2\pi R^5}, \\
 \sigma_{r\theta} &= 0, & \sigma_{\theta z} &= 0,
 \end{aligned} \tag{21}$$

where P is the magnitude of the concentrated force, ν is the Poisson's ratio, μ is the Lamé parameter and R is the Euclidean distance to the origin. The solution has a singularity at the origin, which makes the problem ideal for treatment with adaptive procedures. Furthermore, the closed form solution also allows us to evaluate the accuracy of the numerical solution.

In our setup, we consider only a small part of the problem, i.e. $\varepsilon = 0.1$ away from the singularity, as schematically shown in Fig. 15. From a numerical point of view, we solve the Navier–Cauchy Equation (17) with Dirichlet boundary conditions described in (21), where the domain Ω is defined as a box, i.e. $\Omega = [-1, -\varepsilon] \times [-1, -\varepsilon] \times [-1, -\varepsilon]$.

Although the closed form solution is given in cylindrical coordinate systems, the problem is implemented using cartesian coordinates. We employ the proposed mesh-free *hp*-adaptive solution procedure where the shapes are computed with RBF-FD using the PHS with order $k = 3$ and monomial augmentation restricted to choose between approximation orders $m \in \{2, 4, 6, 8\}$. Other *hp*-refinement related parameters are given in Table 3. For the physical parameters of the problem, the values $P = -1$, $E = 1$ and $\nu = 0.33$ were assumed.

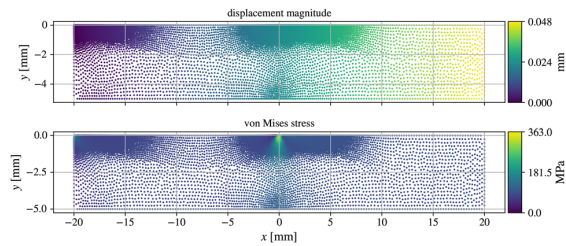


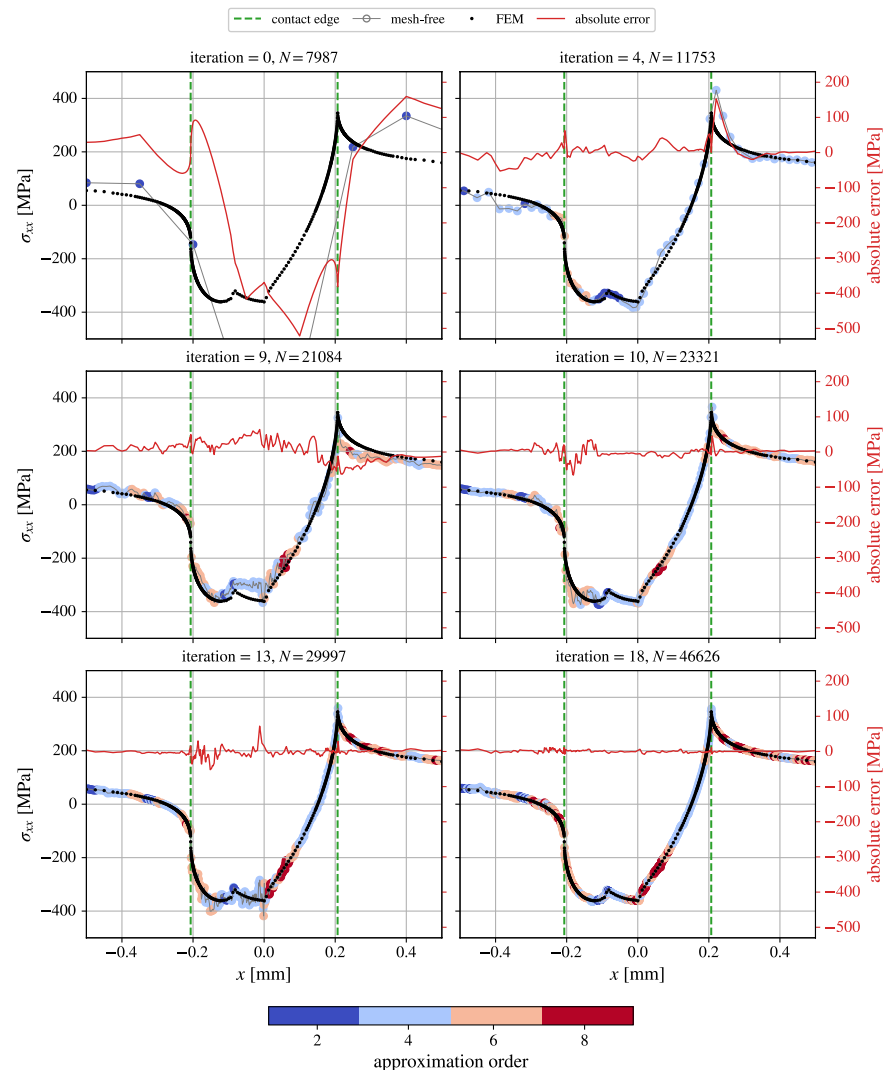
Fig. 12 Example *hp*-refined fretting fatigue contact solution

It is worth mentioning, that the final sparse system was solved using BiCGSTAB with ILUT preconditioner (employed with an initial guess obtained from the previous

adaptivity iteration), where the global tolerance was set to 10^{-15} with a maximum number of 500 iterations and drop-tolerance and fill-factor set to 10^{-6} and 60 respectively. Other possible choices and their effect on the solution procedure are further discussed in Sect. 4.2.2.

Example *hp*-refined numerical solution is given in Fig. 16. We can see that the proposed *hp*-adaptive solution procedure is sufficiently robust to obtain a good solution even for three-dimensional problems with singularities. Additionally, we also observe that the IMEX error indicator successfully identified the singularity, effectively seen as an increase in the local field description in the neighbourhood of the concentrated force applied at the origin.

Fig. 13 Surface traction under the contact for selected iteration steps demonstrating the *hp*-adaptivity process. Colours are used to denote the local approximation orders. Numerical solution is additionally compared against the Abaqus FEM solution, where the red line is used to denote the absolute difference between the two methods. For clarity, the two dashed green lines show the edge contact



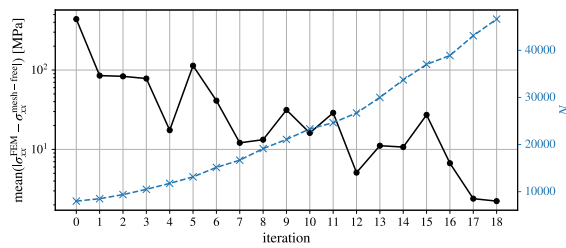


Fig. 14 Mean surface traction difference between the two methods under the contact area

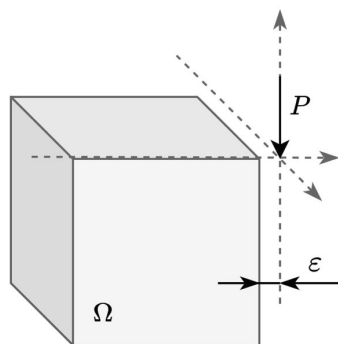


Fig. 15 Schematic presentation of Boussinesq's problem

4.2.1 The von Mises stress along the body diagonal

Figure 17 shows further evaluation of the hp -refined mesh-free numerical solution. Here, the von Mises stress at points near the body diagonal $(-1, -1, -1) \rightarrow (-\epsilon, -\epsilon, -\epsilon)$ is calculated for selected 4 adaptivity iterations and compared to the analytical values in terms of relative error. In addition, the nodes are coloured according to the local approximation order enforced by the hp -adaptive solution procedure. We can see that the highest relative error of approximately 0.3 at the initial state is observed in the neighbourhood of the origin. In the final iteration, the relative error is reduced by about an order of magnitude. We also see that the hp -adaptive solution procedure has found a non-trivial order distribution and that the number of nodes in the neighbourhood of the corner $(-\epsilon, -\epsilon, -\epsilon)$ has increased significantly.

A more quantitative analysis of the mesh-free solution is given in Fig. 18 where the ℓ_1 , ℓ_2 and ℓ_∞ error norms and number of computational nodes vs. adaptivity iteration are shown. Compared to the initial state, the hp -adaptive solution procedure was able to achieve a numerical solution that was almost two orders of magnitude more accurate. In the process, the number of computational nodes increased from 10 500 in the initial state to about 80 000 in the final iteration. However, it is interesting to observe that with

the configuration from Table 3, none of the computational nodes used the approximation with the highest order allowed ($m = 8$). Instead, in the final iteration, there were 130 nodes approximated with $m = 6$, and 5937 with $m = 4$, while the rest were approximated with the second order. Note that, as expected, most of the higher order approximations are near the concentrated force—which is difficult to represent visually, so we only give the descriptive data.

For reference, we take the h -refined solution by Slak et al. [10], who were able to reduce the infinity norm error by about an order of magnitude with $N \approx 140\,000$ nodes in the final iteration. It is perhaps naive to compare this result with ours, since the authors use different marking and refinement strategies and, more importantly, a different error indicator. Nevertheless, the infinity norm error of our hp -refined solution is in the neighbourhood of 10^{-3} compared to theirs at approximately 10^{-2} with almost twice as many computational nodes. We believe our results could be further improved by fine-tuning the free parameters, but we decided to avoid such an approach.

4.2.2 Additional discussion on solving the global sparse system

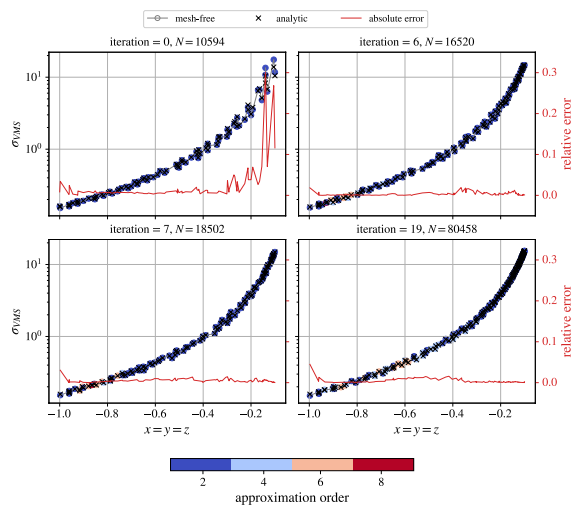
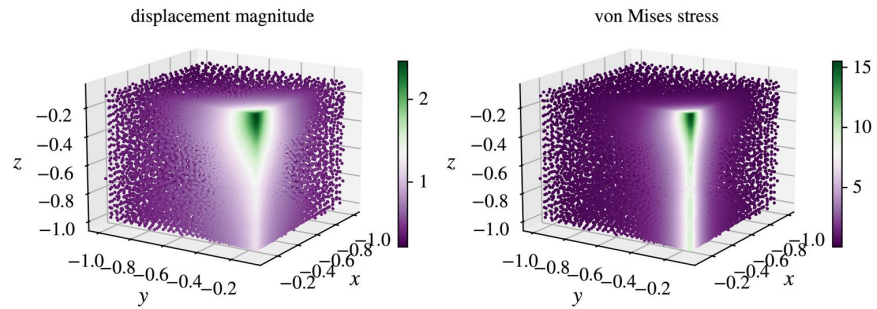
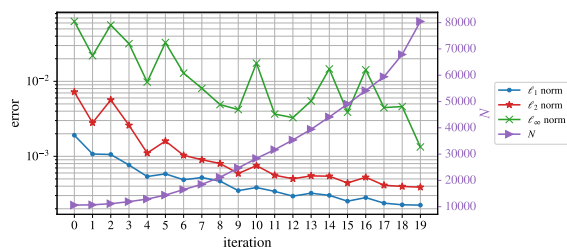
In all previous sections, we have completely neglected the importance of solving the global sparse system in the proposed hp -adaptive solution procedure with a suitable solver. However, inappropriate choice of solver can lead to inaccurate or even unstable behaviour and, most importantly, unreasonably large computational cost. To avoid such flaws, we compared an iterative BiCGSTAB and BiCGSTAB with ILUT preconditioner with two direct solvers—namely the SparseLU and the PardisoLU—on a hp -adaptive solution to the Boussinesq problem, performing 25 adaptivity iterations with approximately 10,000 initial nodes and 135,000 nodes after the last iteration. Note that the iterative BiCGSTAB solver with ILUT preconditioner was employed with an initial guess obtained from the previous adaptivity iteration.

In addition to the discussed solvers, we also tried the SparseQR. While its stability and accuracy were comparable to other solvers, its computational cost was significantly higher and was therefore removed from further analysis and from the list of potential candidates. For all performed tests we used the EIGEN linear algebra library [63].

Let us first examine the sparsity patterns of the systems assembled at different stages of the hp -adaptive process in Fig. 19, where we can see how the system increases in size and also becomes less sparse due to globally decreasing the internodal distance h and increasing the approximation order p . Additionally, the spectra of the matrices are shown in the bottom row of Fig. 19, where we can see that the ratios between the real and imaginary parts of the eigenvalues are in good agreement with previous studies [13, 14, 61].

Table 3 Adaptivity parameters used to obtain solution to Boussinesq's problem

β_h	α_h	λ_h	ϑ_h	β_p	α_p	λ_p	ϑ_p	h_{max}	N_{max}	N_{iter}
10^{-3}	10^{-3}	3.75	1.01	10^{-4}	10^{-2}	3	1.5	0.04	$7 \cdot 10^4$	20

Fig. 16 Example *hp*-refined numerical solution to Boussinesq's problem**Fig. 17** Numerical solution compared to analytical solution at the nodes near the body diagonal $(-1, -1, -1) \rightarrow (-\varepsilon, -\varepsilon, -\varepsilon)$ for selected iterations**Fig. 18** Convergence of numerical solution along with number of computational nodes

Moreover, Fig. 20 presents three different views of the solvers' performance: (i) the achieved accuracy of the final solution for different solvers, (ii) the number of iterations a solver needs to converge, and (iii) the execution times of each solver, each with respect to the *hp*-adaptive iterations. The differences in final accuracy for different solvers are marginal. Perhaps the BiCGSTAB shows better stability behaviour (in terms of error scatter) compared to others. Nevertheless, it is important to observe, that the SparseLU only works until the 15th iteration with approximately 50 000 nodes, at which point our computer ran out of the available 12 Gb memory, which is to be expected due to the computational complexity or SparseLU. PardisoLU, on the other hand, remains stable through all adaptivity iterations.

Generally speaking, the number of iterations BiCGSTAB needs to converge increases with *hp*-adaptivity iterations due to the increasing non-zero elements in the global system. The BiCGSTAB with a ILUT preconditioner shows similar behaviour, but with approximately 2/3 less iterations required. Both direct solvers, of course, require only one "iteration". Finally, the analysis of the execution time shows that the PardisoLU solver is by far the most efficient among all considered candidates.

With all things considered, PardisoLU seems to be the the best candidate for *hp*-adaptive solution procedure. However, the last adaptivity iteration with approximately 115,000 nodes was coincidentally right at the limit of the available 12 Gb RAM memory—using approximately 10.5 Gb. It is therefore expected that like SparseLU, the PardisoLU would soon run out of memory for larger domains. To avoid such problems, we chose to work with a general purpose iterative BiCGSTAB solver with ILUT preconditioner employed with an initial guess, since it shows slightly better computational efficiency than the pure BiCGSTAB and required only 7.5 Gb of RAM for approximately 135,000 nodes in the final adaptivity iteration.

Fig. 19 Global sparse matrix plot (top row) and spectra of the matrices (bottom row) at three selected iterations of the hp -adaptive solution procedure. Note that the spectra are computed for the BiCGSTAB solver with an ILUT preconditioning using an estimate from the previous iteration

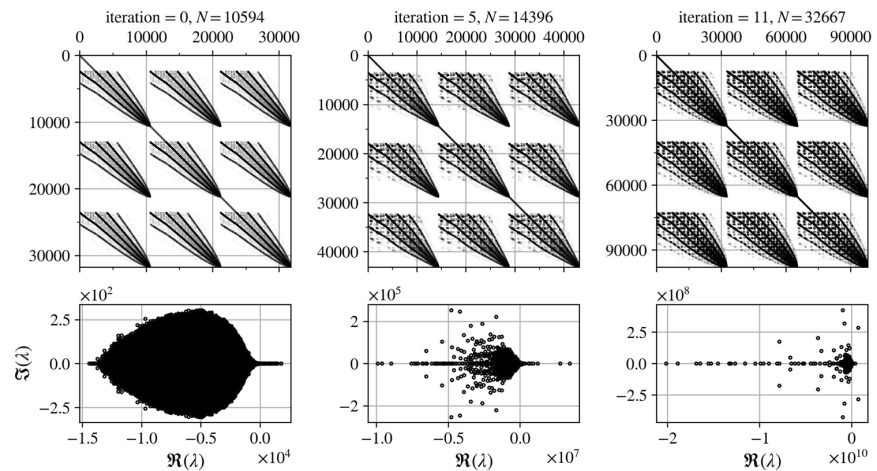
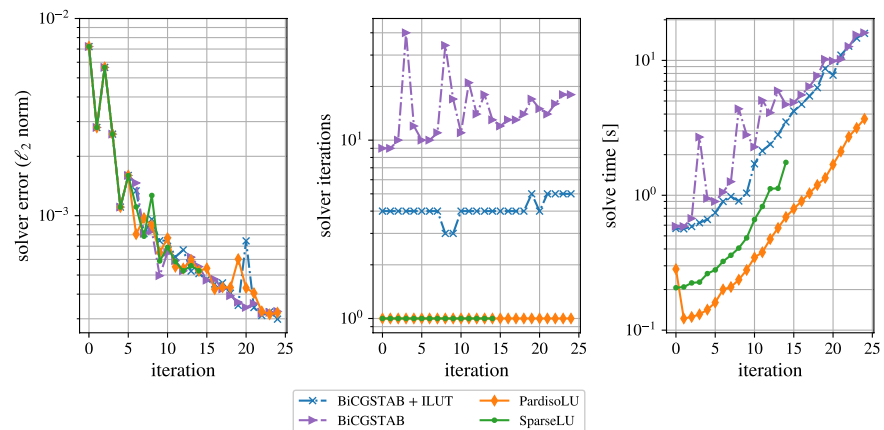


Fig. 20 Error of the final solution with respect to the adaptivity iteration for different solvers (left), number of solver iteration per adaptivity iteration (centre) and solver compute time for each adaptivity iteration (right)



5 Conclusions

In this paper we establish a baseline for hp strong form mesh-free analysis. We have formulated and implemented a hp -adaptive solution procedure and demonstrated its performance in three different numerical experiments.

The cornerstone of the presented hp -adaptive method is an iterative solve–estimate–mark–refine paradigm with the modified Texas Three Step marking strategy. The h -refine of the proposed method relies on an advancing front node positioning algorithm based on Poisson disc sampling, which enables dimension-independent node generation supporting spatially variable density distributions. For the adaptive order of the method, we exploit an elegant control over the order of the approximation via the augmenting monomials in the approximation basis.

We proposed an IMEX error indicator, where the implicit solution of the problem is processed with the higher order

local explicit representation of PDE at hand, e.g. if the implicit solution is computed with a second order approximation, the explicit re-evaluation happens at fourth order. Our analyses show that the proposed error indicator successfully captures main characteristics of error distributions, which suffices for the proposed iterative adaptivity.

The proposed hp -adaptive solution procedure is first demonstrated on a two-dimensional Poisson problem with exponential source and mixed boundary conditions. Further demonstration focuses on linear elasticity problems. First, a 2D fretting fatigue problem – a contact problem with pronounced peaks in the surface stress, and second, a 3D Boussinesq's problem with stress singularity. In both cases, we have demonstrated the advantages of using the proposed hp -adaptive approach.

Although the hp -adaptivity introduces additional steps in the solution procedure and is therefore undoubtedly computationally more expensive per node than the non-adaptive

approach, it is essential in problems that exhibit volatilities in solution in small regions of the domain. For example singularity in the contact problem require excessively detailed numerical analysis near the contact compared to the rest (the bulk) of the domain. Such cases are extremely difficult (or even impossible) to solve without adaptivity, since the minimal uniform h and p distribution required to capture these volatilities would lead to unreasonably high computational complexity. In cases of a smooth solution, however, the benefits of *hp*-adaptivity in most cases do not justify its computational overheads.

We are aware that there are many opportunities for improvement of presented methodology. The IMEX error indicator needs further clarification. Other error indicators should also be implemented and tested. During our experiments, we have found that a marking strategy with more free parameters leads to better accuracy, but is also more difficult to understand and control and can be case dependent. A smarter and more effective refinement and marking strategies are certainly part of future work. These should possibly take into account more information about the method itself, e.g. the dependence of the computational complexity on the approximation order, and most importantly local data regularity to choose between p and h refinement.

One of our goals in future work is also generalisation of the presented *hp*-adaptive solution procedure to time-dependent problems. The most straightforward approach to achieve that is to granularly adapt h and p throughout the simulation. In its simplest form, the proposed *hp*-adaptivity would be performed at each time step, starting with the *hp* distributions of the previous time step and using the same adaptivity parameters for all time steps. A more sophisticated approach would also take into account the desired accuracy during the simulation, resulting in time-dependent adaptivity parameters. For example, if one is only interested in a steady state solution, the desired accuracy would increase with time, reaching its maximum at steady state. Additionally, to perform proper adaptive analysis, the time step should also be adaptive, which requires an additional step in the *hp*-adaptive solution procedure.

Acknowledgements The authors acknowledge the financial support from the Slovenian Research and Innovation Agency (ARIS) research core funding No. P2-0095, and research projects No. J2-3048 and No. N2-0275 (joint research project between National Science Centre, Poland and ARIS, where Polish research group is Funded by National Science Centre, Poland under the OPUS call in the Weave programme 2021/43/I/ST3/00228. This research was funded in whole or in part by National Science Centre (2021/43/I/ST3/00228). For the purpose of Open Access, the author has applied a CC-BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.).

Declarations

Conflict of interest The authors declare that they have no conflict of interest. All the co-authors have confirmed to know the submission of the manuscript by the corresponding author.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Upadhyay BD, Sonigra SS, Daxini SD (2021) Numerical analysis perspective in structural shape optimization: a review post 2000. *Adv Eng Softw* 155:102,992
2. Mitchell WF, McClain MA (2014) A comparison of *hp*-adaptive strategies for elliptic partial differential equations. *ACM Trans Math Softw (TOMS)* 41(1):1–39
3. Segeth K (2010) A review of some a posteriori error estimates for adaptive finite element methods. *Math Comput Simul* 80(8):1589–1600. <https://doi.org/10.1016/j.matcom.2008.12.019>. <https://www.sciencedirect.com/science/article/pii/S0378475408004230>. ESCO 2008 Conference
4. Liu GR, Gu YT (2005) An introduction to meshfree methods and their programming. Springer, Berlin
5. Liu GR (2002) Mesh free methods: moving beyond the finite element method. CRC Press, Boca Raton. <https://doi.org/10.1201/9781420040586>
6. Zienkiewicz OC, Taylor RL, Zhu JZ (2005) The finite element method: its basis and fundamentals. Elsevier, Amsterdam
7. van der Sande K, Fornberg B (2021) Fast variable density 3-d node generation. *SIAM J Sci Comput* 43(1):A242–A257
8. Shankar V, Kirby RM, Fogelson AL (2018) Robust node generation for mesh-free discretizations on irregular domains and surfaces. *SIAM J Sci Comput* 40(4):A2584–A2608
9. Jacquemin T, Suchde P, Bordas SP (2023) Smart cloud collocation: geometry-aware adaptivity directly from CAD. *Comput Aided Design* 154:103409. <https://doi.org/10.1016/j.cad.2022.103409>. <https://linkinghub.elsevier.com/retrieve/pii/S0010448522001427>
10. Slak J, Kosec G (2019) Adaptive radial basis function-generated finite differences method for contact problems. *Int J Numer Methods Eng* 119(7):661–686
11. Davydov O, Oanh DT (2011) Adaptive meshless centres and rbf stencils for poisson equation. *J Comput Phys* 230(2):287–304
12. Jacquemin T, Bordas SPA (2021) A unified algorithm for the selection of collocation stencils for convex, concave, and singular problems. *Int J Numer Methods Eng* 122(16):4292–4312
13. Jančič M, Slak J, Kosec G (2021) Monomial augmentation guidelines for rbf-fd from accuracy versus computational time perspective. *J Sci Comput* 87(1):1–18

14. Bayona V, Flyer N, Fornberg B, Barnett GA (2017) On the role of polynomials in rbf-fd approximations: II. numerical solution of elliptic pdes. *J Comput Phys* 332:257–273
15. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P (1996) Meshless methods: an overview and recent developments. *Comput Methods Appl Mech Eng* 139(1–4):3–47
16. Kosec G, Šarler B (2014) Simulation of macrosegregation with mesosegregates in binary metallic casts by a meshless method. *Eng Anal Bound Elements* 45:36–44. <https://doi.org/10.1016/j.enganabound.2014.01.016>. <https://linkinghub.elsevier.com/retrieve/pii/S0955799714000290>
17. Maksić M, Djurica V, Souvent A, Slak J, Depolli M, Kosec G (2019) Cooling of overhead power lines due to the natural convection. *Int J Electrical Power Energy Syst* 113:333–343. <https://doi.org/10.1016/j.ijepes.2019.05.005>. <https://linkinghub.elsevier.com/retrieve/pii/S0142061518340055>
18. Gui WZ, Babuska I (1985) The h, p and hp versions of the finite element method in 1 dimension. part 3. the adaptive hp version. Tech. rep., Maryland Univ College Park Lab for Numerical Analysis
19. Gui WZ, Babuška I (1986) The h, p and hp versions of the finite element method in 1 dimension. part II. the error analysis of the h and hp versions. *Numerische Mathematik* 49(6):613–657
20. Devloo12 PR, Bravo CM, Rylo EC (2012) Recent developments in hp adaptive refinement
21. Zienkiewicz OC, Zhu JZ (1987) A simple error estimator and adaptive procedure for practical engineering analysis. *Int J Numer Methods Eng* 24(2):337–357
22. González-Estrada OA, Natarajan S, Ródenas JJ, Bordas SP (2021) Error estimation for the polygonal finite element method for smooth and singular linear elasticity. *Comput Math Appl* 92:109–119
23. Thimnejad M, Fallah N, Khoei AR (2015) Adaptive refinement in the meshless finite volume method for elasticity problems. *Comput Math Appl* 69(12):1420–1443. <https://doi.org/10.1016/j.camwa.2015.03.023>
24. Angulo A, Pozo LP, Perazzo F (2009) A posteriori error estimator and an adaptive technique in meshless finite points method. *Eng Anal Bound Elements* 33(11):1322–1338. <https://doi.org/10.1016/j.enganabound.2009.06.004>
25. Oanh DT, Davydov O, Phu HX (2017) Adaptive rbf-fd method for elliptic problems with point singularities in 2d. *Appl Math Comput* 313:474–497
26. Sang-Hoon P, Kie-Chan K, Sung-Kie Y (2003) A posterior error estimates and an adaptive scheme of least-squares meshfree method. *Int J Numer Methods Eng* 58(8):1213–1250. <https://doi.org/10.1002/nme.817>
27. Afshar M, Naisipour M, Amani J (2011) Node moving adaptive refinement strategy for planar elasticity problems using discrete least squares meshless method. *Finite Element Anal Design* 47(12):1315–1325
28. Guo B, Babuška I (1986) The hp version of the finite element method. *Comput Mech* 1(1):21–41
29. Mitchell WF (2016) Performance of hp-adaptive strategies for 3d elliptic problems
30. Tinsley Oden J, Wu W, Ainsworth M (1995) In: Modeling, mesh generation, and adaptive numerical methods for partial differential equations (Springer), pp 347–366
31. Ainsworth M, Senior B (1997) Aspects of an adaptive hp-finite element method: adaptive strategy, conforming approximation and efficient solvers. *Comput Methods Appl Mech Eng* 150(1–4):65–87
32. Houston P, Senior B, Süli E (2003) In: Numerical mathematics and advanced applications (Springer), pp 631–656
33. Houston P, Süli E (2005) A note on the design of hp-adaptive finite element methods for elliptic partial differential equations. *Comput Methods Appl Mech Eng* 194(2–5):229–243
34. Eibner T, Melenk JM (2007) An adaptive strategy for hp-fem based on testing for analyticity. *Comput Mech* 39(5):575–595
35. Bürg M, Dörfler W (2011) Convergence of an adaptive hp finite element strategy in higher space-dimensions. *Appl Numer Math* 61(11):1132–1146
36. Demkowicz L, Rachowicz W, Devloo P (2002) A fully automatic hp-adaptivity. *J Sci Comput* 17(1):117–142
37. Rachowicz W, Pardo D, Demkowicz L (2006) Fully automatic hp-adaptivity in three dimensions. *Comput Methods Appl Mech Eng* 195(37–40):4816–4842
38. Benito J, Urena F, Gavete L, Alvarez R (2003) An h-adaptive method in the generalized finite differences. *Comput Methods Appl Mech Eng* 192(5–6):735–759
39. Liu G, Kee BB, Chun L (2006) A stabilized least-squares radial point collocation method (ls-rpcm) for adaptive analysis. *Comput Methods Appl Mech Eng* 195(37–40):4843–4861
40. Hu W, Trask N, Hu X, Pan W (2019) A spatially adaptive high-order meshless method for fluid-structure interactions. *Comput Methods Appl Mech Eng* 355:67–93
41. Tolstykh A, Shirobokov D (2003) On using radial basis functions in a “finite difference mode” with applications to elasticity problems. *Comput Mech* 33(1):68–79
42. Oanh DT, Tuong NM (2022) An approach to adaptive refinement for the rbf-fd method for 2d elliptic equations. *Appl Numer Math* 178:123–154
43. Tóth B, Düster A (2022) h-adaptive radial basis function finite difference method for linear elasticity problems. *Comput Mech*:1–20
44. Fan L (2019) Adaptive meshless point collocation methods: investigation and application to geometrically non-linear solid mechanics. Ph.D. thesis, Durham University
45. Mishra PK, Ling L, Liu X, Sen MK (2020) Adaptive radial basis function generated finite-difference (rbf-fd) on non-uniform nodes using *p*-refinement. *arXiv preprint arXiv:2004.06319*
46. Milewski S (2021) Higher order schemes introduced to the meshless fdm in elliptic problems. *Eng Anal Bound Elements* 131:100–117
47. Albuquerque-Ferreira A, Ureña M, Ramos H (2021) The generalized finite difference method with third- and fourth-order approximations and treatment of ill-conditioned stars. *Eng Anal Bound Elements* 127:29–39
48. Liszka T, Duarte C, Tworzydło W (1996) hp-meshless cloud method. *Computer Methods Appl Mech Eng* 139(1–4):263–288
49. Jančić M, Slak J, Kosec G (2021) In: 2021 6th International Conference on Smart and Sustainable Technologies (SpliTech), pp 01–06. <https://doi.org/10.23919/SpliTech52315.2021.9566401>
50. Duarte CA, Oden JT (1996) An hp adaptive method using clouds. *Computer Methods Appl Mech Eng* 139(1–4):237–262
51. Slak J, Kosec G (2019) On generation of node distributions for meshless pde discretizations. *SIAM J Sci Comput* 41(5):A3202–A3229
52. Slak J, Kosec G (2021) Medusa: a c++ library for solving pdes using strong form mesh-free methods. *ACM Trans Math Softw (TOMS)* 47(3):1–25
53. Depolli M, Slak J, Kosec G (2022) Parallel domain discretization algorithm for RBF-FD and other meshless numerical methods for solving PDEs. *Comput Struct* 264:106773. (Publisher: Elsevier)
54. Duh U, Kosec G, Slak J (2021) Fast variable density node generation on parametric surfaces with application to mesh-free methods. *SIAM J Sci Comput* 43(2):A980–A1000
55. Wendland H (2004) Scattered data approximation. Cambridge Monographs on Applied and Computational Mathematics.

- Cambridge University Press, Cambridge. <https://doi.org/10.1017/CBO9780511617539>
56. Davydov O, Oanh DT, Tuong NM (2023) Improved stencil selection for meshless finite difference methods in 3d. *J Comput Appl Math* 425:115031. <https://doi.org/10.1016/j.cam.2022.115031>. <https://www.sciencedirect.com/science/article/pii/S037704272200629X>
 57. Jančič M, Strniša F, Kosec G (2022) In: 2022 7th International Conference on Smart and Sustainable Technologies (SpliTech), pp 01–04. <https://doi.org/10.23919/SpliTech55088.2022.9854342>
 58. Heuer N, Mellado ME, Stephan EP (2001) *hp*-adaptive two-level methods for boundary integral equations on curves. *Computing* 67(4):305–334
 59. Bayona V (2019) An insight into rbf-fd approximations augmented with polynomials. *Comput Math Appl* 77(9):2337–2353
 60. Tominec I, Larsson E, Heryudono A (2021) A least squares radial basis function finite difference method with improved stability properties. *SIAM J Sci Comput* 43(2):A1441–A1471. <https://doi.org/10.1137/20M1320079>
 61. Bayona V (2019) Comparison of moving least squares and rbf+poly for interpolation and derivative approximation. *J Sci Comput* 81(1):486–512
 62. Daniel P, Ern A, Smears I, Vohralík M (2018) An adaptive *hp*-refinement strategy with computable guaranteed bound on the error reduction factor. *Comput Math Appl* 76(5):967–983
 63. Guennebaud G, Jacob B, et al (2010) Eigen v3. <http://eigen.tuxfamily.org>
 64. Wakeni MF, Aggarwal A, Kaczmarczyk L, McBride AT, Athanasiadis I, Pearce CJ, Steinmann P (2022) A *p*-adaptive, implicit-explicit mixed finite element method for diffusion-reaction problems. *Int J Numer Methods Eng*
 65. Kosec G, Slak J, Depolli M, Trobec R, Pereira K, Tomar S, Jacquemin T, Bordas SP, Wahab MA (2019) Weak and strong form meshless methods for linear elastic problem under fretting contact conditions. *Tribol Int* 138, 392–402. (Publisher: Elsevier)
 66. Slaughter WS (2002) *Three-dimensional problems*. Birkhäuser, Boston, pp 331–386. https://doi.org/10.1007/978-1-4612-0093-2_9

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Chapter 4

Spatially-Adaptive Approximation Methods

Stencil-size-effect observations from Section 2.2 suggest that a spatially-varying stencil could be beneficial for both the accuracy of the numerical solution and computational complexity of the solution procedure. This has already been studied by Liszka et al. [102] in their attempt of *hp*-adaptive meshless solution procedure, where they proposed to spatially-vary the stencil size to improve the local accuracy of the numerical solution.

Built on Liszka’s idea, this chapter explores the concept of spatially-varying approximation methods aimed to improve one or more aspects toward efficient PDE-solving solution procedures. We exploit the advantages of *local approximation* and test the performance of the solution procedure when the approximation method, and correspondingly the stencil size, is spatially-variable.

Based on the stability analysis from Section 2.3 we propose a hybrid WLS–RBF-FD approximation method in Section 4.1 exploiting the stability of RBF-FD and computational efficiency of WLS on scattered nodes. A step further is done in Section 4.2, where we propose a spatially-variable node regularity and a corresponding spatially-variable combination of RBF-FD on scattered nodes and MON on uniform nodes reducing the computational complexity even further.

4.1 A Hybrid RBF-FD and WLS Mesh-Free Strong-Form Approximation Method

To improve the computational efficiency, we propose a solution procedure with spatially-variable approximation method – employing a different approximation method in different domain regions, depending on the complexity of the underlying problem and geometrical properties. For instance, a more stable method (e.g. RBF-FD) can be used in areas that are possibly detrimental from a stability point of view, while computationally less demanding method (e.g. WLS) can be used in the rest of the domain. Such procedure yields shorter wall-clock times and minimally affects the accuracy of the numerical solution and stability of the solution procedure.

Contributions.

This section represents contribution **C8**. We show that spatially-varying the approximation method has desirable effects on the stability of the solution procedure.

Addressed hypotheses.

The analyses from the following publication again demonstrate the superiority of high-order RBF-FD approximation method, confirming hypothesis **H2**. We also demonstrate

that a hybrid WLS–RBF-FD approximation method can result in a more stable solution procedure, compared to that employing a pure WLS approximation, and computationally cheaper, compared to that employing a pure RBF-FD approximation. The latter observation confirms hypothesis **H7**.

Publications included in this section:

- M. Jančič and G. Kosec, “A hybrid RBF-FD and WLS mesh-free strong-form approximation method,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 1–6. DOI: 10.23919/SpliTech55088.2022.9854278

Regarding my contribution: I made a literature overview of the topic, implemented the solution procedure, planned and executed the analyses and jointly prepared the manuscript with co-authors.

A hybrid RBF-FD and WLS mesh-free strong-form approximation method

Mitja Jančič

“Jožef Stefan” Institute

Parallel and Distributed Systems Laboratory,
Jamova cesta 39, 1000 Ljubljana, Slovenia
and

“Jožef Stefan” International Postgraduate School
Jamova cesta 39, 1000 Ljubljana, Slovenia
Email: mitja.jancic@ijs.si

Gregor Kosec

“Jožef Stefan” Institute

Parallel and Distributed Systems Laboratory,
Jamova cesta 39, 1000 Ljubljana, Slovenia
Email: gregor.kosec@ijs.si

Abstract—Since the advent of mesh-free methods as a tool for the numerical analysis of systems of Partial Differential Equations (PDEs), many variants of differential operator approximation have been proposed. In this work, we propose a local mesh-free strong-form method that combines the stability of Radial Basis Function-Generated Finite Differences (RBF-FD) with the computational effectiveness of Diffuse Approximation Method (DAM), forming a so-called hybrid method. To demonstrate the advantages of a hybrid method, we evaluate its computational complexity and accuracy of the obtained numerical solution by solving a two-dimensional Poisson problem with an exponentially strong source in the computational domain. Finally, we employ the hybrid method to solve a three-dimensional Boussinesq’s problem on an isotropic half-space and show that the implementation overhead can be justified.

Index Terms—mesh-free methods, hybrid, RBF-FD, WLS, strong-form

I. INTRODUCTION

In recent years, mesh-free methods [1] have been increasingly used to obtain a numerical solution to a system of PDEs. They are computationally more complex than traditional mesh-based methods, but the fact that they can operate on scattered nodes makes them very desirable, especially when complex three-dimensional domains are considered.

Since the advent of mesh-free methods in the 1970s, many different variants have been proposed, such as the Finite Point Method [2], the Generalized Finite Difference Method [3], the Diffuse Approximation Method (DAM) [4] and the Radial Basis Function-Generated Finite Differences (RBF-FD) [5], to name but a few of the most commonly used, with recent research exploiting parallelism opportunities offered by a modern computer architecture [6].

While the RBF-FD is known for its high stability, DAM, also known as the Weighted Least Squares (WLS) approach, is known for its low computational complexity. Moreover, the WLS approach has been shown to be incredibly stable for low order approximations but has stability issues for higher

order approximations [7]. On the contrary, the RBF-FD is stable even for higher order approximations. Thus, the aim of this paper is to combine the advantages of the RBF-FD variant (namely the stability) with the computationally efficient WLS variant by proposing a *novel* hybrid WLS – RBF-FD method. This method essentially splits the stencils into two separate sets: One that use the WLS approximation approach to approximate the differential operators and another one, that uses the RBF-FD approximation approach.

The stability and computational complexity of the proposed hybrid method are studied on a solution to a two-dimensional Poisson problem with an exponentially strong source [8]. In addition, we also provide a solution to a three-dimensional Boussinesq’s problem of the concentrated normal traction acting on an isotropic half-space [9], [10]. We show that the hybrid method is more stable than the pure WLS variant and computationally cheaper than the pure RBF-FD variant.

II. SOLUTION PROCEDURE EMPLOYING MESH-FREE METHODS

To obtain a numerical solution $\hat{u}$ to a system of PDEs, three steps are required. First, the computational domain Ω is discretized using a dedicated node positioning algorithm that supports a spatially variable nodal distribution [11] with a quasi-uniform internodal spacing h . An example of nodal distribution is shown in Figure 1. A parallelized version of the same algorithm was recently published in [12], however, parallel execution is already out of the scope of this paper.

After discretizing the domain, the differential operators are approximated. A detailed procedure on differential operator approximation in the context of mesh-free methods is described in the following Section III.

In the final step, the system of PDEs is discretized in spatial and temporal sense, resulting in a global system of linear equations. The system is solved and its solution is proclaimed as the numerical solution $\hat{u}$ of a considered system of PDEs, of course, subject to given initial and boundary conditions.

The authors would like to acknowledge the financial support of the ARRS research core funding No. P2-0095, ARRS project funding No. J2-3048 and the World Federation of Scientists.

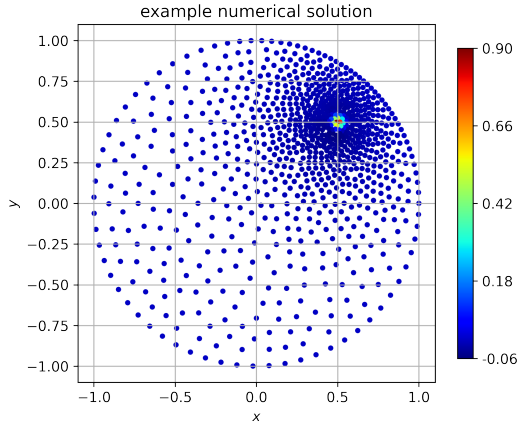


Fig. 1. An example of numerical solution of a Poisson problem on scattered nodes.

III. LINEAR DIFFERENTIAL OPERATOR APPROXIMATION IN THE CONTEXT OF MESH-FREE METHODS

Consider a d -dimensional domain $\Omega \subset \mathbb{R}^d$ with a set of N discretization points $\{\mathbf{x}_j\}_{j=1}^N$. In the context of mesh-free methods, a linear differential operator $\mathcal{L}$ in any node $\mathbf{x}_c \in \Omega$ is approximated over a set of neighboring nodes

$$\widehat{\mathcal{L}}u(\mathbf{x}_c) = \sum_{i=1}^n w_i u(\mathbf{x}_i) \quad (1)$$

for an arbitrary function u , weights w_i yet to be determined and support domain size n also called *stencil size*. It has been reported that a well-designed stencil can significantly reduce the computational cost [13], but, usually, as is also the case in this paper, the closest n nodes are chosen as the stencil to a central node $\mathbf{x}_c$.

The weights from equation (1) are calculated for a given set of s basis functions $\{p_j\}_{j=1}^s$. In the Diffuse Approximation Method, a set of monomials $p_1, \dots, p_s$ with up to and including degree m with $s = \binom{m+d}{d}$ are used as the approximation basis. This essentially means that the approximation (1) can be compactly written as

$$\mathbf{P}\mathbf{w} = \ell_p, \quad (2)$$

where matrix $\mathbf{P}$ is a $n \times s$ matrix of monomials evaluated at stencil nodes,

$$\mathbf{P} = \begin{bmatrix} p_1(\mathbf{x}_1) & \cdots & p_s(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_n) & \cdots & p_s(\mathbf{x}_n) \end{bmatrix} \quad (3)$$

and ℓ_p is the vector of values assembled by applying the considered operator $\mathcal{L}$ to the monomials at a central point $\mathbf{x}_c$

$$\ell_p^i = (\mathcal{L}p_i(\mathbf{x}))|_{\mathbf{x}=\mathbf{x}_c}. \quad (4)$$

When the number of basis functions is equal to the stencil size ($s = n$), the described formulation yields a quadratic

system of equations. Experience shows that the choice of larger support sizes ($n > s$) can be advantageous for accuracy and stability reasons. This leads to an overdetermined linear system, which is usually treated as a minimization of the Weighted Least Squares (WLS) norm. In the remainder of this paper, the above approximation approach, using only monomials as basis functions, will be referred to as the WLS approximation approach or WLS method.

Note that the same approximation procedure (1) can also be used for any other commonly chosen basis functions, such as Multiquadrics, Gaussians, Radial Basis Functions (RBFs). In this paper we focus on two different types of basis: the already presented approach with monomials only and the approach using Polyharmonic Splines (PHS) augmented with monomials. The latter leads to a RBF-FD variant of the mesh-free methods described in the following section.

A. The mesh-free RBF-FD variant

We now take RBFs $\varphi(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_c\|)$ centered at the stencil nodes of a central node $\mathbf{x}_c$. The approximation (1) then takes a compact form

$$\Phi \mathbf{w} = \ell_\varphi \quad (5)$$

for matrix Φ of evaluated radial basis functions

$$\Phi = \begin{bmatrix} \varphi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \cdots & \varphi(\|\mathbf{x}_n - \mathbf{x}_1\|) \\ \vdots & \ddots & \vdots \\ \varphi(\|\mathbf{x}_1 - \mathbf{x}_n\|) & \cdots & \varphi(\|\mathbf{x}_n - \mathbf{x}_n\|) \end{bmatrix} \quad (6)$$

and ℓ_φ is the vector of values assembled by applying the considered operator $\mathcal{L}$ to the RBFs at central point $\mathbf{x}_c$

$$\ell_\varphi^i = (\mathcal{L}\varphi(\|\mathbf{x} - \mathbf{x}_i\|))|_{\mathbf{x}=\mathbf{x}_c}. \quad (7)$$

Different RBFs can be used. To avoid the dependency on a shape parameter, we choose Polyharmonic splines (PHS)

$$\varphi(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \quad (8)$$

where r denotes the Euclidean distance between two nodes. However, the approximation with a pure RBF basis guarantees neither convergent behavior nor solvability. To mitigate these problems, the approximation is augmented with a monomial basis by additionally enforcing an exactness constraint for monomials, as we did in equation (2). This ensures convergent behavior and also allows us to control the order of the approximation, since the approximation order is the same as the order of the augmented monomials. This procedure finally results in a compactly written system

$$\begin{bmatrix} \Phi & \mathbf{P} \\ \mathbf{P}^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \ell_\varphi \\ \ell_p \end{bmatrix} \quad (9)$$

with Lagrangian multipliers $\boldsymbol{\lambda}$. The system (9) is overdetermined and treated as a constraint optimization problem [14]. The weights are obtained by solving the system, while Lagrangian multipliers are discarded.

B. Hybrid WLS-RBF-FD approximation approach

The local RBF-FD systems (9) are clearly larger than the purely monomial systems (2), making the RBF-FD method computationally more expensive. Therefore, our aim is to combine the computational efficiency of WLS approach with the high stability of RBF-FD variant to create a novel *hybrid* method.

The hybrid method has an additional step in the solution procedure, where we need to specify which stencils $\mathcal{N}(\mathbf{x}_i)$ use the WLS approach to approximate the differential operators and which the RBF-FD. This step essentially splits the N discretization nodes of Ω into two parts: N_{WLS} nodes whose stencils use WLS and $N_{\text{RBF-FD}}$ nodes whose stencils use RBF-FD, where $N_{\text{RBF-FD}} + N_{\text{WLS}} = N$.

Assigning a particular approximation type to a particular stencil is not a trivial task. The aim of a hybrid method is to ultimately result in numerical method that is more stable than the pure WLS and computationally less complex than the pure RBF-FD. Therefore, the RBF-FD approximation is naively assigned only to nodes with a high error of the numerical solution $\hat{u}$ expectancy, while the rest are approximated with the WLS approach.

Normally, error indicators, such as ZZ-type [15], are used in such cases. Although using an error indicator makes the most sense and would probably lead to better results, in this paper we make the decision a priori.

From an implementation point of view, only a small amount of overhead is required to implement a hybrid method. The biggest and practically only extra effort we have is when a global system is being assembled, because the sizes of WLS local systems (2) and RBF-FD local systems (9) do not match. Additional zero values have to be assigned in the global matrix to compensate for the mismatching sizes of the local approximations.

Note on the implementation

All elements and corresponding functionality used in this paper are available as part of the *Medusa library* [16].

IV. RESULTS

In this section, an overview of the results is provided. We first study the proposed hybrid method on a two-dimensional Poisson problem with an exponentially strong source in the domain. In particular, we focus on the convergence rates and shape computation times. Finally, as a proof of concept, a three-dimensional Boussinesq's problem is solved in Section IV-B.

All calculations were performed on a single core of a computer with Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz processor and 64 GB of DDR4 memory. The code¹ was compiled with g++ (GCC) 9.3.0 for Linux with -O3 -DNDEBUG flags. The sparse system is solved using the single-threaded LU solver, unless otherwise specified.

¹Source code is available at <https://gitlab.com/e62Lab/public/cp-2022-splitech-hybrid-engine> under the tag *v1.1*.

A. Two-dimensional synthetic example

The proposed hybrid method is studied by solving a synthetic example. We choose a two-dimensional elliptic PDE problem, i.e. a $d = 2$ dimensional Poisson problem, with non-constant Dirichlet boundary conditions in domain Ω . This example is usually used to test adaptive algorithms [8].

The problem is governed by

$$\nabla^2 u(\mathbf{x}) = f_{\text{lap}}(\mathbf{x}) \quad \text{in } \Omega, \quad (10)$$

$$u(\mathbf{x}) = f(\mathbf{x}) \quad \text{on } \partial\Omega, \quad (11)$$

where the domain Ω is a two-dimensional unit disc and the right-hand side is chosen to have an exponentially strong source

$$f(\mathbf{x}) = \exp(-\alpha \|\mathbf{x} - \mathbf{x}_s\|^2), \quad (12)$$

where α determines the strength of the source (for a strong source $\alpha = 10^3$) and $\mathbf{x}_s = \mathbf{1}/2$ is the location of the source. The Laplacian of $f(\mathbf{x})$ can also be calculated analytically

$$f_{\text{lap}} = 4(\alpha^2 \|\mathbf{x} - \mathbf{x}_s\|^2 - \alpha) \exp(-\alpha \|\mathbf{x} - \mathbf{x}_s\|^2). \quad (13)$$

An example solution is shown in Figure 1. The above problem has an analytical solution $u(\mathbf{x}) = f(\mathbf{x})$, which allows us to evaluate the accuracy of the numerically obtained solution $\hat{u}$ in terms of the infinity norm error e_∞ .

The domain Ω was filled with N scattered nodes with a variable node distribution that ensures the best local field description in the neighborhood of the strong source. In this work, the nodal distribution is given by

$$h(\mathbf{x}) = \min(dx + (Dx - dx) \|\mathbf{x} - \mathbf{x}_s\|^{3/2}, dx), \quad (14)$$

for $dx = Dx/5$ and 30 different values of Dx .

The problem was solved using all three previously described mesh-free variants, i.e. with the WLS approach using only monomials up to and including degree $m \in \{2, 4, 6\}$, with the RBF-FD approach using Polyharmonic splines of order $k = 5$ additionally augmented with monomials up to and including the same order m , and finally with a hybrid WLS-RBF-FD with the same approximation order. The stencil size n was determined according to the recommendations of Bayona [17] for a stable RBF-FD approximation

$$n = 2 \binom{m+d}{d}. \quad (15)$$

The division of the nodes into $N_{\text{RBF-FD}}$ RBF-FD nodes and N_{WLS} WLS nodes was done a priori without an error indicator. The largest error of the numerical solution is expected in the neighborhood of the exponentially strong source. We therefore define a circle with radius $r_s = 0.15$ around the strong source $\mathbf{x}_s$. All the stencils with a central node $\mathbf{x}_c$ less than r_s from the source are approximated using the more stable RBF-FD approach, while the rest use the WLS approximation. An example of the distribution of approximation types within the hybrid method is shown in Figure 2.

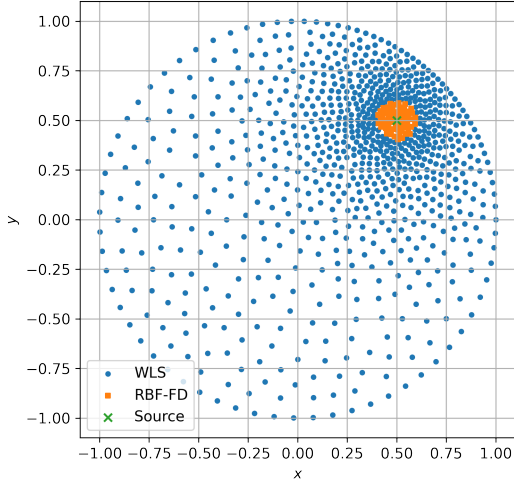


Fig. 2. An example of different approximation methods used within the hybrid WLS-RBF-FD method.

1) *Convergence rates:* In this paper, the error of the numerical solution is evaluated in computational nodes in terms of the infinity norm

$$e_\infty = \frac{\|\hat{u} - u\|_\infty}{\|u\|_\infty}, \quad \|u\|_\infty = \max_{i=1,\dots,N} u_i \quad (16)$$

because it shows the lowest convergence rates and, unlike the 2-norm, does not require averaging. After the solution is obtained, the computational nodes are discarded and the domain is discretized again with the same internodal spacing function h . With the new discretization, the shapes must be recomputed, which essentially allows us to evaluate the dependence of the approximation method on the quality of the node positioning. The process is repeated $N_{\text{runs}} = 100$ times, every time resulting in an approximately the same number of discretization nodes N .

The convergence zones for three different approximation orders and three different mesh-free variants are shown in Figure 3. In the case of a low order approximation ($m = 2$), we can immediately see that all three approximation methods are stable, with the smallest spread around a median of the infinity error norm clearly belonging to the WLS approach. The fact that the lower order WLS approximations are more stable with the WLS variant was already observed by Jančič [7]. Higher order approximations ($m = 4$ and $m = 6$), however, are more stable with the RBF-FD. The stability is further evaluated in Figure 4, making the advantages of a hybrid method in case of a higher order approximation even more evident.

2) *Computational times:* Another advantage of the hybrid method is that it is computationally cheaper than the pure RBF-FD approximation. This is demonstrated in Figure 5, showing the average shape calculation time for 10 runs. We can clearly see that the pure RBF-FD approximation is computationally expensive, while the WLS approach is

computationally more efficient and the hybrid method is somewhere in-between - depending on the $N_{\text{RBF-FD}}/N$ ratio.

B. Benchmark example

As a benchmark case, we chose to solve a three-dimensional Boussinesq's problem of the concentrated normal traction acting on an isotropic half-space [9], [10]. The problem is governed by the Cauchy-Navier equations

$$(\lambda + \mu)\nabla(\nabla \cdot \mathbf{u}) + \mu\nabla^2 \mathbf{u} = \mathbf{f} \quad (17)$$

with unknown displacement vector $\mathbf{u}$, external body force $\mathbf{f}$ and Lamé parameters

$$\lambda = \frac{E\nu}{(1-2\nu)(1+\nu)} \quad \text{and} \quad (18)$$

$$\mu = \frac{E}{2(1+\nu)}, \quad (19)$$

for Young modulus $E = 1$ and Poisson ratio $\nu = 0.33$.

For domain Ω we take a three-dimensional box

$$\Omega = \{(x, y, z) \in \mathbb{R}^3, -0.1 \leq (x, y, z) \leq 1\} \quad (20)$$

and discretize it using h -refinement towards the corner $\mathbf{x}_s = (-0.1, -0.1, -0.1)$ where force $\mathbf{P}$ with magnitude 1 in the $-\hat{\mathbf{e}}_z$ direction is applied. The discretization resulted in a total of $N = 18849$ discretization points.

The problem has a closed form solution [9] for displacements $\mathbf{u}(\mathbf{x}) = \mathbf{u}(x, y, z)$

$$u_x(\mathbf{x}) = x \frac{P}{4\pi\mu} \left(\frac{z}{\|\mathbf{x}\|^3} - \frac{1-2\nu}{\|\mathbf{x}\|(\|\mathbf{x}\|+z)} \right) \quad (21)$$

$$u_y(\mathbf{x}) = y \frac{P}{4\pi\mu} \left(\frac{z}{\|\mathbf{x}\|^3} - \frac{1-2\nu}{\|\mathbf{x}\|(\|\mathbf{x}\|+z)} \right) \quad (22)$$

$$u_z(\mathbf{x}) = \frac{P}{4\pi\mu} \left(\frac{z^2}{\|\mathbf{x}\|^3} + \frac{2(1-\nu)}{\|\mathbf{x}\|^3} \right) \quad (23)$$

allowing us to calculate the infinity norm error in terms of the displacement magnitude.

To solve the sparse system, BiCGSTAB with ILUT preconditioner was used. The global tolerance was set to 10^{-14} with a maximum number of 500 iterations, while the drop tolerance and fill factor were 10^{-5} and 30 respectively.

Results are computed using all three variants described previously, i.e. WLS (with Gaussian weights using $\sigma = 1.5$, essentially increasing the importance of nodes further away from the central stencil node), RBF-FD and a hybrid version of both with $r_s = 0.5$, for monomials of order $m = 4$ and PHS of order $k = 5$. A visual representation of the solution obtained with the hybrid method is shown in Figure 6, while a comparison of important numerical data is given in Table I.

We see that the novel hybrid method was able to obtain a numerical solution of sufficient quality. It is also clear from Table I that RBF-FD was able to achieve the best accuracy - approximately two orders of magnitude better than the hybrid method, but more importantly, the pure WLS

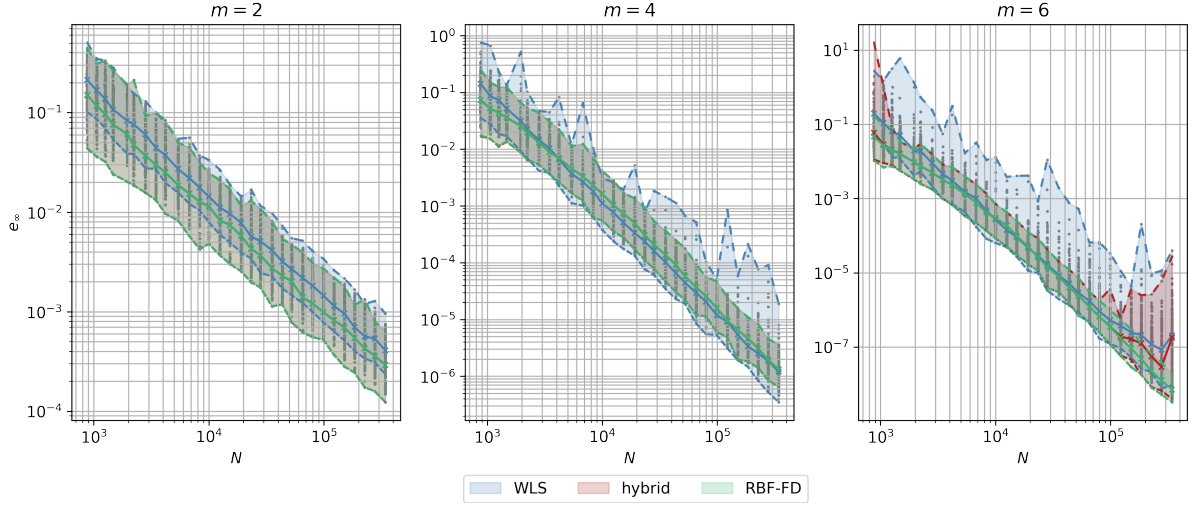


Fig. 3. Convergence rates for WLS approximation approach (blue), RBF-FD approximation approach (green) and a novel hybrid approximation approach (red) for low order approximations $m = 2$ (left) and higher order approximations $m = 4$ (middle) and $m = 6$ right.

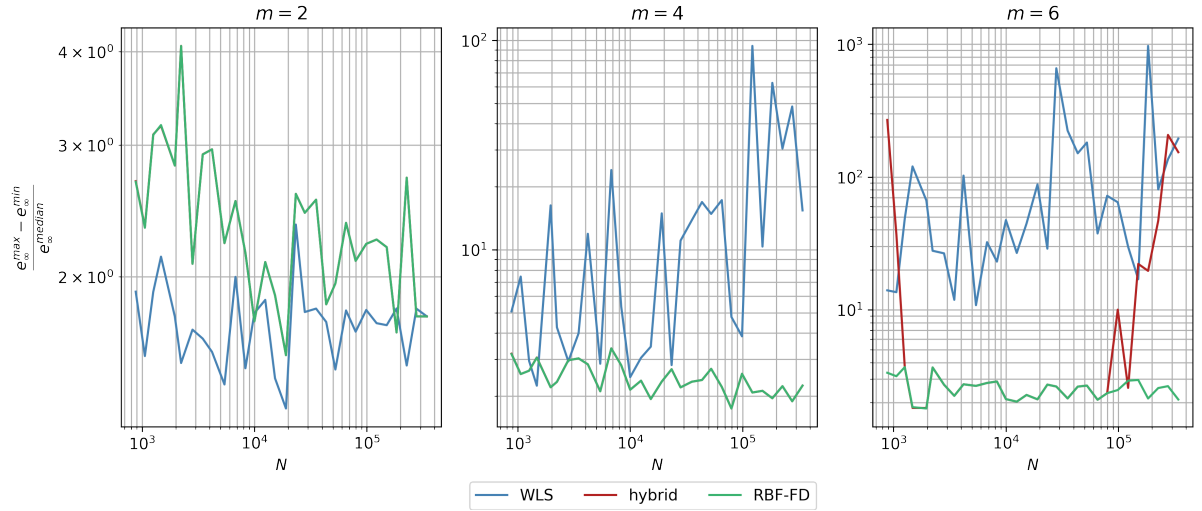


Fig. 4. Normalized spread around a median infinity norm error after $N_{\text{runs}} = 100$ for WLS approximation approach (blue), RBF-FD approximation approach (green) and a novel hybrid approximation approach (red) for low order approximations $m = 2$ (left) and higher order approximations $m = 4$ (middle) and $m = 6$ right.

TABLE I
COMPARISON TABLE FOR THE SOLUTION OF BOUSSINESQ'S PROBLEM.

Approximation	e_∞	t_{shape} [s]	$N_{\text{RBF-FD}}/N \cdot 100$
WLS	NaN	4.74	0.00
RBF-FD	$9.48 \cdot 10^{-5}$	8.22	100.00
hybrid	$2.37 \cdot 10^{-3}$	6.15	34.28

of great importance, because it justifies the effort required to implement a hybrid method. It is also important to observe that less than 35 % of the nodes from the hybrid method used the RBF-FD approximation approach, which is already enough to outperform the WLS in terms of stability and precision, and small enough to outperform the RBF-FD in terms of computation time, reducing it by about 33 %.

V. CONCLUSIONS

A novel WLS–RBF-FD mesh-free method combining the RBF-FD and WLS variants is presented. We demonstrate that approximation approach field to converge. This observation is

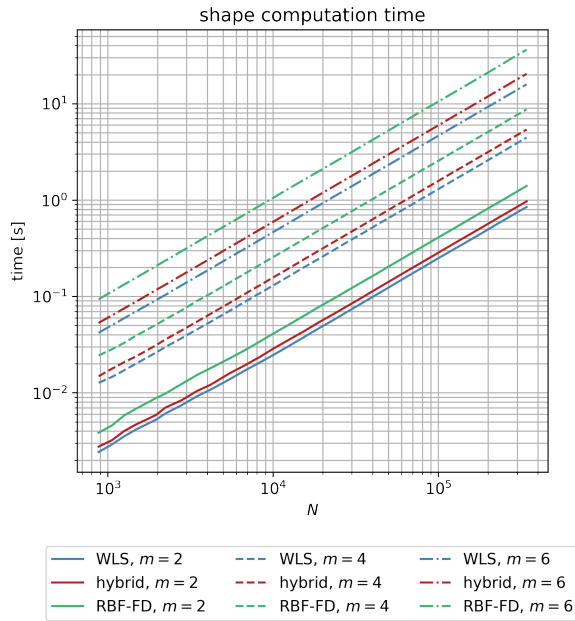


Fig. 5. Shape computation times for WLS approximation approach (blue), RBF-FD approximation approach (green) and a hybrid WLS-RBF-FD approximation approach (red).

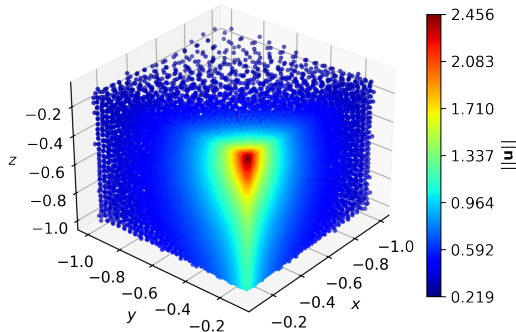


Fig. 6. Example benchmark case. Numerical solution obtained with the proposed WLS-RBF-FD hybrid method.

we can combine the advantages of the two commonly used mesh-free variants with only a small amount of additional work justified for the higher order ($m > 2$) approximations.

Using a two-dimensional synthetic example with exponentially strong source, we show that the newly proposed hybrid method can be successfully used to obtain a numerical solution. We also demonstrate that the hybrid method is indeed computationally cheaper than the pure RBF-FD approach and more stable than the pure WLS approach for higher order approximations. Finally, on a solution to the three-dimensional Boussinesq's problem of the concentrated normal traction

acting on an isotropic half-space we observe that the WLS variant fails to converge, while the hybrid WLS-RBF-FD method converges and reduces the shape computational times for about 33 % compared to the pure RBF-FD.

In this work, the stencils were a priori divided into RBF-FD stencils and WLS stencils. We believe that better results could be obtained by using error indicators.

REFERENCES

- [1] H. Wang and Q.-H. Qin, *Methods of fundamental solutions in solid mechanics*. Elsevier, 2019.
- [2] E. Oñate, S. Idelsohn, O. Zienkiewicz, and R. Taylor, "A finite point method in computational mechanics. applications to convective transport and fluid flow," *International journal for numerical methods in engineering*, vol. 39, no. 22, pp. 3839–3866, 1996.
- [3] L. Gavete, M. Gavete, and J. Benito, "Improvements of generalized finite difference method and comparison with other meshless method," *Applied Mathematical Modelling*, vol. 27, no. 10, pp. 831–847, 2003.
- [4] C. Prax, H. Sadat, and P. Salagnac, "Diffuse approximation method for solving natural convection in porous media," *Transport in Porous Media*, vol. 22, no. 2, pp. 215–223, 1996.
- [5] A. Tolstykh and D. Shirobokov, "On using radial basis functions in a "finite difference mode" with applications to elasticity problems," *Computational Mechanics*, vol. 33, no. 1, pp. 68–79, 2003.
- [6] R. Trobec and M. Depolli, "A k-d tree based partitioning of computational domains for efficient parallel computing," in *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2021, pp. 284–290.
- [7] M. Jančić and G. Kosec, "Stability analysis of rbf-fd and wls based local strong form meshless methods on scattered nodes," 2022.
- [8] W. F. Mitchell, "A collection of 2d elliptic problems for testing adaptive grid refinement algorithms," *Applied mathematics and computation*, vol. 220, pp. 350–364, 2013.
- [9] C. Nwoji, H. Onah, B. Mama, and C. Ike, "Solution of the boussinesq problem of half space using green and zerna displacement potential function method," *The Electronic Journal of Geotechnical Engineering (EJGE)*, vol. 22, no. 11, pp. 4305–4314, 2017.
- [10] J. Slak and G. Kosec, "Adaptive radial basis function-generated finite differences method for contact problems," *International Journal for Numerical Methods in Engineering*, vol. 119, no. 7, pp. 661–686, 2019.
- [11] —, "On generation of node distributions for meshless pde discretizations," *SIAM Journal on Scientific Computing*, vol. 41, no. 5, pp. A3202–A3229, 2019.
- [12] M. Depolli, J. Slak, and G. Kosec, "Parallel domain discretization algorithm for RBF-FD and other meshless numerical methods for solving PDEs," *Computers & Structures*, vol. 264, p. 106773, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0045794922000335>
- [13] O. Davydov, D. T. Oanh, and N. M. Tuong, "Improved stencil selection for meshless finite difference methods in 3d," 2022.
- [14] N. Flyer, B. Fornberg, V. Bayona, and G. A. Barnett, "On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy," *Journal of Computational Physics*, vol. 321, pp. 21–38, 2016.
- [15] D. T. Oanh, O. Davydov, and H. X. Phu, "Adaptive rbf-fd method for elliptic problems with point singularities in 2d," *Applied Mathematics and Computation*, vol. 313, pp. 474–497, 2017.
- [16] J. Slak and G. Kosec, "Medusa: A c++ library for solving pdes using strong form mesh-free methods," *ACM Transactions on Mathematical Software (TOMS)*, vol. 47, no. 3, pp. 1–25, 2021.
- [17] V. Bayona, N. Flyer, B. Fornberg, and G. A. Barnett, "On the role of polynomials in rbf-fd approximations: II. numerical solution of elliptic pdes," *Journal of Computational Physics*, vol. 332, pp. 257–273, 2017.

4.2 Spatially-Varying Meshless Approximation Method Based on Node Regularity

Another approach for enhanced computational efficiency is based on the regularity of computational nodes. Uniform discretization is not only easier to implement but also allows the employment of the computationally cheap MON approximation. However, uniform discretization has two disadvantages. First, it makes it difficult to avoid discretization-related errors in the presence of irregularly shaped domain boundaries, and, second, employing h -adaptive or h -refined solutions becomes non-trivial. Scattered nodes, on the other hand, address both issues, but a more stable approximation method with generally higher computational complexity must be employed.

We therefore propose a hybrid uniform-scattered discretization, positioning scattered nodes only where necessary and uniform nodes elsewhere. On uniform nodes the MON approximation is used, while a stable RBF-FD approximation method is used on scattered nodes. The entire solution procedure employing spatially-variable discretization and correspondingly spatially-variable approximation methods is further discussed in the following publication.

Contributions.

This section represents contribution **C9**, proving that spatially-varying node regularity accompanied with spatially-varying approximation method can improve the computational efficiency with a negligible cost to the accuracy of the numerical solution.

Addressed hypotheses.

Our findings show that spatially-variable node regularity and accompanying spatially-variable approximation method can yield up to 50% shorter execution times with a negligible effect on the accuracy of the numerical solution. Demonstrated on the de Vahl Davis problem, hypothesis **H7** is finally confirmed.

Publications included in this section:

- M. Jančič, M. Rot, and G. Kosec, “Spatially-varying meshless approximation method for enhanced computational efficiency,” in *Computational Science – ICCS 2023*, J. Mikyška, C. de Mulatier, M. Paszynski, *et al.*, Eds., Cham: Springer Nature Switzerland, 2023, pp. 500–514, ISBN: 978-3-031-36027-5

Regarding my contribution: I made a literature overview of the topic, participated in the implementation of the solution procedure, planned and executed the analyses and jointly prepared the manuscript with co-authors.



Spatially-Varying Meshless Approximation Method for Enhanced Computational Efficiency

Mitja Jančič^{1,2(✉)} , Miha Rot^{1,2} , and Gregor Kosec¹

¹ Parallel and Distributed Systems Laboratory, Jožef Stefan Institute, Ljubljana, Slovenia

{mitja.jancic, miha.rot, gregor.kosec}@ijs.si

² Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Abstract. In this paper, we address a way to reduce the total computational cost of meshless approximation by reducing the required stencil size through spatially varying computational node regularity. Rather than covering the entire domain with scattered nodes, only regions with geometric details are covered with scattered nodes, while the rest of the domain is discretized with regular nodes. A simpler approximation using solely monomial basis can be used in regions covered by regular nodes, effectively reducing the required stencil size and computational cost compared to the approximation on scattered nodes where a set of polyharmonic splines is added to ensure convergent behaviour.

The performance of the proposed hybrid scattered-regular approximation approach, in terms of computational efficiency and accuracy of the numerical solution, is studied on natural convection driven fluid flow problems. We start with the solution of the de Vahl Davis benchmark case, defined on a square domain, and continue with two- and three-dimensional irregularly shaped domains. We show that the spatial variation of the two approximation methods can significantly reduce the computational demands, with only a minor impact on the accuracy.

Keywords: Collocation · RBF-FD · RBF · Meshless · Hybrid method · Fluid-flow · Natural convection · Numerical simulation

1 Introduction

Although the meshless methods are formulated without any restrictions regarding the node layouts, it is generally accepted that quasi-uniformly-spaced node sets improve the stability of meshless methods [12, 23]. Nevertheless, even with quasi-uniform nodes generated with recently proposed node positioning algorithms [14, 15, 18], a sufficiently large stencil size is required for stable approximation. A stencil with $n = 2\binom{m+d}{m}$ nodes is recommended [1] for the local Radial Basis Function-generated Finite differences (RBF-FD) [20] method in a d -dimensional domain for approximation order m . The performance of RBF-FD method—with approximation basis consisting of Polyharmonic splines (PHS)

and monomial augmentation with up to and including monomials of degree m —has been demonstrated with scattered nodes on several applications [8, 17, 24]. On the other hand, approximation on regular nodes can be performed with considerably smaller stencil [10] ($n = 5$ in two-dimensional domain) using only monomial basis.

Therefore, a possible way to enhance the overall computational efficiency and consider the discretization-related error is to use regular nodes far away from any geometric irregularities in the domain and scattered nodes in their vicinity. A similar approach, where the approximation method is spatially varied, has already been introduced, e.g., a hybrid FEM-meshless method [6] has been proposed to overcome the issues regarding the unstable Neumann boundary conditions in the context of meshless approximation. Moreover, the authors of [2, 5] proposed a hybrid of Finite Difference (FD) method employed on conventional cartesian grid combined with meshless approximation on scattered nodes. These hybrid approaches are computationally very efficient, however, additional implementation-related burden is required on the transition from cartesian to scattered nodes [9], contrary to the objective of this paper relying solely on the framework of meshless methods.

In this paper we experiment with such *hybrid scattered-regular* method with spatially variable stencil size on solution of natural convection driven fluid flow cases. The solution procedure is first verified on the reference de Vahl Davis case, followed by a demonstration on two- and three-dimensional irregular domains. We show that spatially varying the approximation method can have positive effects on the computational efficiency while maintaining the accuracy of the numerical solution.

2 Numerical Treatment of Partial Differential Equations

To obtain the hybrid scattered-regular domain discretization, we first fill the entire domain with regular nodes. A portion of this regular nodes is then removed in areas where a scattered node placement is desired, i.e., close to the irregular boundaries. Finally, the voids are filled with a dedicated node positioning algorithm [18] that supports variable nodal density and allows us to refine the solution near irregularities in the domain. This approach is rather naive but sufficient for demonstration purposes. A special hybrid fill algorithm is left for future work.

An example of an h -refined domain discretization is shown in Fig. 1. It is worth noting that the width of the scattered node layer δ_h is non-trivial and affects both the stability of the solution procedure and the accuracy of the numerical solution. Although we provide a superficial analysis of δ_h variation in Sects. 4.1 and 4.2, further work is warranted.

After the computational nodes $\mathbf{x}_i \in \Omega$ are obtained, the differential operators $\mathcal{L}$ can be locally approximated in point $\mathbf{x}_c$ over a set of n neighbouring nodes (stencil) $\{\mathbf{x}_i\}_{i=1}^n = \mathcal{N}$, using the following expression

$$(\mathcal{L}u)(\mathbf{x}_c) \approx \sum_{i=1}^n w_i u(\mathbf{x}_i). \quad (1)$$

502 M. Jančíč et al.

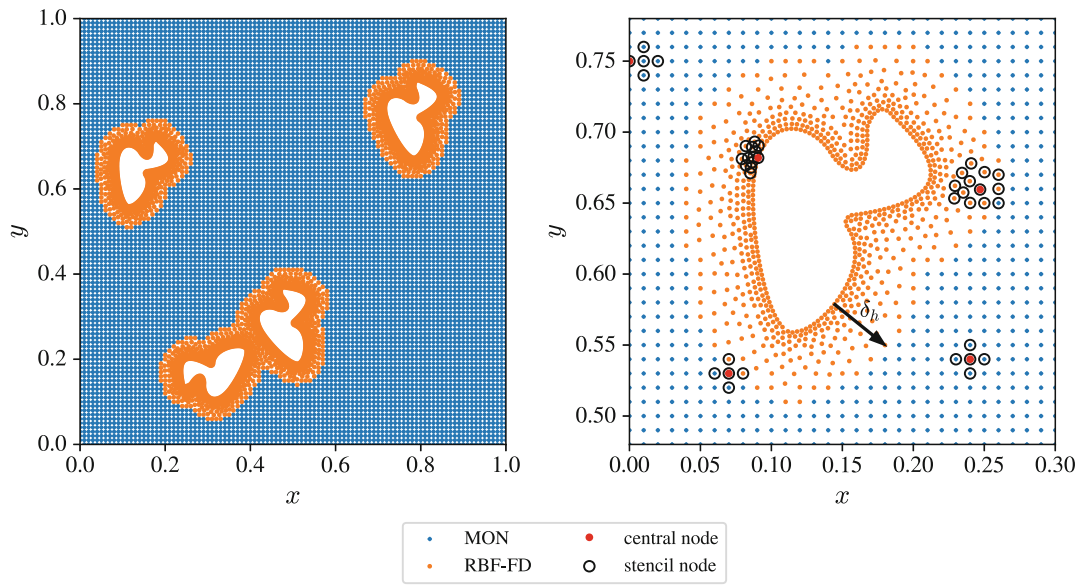


Fig. 1. Irregular domain discretization example (*left*) and spatial distribution of approximation methods along with corresponding example stencils (*right*).

The approximation (1) holds for an arbitrary function u and yet to be determined weights $\mathbf{w}$. To determine the weights, the equality of approximation (1) is enforced for a chosen set of basis functions. Here we will use two variants

- (i) a set of Polyharmonic splines (PHS) augmented with monomials to ensure convergent behaviour [1, 7], effectively resulting in a popular *radial basis function-generated finite differences* (RBF-FD) approximation method [20].
- (ii) a set of monomials centred at the stencil nodes that we will refer to as (MON) [10].

We use the least expensive MON with $2d + 1$ monomial basis functions¹ and the same number of support nodes in each approximation stencil. This setup is fast, but only stable on regular nodes [10, 16]. For the RBF-FD part, we also resort to the minimal configuration required for 2nd-order operators, i.e., 3rd-order PHS augmented with all monomials up to the 2nd-order ($m = 2$). According to the standard recommendations [1], this requires a stencil size of $n = 2 \binom{m+d}{m}$.

Note the significant difference between stencil sizes—5 vs. 12 nodes in 2D—that only increases in higher dimensions (7 vs. 30 in 3D). This results both in faster computation of the weights $\mathbf{w}$ —an $\mathcal{O}(N^3)$ ² operation performed only once for each stencil—and in faster evaluation for the $\mathcal{O}(n)$ explicit operator approx-

¹ In 2D, the 5 basis functions are $\{1, x, y, x^2, y^2\}$. The xy term is not required for regularly placed nodes and its omission allows us to use the smaller and completely symmetric 5-node stencil.

² $N_{\text{RBF-FD}} \sim 3N_{\text{MON}}$ due to the larger stencil size and the extra PHS in the approximation basis.

imation (1) performed many times during the explicit time stepping. Therefore, a spatially varying node regularity can have desirable consequences on the discretization-related errors and computational efficiency of the solution procedure.

2.1 Computational Stability

By enforcing the equality of approximation (1), we obtain a linear system $\mathbf{M}\mathbf{w} = \ell$. Solving the system provides us with the approximation weights $\mathbf{w}$, but the stability of such procedure can be uncertain and is usually estimated via the condition number $\kappa(\mathbf{M}) = \|\mathbf{M}\| \|\mathbf{M}^{-1}\|$ of matrix $\mathbf{M}$, where $\|\cdot\|$ denotes the L^2 norm.

A spatial distribution of condition numbers is shown in Fig. 2. It can be observed that the RBF-FD approximation method generally results in higher condition numbers than the MON approach. This could be due to the fact that the matrices $\mathbf{M}$ for the RBF-FD part are significantly larger and based on scattered nodes. Nevertheless, it is important to note that the transition from regular to scattered nodes does not appear to affect the conditionality of the matrices.

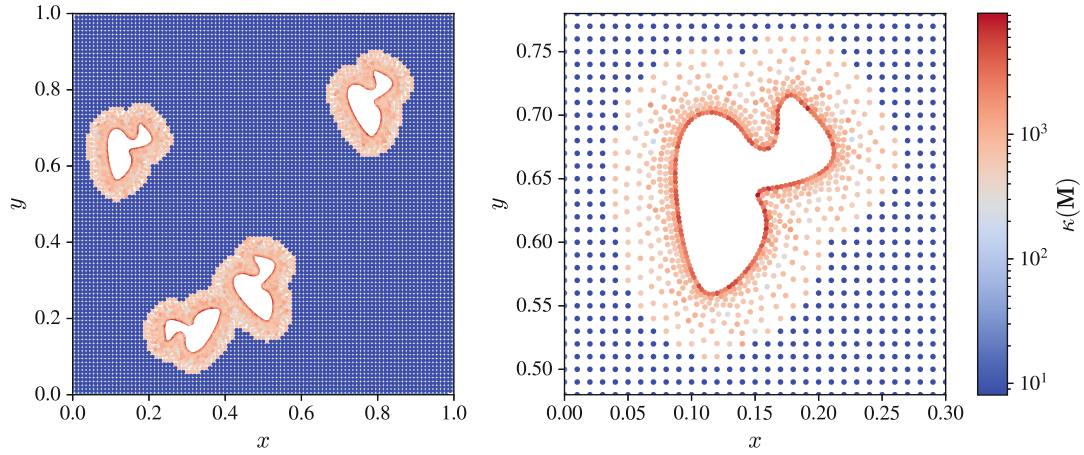


Fig. 2. Condition numbers $\kappa(\mathbf{M})$ for the Laplacian operator: entire computational domain (*left*) and a zoomed-in section around the irregularly shaped obstacle (*right*).

2.2 Implementation Details

The entire solution procedure employing the hybrid scattered-regular method is implemented in C++. The project's implementation³ is strongly dependent on our in-house developed meshless C++ framework *Medusa library* [19] supporting all building blocks of the solution procedure, i.e., differential operator approximations, node positioning algorithms, etc.

³ Source code is available at http://gitlab.com/e62Lab/public/2023_cp_iccs_hybrid_nodes under tag *v1.1*.

504 M. Jančíč et al.

We used `g++ 11.3.0` for Linux to compile the code with `-O3 -DNDEBUG` flags on Intel(R) Xeon(R) CPU E5520 computer. To improve the timing accuracy we run the otherwise parallel code in a single thread with the CPU frequency fixed at 2.27 GHz, disabled boost functionality and assured CPU affinity using the `taskset` command. Post-processing was done using Python 3.10.6 and Jupyter notebooks, also available in the provided git repository.

3 Governing Problem

To objectively assess the advantages of the hybrid method, we focus on the natural convection problem that is governed by a system of three PDEs that describe the continuity of mass, the conservation of momentum and the transfer of heat

$$\nabla \cdot \mathbf{v} = 0, \quad (2)$$

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \nabla \cdot (\text{Pr} \nabla \mathbf{v}) - \text{RaPr} \mathbf{g} T_{\Delta}, \quad (3)$$

$$\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T = \nabla \cdot (\nabla T), \quad (4)$$

where a dimensionless nomenclature using Rayleigh (Ra) and Prandtl (Pr) numbers is used [11, 22].

The temporal discretization of the governing equations is solved with the explicit Euler time stepping where we first update the velocity using the previous step temperature field in the Boussinesq term [21]. The pressure-velocity coupling is performed using the Chorin's projection method [3] under the premise that the pressure term of the Navier-Stokes equation can be treated separately from other forces and used to impose the incompressibility condition. The time step is a function of internodal spacing h , and is defined as $dt = 0.1 \frac{h^2}{2}$ to assure stability.

4 Numerical Results

The governing problem presented in Sect. 3 is solved on different geometries employing (i) MON, (ii) RBF-FD and (iii) their spatially-varying combination. The performance of each approach is evaluated in terms of accuracy of the numerical solution and execution times. Unless otherwise specified, the MON method is employed using the monomial approximation basis omitting the mixed terms, while the RBF-FD approximation basis consists of Polyharmonic splines or order $k = 3$ augmented with monomials up to and including order $m = 2$.

4.1 The de Vahl Davis Problem

First, we solve the standard de Vahl Davis benchmark problem [22]. The main purpose of solving this problem is to establish confidence in the presented solution procedure and to shed some light on the behaviour of considered approximation methods, the stability of the solution procedure and finally on the computational efficiency. Furthermore, the de Vahl Davis problem was chosen as the basic test case, because the regularity of the domain shape allows us to efficiently discretize it using exclusively scattered or regular nodes and compare the solutions to that obtained with the hybrid scattered-regular discretization.

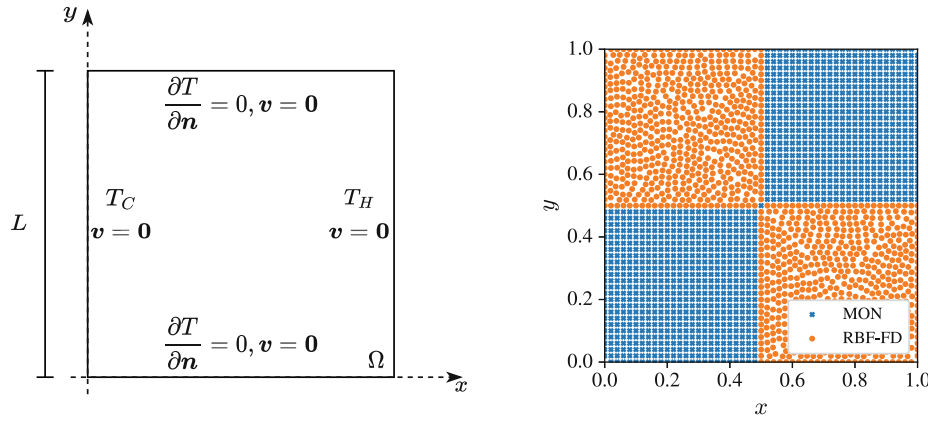


Fig. 3. The de Vahl Davis sketch (*left*) and example hybrid scattered-regular domain discretization (*right*).

For a schematic representation of the problem, see Fig. 3 (*left*). The domain is a unit box $\Omega = [0, 1] \times [0, 1]$, where the left wall is kept at a constant temperature $T_C = -0.5$, while the right wall is kept at a higher constant temperature $T_H = 0.5$. The upper and lower boundaries are insulated, and no-slip boundary condition for velocity is imposed on all walls. Both the velocity and temperature fields are initially set to zero.

To test the performance of the proposed hybrid scattered-regular approximation method, we divide the domain Ω into quarters, where each quarter is discretized using either scattered or regular nodes – see Fig. 3 (*right*) for clarity.

An example solution for $Ra = 10^6$ and $Pr = 0.71$ at a dimensionless time $t = 0.15$ with approximately $N = 15\,800$ discretization nodes is shown in Fig. 4.

We use the Nusselt number—the ratio between convective and conductive heat transfer—to determine when a steady state has been reached and as a convenient scalar value for comparison with reference solutions. In the following analyses, the average Nusselt number ($\overline{Nu}$) is calculated as the average of the Nusselt values at the cold wall nodes

$$Nu = \frac{L}{T_H - T_C} \left| \frac{\partial T}{\partial \mathbf{n}} \right|_{x=0}. \quad (5)$$

506 M. Jančíč et al.

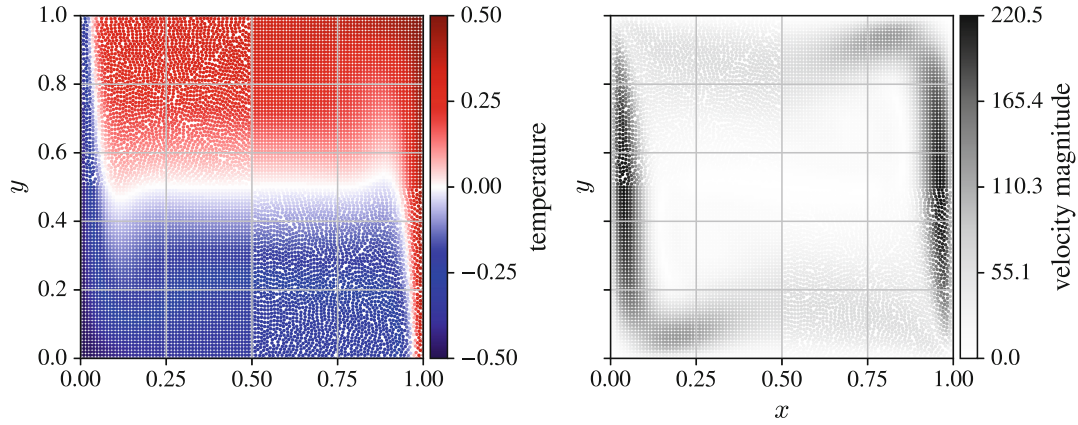


Fig. 4. Example solution at the stationary state. Temperature field (*left*) and velocity magnitude (*right*).

Its evolution over time is shown in Fig. 5. In addition, three reference results are also added to the figure. We are pleased to see that our results are in good agreement with the reference solutions from the literature.

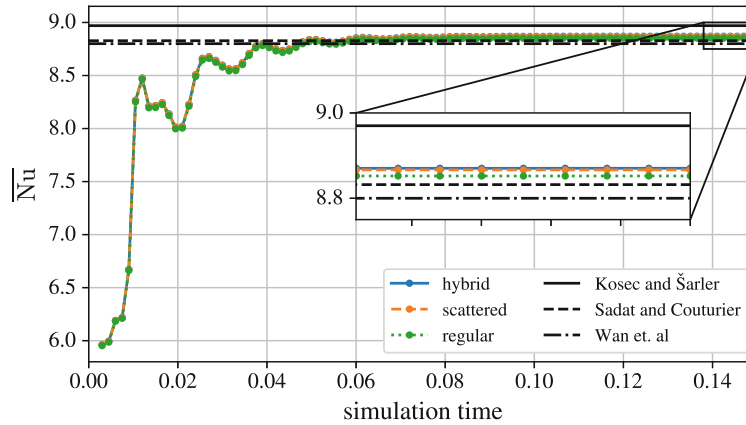


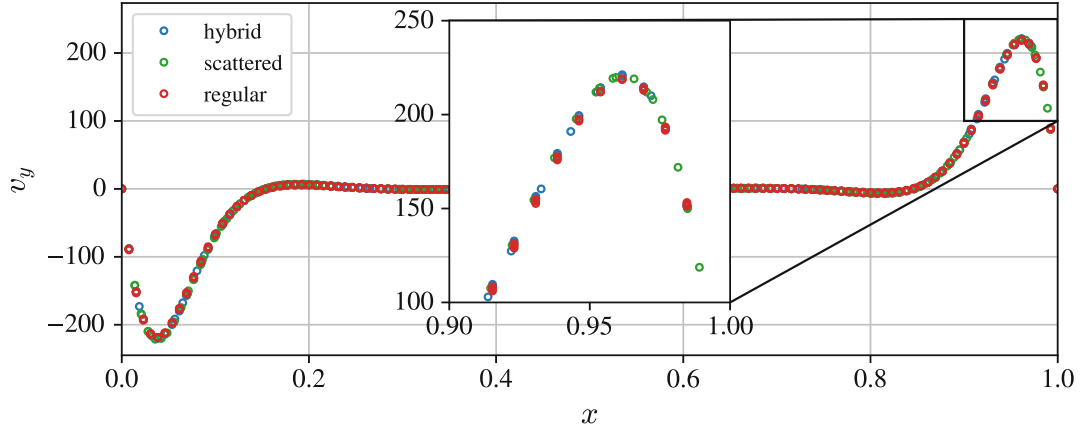
Fig. 5. Time evolution of the average Nusselt number along the cold edge calculated with the densest considered discretization. Three reference results Kosec and Šarler [11], Sadat and Couturier [13] and Wan et. al. [4] are also added.

Moreover, Fig. 5 also shows the time evolution of the average Nusselt number value for cases where the entire domain is discretized using either scattered or regular nodes. We find that all—hybrid, purely scattered and purely regular domain discretizations—yield results in good agreement with the references. More importantly, the hybrid method shows significantly shorter computational time (about 50%) than that required by the scattered discretization employing RBF-FD, as can be seen in Table 1 for the densest considered discretization with $h = 0.00398$.

Table 1. Average Nusselt along the cold edge along with execution times and number of discretization nodes.

Approximation	$\overline{Nu}$	execution time [h]	N
scattered	8.867	6.23	55 477
regular	8.852	2.42	64 005
hybrid	8.870	3.11	59 694
Kosec and Šarler (2007) [11]	8.97	/	10201
Sadat and Couturier (2000) [13]	8.828	/	22801
Wan et. al. (2001) [4]	8.8	/	10201

To further validate the hybrid method, we show in Fig. 6 the vertical component of the velocity field across the section $y = 0.5$. It is important to observe that the results for the hybrid, scattered and regular approaches overlap, which means that the resulting velocity fields for the three approaches are indeed comparable.

**Fig. 6.** Vertical velocity component values at nodes close to the vertical midpoint of the domain, i.e., $|y - 0.5| \leq h$ for purely scattered, purely regular and hybrid discretizations.

As a final remark, we also study the convergence of the average Nusselt number with respect to the number of discretization nodes in Fig. 7 (*left*), where we confirm that all our discretization strategies converge to a similar value that is consistent with the reference values. Moreover, to evaluate the computational efficiency of the hybrid approach, the execution times are shown on the right. Note that the same values for h were used for all discretization strategies and the difference in the total number of nodes is caused by the lower density of scattered nodes at the same internodal distance.

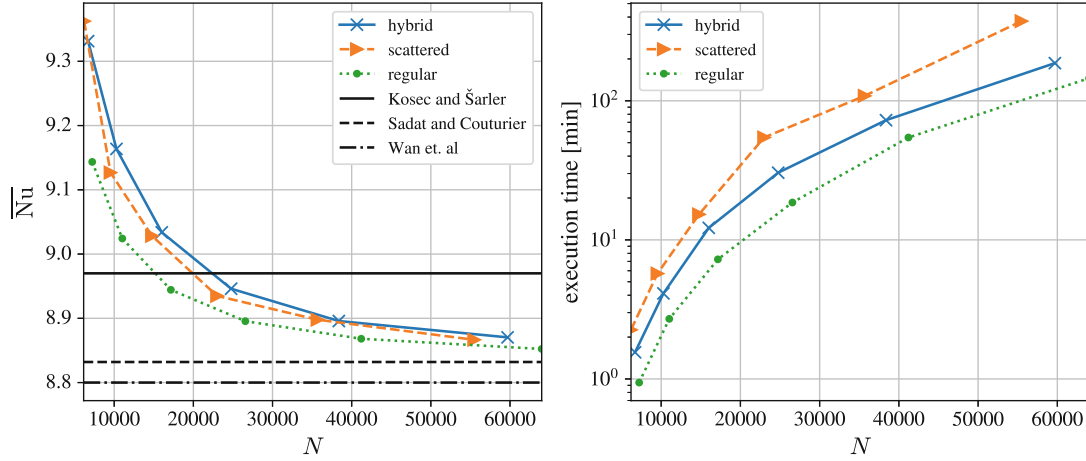


Fig. 7. Convergence of average Nusselt number with respect to discretization quality (*left*) and corresponding execution times (*right*).

The Effect of the Scattered Nodes Layer Width δ_h . To study the effect of the width of the scattered node layer δ_h , we consider two cases. In both cases, the domain from Fig. 3 is split into two parts at a distance $h\delta_h$ from the origin in the lower left corner. In the first scenario, the split is horizontal, resulting in scattered nodes below the imaginary split and regular nodes above it. In the second scenario, the split is vertical, resulting in scattered nodes to the left of it and regular nodes to the right of it. In both cases, the domain is discretized with purely regular nodes when $h\delta_h = 0$ and with purely scattered nodes when $h\delta_h = L$.

In Fig. 8, we show how the width of the scattered node layer affects the average Nusselt number in stationary state for approximately 40 000 discretization nodes. It is clear that even the smallest values of δ_h yield satisfying results. However, it is interesting to observe that the accuracy is significantly affected when the boundary between regular and scattered nodes runs across the region with the largest velocity magnitudes, i.e., the first and last couple of vertical split data points in Fig. 8.

4.2 Natural Convection on Irregularly Shaped Domains

In the previous section we demonstrated that the hybrid scattered-regular approximation method is computationally more efficient than the pure RBF-FD approximation with only minor differences in the resulting fields. However, to truly exploit the advantages of the hybrid method, irregular domains must be studied. Therefore, in this section, the hybrid scattered-regular approach is

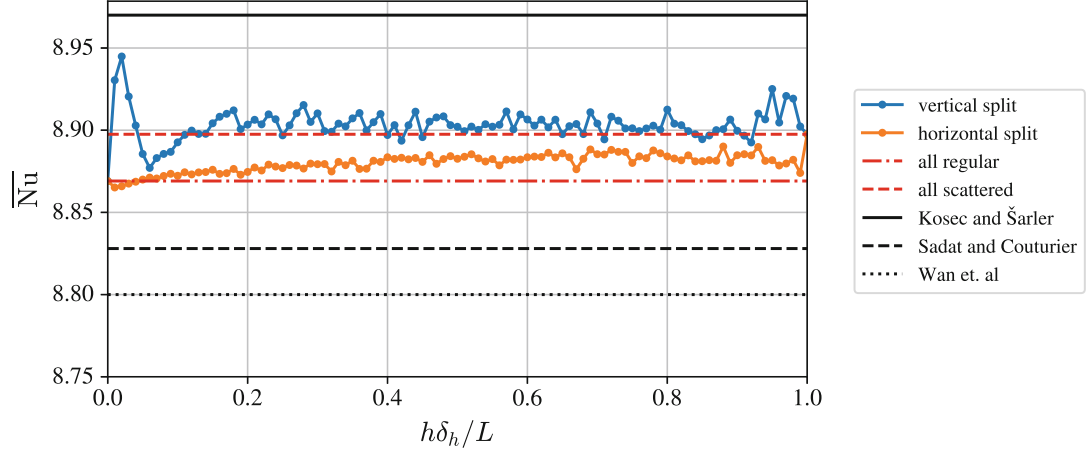


Fig. 8. Demonstration of the scattered node layer width (δ_h) effect on the accuracy of the numerical solution.

employed on an irregularly shaped domain. Let the computational domain Ω be a difference between the two-dimensional unit box $\Omega = [0, 1] \times [0, 1]$ and 4 randomly positioned duck-shaped obstacles introducing the domain irregularity.

The dynamics of the problem are governed by the same set of Eqs. (2–4) as in the previous section. This time, however, all the boundaries of the box are insulated. The obstacles, on the other hand, are subject to Dirichlet boundary conditions, with half of them at $T_C = 0$ and the other half at $T_H = 1$. The initial temperature is set to $T_{\text{init}} = 0$.

We have chosen such a problem because it allows us to further explore the advantages of the proposed hybrid scattered-regular discretization. Generally speaking, the duck-shaped obstacles within the computational domain represent an arbitrarily complex shape that requires scattered nodes for accurate description, i.e., reduced discretization-related error. Moreover, by using scattered nodes near the irregularly shaped domain boundaries, we can further improve the local field description in their vicinity by employing a h -refined discretization. Specifically, we employ h -refinement towards the obstacles with linearly decreasing internodal distance from $h_r = 0.01$ (regular nodes) towards $h_s = h_r/3$ (irregular boundary) over a distance of $h_r\delta_h$. The refinement distance and the width of the scattered node layer are the same, except in the case of fully scattered discretization. Such setup effectively resulted in approximately $N = 11\,600$ computational nodes ($N_s = 3149$ scattered nodes and $N_r = 8507$ regular nodes), as shown in Fig. 1 for a scattered node layer width $\delta_h = 4$. Note that the time step is based on the smallest h , i.e., $dt = 0.1 \frac{h_s^2}{2}$.

510 M. Jančíč et al.

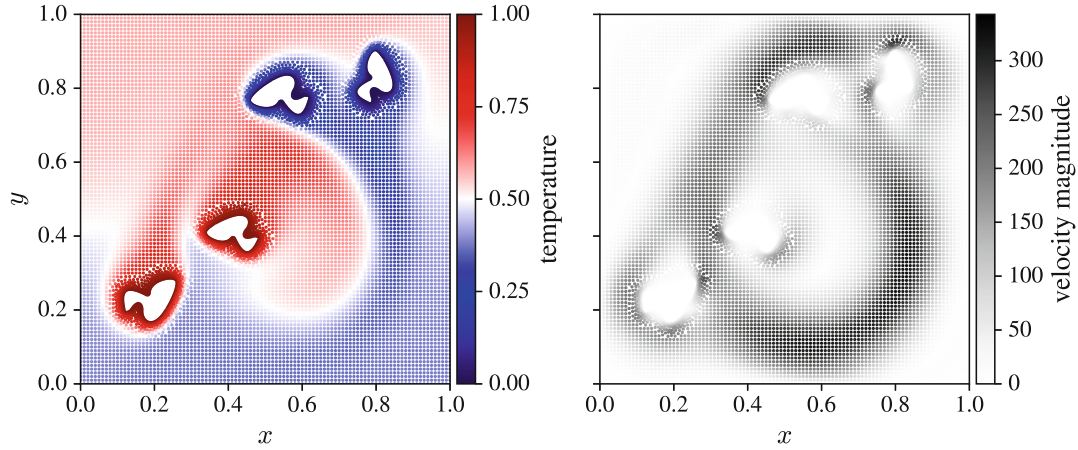


Fig. 9. Example solution on irregular domain. Temperature field (*left*) and velocity magnitude (*right*).

Figure 9 shows an example solution for an irregularly shaped domain. The hybrid scattered-regular solution procedure was again able to obtain a reasonable numerical solution.

Furthermore, Fig. 10 (*left*) shows the average Nusselt number along the cold duck edges where we can observe that a stationary state has been reached. The steady state values for all considered discretizations match closely but it is interesting to note that in the early stage of flow formation, the fully scattered solutions with different refinement distance δ_h differ significantly more than the hybrid and the fully scattered solutions with the same refinement strategy.

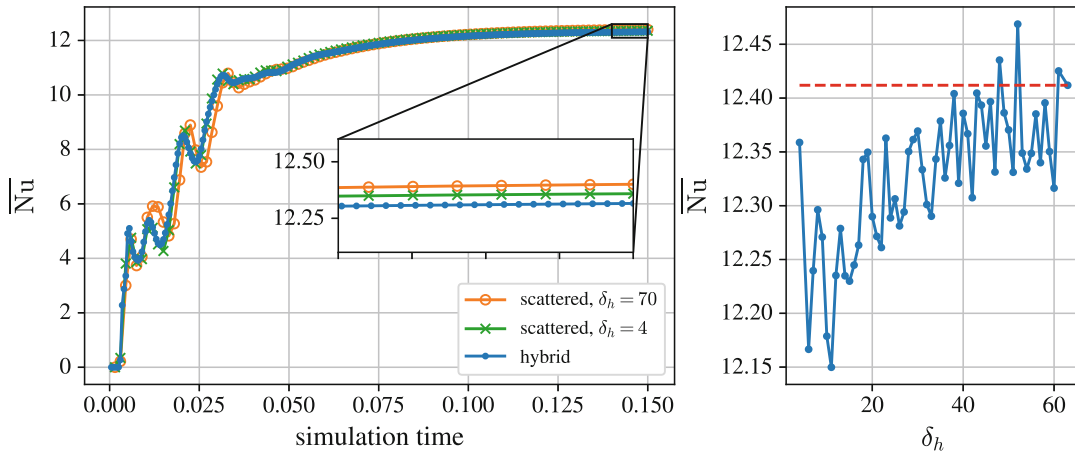


Fig. 10. Time evolution of the average Nusselt number calculated on the cold duck-shaped obstacles of an irregularly shaped domain. (*left*) Changes in stationary state average Nusselt number as the scattered node layer width δ_h increases. (*right*)

It is perhaps more important to note that the execution times gathered in Table 2 show that the hybrid method effectively reduces the execution time for approximately 35 %. The pure regular discretization with MON approximation is omitted from the table because a stable numerical solution could not be obtained.

Table 2. Average Nusselt along the cold duck edges along with execution times. Note that all values in the table were obtained for $\delta_h = 4$.

Approximation	$\overline{\text{Nu}}$	execution time [min]	N
scattered	12.32	46.31	10 534
hybrid	12.36	29.11	11 535

The Effect of the Scattered Nodes Layer Width δ_h . To justify the use of $\delta_h = 4$, we show in Fig. 10 (*right*) that the average value of the Nusselt number at steady state for different values of δ_h . In the worst case, the difference is <2%, justifying the use of the computationally cheaper smaller δ_h . Note that in this particular domain setup, $\delta_h > 64$ already yields a purely scattered domain discretization, while the minimum stable value is $\delta_h = 4$. Note also that the general increase of the Nusselt number with respect to the width of the scattered node layers δ_h may also exhibit other confounding factors. An increase in δ_h leads to a finer domain discretization due to a more gradual refinement, i.e., a fully scattered discretization using $\delta_h = 70$ results in about 35 000 discretization points compared to 11 600 at $\delta_h = 4$, while a decrease in δ_h leads to a more aggressive refinement that could also have a negative effect. This can be supported by observing the difference between the results for the two fully scattered discretizations in Fig. 10.

4.3 Application to Three-Dimensional Irregular Domains

As a final demonstrative example, we employ the proposed hybrid scattered-regular approximation method on a three-dimensional irregular domain. The computational domain Ω is a difference between the three-dimensional unit box $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ and 4 randomly positioned and sized spheres introducing the domain irregularity.

The dynamics are governed by the same set of Eqs. (2–4) as in the two-dimensional case from Sect. 4.2. To improve the quality of the local field description near the irregularly shaped domain boundaries, h -refinement is employed with a linearly decreasing internodal distance from $h_r = 0.025$ (regular nodes) towards $h_s = h_r/2$ (spherical boundaries). Two spheres were set to a constant temperature $T_C = 0$ and the remaining two to $T_H = 1$. The Rayleigh number was set to 10^4 .

Although difficult to visualize, an example solution is shown in Fig. 11. Using the hybrid scattered-regular domain discretization, the solution procedure was again able to obtain a reasonable numerical solution.

512 M. Jančíč et al.

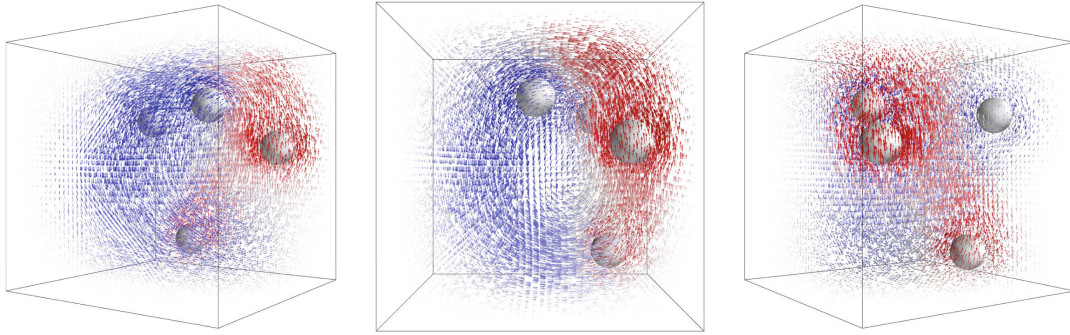


Fig. 11. Example solution viewed from three different angles. The arrows show the velocity in computational nodes and are coloured according to the temperature in that node. The values range from dark blue for T_C to dark red for T_H . For clarity, only a third of the nodes is visualized. (Color figure online)

Note that the scattered method took about 48 h and the hybrid scattered-regular approximation method took 20 h to simulate 1 dimensionless time unit with the dimensionless time step $dt = 7.8125 \cdot 10^{-6}$ and about 75 000 computational nodes with $\delta_h = 4$. For clarity, the data is also gathered in Table 3. Note that the pure regular discretization with MON approximation is again omitted from the table because a stable numerical solution could not be obtained.

Table 3. Average Nusselt along the cold spheres, execution time, and number of computational nodes.

Approximation	$\overline{Nu}$	execution time [h]	N
scattered	7.36	48.12	65 526
hybrid	6.91	20.54	74 137

5 Conclusions

We proposed a computationally efficient approach to the numerical treatment of problems in which most of the domain can be efficiently discretized with regularly positioned nodes, while scattered nodes are used near irregularly shaped domain boundaries to reduce the discretization-related errors. The computational effectiveness of the spatially-varying approximation method, employing FD-like approximation on regular nodes and RBF-FD on scattered nodes, is demonstrated on a solution to a two-dimensional de Vahl Davis natural convection problem.

We show that the proposed hybrid method, can significantly improve the computational efficiency compared to the pure RBF-FD approximation, while

introducing minimal cost to the accuracy of the numerical solution. A convergence analysis from Fig. 7 shows good agreement with the reference de Vahl Davis solutions.

In the continuation, the hybrid method is applied to a more general natural convection problem in two- and three-dimensional irregular domains, where the elegant mathematical formulation of the meshless methods is further exposed by introducing h -refinement towards the irregularly shaped obstacles. In both cases, the hybrid method successfully obtained the numerical solution and proved to be computationally efficient, with execution time gains nearing 50 %.

Nevertheless, the scattered node layer width and the aggressiveness of h -refinement near the irregularly shaped domain boundaries should be further investigated, as both affect the computational efficiency and stability of the solution procedure. In addition, future work should also include more difficult problems, such as mixed convection problems and a detailed analysis of possible surface effects, e.g. scattering, at the transition layer between the scattered and regular domains.

Acknowledgements. The authors would like to acknowledge the financial support of Slovenian Research Agency (ARRS) in the framework of the research core funding No. P2-0095, the Young Researcher program PR-10468 and research project J2-3048.

Conflict of Interest. The authors declare that they have no conflict of interest. All the co-authors have confirmed to know the submission of the manuscript by the corresponding author.

References

1. Bayona, V., Flyer, N., Fornberg, B., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs. *J. Comput. Phys.* **332**, 257–273 (2017)
2. Bourantas, G., et al.: Strong-form approach to elasticity: hybrid finite difference-meshless collocation method (FDMCM). *Appl. Math. Model.* **57**, 316–338 (2018). <https://doi.org/10.1016/j.apm.2017.09.028>
3. Chorin, A.J.: Numerical solution of the Navier-Stokes equations. *Math. Comput.* **22**(104), 745–762 (1968)
4. Wan, D.C., Patnaik, B.S.V., Wei, G.W.: A new benchmark quality solution for the buoyancy-driven cavity by discrete singular convolution. *Numer. Heat Transfer Part B: Fundam.* **40**(3), 199–228 (2001). <https://doi.org/10.1080/104077901752379620>
5. Ding, H., Shu, C., Yeo, K., Xu, D.: Simulation of incompressible viscous flows past a circular cylinder by hybrid FD scheme and meshless least square-based finite difference method. *Comput. Methods Appl. Mech. Eng.* **193**(9–11), 727–744 (2004)
6. El Kadmiri, R., Belaasilia, Y., Timesli, A., Kadiri, M.S.: A hybrid algorithm using the fem-meshless method to solve nonlinear structural problems. *Eng. Anal. Boundary Elem.* **140**, 531–543 (2022). <https://doi.org/10.1016/j.enganabound.2022.04.018>

514 M. Jančič et al.

7. Flyer, N., Fornberg, B., Bayona, V., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. *J. Computat. Phys.* **321**, 21–38 (2016)
8. Fornberg, B., Flyer, N.: *A Primer on Radial Basis Functions with Applications to the Geosciences*. SIAM (2015)
9. Javed, A., Djidjeli, K., Xing, J., Cox, S.: A hybrid mesh free local RBF-cartesian FD scheme for incompressible flow around solid bodies. *Int. J. Math. Comput. Nat. Phys. Eng.* **7**, 957–966 (2013)
10. Kosec, G.: A local numerical solution of a fluid-flow problem on an irregular domain. *Adv. Eng. Softw.* **120**, 36–44 (2018)
11. Kosec, G., Šarler, B.: Solution of thermo-fluid problems by collocation with local pressure correction. *Int. J. Numer. Methods Heat Fluid Flow* **18**, 868–882 (2008)
12. Liu, G.R.: *Meshfree Methods: Moving Beyond the Finite Element Method*. CRC Press (2009)
13. Sadat, H., Couturier, S.: Performance and accuracy of a meshless method for laminar natural convection. *Numer. Heat Transfer Part B: Fundam.* **37**(4), 455–467 (2000). <https://doi.org/10.1080/10407790050051146>
14. van der Sande, K., Fornberg, B.: Fast variable density 3-D node generation. *SIAM J. Sci. Comput.* **43**(1), A242–A257 (2021)
15. Shankar, V., Kirby, R.M., Fogelson, A.L.: Robust node generation for meshfree discretizations on irregular domains and surfaces. *SIAM J. Sci. Comput.* **40**(4), 2584–2608 (2018). <https://doi.org/10.1137/17m114090x>
16. Slak, J., Kosec, G.: Refined meshless local strong form solution of Cauchy-Navier equation on an irregular domain. *Eng. Anal. Boundary Elem.* **100**, 3–13 (2019). <https://doi.org/10.1016/j.enganabound.2018.01.001>
17. Slak, J., Kosec, G.: Adaptive radial basis function-generated finite differences method for contact problems. *Int. J. Numer. Meth. Eng.* **119**(7), 661–686 (2019). <https://doi.org/10.1002/nme.6067>
18. Slak, J., Kosec, G.: On generation of node distributions for meshless PDE discretizations. *SIAM J. Sci. Comput.* **41**(5), A3202–A3229 (2019)
19. Slak, J., Kosec, G.: Medusa: a C++ library for solving PDEs using strong form mesh-free methods. *ACM Trans. Math. Softw. (TOMS)* **47**(3), 1–25 (2021)
20. Tolstykh, A., Shirobokov, D.: On using radial basis functions in a “finite difference mode” with applications to elasticity problems. *Comput. Mech.* **33**(1), 68–79 (2003)
21. Tritton, D.J.: *Physical Fluid Dynamics*. Oxford Science Publ, Clarendon Press (1988). <https://doi.org/10.1007/978-94-009-9992-3>
22. de Vahl Davis, G.: Natural convection of air in a square cavity: a bench mark numerical solution. *Int. J. Numer. Meth. Fluids* **3**(3), 249–264 (1983)
23. Wendland, H.: *Scattered Data Approximation*, vol. 17. Cambridge University Press (2004)
24. Zamolo, R., Nobile, E.: Solution of incompressible fluid flow problems with heat transfer by means of an efficient RBF-FD meshless approach. *Numer. Heat Transf. Part B: Fundam.* **75**(1), 19–42 (2019)

Chapter 5

Conclusions and Future Work

In this thesis, we studied some approaches toward efficient solution procedures for solving PDEs. We began by investigating the fundamental properties of two commonly used approximation methods with the ability to operate on scattered nodes: the RBF-FD and the WLS approximation methods. Between the two, the RBF-FD method showed superior performance by providing more accurate solutions and greater stability of the solution procedure, especially when dealing with high-order approximations. We also showed that the approximation order is easily controlled by the highest monomial degree in the approximation basis, and found that there exists an optimal setup in terms of execution time and targeted solution accuracy. Our studies also include the stencil size influence on the performance of the linear differential operator approximation in terms of the approximation accuracy.

In the second step of our research, we proposed to spatially vary the approximation order using an a priori defined approximation-order-distribution. We then developed an original IMEX error indicator based on high-order approximations, which made it possible to implement and study a meshless *hp*-adaptive solution procedure – a procedure that simultaneously adjusts the spatial discretization resolution and the approximation order for more efficient PDE-solving solution procedures.

In the last chapter, we proposed an adaptive solution procedure based on spatially varying the approximation method. We demonstrated the advantages of a hybrid WLS–RBF-FD approximation method and extended the idea to spatially-variable node regularity employing the RBF-FD approximation on scattered nodes and the MON approximation on uniform nodes. Both approaches were again able to improve the efficiency of the PDE-solving solution procedure.

While we have successfully identified and demonstrated some opportunities to modify the solution procedures for an efficient PDE-solving, in the process, we have also observed many possibilities for future work. Conclusions and some opportunities for future work are discussed in the following sections.

5.1 Summary of Conclusions

Fundamentals of Meshless Approximation

Our research began with a series of performance analyses of different meshless approximation methods, where we observed several important characteristics in the process. In particular, we focused on the behaviour of high-order approximations on scattered nodes.

We have shown that there exists an optimal augmenting monomial degree in the RBF-FD approximation basis in terms of the computational complexity of the solution procedure

and the target accuracy of the numerical solution. This allowed us to provide guidelines for monomial augmentation to achieve the targeted accuracy of the numerical solution in a time-efficient manner. In the process, we also demonstrated that the order of approximation can easily be controlled by the highest augmenting monomial degree – a property that is later crucial for the implementation of an *hp*-adaptive solution procedure.

The fact that the stencil size can notably affect both the stability of the solution procedure and the accuracy of the numerical solution has already been observed by Bayona. Our contribution to this topic is based on the observation that the Bayona’s recommended stencil size $n = 2\binom{m+d}{d}$ for a stable RBF-FD approximation is by no means optimal. We have shown that already a small change in the stencil size can lead to numerical solutions that are almost an order of magnitude more accurate. In doing so, we observed an interesting oscillatory behaviour of the approximation error. Although this observation is not yet fully understood, we believe it could open up new opportunities toward optimal stencil size research.

Another contribution that falls under the general understanding of meshless approximations was obtained by directly comparing the performance of the RBF-FD and WLS approximations. Our analyses showed that the larger approximation basis in RBF-FD yields more stable and more accurate high-order approximations, while low-order approximations are better obtained with the WLS approximation. This observation made the RBF-FD approximation a better candidate for the development of *hp*-adaptive solution procedure, where stable and accurate high-order approximations are crucial.

***hp*-Adaptive Solution Procedure**

To the best of our knowledge, the research in this thesis represents the first true *hp*-adaptive solution procedure employing meshless methods. Although some aspects are still in need of further research, we were able to draw several conclusions from it.

First, we have shown that a spatially varying approximation order leading to *p*-refined numerical solutions has desirable effects toward efficient PDE-solving solution procedures. Enforcing high-order approximations in domain regions where the error of the numerical solution is high locally reduces the error of the numerical solution and thus has a similar effect to improving the spatial discretization. We demonstrated that a well-thought-out spatial distribution of high-order approximations can reduce the ℓ_∞ -norm of the solution’s error, while minimally increasing the computational complexity as a consequence of the high-order approximations requiring larger stencil sizes.

The drawback of the proposed *p*-refined solution procedure is that the spatial distribution of the approximation order was defined a priori, contrary to adaptive solution procedures that aim to eliminate the need for human intervention in the solution procedure by automatically identifying the domain regions with high error of the numerical solution. A commonly used tool for identifying regions in need of refinement is an error indicator. We have proposed an original error indicator based on the high order approximations. The proposed IMEX error indicator makes use of the implicitly obtained numerical solution and explicit operators (approximated by a higher order approximation basis) to reconstruct the right-hand side of the governing PDE. The true and the reconstructed right-hand sides are then compared. The idea is similar to the one behind the ZZ-type indicators, where the deviation of the recovered high-order solution from the computed solution characterises the error. We have used a Poisson problem with an exponentially strong source within the domain to demonstrate that the proposed IMEX error indicator is able to successfully identify the regions with high solution error.

Equipped with the tool for automated localization of domain regions with lowest accuracy of the numerical solution, we then had all the building blocks required for the

implementation of a *hp*-adaptive solution procedure. We chose to work with the RBF-FD approximations and grounded the proposed *hp*-adaptive solution procedure on a well-established *solve-estimate-mark-refine* paradigm. In the solve stage, the implicit solution of the governing problem is obtained using the RBF-FD approximation. In the estimate stage, the IMEX error indicator is used to identify the regions with the likely large numerical solution error. Afterwards, the marking strategy is used in the mark stage to (i) define which nodes are subject to (de-)refinement and (ii) which type of adaptivity should be invoked, i.e. h or p . In our work, the marking strategy is based on the Texas Three Step algorithm, which is a very easy-to-implement strategy that gives good enough results for demonstration purposes, but is conceptually flawed to achieve more optimal results. This is particularly evident in the proximity of singularities, where a better spatial discretization should be preferred. Instead, the proposed marking strategy also forces higher approximation orders. Nevertheless, with the proposed solution procedure we were able to clearly demonstrate the advantages of simultaneously varying the discretization quality and the approximation order on a set of two- and three-dimensional linear elasticity problems.

During our implementation of the *hp*-adaptive solution procedure, we also conducted a brief study on the efficiency of the final large sparse system. We studied different solvers (direct and iterative) in terms of the solver error and execution times and found that the direct solvers are generally faster, but require significantly more memory for large domains. Therefore, we chose to work with the general-purpose iterative BiCGSTAB solver to avoid memory-related problems, but at the cost of longer execution times.

Spatially-Adaptive Approximation Methods

Through our research, we have found another opportunity to enhance the efficiency of PDE-solving solution procedures. Specifically, we have proposed to spatially vary the approximation method. While such procedures have already been proposed by other researchers, to the best of our knowledge, our attempt – which is entirely based on the framework of meshless methods – does not require any special treatment on the transition between one or more approximation methods.

For instance, in the first chapter, we concluded that RBF-FD is more stable than WLS, but WLS is computationally cheaper. This observation led us to the idea of using a hybrid approximation method that combines the advantages of both methods. In our implementation, we used RBF-FD in regions where stability was important and WLS in regions where stability issues were not expected. The proposed hybrid WLS–RBF-FD approximation method was tested on a Poisson problem and showed improvements in the efficiency of the PDE-solving solution procedure while maintaining the accuracy of the numerical solution. The performance was then also demonstrated on a three-dimensional contact problem, where we showed that the pure WLS approximation is not stable enough to obtain a numerical solution, while the hybrid approximation method is stable enough and reduces the wall-clock times by about 33%, compared to the pure RBF-FD method.

In the second approach, our attention shifted to the second-order RBF-FD approximations with a recommended support size $n = 12$ in two-dimensional domains and to the cost-effective MON approximation, which uses only monomials in the approximation basis and requires a smaller stencil size of $n = 5$. While RBF-FD can operate with scattered nodes, the stability of MON is limited to uniform nodes. This led us to the idea of using a hybrid scattered-uniform discretization that employs RBF-FD approximation on scattered nodes and the MON approximation on uniform nodes. Here, scattered nodes are positioned in such a way that the irregularly shaped domain boundaries are accurately discretized and local application of h -refinement is possible, while uniform nodes are positioned elsewhere. The proposed hybrid scattered-uniform discretization method has been tested on a solu-

tion to the de Vahl Davis problem in two- and three-dimensional domains. We have shown that the hybrid approach was able to improve the efficiency of the solution procedure and reduce the execution time by almost 50%.

5.2 Future Work Opportunities

Further analysis of RBF-FD and WLS approximation methods: We have examined the differences between RBF-FD and WLS approximation methods, but our analyses are by no means complete. Thus, we would like to extend the comparison by investigating the stencil size, shape parameters (present in the WLS approximation) and approximation stability in case of non-optimal domain discretizations. We would also like to better understand the oscillatory behaviour of the solution-error observed in our analyses with RBF-FD. Perhaps more complicated problems and domain shapes should also be included in the analyses.

Performance of the IMEX error indicator: With regard to the proposed IMEX error indicator, it is important that its theoretical background is well-understood and further investigated. The performance of IMEX should be further tested across a wider range of examples. For example, application to fluid flow problems could further verify our governing idea. Given that high-order approximations near singularities are conceptually flawed, developing an error indicator that considers such scenarios could lead to an even more robust implementation.

The marking strategy: The Texas Three Step marking strategy used in our work is not optimal for any of the problems addressed; rather, it is a starting point for the development of the meshless hp -adaptive methods. A much better marking strategy would take into account the smoothness of the solution and act accordingly, i.e. in areas with singularity the h -refinement would be preferred, while in areas with smooth solution the p -refinement would be preferred. In FEM, strategies that use local Sobolev regularity estimates to choose between the h - and the p -refinement for a given finite element have been proposed.

In our work, we wanted to keep things as simple as possible, so we used the Texas Three Step marking strategy as the cornerstone of our strategy. Nevertheless, marking strategies based on local data regularity should certainly be one of the first steps in future development of meshless hp -adaptive solution procedures.

The hp -adaptive solution procedure: The proposed hp -adaptive solution procedure needs further testing on various problems, perhaps fluid flow problems. Future work should also consider time-dependent problems, which have only been briefly mentioned in our research. The simplest approach to generalise the presented hp -adaptive solution procedure to time-dependent problems would be a granular adjustment of h and p throughout the simulation. In its simplest form, the proposed hp -adaptivity would be performed at each time step, starting with the hp distributions of the previous time step and using the same adaptivity parameters for all time steps. A more sophisticated approach would also take into account the desired accuracy during the simulation, resulting in time-dependent adaptivity parameters. Additionally, to perform a proper adaptive analysis, the time step should also be adaptive, which would require an additional step in the hp -adaptive solution procedure.

Many open questions about hp -adaptive solutions of time-dependent problems have been left for future research.

Spatially-variable approximation approach: We have proposed a spatially-variable approximation approach that can improve the efficiency of PDE-solving solution procedures. However, further research considering other approximation methods is needed. We would also like to analyse the required width of the scattered node layer in the case of hybrid scattered-uniform discretizations and ensure that no additional treatment is required on the transition between approximation methods or different node regularities.

References

- [1] K. W. Morton and D. F. Mayers, *Numerical Solution of Partial Differential Equations: An Introduction*, 2nd ed. Cambridge University Press, 2005. DOI: 10.1017/CB09780511812248.
- [2] B. Plestenjak, *Razširjen uvod v numerične metode* (Matematika - fizika : zbirka univerzitetnih učbenikov in monografij / DMFA - založništvo). DMFA - založništvo, 2015, ISBN: 9789612122645. [Online]. Available: <https://books.google.si/books?id=fh08jwEACAAJ>.
- [3] G. D. Smith, G. D. Smith, and G. D. S. Smith, *Numerical solution of partial differential equations: finite difference methods*. Oxford university press, 1985.
- [4] K.-J. Bathe and E. L. Wilson, *Numerical methods in finite element analysis*. Prentice-Hall, 1976.
- [5] B. D. Upadhyay, S. S. Sonigra, and S. D. Daxini, “Numerical analysis perspective in structural shape optimization: A review post 2000,” *Advances in Engineering Software*, vol. 155, p. 102992, 2021.
- [6] W. F. Mitchell and M. A. McClain, “A comparison of hp-adaptive strategies for elliptic partial differential equations,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 41, no. 1, pp. 1–39, 2014.
- [7] K. Segeth, “A review of some a posteriori error estimates for adaptive finite element methods,” *Mathematics and Computers in Simulation*, vol. 80, no. 8, pp. 1589–1600, 2010, ESCO 2008 Conference, ISSN: 0378-4754. DOI: <https://doi.org/10.1016/j.matcom.2008.12.019>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378475408004230>.
- [8] G.-R. Liu and Y.-T. Gu, *An introduction to meshfree methods and their programming*. Springer Science & Business Media, 2005.
- [9] G.-R. Liu, *Mesh free methods: moving beyond the finite element method*. CRC press, 2002. DOI: 10.1201/9781420040586.
- [10] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, *The finite element method: its basis and fundamentals*. Elsevier, 2005.
- [11] K. van der Sande and B. Fornberg, “Fast variable density 3-D node generation,” *SIAM Journal on Scientific Computing*, vol. 43, no. 1, A242–A257, 2021.
- [12] V. Shankar, R. M. Kirby, and A. L. Fogelson, “Robust node generation for mesh-free discretizations on irregular domains and surfaces,” *SIAM Journal on Scientific Computing*, vol. 40, no. 4, A2584–A2608, 2018.
- [13] G.-R. Liu, *Meshfree methods: moving beyond the finite element method*. Taylor & Francis, 2009.

- [14] V. Shankar, G. B. Wright, R. M. Kirby, and A. L. Fogelson, “A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction–diffusion equations on surfaces,” *Journal of scientific computing*, vol. 63, no. 3, pp. 745–768, 2015.
- [15] X.-Y. Li, S.-H. Teng, and A. Ungor, “Point placement for meshless methods using sphere packing and advancing front methods,” in *ICCES’00, Los Angeles, CA*, Citeseer, 2000.
- [16] R. Löhner and E. Oñate, “A general advancing front technique for filling space with arbitrary objects,” *International journal for numerical methods in engineering*, vol. 61, no. 12, pp. 1977–1991, 2004.
- [17] G. Kosec, “A local numerical solution of a fluid-flow problem on an irregular domain,” *Adv. Eng. Software*, vol. 120, pp. 36–44, 2018. DOI: 10.1016/j.advengsoft.2016.05.010.
- [18] R. Bridson, “Fast poisson disk sampling in arbitrary dimensions,” *SIGGRAPH sketches*, vol. 10, pp. 1 278 780–1 278 807, 2007.
- [19] J. Slak and G. Kosec, “On generation of node distributions for meshless PDE discretizations,” *SIAM Journal on Scientific Computing*, vol. 41, no. 5, A3202–A3229, 2019.
- [20] M. Depolli, J. Slak, and G. Kosec, “Parallel domain discretization algorithm for rbf-fd and other meshless numerical methods for solving PDEs,” *Computers & Structures*, vol. 264, p. 106 773, 2022, ISSN: 0045-7949. DOI: <https://doi.org/10.1016/j.compstruc.2022.106773>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0045794922000335>.
- [21] T. Jacquemin, P. Suchde, and S. P. Bordas, “Smart Cloud Collocation: Geometry-Aware Adaptivity Directly From CAD,” in *Computer-Aided Design*, vol. 154, p. 103 409, Jan. 2023, ISSN: 00104485. DOI: 10.1016/j.cad.2022.103409. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0010448522001427> (visited on 10/10/2022).
- [22] Z. Han, A. Rajendran, and S. N. Atluri, “Meshless local petrov-galerkin (MLPG) approaches for solving nonlinear problems with large deformations and rotations,” *Computer Modeling in Engineering and Sciences*, vol. 10, no. 1, p. 1, 2005.
- [23] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl, “Meshless methods: An overview and recent developments,” *Computer methods in applied mechanics and engineering*, vol. 139, no. 1-4, pp. 3–47, 1996.
- [24] A. Arefmanesh and M. Nikfar, “Analysis of natural convection in a nanofluid-filled triangular enclosure induced by cold and hot sources on the walls using stabilised mlpg method,” *The Canadian journal of chemical engineering*, vol. 91, no. 10, pp. 1711–1728, 2013.
- [25] G. Greto and S. Kulasegaram, “An efficient and stabilised sph method for large strain metal plastic deformations,” *Computational Particle Mechanics*, vol. 7, no. 3, pp. 523–539, 2020.
- [26] J. Slak and G. Kosec, “Adaptive radial basis function-generated finite differences method for contact problems,” *International Journal for Numerical Methods in Engineering*, vol. 119, no. 7, pp. 661–686, 2019.
- [27] M. Jančič, J. Slak, and G. Kosec, “ p -refined RBF-FD solution of a poisson problem,” in *2021 6th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2021, pp. 01–06. DOI: 10.23919/SpliTech52315.2021.9566401.

- [28] O. Davydov and D. T. Oanh, “Adaptive meshless centres and RBF stencils for poisson equation,” *Journal of Computational Physics*, vol. 230, no. 2, pp. 287–304, 2011.
- [29] T. Jacquemin and S. P. A. Bordas, “A unified algorithm for the selection of collocation stencils for convex, concave, and singular problems,” *International Journal for Numerical Methods in Engineering*, vol. 122, no. 16, pp. 4292–4312, 2021.
- [30] V. Bayona, N. Flyer, B. Fornberg, and G. A. Barnett, “On the role of polynomials in RBF-FD approximations: Ii. numerical solution of elliptic pdes,” *Journal of Computational Physics*, vol. 332, pp. 257–273, 2017.
- [31] N. Flyer, B. Fornberg, V. Bayona, and G. A. Barnett, “On the role of polynomials in RBF-FD approximations: I. interpolation and accuracy,” *Journal of Computational Physics*, vol. 321, pp. 21–38, 2016.
- [32] M. Jančič, J. Slak, and G. Kosec, “Monomial augmentation guidelines for RBF-FD from accuracy versus computational time perspective,” *Journal of Scientific Computing*, vol. 87, no. 1, pp. 1–18, 2021.
- [33] R. Trobec and G. Kosec, *Parallel scientific computing: theory, algorithms, and applications of mesh based and meshless methods*. Springer, 2015.
- [34] P. S. Jensen, “Finite difference techniques for variable grids,” *Computers & Structures*, vol. 2, no. 1-2, pp. 17–29, 1972.
- [35] T. Liszka and J. Orkisz, “The finite difference method at arbitrary irregular grids and its application in applied mechanics,” *Computers & Structures*, vol. 11, no. 1-2, pp. 83–95, 1980.
- [36] L. B. Lucy, “A numerical approach to the testing of the fission hypothesis,” *The astronomical journal*, vol. 82, pp. 1013–1024, 1977.
- [37] R. A. Gingold and J. J. Monaghan, “Smoothed particle hydrodynamics: Theory and application to non-spherical stars,” *Monthly notices of the royal astronomical society*, vol. 181, no. 3, pp. 375–389, 1977.
- [38] J. J. Monaghan, “Why particle methods work,” *SIAM Journal on Scientific and Statistical Computing*, vol. 3, no. 4, pp. 422–433, 1982.
- [39] J. J. Monaghan, “An introduction to SPH,” *Comput. Phys. Comm.*, vol. 48, pp. 89–96, 1988.
- [40] J. Bonet and S. Kulasegaram, “Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations,” *International journal for numerical methods in engineering*, vol. 47, no. 6, pp. 1189–1214, 2000.
- [41] F. A. Allahdadi, T. C. Carney, J. R. Hipp, L. D. Libersky, and A. G. Petschek, “High strain lagrangian hydrodynamics: A three dimensional SPH code for dynamic material response,” Phillips Lab Kirtland AFB NM, Tech. Rep., 1993.
- [42] B. Nayroles, G. Touzot, and P. Villon, “Generalizing the finite element method: Diffuse approximation and diffuse elements,” *Computational Mechanics*, vol. 10, no. 5, pp. 307–318, 1992. DOI: 10.1007/bf00364252.
- [43] T. Belytschko, Y. Y. Lu, and L. Gu, “Element-free galerkin methods,” *International journal for numerical methods in engineering*, vol. 37, no. 2, pp. 229–256, 1994.
- [44] J. Slak, “Adaptive RBF-FD method: PhD thesis,” Ph.D. dissertation, Univerza v Ljubljani, 2020. [Online]. Available: <https://repozitorij.uni-lj.si/IzpisGradiva.php?lang=slv&id=121511>.

- [45] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, and R. L. Taylor, "A finite point method in computational mechanics. Applications to convective transport and fluid flow," *International Journal for Numerical Methods in Engineering*, vol. 39, no. 22, pp. 3839–3866, 1996. DOI: 10.1002/(sici)1097-0207(19961130)39:22<3839::aid-nme27>3.0.co;2-r.
- [46] V. P. Nguyen, T. Rabczuk, S. Bordas, and M. Duflot, "Meshless methods: A review and computer implementation aspects," *Mathematics and Computers in Simulation*, vol. 79, no. 3, pp. 763–813, 2008. DOI: 10.1016/j.matcom.2008.01.003.
- [47] E. Ortega, E. Oñate, and S. Idelsohn, "An improved finite point method for tridimensional potential flows," *Computational Mechanics*, vol. 40, no. 6, pp. 949–963, 2007. DOI: 10.1007/s00466-006-0154-6.
- [48] C. Prax, H. Sadat, and P. Salagnac, "Diffuse approximation method for solving natural convection in porous media," *Transport in Porous Media*, vol. 22, no. 2, pp. 215–223, 1996.
- [49] W. K. Liu, S. Jun, and Y. F. Zhang, "Reproducing kernel particle methods," *International journal for numerical methods in fluids*, vol. 20, no. 8-9, pp. 1081–1106, 1995.
- [50] J. M. Melenk and I. Babuška, "The partition of unity finite element method: Basic theory and applications," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 289–314, Dec. 1996. DOI: 10.1016/s0045-7825(96)01087-0.
- [51] I. Babuška and J. M. Melenk, "The partition of unity method," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 4, pp. 727–758, 1997. DOI: 10.1002/(sici)1097-0207(19970228)40:4<727::aid-nme86>3.0.co;2-n.
- [52] S. N. Atluri and T. Zhu, "A new meshless local petrov-galerkin (MLPG) approach in computational mechanics," *Computational mechanics*, vol. 22, no. 2, pp. 117–127, 1998.
- [53] S. Atluri and T.-L. Zhu, "The meshless local petrov-galerkin (MLPG) approach for solving problems in elasto-statics," *Computational Mechanics*, vol. 25, no. 2-3, pp. 169–179, 2000.
- [54] Y. U. X. Mukherjee and S. Mukherjee, "The boundary node method for potential problems," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 5, pp. 797–815, 1997. DOI: 10.1002/(sici)1097-0207(19970315)40:5<797::aid-nme89>3.0.co;2-#.
- [55] T. Zhu, J.-D. Zhang, and S. N. Atluri, "A local boundary integral equation (LBIE) method in computational mechanics, and a meshless discretization approach," *Computational Mechanics*, vol. 21, no. 3, pp. 223–235, 1998. DOI: 10.1007/s004660050297.
- [56] E. Oñate and S. Idelsohn, "A mesh-free finite point method for advective-diffusive transport and fluid flow problems," *Computational Mechanics*, vol. 21, no. 4-5, pp. 283–292, 1998.
- [57] R. Löhner, C. Sacco, E. Onate, and S. Idelsohn, "A finite point method for compressible flow," *International Journal for Numerical Methods in Engineering*, vol. 53, no. 8, pp. 1765–1779, 2002.
- [58] 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*, vol. 19, no. 8-9, pp. 127–145, 1990.

- [59] C. Franke and R. Schaback, “Solving partial differential equations by collocation using radial basis functions,” *Applied Mathematics and Computation*, vol. 93, no. 1, pp. 73–82, 1998.
- [60] X. Zhang, K. Z. Song, M. W. Lu, and X. Liu, “Meshless methods based on collocation with radial basis functions,” *Computational mechanics*, vol. 26, pp. 333–343, 2000.
- [61] X. Liu, G. Liu, K. Tai, and K. Lam, “Radial point interpolation collocation method (rpim) for partial differential equations,” *Computers & Mathematics with Applications*, vol. 50, no. 8-9, pp. 1425–1442, 2005.
- [62] A. Tolstykh and D. Shirobokov, “On using radial basis functions in a “finite difference mode” with applications to elasticity problems,” *Computational Mechanics*, vol. 33, no. 1, pp. 68–79, 2003.
- [63] D. T. Oanh, O. Davydov, and H. X. Phu, “Adaptive rbf-fd method for elliptic problems with point singularities in 2D,” *Applied Mathematics and Computation*, vol. 313, pp. 474–497, 2017.
- [64] O. Davydov, D. T. Oanh, and N. M. Tuong, “Improved stencil selection for meshless finite difference methods in 3d,” *Journal of Computational and Applied Mathematics*, vol. 425, p. 115 031, 2023.
- [65] S. Shahane, A. Radhakrishnan, and S. P. Vanka, “A high-order accurate meshless method for solution of incompressible fluid flow problems,” *Journal of Computational Physics*, vol. 445, p. 110 623, 2021.
- [66] S. Le Borne and W. Leinen, “Guidelines for RBF-FD discretization: Numerical experiments on the interplay of a multitude of parameter choices,” *Journal of Scientific Computing*, vol. 95, no. 1, p. 8, 2023.
- [67] G. E. Fasshauer, *Meshfree approximation methods with MATLAB* (Interdisciplinary Mathematical Sciences 6). World Scientific, 2007. DOI: 10.1142/6437.
- [68] J.-S. Chen and T. Belytschko, “Meshless and meshfree methods,” in *Encyclopedia of Applied and Computational Mathematics*, B. Engquist, Ed., Springer Berlin Heidelberg, 2015, pp. 886–894. DOI: 10.1007/978-3-540-70529-1_531.
- [69] V. G. Patel and N. V. Rachchh, “Meshless method—review on recent developments,” *Materials today: proceedings*, vol. 26, pp. 1598–1603, 2020.
- [70] X. Li and H. Dong, “Error analysis of the meshless finite point method,” *Applied Mathematics and Computation*, vol. 382, p. 125 326, 2020.
- [71] P. Suchde and J. Kuhnert, “A fully lagrangian meshfree framework for PDEs on evolving surfaces,” *Journal of Computational Physics*, vol. 395, pp. 38–59, 2019.
- [72] A. Albuquerque-Ferreira, M. Ureña, and H. Ramos, “The generalized finite difference method with third-and fourth-order approximations and treatment of ill-conditioned stars,” *Engineering Analysis with Boundary Elements*, vol. 127, pp. 29–39, 2021.
- [73] I. Tominec, E. Larsson, and A. Heryudono, “A least squares radial basis function finite difference method with improved stability properties,” *SIAM Journal on Scientific Computing*, vol. 43, no. 2, A1441–A1471, 2021.
- [74] O. C. Zienkiewicz and J. Z. Zhu, “A simple error estimator and adaptive procedure for practical engineering analysis,” *International journal for numerical methods in engineering*, vol. 24, no. 2, pp. 337–357, 1987.

- [75] O. A. González-Estrada, S. Natarajan, J. J. Ródenas, and S. P. Bordas, “Error estimation for the polygonal finite element method for smooth and singular linear elasticity,” *Computers & Mathematics with Applications*, vol. 92, pp. 109–119, 2021.
- [76] M. Thimnejad, N. Fallah, and A. R. Khoei, “Adaptive refinement in the meshless finite volume method for elasticity problems,” *Computers & Mathematics with Applications*, vol. 69, no. 12, pp. 1420–1443, 2015. DOI: 10.1016/j.camwa.2015.03.023.
- [77] A. Angulo, L. P. Pozo, and F. Perazzo, “A posteriori error estimator and an adaptive technique in meshless finite points method,” *Engineering Analysis with Boundary Elements*, vol. 33, no. 11, pp. 1322–1338, 2009. DOI: 10.1016/j.enganabound.2009.06.004.
- [78] M. Afshar, M. Naisipour, and J. Amani, “Node moving adaptive refinement strategy for planar elasticity problems using discrete least squares meshless method,” *Finite Elements in Analysis and Design*, vol. 47, no. 12, pp. 1315–1325, 2011.
- [79] W.-z. Gui and I. Babuska, “The h, p and hp versions of the Finite Element Method in 1 Dimension. part 3. the adaptive hp version,” MARYLAND UNIV COLLEGE PARK LAB FOR NUMERICAL ANALYSIS, Tech. Rep., 1985.
- [80] W.-Z. Gui and I. Babuška, “The h, p and hp versions of the finite element method in 1 dimension. part ii. the error analysis of the h and hp versions,” *Numerische Mathematik*, vol. 49, no. 6, pp. 613–657, 1986.
- [81] P. R. Devloo¹², C. M. Bravo, and E. C. Rylo, “Recent developments in hp adaptive refinement,”
- [82] B. Guo and I. Babuška, “The hp version of the finite element method,” *Computational Mechanics*, vol. 1, no. 1, pp. 21–41, 1986.
- [83] W. F. Mitchell, “Performance of hp-adaptive strategies for 3d elliptic problems,” 2016.
- [84] J. Tinsley Oden, W. Wu, and M. Ainsworth, “Three-step hp adaptive strategy for the incompressible navier-stokes equations,” in *Modeling, mesh generation, and adaptive numerical methods for partial differential equations*, Springer, 1995, pp. 347–366.
- [85] M. Ainsworth and B. Senior, “Aspects of an adaptive hp-finite element method: Adaptive strategy, conforming approximation and efficient solvers,” *Computer Methods in Applied Mechanics and Engineering*, vol. 150, no. 1-4, pp. 65–87, 1997.
- [86] P. Houston, B. Senior, and E. Süli, “Sobolev regularity estimation for hp-adaptive finite element methods,” in *Numerical mathematics and advanced applications*, Springer, 2003, pp. 631–656.
- [87] P. Houston and E. Süli, “A note on the design of hp-adaptive finite element methods for elliptic partial differential equations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 2-5, pp. 229–243, 2005.
- [88] T. Eibner and J. M. Melenk, “An adaptive strategy for hp-FEM based on testing for analyticity,” *Computational Mechanics*, vol. 39, no. 5, pp. 575–595, 2007.
- [89] M. Bürg and W. Dörfler, “Convergence of an adaptive hp finite element strategy in higher space-dimensions,” *Applied numerical mathematics*, vol. 61, no. 11, pp. 1132–1146, 2011.
- [90] L. Demkowicz, W. Rachowicz, and P. Devloo, “A fully automatic hp-adaptivity,” *Journal of Scientific Computing*, vol. 17, no. 1, pp. 117–142, 2002.

- [91] W. Rachowicz, D. Pardo, and L. Demkowicz, “Fully automatic hp-adaptivity in three dimensions,” *Computer methods in applied mechanics and engineering*, vol. 195, no. 37-40, pp. 4816–4842, 2006.
- [92] G. Kosec and B. Šarler, “H-adaptive local radial basis function collocation meshless method,” *Computers Materials and Continua*, vol. 26, no. 3, p. 227, 2011.
- [93] J. Benito, F. Urena, L. Gavete, and R. Alvarez, “An h-adaptive method in the generalized finite differences,” *Computer methods in applied mechanics and engineering*, vol. 192, no. 5-6, pp. 735–759, 2003.
- [94] G. Liu, B. B. Kee, and L. Chun, “A stabilized least-squares radial point collocation method (ls-rpcm) for adaptive analysis,” *Computer methods in applied mechanics and engineering*, vol. 195, no. 37-40, pp. 4843–4861, 2006.
- [95] B. Tóth and A. Düster, “H-adaptive radial basis function finite difference method for linear elasticity problems,” *Computational Mechanics*, pp. 1–20, 2022.
- [96] W. Hu, N. Trask, X. Hu, and W. Pan, “A spatially adaptive high-order meshless method for fluid–structure interactions,” *Computer Methods in Applied Mechanics and Engineering*, vol. 355, pp. 67–93, 2019.
- [97] D. T. Oanh and N. M. Tuong, “An approach to adaptive refinement for the rbf-fd method for 2d elliptic equations,” *Applied Numerical Mathematics*, vol. 178, pp. 123–154, 2022.
- [98] R. Cavoretto and A. De Rossi, “Adaptive meshless refinement schemes for RBF-PUM collocation,” *Applied Mathematics Letters*, vol. 90, pp. 131–138, 2019.
- [99] L. Fan, “Adaptive meshless point collocation methods: Investigation and application to geometrically non-linear solid mechanics,” Ph.D. dissertation, Durham University, 2019.
- [100] P. K. Mishra, L. Ling, X. Liu, and M. K. Sen, “Adaptive radial basis function generated finite-difference (RBF-FD) on non-uniform nodes using p -refinement,” *arXiv preprint arXiv:2004.06319*, 2020.
- [101] S. Milewski, “Higher order schemes introduced to the meshless fdm in elliptic problems,” *Engineering Analysis with Boundary Elements*, vol. 131, pp. 100–117, 2021.
- [102] T. Liszka, C. Duarte, and W. Tworzydło, “Hp-meshless cloud method,” *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 263–288, 1996.
- [103] C. A. Duarte and J. T. Oden, “An hp adaptive method using clouds,” *Computer methods in applied mechanics and engineering*, vol. 139, no. 1-4, pp. 237–262, 1996.
- [104] R. El Kadmiri, Y. Belaasilia, A. Timesli, and M. S. Kadiri, “A hybrid algorithm using the FEM-MESHLESS method to solve nonlinear structural problems,” *Engineering Analysis with Boundary Elements*, vol. 140, pp. 531–543, 2022, ISSN: 0955-7997. DOI: 10.1016/j.enganabound.2022.04.018.
- [105] B. Zhou, C. Zhang, and F. Zhao, “A finite element-meshless hybrid method (FEMLHM) of elasticity problem and its applications,” *Mechanics of Solids*, pp. 1–20, 2023.
- [106] H. Ding, C. Shu, K. Yeo, and D. Xu, “Simulation of incompressible viscous flows past a circular cylinder by hybrid FD scheme and meshless least square-based finite difference method,” *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 9-11, pp. 727–744, 2004.

- [107] G. Bourantas, K. Mountris, V. Loukopoulos, *et al.*, “Strong-form approach to elasticity: Hybrid finite difference-meshless collocation method (FDMCM),” *Applied Mathematical Modelling*, vol. 57, pp. 316–338, 2018, ISSN: 0307-904X. DOI: 10.1016/j.apm.2017.09.028.
- [108] A. Javed, K. Djidjeli, J. Xing, and S. Cox, “A hybrid mesh free local RBF-Cartesian FD scheme for incompressible flow around solid bodies,” *International Journal of Mathematical, Computational, Natural and Physical Engineering*, vol. 7, pp. 957–966, 2013.
- [109] B. Fornberg and N. Flyer, *A primer on radial basis functions with applications to the geosciences*. SIAM, 2015.
- [110] R. Zamolo and E. Nobile, “Solution of incompressible fluid flow problems with heat transfer by means of an efficient RBF-FD meshless approach,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 75, no. 1, pp. 19–42, 2019.
- [111] M. Jančič and G. Kosec, “Stability analysis of RBF-FD and WLS based local strong form meshless methods on scattered nodes,” in *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 275–280. DOI: 10.23919/MIPRO55190.2022.9803334.
- [112] A. I. Tolstykh, “On using RBF-based differencing formulas for unstructured and mixed structured-unstructured grid calculations,” in *16th IMACS World Congress on Scientific Computation, Applied Mathematics and Simulation: Lausanne, Switzerland, August 21–25*, M. Deville, Ed., 2000, pp. 4606–4624.
- [113] G. Kosec and B. Sarler, “Local RBF collocation method for darcy flow,” *Computer Modeling in Engineering and Sciences*, vol. 25, no. 3, p. 197, 2008.
- [114] A. Kolar-Požun, M. Jančič, M. Rot, and G. Kosec, “Oscillatory behaviour of the rbf-fd approximation accuracy under increasing stencil size,” in *Computational Science – ICCS 2023*, J. Mikyška, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Slood, Eds., Cham: Springer Nature Switzerland, 2023, pp. 515–522, ISBN: 978-3-031-36027-5.
- [115] M. Jančič, F. Strniša, and G. Kosec, “Implicit-explicit error indicator based on approximation order,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 01–04. DOI: 10.23919/SpliTech55088.2022.9854342.
- [116] J. A. Reeger, *Adaptivity in local kernel based methods for approximating the action of linear operators*, 2023. arXiv: 2308.15325 [math.NA].
- [117] J. K. Hunter and B. Nachtergaele, *Applied analysis*. World Scientific Publishing Company, 2001.
- [118] F. Dell’Accio, F. Di Tommaso, N. Siar, and M. Vianello, “Numerical differentiation on scattered data through multivariate polynomial interpolation,” *BIT Numerical Mathematics*, pp. 1–29, 2021.
- [119] V. Bayona, “An insight into rbf-fd approximations augmented with polynomials,” *Computers & Mathematics with Applications*, vol. 77, no. 9, pp. 2337–2353, 2019.
- [120] M. Jančič and G. Kosec, “Strong form mesh-free *hp*-adaptive solution of linear elasticity problem,” *Engineering with Computers*, May 2023, ISSN: 1435-5663. DOI: 10.1007/s00366-023-01843-6. [Online]. Available: <https://doi.org/10.1007/s00366-023-01843-6>.

- [121] M. Jančič and G. Kosec, “A hybrid RBF-FD and WLS mesh-free strong-form approximation method,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 1–6. DOI: 10.23919/SpliTech55088.2022.9854278.
- [122] M. Jančič, M. Rot, and G. Kosec, “Spatially-varying meshless approximation method for enhanced computational efficiency,” in *Computational Science – ICCS 2023*, J. Mikyška, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Slood, Eds., Cham: Springer Nature Switzerland, 2023, pp. 500–514, ISBN: 978-3-031-36027-5.

Bibliography

Publications Related to the Thesis

Journal Articles

- M. Jančič, J. Slak, and G. Kosec, “Monomial augmentation guidelines for RBF-FD from accuracy versus computational time perspective,” *Journal of Scientific Computing*, vol. 87, no. 1, pp. 1–18, 2021.
- M. Jančič and G. Kosec, “Strong form mesh-free *hp*-adaptive solution of linear elasticity problem,” *Engineering with Computers*, May 2023, ISSN: 1435-5663. DOI: 10.1007/s00366-023-01843-6. [Online]. Available: <https://doi.org/10.1007/s00366-023-01843-6>.

Conference Papers

- M. Jančič, J. Slak, and G. Kosec, “*p*-refined RBF-FD solution of a poisson problem,” in *2021 6th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2021, pp. 01–06. DOI: 10.23919/SpliTech52315.2021.9566401.
- M. Jančič and G. Kosec, “Stability analysis of RBF-FD and WLS based local strong form meshless methods on scattered nodes,” in *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 275–280. DOI: 10.23919/MIPRO55190.2022.9803334.
- M. Jančič, F. Strniša, and G. Kosec, “Implicit-explicit error indicator based on approximation order,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 01–04. DOI: 10.23919/SpliTech55088.2022.9854342.
- A. Kolar-Požun, M. Jančič, M. Rot, and G. Kosec, “Oscillatory behaviour of the rbf-fd approximation accuracy under increasing stencil size,” in *Computational Science – ICCS 2023*, J. Mikyška, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Sloot, Eds., Cham: Springer Nature Switzerland, 2023, pp. 515–522, ISBN: 978-3-031-36027-5.
- M. Jančič, M. Rot, and G. Kosec, “Spatially-varying meshless approximation method for enhanced computational efficiency,” in *Computational Science – ICCS 2023*, J. Mikyška, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Sloot, Eds., Cham: Springer Nature Switzerland, 2023, pp. 500–514, ISBN: 978-3-031-36027-5.
- M. Jančič and G. Kosec, “A hybrid RBF-FD and WLS mesh-free strong-form approximation method,” in *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*, 2022, pp. 1–6. DOI: 10.23919/SpliTech55088.2022.9854278.

Other Publications

Journal Articles

- A. Rashkovska, M. Jančič, M. Depolli, J. Kosmač, and G. Kosec, “Uncertainty assessment of dynamic thermal line rating for operational use at transmission system operators,” *IEEE Transactions on Power Systems*, vol. 37, no. 6, pp. 4642–4650, 2022. DOI: 10.1109/TPWRS.2022.3144740.

Conference Papers

- M. Jančič, M. Založnik, and G. Kosec, “A sharp-interface mesoscopic model for dendritic growth,” *IOP Conference Series: Materials Science and Engineering*, vol. 1274, no. 1, p. 012046, Jan. 2023. DOI: 10.1088/1757-899X/1274/1/012046. [Online]. Available: <https://dx.doi.org/10.1088/1757-899X/1274/1/012046>.
- F. Strniša, M. Jančič, and G. Kosec, “A meshless solution of a small-strain plasticity problem,” in *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 257–262. DOI: 10.23919/MIPRO55190.2022.9803585.
- F. Strniša, M. Jančič, and G. Kosec, “Meshless simulation of infinitesimal elastoplastic deformation of a 3d body,” in *CCC:3, PROCEEDINGS OF THE FOURTEENTH INTERNATIONAL CONFERENCE ON COMPUTATIONAL STRUCTURES TECHNOLOGY*, 2022. DOI: 10.4203/ccc.3.3.4.
- M. Jančič, V. Cvrtila, and G. Kosec, “Discretized boundary surface reconstruction,” in *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, IEEE, 2021, pp. 278–283.
- M. Jančič, J. Slak, and G. Kosec, “GPU accelerated RBF-FD solution of poisson’s equation,” in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, IEEE, 2020, pp. 214–218.
- M. Jančič, J. Kocijan, and B. Grašič, “Identification of atmospheric variable using deep gaussian processes,” *IFAC-PapersOnLine*, vol. 51, no. 5, pp. 43–48, 2018.

Biography

Mitja Jančič received his Bachelor's degree (2015) degree from the Faculty of Mathematics and Physics at the University of Ljubljana, where he studied physics. He continued his studies with a Master's degree at the Faculty of Mechanical Engineering at the University of Ljubljana, where he studied mechanical engineering. In 2019, he joined the Department of Communication Systems (E6) at the Jožef Stefan Institute as a researcher and enrolled in doctoral studies at the Jožef Stefan International Postgraduate School in Information and Communication Technologies study programme.

Under the supervision of Dr. Gregor Kosec, his work began with the numerical treatment of PDEs, where he became familiar with the latest developments in meshless methods. From then on, his research focused on adaptive solution procedures, concentrating on various solution procedures for efficient solving of PDE systems. In particular, the *hp*-adaptive solution procedure, which simultaneously adjusts the spatial discretization and the approximation order, and adaptive methods that spatially vary the approximation method.

He presented some of his research milestones at international conferences (e.g. ICCS, MIPRO, SpliTech, ICASP). His most recognizable achievement is a published article on a strong-form mesh-free *hp*-adaptive solution procedure in the highly ranked journal *Engineering with Computers*. During his research he has also participated in several industrial projects.

Mitja was also awarded a 12-month scholarship from the World Federation of Scientists (WFS).