

5-2014

Adaptive Method of Approximate Particular Solution for One-Dimensional Differential Equations

Yichuan Dong

University of Southern Mississippi

Follow this and additional works at: https://aquila.usm.edu/masters_theses

Recommended Citation

Dong, Yichuan, "Adaptive Method of Approximate Particular Solution for One-Dimensional Differential Equations" (2014). *Master's Theses*. 9.

https://aquila.usm.edu/masters_theses/9

This Masters Thesis is brought to you for free and open access by The Aquila Digital Community. It has been accepted for inclusion in Master's Theses by an authorized administrator of The Aquila Digital Community. For more information, please contact Joshua.Cromwell@usm.edu.

The University of Southern Mississippi

ADAPTIVE METHOD OF APPROXIMATE PARTICULAR SOLUTION
FOR ONE-DIMENSIONAL DIFFERENTIAL EQUATIONS

by

Yichuan Dong

A Thesis

Submitted to the Graduate School
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Master of Science

Approved:

Dr. Huiqing Zhu

Dr. C.S. Chen
Professor,
Department of Mathematics

Dr. Haiyan Tian
Associate Professor,
Department of Mathematics

ABSTRACT

ADAPTIVE METHOD OF APPROXIMATE PARTICULAR SOLUTION FOR ONE-DIMENSIONAL DIFFERENTIAL EQUATIONS

by Yichuan Dong

May 2014

An adaptive algorithm for the method Approximate Particular Solution(MAPS) using radial basis functions for solving boundary value problems is discussed in this work. The goal of the adaptive algorithm is to construct an optimal collocation points distribution that gives the required accuracy with the smallest number of degrees of freedom. I proposed the formulation of the adaptive MAPS for second order boundary value problems in an arbitrary dimensional setting. Then I applied this method to three different boundary value problems in one-dimensional setting. The performance of the adaptive method has been demonstrated by numerical experiments.

ACKNOWLEDGMENTS

First, I would like to thank all of those who have assisted me in writing this thesis. I would like to thank my advisor, Dr. Huiqing Zhu for his help, suggestions, and support during my research. I am deeply grateful to him for the time and effort he has put into my thesis over the past year. He is very patient and always encourages me during the process of writing and researching. I really appreciate the time I spent working with him.

Secondly, I would like to thank my thesis committee from the Department of Mathematics. I first learned the concept of the meshfree method in Dr. C.S.Chen's class, and I learned a lot from the lectures and other Ph.D. students. I really appreciate all of the comments and suggestions that I have received.

Furthermore, I really appreciate the help I received from my teachers and classmates here. So many thanks go to my friends in The Mathzone. Leishin Kuo, Rodney T Lyons, Suanrong Chen, Leo Shen, Anup Raja Lamichhane, Tuisi Upadhyay, Vika Veunvi, Partrick Lambert, and Balaram Ghimire, who have helped and supported me a lot over the past two years.

Finally, I want to thank my family: my parents in China, who have supported me during my stay at The University of Southern Mississippi and have always encouraged to me to work hard to achieve my goals. My brother is in Atlanta pursuing his Ph.D. degree now, but he has always taken care of me. My family's love is the best gift of all.

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGMENTS	iii
LIST OF TABLES	v
LIST OF ILLUSTRATIONS	vi
LIST OF ABBREVIATIONS	vii
NOTATION AND GLOSSARY	viii
1 BACKGROUND	1
1.1 Introduction	1
1.2 One-dimensional differential equations	2
1.3 A brief review of adaptive algorithm	4
2 Adaptive Technique for MAPS and LMAPS	6
2.1 Radial basis functions (RBFs)	6
2.2 Method of approximated particular solutions (MAPS)	7
2.3 Adaptive schemes for the MAPS	10
3 Numerical Experiments	12
3.1 Examples of Adaptive MAPS	12
4 Conclusions and Remarks	27
Bibliography	28

LIST OF TABLES

Table

2.1	Examples of Radial Basis Functions	7
3.1	(Example 1) A comparison of four methods	16
3.2	(Example 2) A comparison of four methods when $\eta = 10$	19
3.3	(Example 2) A comparison of four methods when $\eta = 200$	20
3.4	(Example 3) A comparison of four methods when $\varepsilon = 10^{-2}$	23
3.5	(Example 3) A comparison of four methods when $\varepsilon = 10^{-6}$	25

LIST OF ILLUSTRATIONS

Figure

3.1	(Example 1) Graph of the exact solution.	13
3.2	(Example 1) Sets of adaptive collocation points (Left); and sets of uniformly distributed collocation points (Right).	14
3.3	(Example 1) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right).	14
3.4	(Example 1) Error of adaptive MAPS method versus non-adaptive MAPS method(left);Error of adaptive Kansa’s method versus non-adaptive Kansa’s method(right).	15
3.5	(Example 1) Error of adaptive MAPS method versus non-adaptive MAPS method(left); Error of adaptive Kansa’s method versus non-adaptive Kansa’s method(right).	16
3.6	(Example 2) Exact solutions when $\eta = 10$ (Left), and when $\eta = 200$ (Right). .	17
3.7	(Example 2) Collocation points using adaptive MAPS when $\eta = 10$ (Left), and when $\eta = 200$ (Right).	18
3.8	(Example 2) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right) when $\eta = 10$	18
3.9	(Example 2) Error curves of four methods when $\eta = 10$	19
3.10	(Example 2) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right) when $\eta = 200$	20
3.11	(Example 2) Error curves of four methods when $\eta = 200$	21
3.12	(Example 3) Exact solution (Left) and the adaption of nodes using MAPS (Right) when $\varepsilon = 10^{-2}$	22
3.13	(Example 3) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); Errors of MAPS when uniformly increasing the number of collocation points (Right) when $\varepsilon = 10^{-2}$	22
3.14	(Example 3) Error curves of four methods when $\varepsilon = 10^{-2}$	23
3.15	(Example 3) Exact solution (Left) and the adaption of nodes using MAPS (Right) when $\varepsilon = 10^{-6}$	24
3.16	(Example 3) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); Errors of MAPS when uniformly increasing the number of collocation points (Right) when $\varepsilon = 10^{-6}$	24
3.17	(Example 3) Error curves of four methods when $\varepsilon = 10^{-6}$	25

LIST OF ABBREVIATIONS

MQ	-	Multiquadrics
PDE	-	Partial Differential Equation
RBF	-	Radial Basis Function
MPS	-	Method of Particular Solutions
MFS	-	Method of Fundamental Solutions
FEM	-	Finite Element Method
BEM	-	Boundary Element Method
BKM	-	Boundary Knot Method
MAPS	-	Method of Approximate Particular Solutions
LIMQ	-	Local Inverse Multiquadrics
LMAPS	-	Localized Method of Approximate Particular Solutions
RBFCM	-	Radial Basis Function Collocation Method
MLPGM	-	Meshless Local Petrov-Galerkin Method
ELMAPS	-	Explicit Localized Method of Approximate Particular Solutions

NOTATION AND GLOSSARY

General Usage and Terminology

The notation used in this text represents fairly standard mathematical and computational usage. In many cases these fields tend to use different preferred notation to indicate the same concept, and these have been reconciled to the extent possible, given the interdisciplinary nature of the material. In particular, the notation for partial derivatives varies extensively, and the notation used is chosen for stylistic convenience based on the application. While it would be convenient to utilize a standard nomenclature for this important symbol, the many alternatives currently in the published literature will continue to be utilized.

The blackboard fonts are used to denote standard sets of numbers: \mathbb{R} for the field of real numbers, \mathbb{C} for the complex field, \mathbb{Z} for the integers, and \mathbb{Q} for the rationals. The capital letters, A, B, \dots are used to denote matrices, including capital greek letters, e.g., Λ for a diagonal matrix. Functions which are denoted in boldface type typically represent vector valued functions, and real valued functions usually are set in lower case roman or greek letters. Caligraphic letters, e.g., \mathcal{V} , are used to denote spaces such as \mathcal{V} denoting a vector space, \mathcal{H} denoting a Hilbert space, or \mathcal{F} denoting a general function space. Lower case letters such as i, j, k, l, m, n and sometimes p and d are used to denote indices.

Vectors are typeset in square brackets, e.g., $[\cdot]$, and matrices are typeset in parentheses, e.g., (\cdot) . In general the norms are typeset using double pairs of lines, e.g., $\|\cdot\|$, and the absolute value of numbers is denoted using a single pairs of lines, e.g., $|\cdot|$. Single pairs of lines around matrices indicates the determinant of the matrix.

Chapter 1

BACKGROUND

1.1 Introduction

Differential equations play a prominent role in many disciplines including engineering, economics, physics, and biology. It is always preferable to obtain a precise analytical solution for a given differential equation. However, many of them cannot be solved exactly by using analytical methods, which motivates us to develop some numerical algorithms to solve differential equations that analytical solutions cannot derive.

Radial Basis Function(RBF) has been praised for its ease of implementation and simplicity in data approximation [1], and it becomes the choice as a method used for the numerical solution of differential equations[2]. Meshless methods using RBF have drawn great attention for numerically solving partial differential equations during the past decade. Although meshless methods are relatively new numerical algorithms developed in recent years, these methods have been applied to problems in fluid dynamics, solid mechanics, and other fields. Various meshless methods have been developed and successfully applied to solving some challenging problems in different areas of science and engineering. Recently, the method of approximate particular solutions (MAPS) and a localized scheme for the method of approximate particular solutions (LMAPS) have been introduced to evaluate the partial solution of the given differential equations.

The first meshless method appears to be the smooth particle hydrodynamics (SPH) method introduced in 1977, which was initially developed for solving astrophysical problems. This method was further discussed by Gingold and Monaghan by using kernel

approximations. Since then, many variations and advancements have been extensively studied. Kansa proposed the radial basis function (RBF) collocation method for solving elliptical, hyperbolic, and parabolic differential equations[12]. Kansa's method was extended to various differential equations, including nonlinear differential equations.

MAPS that was proposed by Chen et. al[3] originally stems from the Method of Particular Solutions. However, the global matrices formed by the radial basis function approximations are plagued by ill-conditioning, especially when a large number of collocation points are used. It seems that if certain RBFs are used, the most important technique to obtain the numerical approximation is to increase the number of collocation points. In this situation, localized methods will be more desirable because the matrices will be sparse, instead of dense. In this way, the issues that present in other methods can be avoided by using localized domains. For more details about localized meshless methods using RBFs, the author refer the reader to[3, 24, 25].

1.2 One-dimensional differential equations

This thesis focuses on the numerical solution of one-dimensional differential equations by using adaptive MAPS/LMAPS. The author consider the singularly perturbed linear two-point boundary value problem (BVP):

$$\begin{aligned} -\varepsilon u'' + p(x)u' + q(x)u &= f \quad \text{in } (a, b), \\ u(a) &= \alpha \quad u(b) = \beta. \end{aligned}$$

where $\varepsilon > 0$ denotes a fixed constant, a , b , α and β , are given constants. The function $p(x)$, $q(x)$ and f are sufficiently smooth on (a, b) .

For the above problem, if the author set $p(x) = 0$, $q(x) = 0$, when $\varepsilon = 1$, it becomes the Poisson equation. The Poisson equation is a differential equation of elliptic type with broad application in electrostatics, mechanical engineering, and theoretical physics. It is

commonly used to model diffusion and to study physical problems involving differential equations, for example, electromagnetic wave propagation and acoustics. The Poisson equation, that represents the time-independent form of the original equation, is obtained by applying the technique of separation of variables to reduce the complexity of analysis.

It is well-known that when the parameter ε is very small, the solution's derivatives change dramatically within a very narrow interval, which is the so-called boundary layer (if happens near the boundary) or interior layer (if happens away from the boundary). This type of problems is called a singularly perturbed problem in the literature; it is a practically important but still difficult subject in applications such as fluid dynamics, electromagnetism, semiconductor research, genetics, chemotaxis, and computational biology. The presence of boundary layers in fluid dynamics was verified by Prandtl who gave his seminal paper to the Third International Congress of Mathematicians in Heidelberg in 1904. With the development of science and technology, researchers found that singular perturbation, the term which was first used by Friedrichs and Wasow, is widespread in nature.

The existence of boundary layers or interior layers usually incurs nonphysical excessive numerical oscillations in the standard finite element (FEM) approximation and finite difference approximation. To overcome this difficulty, many stabilization techniques have been proposed to improve the numerical solutions, such as upwind method, Petrov-Galerkin method, streamline diffusion FEM, discontinuous Galerkin method, and graded mesh method, to name a few. In the past decade, the most popular strategies is to construct a piecewise uniform meshes so that the numerical results are uniformly convergent. A direct drawback of these methods is the choice of problem-dependent parameters that need to be wisely selected based on the a priori knowledge of the layers, which is not available for most complex problems.

Another popular approach is the use of adaptive method, which will be briefly reviewed in the next section.

1.3 A brief review of adaptive algorithm

The goal of an adaptive algorithm is to construct an optimal point distribution that gives the required accuracy with the smallest number of degrees of freedom. In an adaptive scheme, more nodes are added on those parts of the domain with more detail, and simultaneously, enough nodes are kept in smooth regions. It is a fundamental task to construct such distributions for the solution of the problems while having nearly singularities or singular perturbations, which generate non-uniform data center distributions adapt to the changes in the sharp transition region.

A considerable effort has gone into the development and implementation of adaptive methods using mesh-based methods for various types of differential equations arisen in the physical world, like shock wave formation, crack front stress intensity, concentration, etc. In recent years, some researchers have used numerical methods with RBFs in adaptive schemes. Hon [9] used adaptive MQ scheme and showed that it can achieve superior accuracy for problems with one layer and relatively smooth solutions. This paper proposes an adaptive MQ-RBF technique using a posteriori indicator based on the weak formulation of the governing equation to add collocation points [9, 10].

An adaptive scheme based on greedy algorithm was developed by Schaback and Wendland[21] and Hon et al. [9]. They obtained a linear convergence in interpolation and collocation problems. Hon [9, 10] proposed an adaptive RBF collocation method using MQ-RBFs where a posterior indicator uses a weak formulation of the governing equation. This indicator detects a sharp transition and adds new collocation points according to the feedback from the error indicator. Later on, Ling and Trummer [14] modified Hon's indicator to fit transformed boundary value problems. Likewise Sarra [19] introduced an adaptive RBF collocation method based on the simple equidistribution of an arc length algorithm. It has been successfully applied for the solution of the nearly singular time-dependent diffusion-advection and Burger's equations in a one-dimensional setting. Meanwhile, a

dynamic adaptive scheme was proposed by Wu[22, 23] for time-dependent PDEs. In the literature, many of these methods that are formulated are one-dimensional, and the authors discussed how to extend their schemes to higher dimensions.

Chapter 2

Adaptive Technique for MAPS and LMAPS

Over the past two decades Radial Basis Functions (RBFs) have played an important role in the development of meshless methods. Since they can be applied to various partial Differential Equations (PDEs), RBFs become an important tool in these meshless methods. The RBF is especially suitable for problems in high-dimensions or problems on complex domains. In 1991, Kansa [12] first applied collocation methods using RBFs to PDEs in fluid dynamics problems. Since then, a lot of approximate methods employing RBFs were proposed including the meshless local petrov-Galerkin Method (MLPGM)[5], the method of fundamental solutions(MFS), the method of particular solutions (MPS)[24], the method of approximate particular solutions (MAPS)[24], the Boundary Knot Method (BKM), and the differential quadrature method (DQ).

This study will discuss MAPS in solving second order PDEs in one-dimensional domain first and then introduce an adaptive scheme upon these two methods.

2.1 Radial basis functions (RBFs)

The concept of radial basis functions (RBFs) was proposed in the 1980s when they were first investigated by Powell [23] and Micchelli [19]. Since then, extensive studies have been done in this research field [?]. According to the definition provided recently in Yao's dissertation [24], the function $\phi(r)$ is a RBF in n -dimensional space for any positive integer n . Here, $r := \|\mathbf{x} - \mathbf{p}\|$ denotes the Euclidean norm of two arbitrary points $\mathbf{x}, \mathbf{p} \in \mathbb{R}^n$. Usually, \mathbf{p} denotes the center of the RBF ϕ . For the remainder of this study, the radial basis function

will be indicated as $\phi(r)$. Table 2.1 gives a number of examples of RBFs which constant $c > 0$ is the shape parameter of RBFs.

Table 2.1: Examples of Radial Basis Functions

RBF Names	Formula
linear	r
cubic	r^3
Gaussian	$\exp(-cr^2)$
Multiquadrics	$\sqrt{r^2 + c^2}$
Inverse Multiquadrics	$1/\sqrt{r^2 + c^2}$
Thin-plate spline	$r^2 \ln(r)$

To generate an approximation of a function $u(\mathbf{x})$ using a RBF, the first step is to identify a finite set of distinct points $\{\mathbf{p}_j\}_{j=1}^N$ in \mathbb{R}^n . Based on the definition of a RBF, each one of the points in this set is a center. Collectively, these centers can be used to define an RBF approximation as follows:

$$\tilde{u}(\mathbf{x}) := \sum_{j=1}^N \alpha_j \phi(\|\mathbf{x} - \mathbf{p}_j\|), \quad (2.1.1)$$

for any $\mathbf{x} \in \mathbb{R}^n$.

Then, the coefficients α_j can be found from a linear system once the function values of u are given at a finite set of points. For instance, if $u(x_i)$, $i = 1, 2, \dots, N$, are known, finding α would require the solution of an $N \times N$ linear system $A\boldsymbol{\alpha} = \mathbf{U}$, where $A = (\phi(\|\mathbf{x}_i - \mathbf{p}_j\|))_{N \times N}$, $\boldsymbol{\alpha} = (\alpha_j)_{N \times 1}$, $\mathbf{U} = (U_j)_{N \times 1}$.

In this study, the author particularly interested in the Multiquadrics RBF (MQ). As a class of monotonically functions of distance r , MQ is proven to be conditionally positive definite, so that the matrix for the interpolation problem is invertible. The MQ is one of the most popular RBFs used in applications for its high-order rate of convergence.

2.2 Method of approximated particular solutions (MAPS)

The author will introduce the MAPS for problem (1.2) in this section. It is worth pointing out that the problem we consider here involves a diffusion-convection-reaction equation. Since

MAPS using radial basis functions actually can be applied to any dimensional problems, the author give a general introduction based on a n -dimensional model problem

$$\begin{aligned} -\varepsilon\Delta u + \mathbf{p}(\mathbf{x})\nabla u + q(\mathbf{x})u &= f(\mathbf{x}) \quad \text{in } \Omega, \\ u &= g(\mathbf{x}) \quad \text{on } \partial\Omega, \end{aligned} \tag{2.2.1}$$

where $\mathbf{p}(\mathbf{x})$, $q(\mathbf{x})$, $f(\mathbf{x})$, and $g(\mathbf{x})$ will become $p(x)$, $q(x)$, $f(x)$, and $g(x)$; ∇u and Δu will become u' and u'' when $n = 1$.

There are actually two ways to implement the MAPS. The first method is to directly find a particular solution for the differential equation (2.2.1), which may not be possible in many circumstances. Readers are referred to some recent publications [15]. Another method is to use the particular solution of Poisson's equation for the diffusion-convection-reaction equation given in problem (2.2.1).

The author will only use the second method in this study. To be more specific, the author first define the set of collocation points $\{\mathbf{p}_j\}_1^N = \{\mathbf{x}_j\}_1^N$ that locate both in the domain and on the boundary. The interior points are recorded as $\{\mathbf{x}_j\}_1^{N_i}$, and the collocation points on the boundary will be denoted as $\{\mathbf{x}_j\}_{N_i+1}^{N_i+N_b}$. We set $N = N_i + N_b$. The whole set of collocation points will be used to approximate the solution $u(\mathbf{x})$ of problem (2.2.1). Consequently, the author define an RBF approximation \tilde{u} to the solution of problem (2.2.1) as be $\sum \alpha_j \Phi_j$, where Φ_j is a particular solution of $\Delta v = \phi_j$, which means that Φ_j and ϕ_j satisfy

$$\Delta\Phi_j = \phi_j.$$

Here ϕ_j denotes $\phi(\|\mathbf{x} - \mathbf{p}_j\|)$ for any \mathbf{x} inside the domain.

As a consequence, it follows that

$$\Delta\tilde{u} = \sum_{j=1}^N \alpha_j \phi_j.$$

Substituting u in (2.2.1) with \tilde{u} and setting the equation to the set of collocation points, one has

$$\begin{aligned} \sum_{j=1}^N \alpha_j [\phi_j(\mathbf{x}_i) + \mathbf{p}(\mathbf{x}_i) \cdot \nabla \Phi_j(\mathbf{x}_i) + q(\mathbf{x}_i) \Phi_j(\mathbf{x}_i)] &= f(\mathbf{x}_i) \quad \forall i = 1, \dots, N_i, \\ \sum_{j=1}^N \alpha_j \Phi_j(\mathbf{x}_i) &= g(\mathbf{x}_i) \quad \forall i = N_i + 1, \dots, N, \end{aligned} \quad (2.2.2)$$

where $\Phi_j(\mathbf{x}_i) := \Phi_j(\|\mathbf{x}_i - \mathbf{p}_j\|)$. The above system of equations can be written as a linear system with variable $\boldsymbol{\alpha}$, i.e.,

$$A\boldsymbol{\alpha} = F, \quad (2.2.3)$$

where

$$A = \begin{pmatrix} B \\ C \end{pmatrix}, \quad B = [\phi_j(\mathbf{x}_i) + \mathbf{p}(\mathbf{x}_i) \cdot \nabla \Phi_j(\mathbf{x}_i) + q(\mathbf{x}_i) \Phi_j(\mathbf{x}_i)]_{N_i \times N}, \quad C = [\Phi_j(\mathbf{x}_i)]_{N_b \times N}.$$

Meanwhile, the right hand side term is defined as follows:

$$F = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_{N_i}), g(\mathbf{x}_{N_i+1}), \dots, g(\mathbf{x}_N)]^T.$$

If the author let ϕ be the MQ-RBF, then the author is going to solve the matrix by using the following expressions for ϕ and Φ :

- when $n = 1$,

$$\begin{aligned} \Phi(r) &= \frac{1}{6}(-3c^2\sqrt{r^2+c^2} + (r^2+c^2)(3/2) + 3c^2r\ln(\sqrt{(c^2+r^2)}+r)), \\ \phi(r) &= \sqrt{r^2+c^2}. \end{aligned} \quad (2.2.4)$$

- when $n = 2$,

$$\begin{aligned}\Phi(r) &= \frac{1}{9}(4c^2 + r^2)\sqrt{r^2 + c^2} - \frac{c^3}{3}\ln(\sqrt{c^2 + r^2} + c), \\ \phi(r) &= \sqrt{r^2 + c^2}.\end{aligned}\tag{2.2.5}$$

- when $n = 3$,

$$\begin{aligned}\Phi(r) &= \begin{cases} \frac{2r^2 + 5c^2}{24}\sqrt{r^2 + c^2} + \frac{c^4 \ln(\sqrt{c^2 + r^2} + r)}{8r} - \frac{c^3}{3} - \frac{c^4 \ln c}{8r}, & \text{when } r \neq 0, \\ 0 & \text{when } r = 0, \end{cases} \\ \phi(r) &= \sqrt{r^2 + c^2}.\end{aligned}\tag{2.2.6}$$

2.3 Adaptive schemes for the MAPS

As mentioned in Chapter 1, the purpose of adding adaptive algorithms to MAPS is to increase nodes only on those parts of the computational domain where the numerical approximation requires to be more accurate. This is obviously different from uniformly increasing nodes on the whole computational domain. Therefore this strategy is especially suitable for singularly perturbed problems since only a narrow subinterval needs to be treated specially.

In this work, the author apply the adaptive MAPS to the problem (1.2). It is numerically verified in Chapter 3 that this refinement method efficiently increases the spatial density distribution of nodes in regions where the solution presents sharp gradients and simultaneously keeps enough data points in smooth regions. Moreover, it is verified that the number of nodes required to meet a prescribed error value is considerably reduced when using the proposed node refinement algorithm, compared with respect to the uniform grid refinement method applied under the same conditions.

Next, the author is going to present the process of the adaptive MAPS method, which will be used in the numerical experiments. The author start from setting up initial collo-

collocation points, which is usually a uniformly distributed set of points $\Xi = \{x_j\}_{j=0}^N$ in the computational domain $[a, b]$. Also, denote by $x_{j+\frac{1}{2}} = (x_j + x_{j+1})/2$ the middle point of x_j and x_{j+1} , and by $\Xi^* = \{x_{j+\frac{1}{2}}\}_{j=0}^{N-1}$ the set of all middle points. Before the author list all steps of the adaptive method, the author define the error indicator δ , which can define the rule to flag the new nodes to be added.

Marking Rule: Let $\delta_{max} = \max \delta(x)$ be for all x belonging to the set of all middle points Ξ^* , and let $\lambda > 0$ be a threshold value. The author says that a node $x \in \Xi^*$ is flagged to be refined *if* $\delta(x) > \lambda \delta_{max}$.

The implementation of the MAPS can be formulated into the following three-steps procedure:

- Step 1: Compute the numerical solution \tilde{u} on the given set of collocation points, which is denoted by x_0, x_1, \dots, x_N , with $x_0 = a$ and $x_N = b$;
- Step 2: Compute the value of the error indicator δ at each midpoints for any $j = 0, \dots, N - 1$. Use the Marking Rule to flag new nodes;
- Step 3: Add new nodes to Ξ and update Ξ^* . Then return to Step 1 and repeat this process till the stopping criterion is satisfied.

There are different ways to define the stopping criterion. For example, in[2] the author used a stopping criterion that the maximum of the difference of the numerical solution and its interpolation at test points should be smaller than a given small number. In this study, we just simply use a fixed number of refinements as stopping criterion.

The author define $\delta(x_{j+\frac{1}{2}}) = |x_{j+1} - x_j|$. It is one of the simplest ways to define an error indicator. The meaning of this definition is to use the difference of two adjacent approximate values of solution as a measurement to the derivative. There are many different kinds of error indicators developed to fit different types of methods and problems in the literature.

Chapter 3

Numerical Experiments

This chapter presents the results of numerical experiments using the adaptive MAPS in a one-dimensional setting. Comparisons with non-adaptive MAPS and non-adaptive Kansa's method on uniformly distributed collocation points are provided. The author also try to compare the adaptive MAPS method with adaptive Kansa method in the examples. The error that appears in this chapter is the maximum of absolute values of exact solution and numerical solution at 1000 test points randomly chosen from the computational domain $[a, b]$.

In all examples, the author define the shape parameter as follows

$$c(x_j) = c_p(x_{j+1} - x_{j-1}),$$

for any $j = 1, \dots, N - 1$. When $j = 0$, we let $c = c_p(x_2 - x_0)$. When $j = N$, we let $c = c_p(x_N - x_{N-2})$.

3.1 Examples of Adaptive MAPS

First of all, the author use an example to show that the adaptive algorithm works for Poisson's equation. Starting from a uniformly distributed set of collocation points, more new collocation points will be added to positions where the solution of the problem has large derivative.

Example 1: For the BVP problem (1.2), the author first set ε equals to 1; $p(x)$, $q(x)$ are

equal to zero:

$$\begin{aligned} -u'' &= f \quad \text{in } (0,1), \\ u(0) &= 0 \quad u(1) = 0. \end{aligned}$$

The exact solution is selected to be

$$u(x) = \sin(2\pi x), \tag{3.1.1}$$

which is plotted in Figure 3.1.

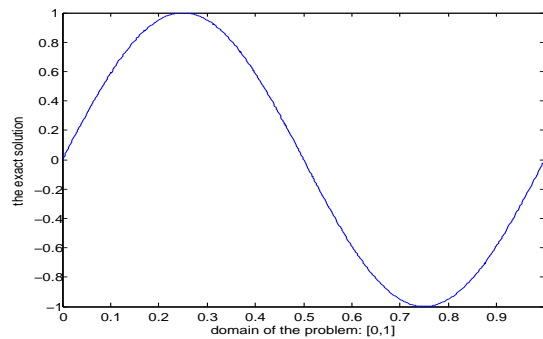


Figure 3.1: (Example 1) Graph of the exact solution.

The author set $\lambda = 0.3$ and the parameter c_p to be 5. The initial set of collocations are distributed uniformly with 11 points in total. The adaptive algorithm generates a series of non-uniform set of collocation points with total number of points $N = 11, 21, 37$ which is shown in Figure 3.2 (Left). It takes three steps to satisfy the stop criterion. Figure 3.2 (Right) plots sets of uniformly distributed collocation points with total number of points $N = 11, 21, 31, 41, 51$.

It may be interesting to take a look at errors at 1000 test points on the domain $[0, 1]$ when adaptive algorithm is used (see Figure 3.3(Left)). The error of the adaptive MAPS decreases rapidly. But in Figure 3.3 (Right), the decreasing rate of the errors of non-adaptive MAPS

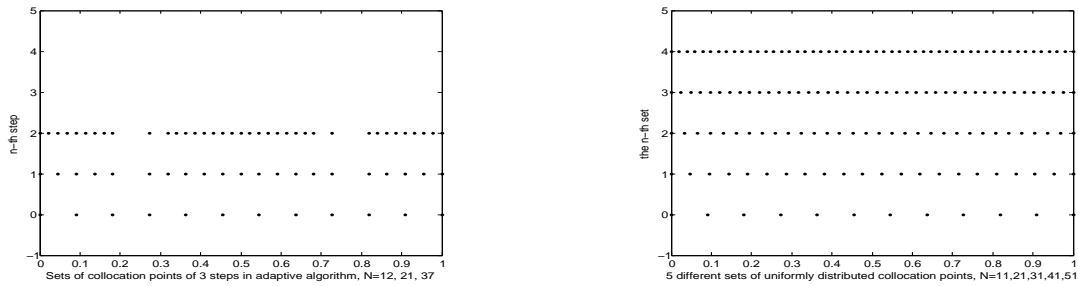


Figure 3.2: (Example 1) Sets of adaptive collocation points (Left); and sets of uniformly distributed collocation points (Right).

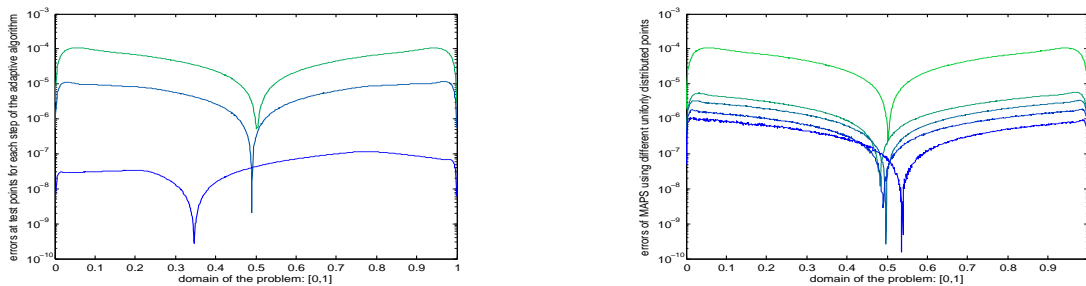


Figure 3.3: (Example 1) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right).

becomes slow.

The Figure 3.4 shows the maximal error of adaptive MAPS approximation versus non-adaptive MAPS approximation using a uniformly distributed set of collocation points. It can be seen that the error of adaptive MAPS approximation decreases faster than non-adaptive MAPS approximation when more collocation points are used. This verifies that adaptive method yields more accurate results than the method in which computational unknowns are

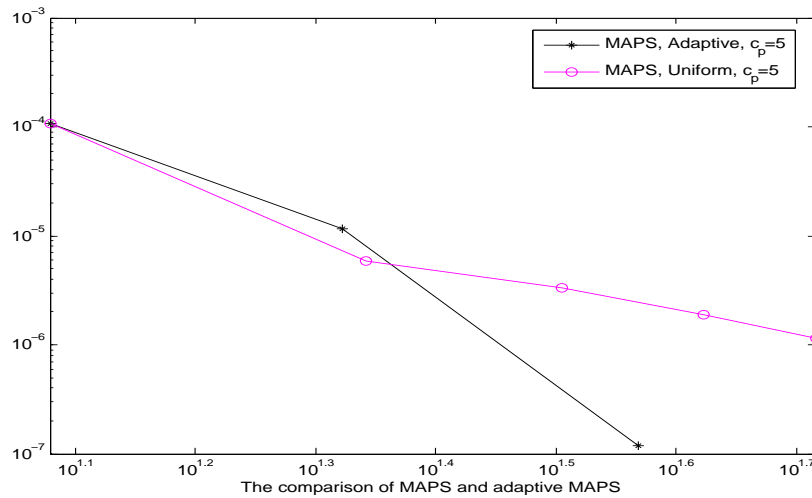


Figure 3.4: (Example 1) Error of adaptive MAPS method versus non-adaptive MAPS method(left);Error of adaptive Kansa's method versus non-adaptive Kansa's method(right).

uniformly increased. To compare with Kansa's method, the author also present in Figure 3.5 the maximal error of adaptive Kansa's approximation versus non-adaptive Kansa's approximation using uniform distributed set of collocation points. It seems that the error of Kansa's method is comparable with that of non-adaptive Kansa's method, which is possible since the exact solution is quite smooth, so the adaptive methods refine the set of collocation points in a similar way to uniform refinement.

To show the computational results more clearly, the author put data from Figure 3.4 and Figure 3.5 into Table 3.1.

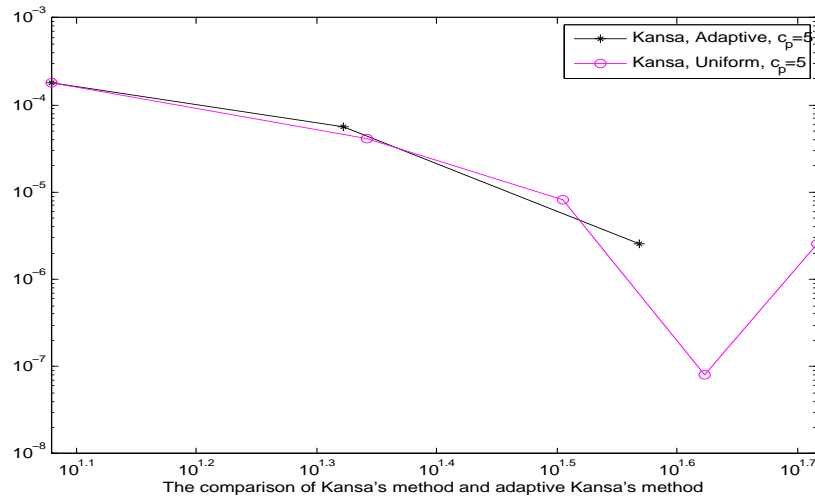


Figure 3.5: (Example 1) Error of adaptive MAPS method versus non-adaptive MAPS method(left); Error of adaptive Kansa's method versus non-adaptive Kansa's method(right).

Table 3.1: (Example 1) A comparison of four methods

Step	adaptive MAPS		adaptive Kansa's method		MAPS		Kansa's method	
	N	error	N	error	N	error	N	error
1	11	1.0753e-4	11	1.7909e-4	11	1.0753e-4	11	1.7909e-4
2	21	1.1680e-5	21	5.6146e-5	21	5.7480e-6	21	4.0664e-5
3	37	1.1752e-7	37	2.5449e-6	31	3.3801e-6	31	8.2115e-6
4					41	1.9032e-6	41	8.0922e-8
5					51	1.1522e-6	51	2.5887e-6

Example 2: The author consider Problem (3.1) again with another right-hand side term

$$f(x) = -2\eta u(x)(\eta - \eta u(x)^2).$$

The exact solution is

$$u(x) = \tanh(\eta(x - 0.5)),$$

which is plotted in Figure 3.6 when $\eta = 10$ and 200.

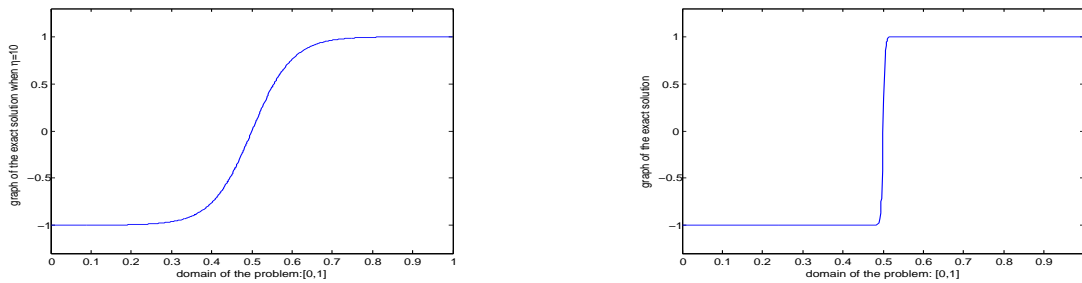


Figure 3.6: (Example 2) Exact solutions when $\eta = 10$ (Left), and when $\eta = 200$ (Right).

Note that there is an interior layer inside the domain at $x = 0.5$ when η becomes large so that it turns out to be very difficult to capture the rapid change of the function values using MAPS or Kansa's method on uniformly distributed collocation points (see the upper two curves in Figure 3.9). Therefore, the author set initial set of collocation points with $N = 11$ and $\lambda = 0.3$ when $\eta = 10$, but $N = 101$ and $\lambda = 0.1$ when $\eta = 200$. The adaption of collocation points are presented in Figure 3.7.

Next, let us take a look at the convergence behaviors of adaptive MAPS and non-adaptive MAPS. When $\eta = 10$, the plot the errors of adaptive MAPS at each step of the adaptive algorithm and errors of MAPS using different sets of uniformly distributed collocation points in Figure 3.10. It is observed that the non-adaptive MAPS produced better results than adaptive MAPS when roughly the same number of collocation points were used. The reason is that the solution does not really have a rapid change at the midpoint of the domain when $\eta = 10$. In this situation, non-adaptive MAPS using uniformly distributed collocation

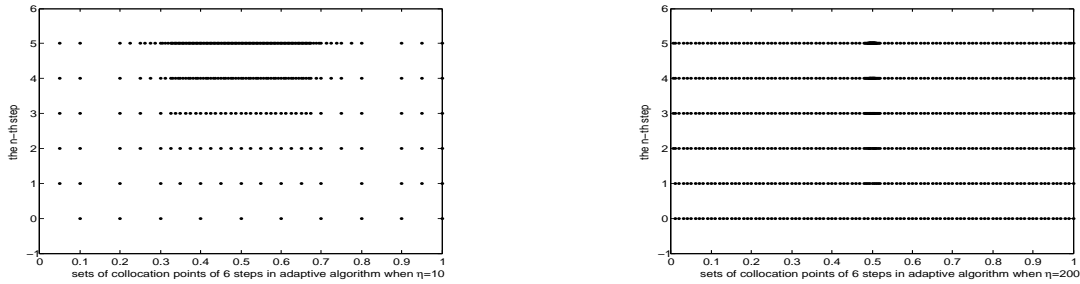


Figure 3.7: (Example 2) Collocation points using adaptive MAPS when $\eta = 10$ (Left), and when $\eta = 200$ (Right).

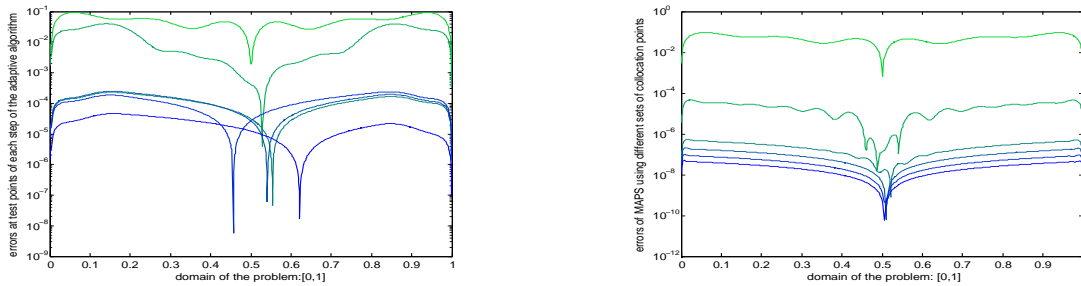


Figure 3.8: (Example 2) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right) when $\eta = 10$.

points may achieve better results because its results should be affected much less by the choice of the shape parameter.

The errors of adaptive MAPS, non-adaptive MAPS, Kansa's method, and nonadaptive Kansa's method are presented in Table 3.2 and in Figure 3.9. Here, non-adaptive methods

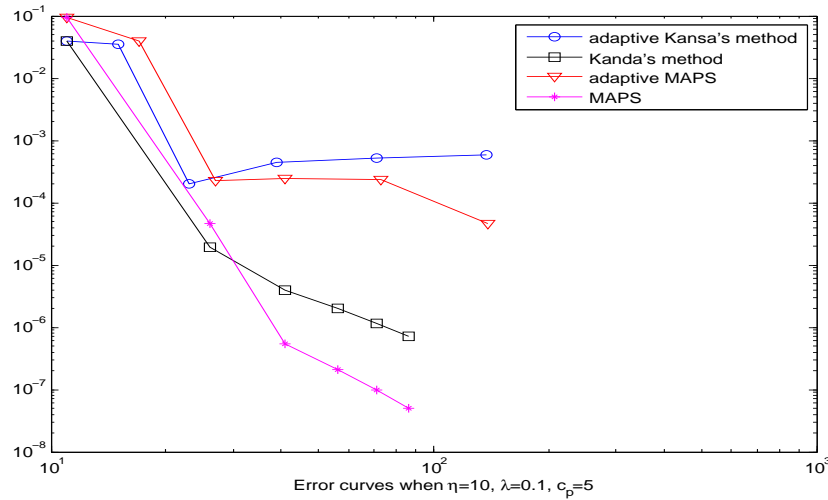


Figure 3.9: (Example 2) Error curves of four methods when $\eta = 10$.

are implemented on uniform distributed points.

Table 3.2: (Example 2) A comparison of four methods when $\eta = 10$

Step	adaptive MAPS		adaptive Kansa's method		MAPS		Kansa's method	
	N	error	N	error	N	error	N	error
1	11	9.3362e-2	11	3.9850e-2	11	9.3370e-2	11	3.9849e-2
2	17	4.0108e-2	15	3.4263e-2	26	4.5736e-5	26	1.8921e-5
3	27	2.2996e-4	23	1.9800e-4	41	5.4340e-7	41	3.9379e-6
4	41	2.4849e-4	39	4.4475e-4	56	2.1047e-7	56	2.0141e-6
5	71	2.3801e-4	71	5.2438e-4	71	9.7733e-8	71	1.1585e-6
6	139	4.6047e-5	137	5.8974e-4	86	5.0831e-8	86	7.2583e-7

Now, the author consider the problem when $\eta = 200$ which incurs a rapid change to the solution. In this case, it is extremely important to know where new nodes should be added. From Figure 3.7 (Right), it can be seen that the new nodes were concentrated on a very

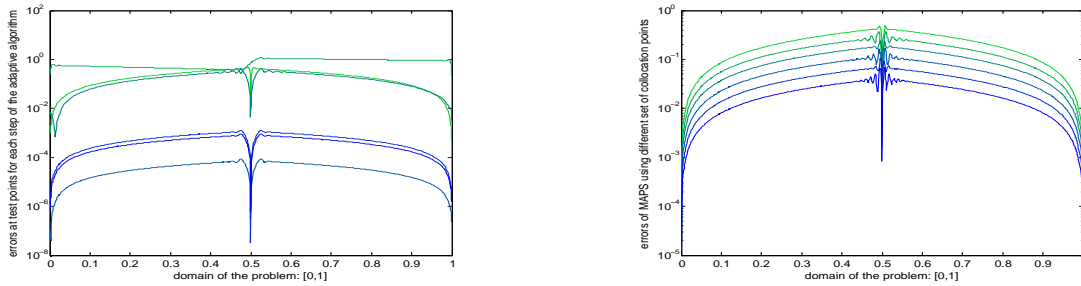


Figure 3.10: (Example 2) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); and errors of MAPS using different sets of uniformly distributed collocation points (Right) when $\eta = 200$.

narrow subdomain near the interior layer. As a consequence, the error curves of adaptive MAPS in Table 3.3 and in Figure 3.11 show that the error decreases rapidly as new nodes are added to the set of collocation points.

Table 3.3: (Example 2) A comparison of four methods when $\eta = 200$

Step	adaptive MAPS		adaptive Kansa's method		MAPS		Kansa's method	
	N	error	N	error	N	error	N	error
1	101	4.7919e-1	101	4.7937e-1	101	4.7887e-1	101	4.7896e-1
2	105	1.1801e-0	105	4.6086e-1	116	3.7525e-1	116	3.7453e-1
3	111	4.2167e-1	111	1.3673e-1	131	2.3831e-1	131	2.3813e-1
4	121	8.6705e-5	121	1.3622e-4	146	1.6860e-1	146	1.6909e-1
5	135	1.2915e-3	135	8.9990e-3	161	1.0581e-1	161	1.0548e-1
6	165	9.1958e-4	167	1.2936e-2	176	6.8635e-2	176	6.8337e-2

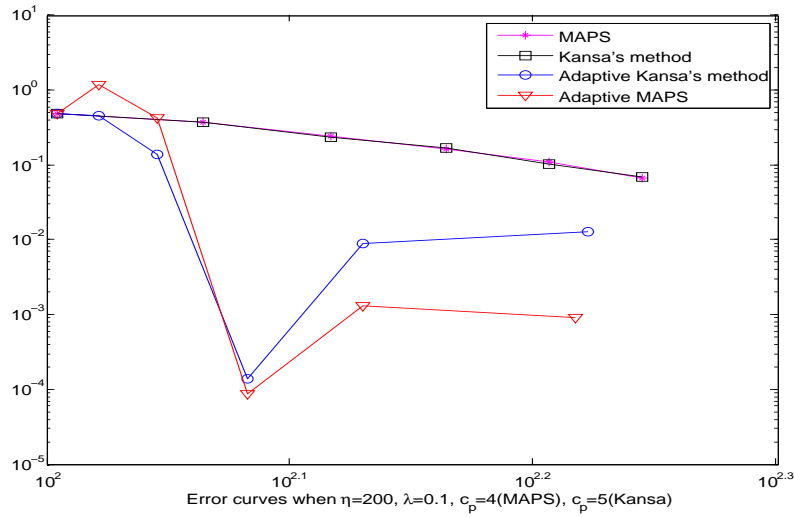


Figure 3.11: (Example 2) Error curves of four methods when $\eta = 200$.

Example 3: In this example, the author set $p(x)$ equals to zero, $q(x)$ equals to one:

$$-\varepsilon u'' + u = 0 \quad \text{in } (0, 1),$$

$$u(0) = 0 \quad u(1) = 1.$$

the exact solution is selected to be

$$u(x) = \frac{1 - e^{-x/\sqrt{\varepsilon}}}{1 - e^{-1/\sqrt{\varepsilon}}}. \quad (3.1.2)$$

First, let us consider the numerical methods when $\varepsilon = 10^{-2}$. The exact solution is plotted in Figure 3.12 (Left).

The author set $\lambda = 0.5$, and $c_p = 10$ for adaptive MAPS, and other methods appeared in

comparison 3.14. Figure 3.12 (Right) shows the distribution of collocation points in each step.

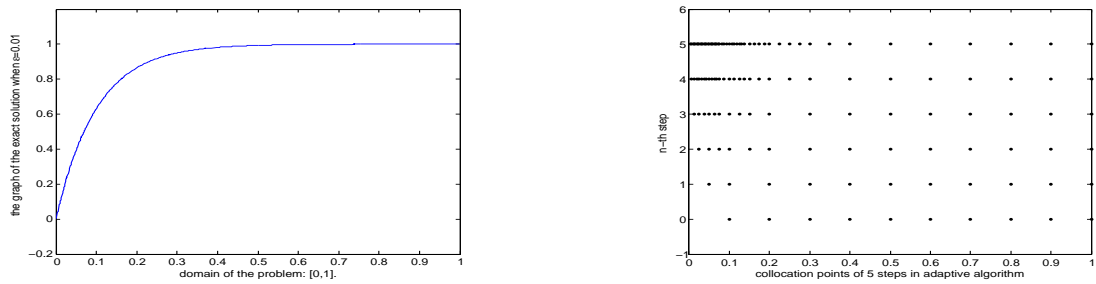


Figure 3.12: (Example 3) Exact solution (Left) and the adaption of nodes using MAPS (Right) when $\varepsilon = 10^{-2}$.

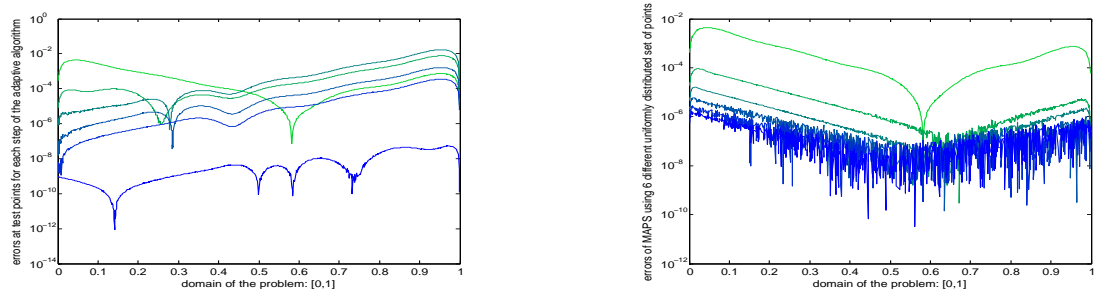


Figure 3.13: (Example 3) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); Errors of MAPS when uniformly increasing the number of collocation points (Right) when $\varepsilon = 10^{-2}$.

Figure 3.13 (Left) shows the errors of adaptive MAPS on $[0, 1]$ at each step of the

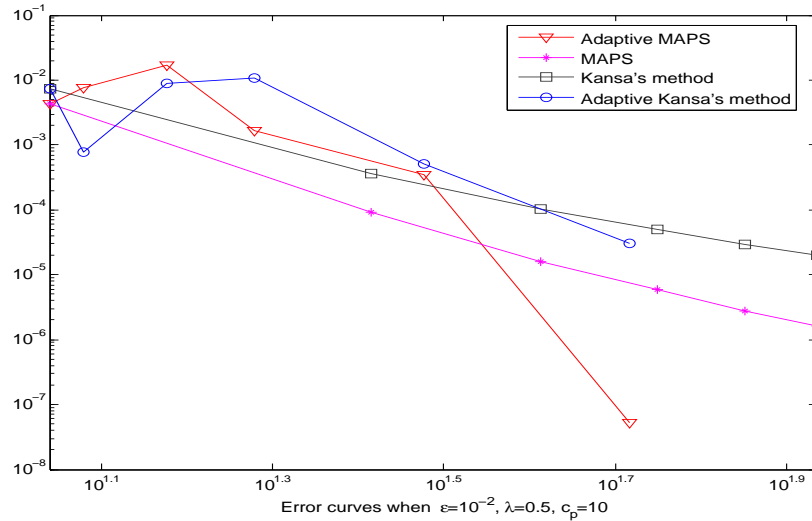


Figure 3.14: (Example 3) Error curves of four methods when $\varepsilon = 10^{-2}$.

Table 3.4: (Example 3) A comparison of four methods when $\varepsilon = 10^{-2}$

Step	adaptive MAPS		adaptive Kansa's method		MAPS		Kansa's method	
	N	error	N	error	N	error	N	error
1	11	4.3459e-3	11	7.3556e-3	11	4.3459e-3	11	7.3484e-3
2	12	7.6332e-3	12	7.8780e-4	26	9.1573e-5	26	3.6138e-4
3	15	1.6966e-2	15	8.9648e-3	41	1.6058e-5	41	1.0244e-4
4	19	1.6390e-3	19	1.0883e-2	56	5.6531e-6	56	4.8830e-5
5	30	3.8851e-4	30	5.0847e-4	71	2.6818e-6	71	2.9680e-5
5	52	2.6486e-7	52	3.0549e-5	86	1.7489e-6	86	2.0374e-5

adaptive algorithm. Similarly, the errors of MAPS using a different uniform distributed set of collocation points on $[0, 1]$ are plotted in Figure3.13 (Right). The maximums of these curves are taken to plot Figure3.14. Convergent curves show that the error of adaptive MAPS can be smaller than 10^{-7} , while

Another observation is that the adaptive MAPS uses less collocation points than non-

adaptive MAPS. A similar conclusion can be drawn for adaptive Kansa's method and non-adaptive Kansa's method.

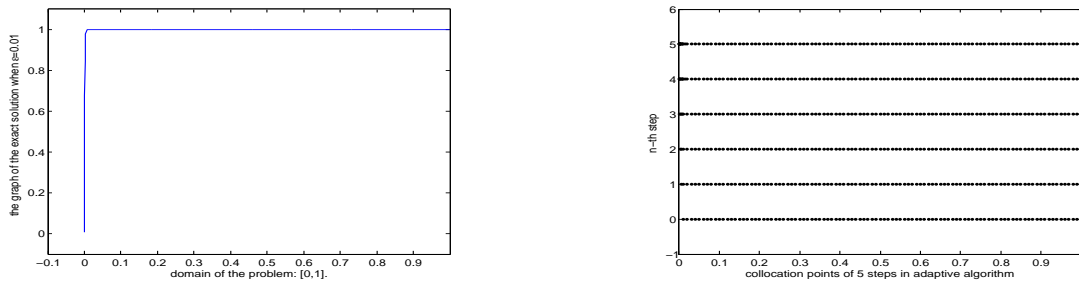


Figure 3.15: (Example 3) Exact solution (Left) and the adaption of nodes using MAPS (Right) when $\epsilon = 10^{-6}$.

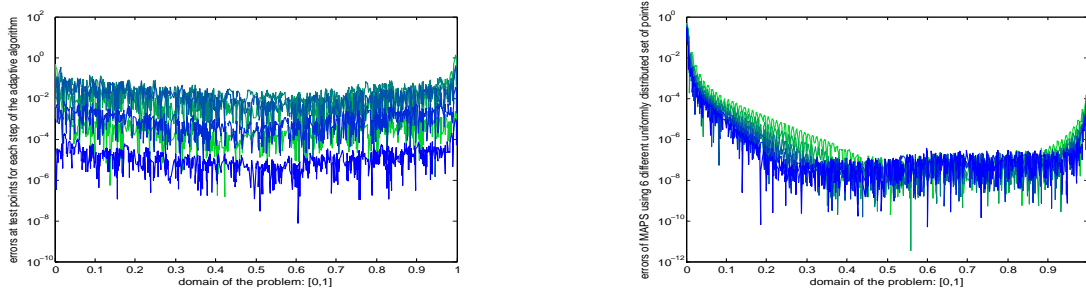


Figure 3.16: (Example 3) Errors of adaptive MAPS at each step of the adaptive algorithm (Left); Errors of MAPS when uniformly increasing the number of collocation points (Right) when $\epsilon = 10^{-6}$.

Secondly, let us consider $\epsilon = 10^{-6}$, which means that the problem has strong singularity

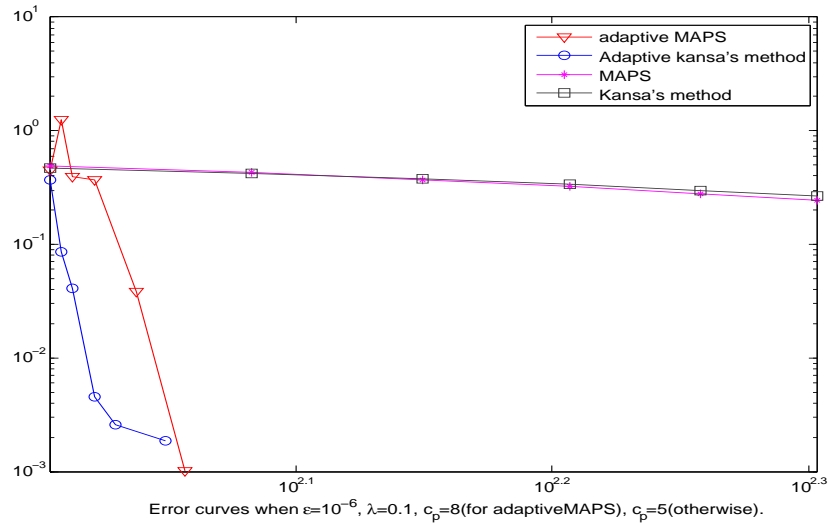


Figure 3.17: (Example 3) Error curves of four methods when $\varepsilon = 10^{-6}$.

Table 3.5: (Example 3) A comparison of four methods when $\varepsilon = 10^{-6}$

Step	adaptive MAPS		adaptive Kansa's method		MAPS		Kansa's method	
	N	error	N	error	N	error	N	error
1	101	4.3102e-1	101	5.0214e-1	101	4.8730e-1	101	5.2102e-1
2	102	7.5386e-1	102	2.3092e-1	121	4.3810e-1	121	4.6912e-1
3	103	7.3954e-1	103	4.3215e-2	141	3.9391e-1	141	4.2140e-1
4	105	1.4122e-2	105	4.5064e-3	161	3.5337e-1	161	3.7757e-1
5	107	3.8752e-2	107	2.5710e-3	181	3.1635e-1	181	3.3750e-1
5	112	5.8615e-4	112	1.8998e-3	201	2.8269e-1	201	3.0104e-1

at $x = 0$. The exact solution is plotted in Figure 3.15 (Left). Other figures are arranged in the same way as for the case when $\varepsilon = 10^{-2}$. The author set $\lambda = 0.1$, and $c_p = 8$ for adaptive MAPS, and $c_p = 5$ other methods appeared in comparison Figure 3.17.

Figure 3.17 illustrates that MAPS and Kansa's method using uniformly distributed collocation points cannot produce an accurate approximation unless they use a very large

member of collocation points to resolve the boundary layer. On the other hand, the adaptive MAPS and the adaptive Kansa's method can generate good approximations with error $\sim 10^{-3}$ only using about 110 collocation points.

Chapter 4

Conclusions and Remarks

We introduced an adaptive MAPS in this work. Based on the numerical results we obtained in chapter 3, we have the following conclusions:

1. Adaptive MAPS has been formulated for second order boundary value problems. This method can actually be extended to higher order differential equations and time-dependent problems. In this work, we only consider most simplest error indicator and stop criterion. There are also many other choices for the error indicator and stop criterion.
2. It has been numerically verified that adaptive MAPS and adaptive Kansa can both be used for computing numerical approximations to the solution of the differential equations. It has been found that if the solution has a boundary layer or interior layer, adaptive methods are more accurate than non-adaptive method when the same number of collocation points are used.
3. The performance of the adaptive methods depends on the number of collocation points in a initial set, the parameter c_p , and the parameter λ in error indicator. The choice of an optimal shape parameter for adaptive RBF collocation method is another interesting research topic that has already been extensively discussed for non-adaptive RBF collocation method.
4. The investigation of adaptive localized MAPS method will be a future research project. It will be be more convincible to study the adaptive localized MAPS in a domain with higher dimension and complex boundary.

BIBLIOGRAPHY

- [1] M.D. Buhmann, *Radial Basis Functions: Theory and Implementations*, Cambridge University Press, Cambridge, 2003.
- [2] D. S. Broomhead, and David Lowe, *Radial basis functions, multi-variable functional interpolation and adaptive networks*, (Technical Report 4148). RSRE. 1988.
- [3] C.S. Chen, C.M. Fan, and P.H. Wen, *The Method of Particular Solutions for Solving Certain Partial Differential Equations*, Numerical Methods for Partial Differential Equations, 28:506–522, 2012.
- [4] C.S. Chen, Y.C. Hon, and R.A. Schaback, *Scientific Computing with Radial Basis Functions*, University of Southern Mississippi, 2012.
- [5] S.N. Atluri and T.-L. Zhu. *The meshless local Petrov-Galerkin (MLPG) approach in Computational mechanics*, Computational Mechanics, 22:117–127, 1998.
- [6] Stanley J. Farlow, *Partial Differential Equations for Scientists and Engineers*. New York: Dover Publications, Inc. 1982.
- [7] E.C. Gartland Jr., *Graded mesh difference schemes for singularly perturbed two point boundary value problems*, Math. Comp., 51:631–657, 1988.
- [8] E.E. Hart,, S.J. Cox, K. Djidjeli, and V.O. Kubytskyi, *Solving an eigenvalue problem with a periodic domain using radial basis functions*, Engineering Analysis with Boundary Elements, 33:258–262, 2009.
- [9] Y.C. Hon, R. Schaback, X. Zhou, *An adaptive greedy algorithm for solving large RBF collocation problems*, Numer. Algorithms, 32(1):13–25, 2010.

- [10] Y.C. Hon, *Multiquadric collocation method with adaptive technique for problems with boundary layer*, Int. J. Appl. Sci. Comput., 6(3):173–184, 1999.
- [11] Y.C. Hon and W. Chen. *Boundary knot method for 2D and 3D Helmholtz and convection diffusion problems with complicated geometry*, J. Numer. Methd. Engng., 31(7):614–623, 2007.
- [12] E.J. Kansa, *Multiquadratic - a scattered data approximation scheme with applications to computational fluid dynamics - I. Surface approximations and partial derivatives estimates*, Comput. Math. Appl., 19: 127–45, 1990.
- [13] C.K. Leen, X. Liu, and S.C. Fan. *Local multiquadric approximation for solving boundary value problems* Computational Mechanics, 30:396–409, 2003.
- [14] L. Ling, M.R. Trummer, *Adaptive multiquadric collocation for boundary layer problems*. J. Comput. Appl. Math., 188:265–282, 2006.
- [15] Charles A. Micchelli, *Interpolation of Scattered Data: Distance Matrices and Conditionally Positive Definite Functions*, Constructive Approximation, 2: 11–22, 1986.
- [16] J.J.H. Miller, E.O’Riordan, G.I. Shinshkin, *On piecewise uniform meshes for upwind and central difference operators for solving singularly perturbed problems*, IMA J. Numer. Anal., 15:89–99, 1995.
- [17] Rodrigo B. Platte, *Accuracy and Stability of Global Radial Basis Function Methods for the Numerical Solution of Partial Differential Equations*, Dissertation. University of Delaware, 2005.
- [18] M.J.D. Powell, *Radial basis functions for multivariable interpolation: A review*. IMA conference on "Algorithms for the Approximation of Functions and Data." RMCS Shrivenham, 1985.
- [19] S.A. Sarra, *Adaptive radial basis function methods for time dependent partial differential equations*, Appl. Numer. Math., 54(1):79–94, 2005.

- [20] B. Sarler, T. Tran-Cong, and C.S. Chen, *Meshfree direct and indirect local radial basis function collocation method for diffusion problems*. Computers and Mathematics with Applications. 51:1269–1282, 2006.
- [21] R. Schaback, H. Wendland, *Adaptive greedy techniques for approximate solution of large RBF systems*, Numer. Algorithms, 24(3):239–254, 2000.
- [22] Z. Wu, *Dynamically knots setting in meshless method for solving time dependent propagation equation*, Comput. Methods Appl. Mech. Eng., 193(12):1221–1229, 2004.
- [23] Z. Wu, *Dynamically knot and shape parameter setting for simulating shock wave by using multi-quadric quasi-interpolation*, Eng. Anal. Bound. Elem. 29(4):354-358, 2004. [18] BozziniM, LenarduzziL,
- [24] Guangming Yao, *Local Radial Basis Function Methods for Solving Partial Differential Equations*. Diss. University of Southern Mississippi, 2010.
- [25] Guangming Yao, C.S. Chen, and Joseph Kolibal, *A localized approach for the method of approximated particular solutions*, Computers and Mathematics with Applications, 61:2376–2387, 2011.