12-14-2015

Fast Methods for Variable-Coefficient Peridynamic and Non-Local Diffusion Models

Che Wang
University of South Carolina - Columbia

Fast methods for variable-coefficient peridynamic and non-local diffusion models

by

Che Wang

Bachelor of Science
Shandong University, 2009
Bachelor of Finance
Shandong University, 2009

Submitted in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy in

Mathematics

College of Arts and Sciences

University of South Carolina

2015

Accepted by:

Hong Wang, Major Professor

Qi Wang, Committee Member

LiLi Ju, Committee Member

Yi Sun, Committee Member

GuiRen Wang, Committee Member

Lacy Ford, Senior Vice Provost and Dean of Graduate Studies

Acknowledgments

I would like to express my sincere gratitude to my advisor Professor Hong Wang for his continuous encouragement and help of my Ph.D study and related research, for his motivation, patience, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

I would like to thank my dissertation committee members: Professors Qi Wang, LiLi Ju, Yi Sun, and GuiRen Wang, for their insightful comments and valuable suggestions.

I would like to thank all my friends, especially Wei Cheng, XiaoDi Deng, GenSong Gao, Hao Tian, YongQiang Ren, LiWei Wang, Su Yang, Jia Zhao, and XuHao Zhang. They were always willing to help and give their best suggestions.

Last but not the least, I am deeply thankful to my family for their love, support, and sacrifices.

Abstract

In previous studies, scientists developed the classical solid mechanic theory. The model has been widely used in scientific research and practical production. The main assumption of the classical theory of solid mechanics is that all internal forces act through zero distance. Because of this assumption, the mathematical model always leads to partial differential equations, which meet with problems when describing the spontaneous formation of discontinuities and other singularities.

A peridynamic model was proposed as a reformation of solid mechanics [40, 41, 43, 44, 45], which leads to a non-local framework that does not explicitly involve the notion of deformation gradients, and so provides a more accurate description of the problems[5, 15, 27].

With a steady state, the two dimensional non-local diffusion model has the same properties as the peridynamic model[13, 14]. Hence, we can consider it as a scalar-valued version of a peridynamic model. Our discussion will focus on the one dimensional peridynamic model and two dimensional non-local diffusion model.

Starting in the 1970s, scientists began to focus on the research of numerical simulation of integral or boundary integral equations by collocation methods and Galerkin finite element methods[7, 24, 25, 47]. After the peridynamic model was developed, an enormous amount of research effort went to the peridynamic and its numerical simulation[17, 18, 42]. It was found that there are close relations between the peridynamic model[22], non-local diffusion model and fractional partial differential equations [12, 13, 19, 28, 29, 30, 31, 32, 33, 35, 37]. In contrast to those for classical elasticity models of solid mechanics and integer-order partial differential equations, numerical

methods for peridynamic models, like those for space-fractional partial differential equations [19, 28, 29, 30, 32, 33], usually generate dense or full stiffness matrices for which widely used direct solvers typically require $O(N^3)$ operations and $O(N^2)$ memory storage where N refers to the number of unknowns.

A simplified peridynamic model was proposed to reduce the computational cost and memory requirement of the corresponding numerical methods, in which the horizon of the material δ in the peridynamic model was assumed to be $\delta = O(N^{-1})$ [10]. The advantage of the simplified model was that it reduced the computational cost and memory requirement to $O(N)$, but at the cost of a reduced convergence rate of their numerical approximation. Furthermore, it is not clear from the physical relevance that the material property (the radius of the horizon) can be assumed to be of the same order as the numerical mesh size.

In previous research, we developed a fast numerical method for the constant-coefficient one dimensional peridynamic model and two dimensional non-local diffusion model[50, 49, 46], which reduced the memory requirement from $O(N^2)$ to $O(N)$, and the computational cost from $O(N^3)$ to $O(NlogN)$. These works relied heavily on the Toeplitz-like structure of the stiffness matrix like fractional partial differential equations[48, 53, 51, 52].

After Mengesha and Du developed the variable-coefficient peridynamic model[34], we found that the stiffness matrix was no longer in Toeplitz structure, and our fast method could not be applied. Moreover, the variable coefficient also increased the computational work of evaluating the entries of the stiffness matrix. Since our fast method significantly reduced the computational complexity and memory requirement of the constant coefficient models, we hope to reform the stiffness matrix in order to explore useful structure and improve the efficiency of the numerical simulation.

In this thesis, we will briefly introduce the tools we will use in developing the fast method in Chapter 1, and show the general idea of our fast method by implementing it on a one dimensional peridynamic model in Chapter two. In Chapter three, we will derive the fast method for a special case of a one dimensional peridynamic model by collocation numerical simulation. Finally, we will extend the fast method to a two dimensional non-local diffusion model in Chapter 4.

Table of Contents

List of Tables

Chapter 1

Review of useful tools

1.1 Introduction

In our fast method, several numerical strategies and methods are used, such as collocation method, Toeplitz matrix, and Krylov subspace iteration method[1, 8, 11, 20, 26, 36, 38]. We shall give a brief introduction of those tools in this chapter.

1.2 Recall about collocation method

The collocation method has been used in many problems in applied mathematics. It gives us a clear idea when facing some complicate problems. Here let us recall the general idea of the collocation method. For the equation system

$$\mathcal{A}x = b \tag{1.1}$$

with any linear operator $\mathcal{A}$, such as an integral operator or differential operator, where b is given and x is going to be solved. To solve the problem numerically, we selected a set of base vectors $\{v_1, v_2, ..., v_n\}$, and x can be written as

$$x = c_1v_1 + c_2v_2 + ... + c_nv_n, \tag{1.2}$$

since $\mathcal{A}$ is a linear operator, we can rewrite

$$\mathcal{A}x = \sum_{j=1}^{n} c_j\mathcal{A}v_j, \tag{1.3}$$

then the equation (1.1) shall be written as

$$\sum_{j=1}^{n} c_j\mathcal{A}v_j = b. \tag{1.4}$$

As $\{v_1, v_2, ..., v_n\}$ are selected base vectors, the problem now is coming to solve for those coefficients $c_1, c_2, ..., c_3$. But it seems impossible to solve the equation system (1.4). Here we try to make (1.4)almost true.

We now suppose that in some selected points t_i, the right hand side term b has the same value with left hand side $\sum_{j=1}^{n} c_j \mathcal{A} v_j$, since the base vectors v_j, x and b, are all functions on the same domain. It can be written as

$$\sum_{j=1}^{n} c_j (\mathcal{A} v_j)(t_i) = b(t_i), 1 \leq i \leq n. \tag{1.5}$$

Then the previous equation system becomes to a linear equation system, which involves n linear equations. We can find out n unknown coefficients c_j, by solving the new equation system. This equation system can obviously be solved as long as the stiffness matrix with non-singular entries $(\mathcal{A} v_j)(t_i)$, which results from the chosen basic functions v_j, and selected points t_i.

1.3 Introduction of The Toeplitz matrix

Defination. An n-by-n matrix $\mathcal{A}_n = [a_{k,j}; k, j = 0, 1, ..., n-1]$, where $a_{k,j} = a_{k-j}$ is said to be Toeplitz if

$$\mathcal{A}_n = \begin{pmatrix} a_0 & a_{-1} & \dots & a_{2-n} & a_{1-n} \\ a_1 & a_0 & a_{-1} & \dots & a_{2-n} \\ \vdots & a_{-1} & a_0 & \ddots & \vdots \\ a_{n-2} & & \ddots & \ddots & a_{-1} \\ a_{n-1} & a_{n-2} & \dots & a_1 & a_0 \end{pmatrix}; \tag{1.6}$$

i.e., $\mathcal{A}_n$ is constant along with its diagonals. The name Toeplitz was first used in the early 1900s because Otto Toeplitz's work on bilinear forms of Laurent series.

Toeplitz matrices can be found in many different applications. For example, to consider a matrix and vector formulation for a discrete time convolution, the discrete time input and discrete time filter can be denoted as a column vector

$$x = (x_0, x_1, ..., x_{n-1})' = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{pmatrix} \tag{1.7}$$

and a Toeplitz matrix

$$T_n = \begin{pmatrix} t_0 & 0 & 0 & \dots & 0 \\ t_1 & t_0 & 0 & & \\ t_2 & t_1 & t_0 & & \vdots \\ \dots & & & \ddots & \\ t_{n-1} & & & \dots & t_0 \end{pmatrix} \tag{1.8}$$

. Then the matrix and vector formulation can be represented as a product of T_n and x, where

$$y = T_n x = \begin{pmatrix} t_0 & 0 & 0 & \dots & 0 \\ t_1 & t_0 & 0 & & \\ t_2 & t_1 & t_0 & & \vdots \\ \dots & & & \ddots & \\ t_{n-1} & & & \dots & t_0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{pmatrix} = \begin{pmatrix} x_0 t_0 \\ t_1 x_0 + t_0 x_1 \\ \sum_{i=0}^{2} t_{2-i} x_i \\ \vdots \\ \sum_{i=0}^{n-1} t_{2-i} x_i \end{pmatrix} \tag{1.9}$$

, and each entry $y_k = \sum_{i=0}^{k} t_{k-i} x_i$ can be seen as the output of the discrete time and causal time invariant filter h with "impulse response" t_k.

We can easily find more applications of Toeplitz system in engineering and mathematics. In the signal Processing, let a discrete time random process X_n, and $m_k = E(X_K)$, $K_X(k,j) = E[(X_k - m_k)(X_j - m_j)]$ be the mean function and covariance function of the process. Many signal process theories are based on the assumption that the mean is constant and the covariance is Toeplitz, which is $K_X(k,j) = K_X(k-j)$. Since we assume that the covariance is Toeplitz, the covariance stationary and second order stationary are also used. Then the n by n covariance matrices $K_n = [K_X(k,j); k,j = 0,1,...,n-1]$ are Toeplitz matrices.

There is a very useful special case of Toeplitz matrix called circulant matrix. A matrix C_n is called circulant matrix, if it is in the form below

$$C_n = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & \dots & t_{-(n-1)} \\ t_{-(n-1)} & t_0 & t_{-1} & & \\ t_{-(n-2)} & t_{-(n-1)} & t_0 & & \vdots \\ \dots & & & \ddots & \\ t_{-1} & t_{-2} & & \dots & t_0 \end{pmatrix} \tag{1.10}$$

where $t_k = t_{-(n-k)} = t_{k-n}$ for $k = 1,2,...,n-1$. This kind of matrix may be arose in the applications which are related to the discrete Fourier transformation and the study of cyclic codes for error correction.

We are going to use the special structure of Toeplitz matrices and circulant matrices to derive our fast faithful method, in solving the Toeplitz system $\mathcal{A}x = b$.

Many works have already been done in a variety of applications. In time series analysis, we need to solve Toeplitz systems in order to get the unknown parameters of stationary autoregressive models. We shall need to obtain the filter coefficients in designing recursive digital filters by solving Toeplitz systems. And also, in this paper, solving Toeplitz systems is the key part in order to get the numerical solution of the non-local diffusion model.

1.4 Motivation of accelerating the computing of solving Toeplitz systems

At the first seeing of a Teoplitz matrix system $\mathcal{A}x = b$ with an n by n Toeplitz matrix $\mathcal{A}$, where

$$\mathcal{A} = \begin{pmatrix} a_0 & a_{-1} & \dots & a_{2-n} & a_{1-n} \\ a_1 & a_0 & a_{-1} & \dots & a_{2-n} \\ \vdots & a_{-1} & a_0 & \ddots & \vdots \\ a_{n-2} & & \ddots & \ddots & a_{-1} \\ a_{n-1} & a_{n-2} & \dots & a_1 & a_0 \end{pmatrix} \tag{1.11}$$

, as a full matrix, if we consider the direct solver such as Gaussian elimination, the matrix equation system will result in an algorithm of $\mathcal{O}(n^3)$ depending on our discretization. If the number of partition n goes to large, the cost will become very expansive. And in normal cases, we need $\mathcal{O}(n^2)$ of storage space.

Whereas the special structure of Toeplitz matrices, the matrix in our problem is actually just determined by $(2n-1)$ entries. To consider a more efficient storage, we can simply store these information by a $(2n-1)$ vector, but it will bring some difficulties when we need to implement the direct solver. And in the consideration of computational cost, we for surely expect to obtain the solution of the Toeplitz system in less than $\mathcal{O}(n^3)$ operations.

There are some early works focusing on the direct method for solving Toeplitz systems[4, 9, 20]. From 1940s on, the complexity has been decreased to $\mathcal{O}(n^2)$ operations by numbers of mathematicians such as Levinson(1946), Baxter(1961), Trench(1964) and Zohar(1974). The invertibility of the $(n-1)$ by $(n-1)$ principle sub-matrix of $\mathcal{A}$ is required by these algorithms. The computational cost has been reduced to $\mathcal{O}(n\log^2 n)$ by Brent, Gustavson, and Yun(1980), Bitmead and Anderson(1980), Morf(1980), de Hoog(1987), and Ammar and Gragg(1988) around 1980th. The invertibility of the $\lfloor n/2 \rfloor$ by $\lfloor n/2 \rfloor$ principle sub-matrix of $\mathcal{A}$ is required by these algorithms.

In this paper, the non-local diffusion model is being discussed, some works also have been done to accelerate the computation. For peridynamic model, Chen and Gunzburger proposed a finite element method by assuming the horizon of the material $\delta = Mh$. Here M is a constant number, this method reduces the computational cost to $M^2\mathcal{O}(N)$, and reduces the memory cost to $M\mathcal{O}(N)$. As we derived in previous chapter, in order to solve the non-local diffusion model numerically, we discretized the integral equation, then trying to solve the matrix system $\mathcal{A}x = b$, where the stiffness matrix $\mathcal{A}$ is a Toeplitz matrix.

If we did not consider the Toeplitz structure of the stiffness matrix, we suppose to use conjugate gradient method to solve the system $\mathcal{A}x = b$. Since the stiffness matrix is dense, and the computation of the matrix-vector multiplication will be $\mathcal{O}(N^2)$, but others only need $\mathcal{O}(N)$ computational work. Therefore, by considering the Krylov subspace method, the key point is to use the special structure of the stiffness matrix, and to accelerate the computation of matrix-vector multiplication. Then we can improve the computational work of the whole scheme.

1.5 The conjugate gradient method

In searching of numerical solutions for mathematical models, such as solving differential and integral equations, we always come to solve the matrix equation $\mathcal{A}x = b$. Many direct solvers have been explored, and lots of special methods depend on the structures of the stiffness matrices those have been used in this area. In this paper, the problem is about the Toeplitz matrix, which can be embedded into a symmetric matrix. Hence we are going to consider one of the most popular method, the Hestenes-Stiefel conjugate gradient method.

The method of successive over-relaxation, Chebyshev semi-iterative and many related methods have been used in these kind of problems. The difficulty is that they usually depend on parameters, sometimes hard to be chosen properly. For example, the Chebyshev acceleration scheme needs good estimates of the largest and smallest eigenvalues of the underlying iteration matrix $M^{-1}N$. However, the conjugate gradient method will not have these kind of difficulties.

We shall recall the conjugate gradient method in this section.

Discussion of solving a linear system

To solve the matrix equation $\mathcal{A}x = b$ with a positive definite stiffness matrix, we are going to consider the function

$$\phi(x) = \frac{1}{2}x^T\mathcal{A}x - x^T b, \tag{1.12}$$

where $b \in \mathbb{R}$, and $\mathcal{A} \in \mathbb{R}^{n\times n}$ which is positive definite and symmetric matrix. Because $x = \mathcal{A}^{-1}b$ minimized the equation ϕx, where $\phi(x) = -b^T\mathcal{A}b/2$. Solving the problem $\mathcal{A}x = b$ is equivalent to minimize the equation $\phi(x)$, if $\mathcal{A}$ is symmetric and positive definite.

We shall a searching direction in order to use the iteration method, the up coming idea is the method of steepest descent, at some point x_p, the function $\phi(x)$ shall decreases most rapidly in the direction of the negative gradient,

$$-\nabla\phi(x_p) = b - \mathcal{A}x_p. \tag{1.13}$$

Here we name that

$$r_p = -\nabla\phi(x_p) = b - \mathcal{A}x_p, \tag{1.14}$$

the residual of x_p. There is a positive α, such that $\phi(x_p + \alpha r_p) < \phi(x_p)$, if r_p is not zero. Then we define

$$\alpha = r_p^T r_p / r_p^T\mathcal{A}r_p$$

, and plug it back to $\phi(x_p + \alpha r_p)$. We have

$$\phi(x_p + \alpha r_p) = \phi(x_p) - \alpha r_p^T r_p + \frac{1}{2}\alpha^2 r_p^T r_p / r_p^T\mathcal{A}r_p$$

.

Now we can conclude a simplest iteration scheme as the following

$$\begin{aligned}
&set\ x_0 = initial\ guess \\
&r_0 = b - \mathcal{A}x_0 \\
&k = 0 \\
&while\ r_k \neq 0 \\
&k = k + 1 \\
&\alpha_k = r_{k-1}^T r_{k-1} / r_{k-1}^T \mathcal{A} r_{k-1} \\
&x_k = x_{k-1} + \alpha_k r_k \\
&r_k = b - \mathcal{A}x_k \\
&end
\end{aligned} \tag{1.15}$$

and the global convergence can be given as

$$(\phi(x_k) + \frac{1}{2}b^T\mathcal{A}^{-1}b) \leq (1 - \frac{1}{\kappa_2(\mathcal{A})})(\phi(x_{k-1}) + \frac{1}{2}b^T\mathcal{A}^{-1}b). \tag{1.16}$$

However, we can also find that when the condition number $\kappa_2(\mathcal{A}) = \lambda_1(\mathcal{A})/\lambda_n(\mathcal{A})$ is large, the rate of convergent will slow down significantly.

This means that in searching of the lowest point in a relatively flat, steep-sided valley. The method of steepest descent always made us forth across or traverse back the valley, but not go down to the valley. In another word, the gradient direction is not always a good direction to search for the result. In order to avoid this kind of problem, we are going to find a direction which not necessarily depends on the residual.

We suppose to search the result which shall minimize $\phi(x)$ along a set of directions $\{p_1, p_2, ...\}$, these directions do not necessarily correspond to the residuals $\{r_0, r_1, ...\}$. Then we set

$$\begin{aligned}\alpha &= \alpha_k \\ &= p_k^T r_{k-1}/p_k^T \mathcal{A} p_k.\end{aligned} \tag{1.17}$$

This shall minimize $\phi(x_k + \alpha_k)$, and we can easily see that

$$\phi(x_{k-1} + \alpha_k p_k) = \phi(x_{k-1}) - \frac{(p_k^T r_{k-1})^2}{2 p_k^T \mathcal{A} p_k}, \tag{1.18}$$

and we need to set p_k that is not to be orthogonal to r_{k-1}, in order to make sure that $\phi(x)$ is successively reduced in these directions. Then the following scheme can be given

$$\begin{aligned}
& x_0 = initial\ guess \\
& r_0 = b - \mathcal{A} x_0 \\
& k = 0 \\
& while\ r_k \neq 0 \\
& k = k + 1 \\
& Choose\ a\ direction\ p_k\ such\ that\ p_k^T r_{k-1} \neq 0 \\
& \alpha_k = p_k^T r_{k-1}/p_k^T \mathcal{A} p_k \\
& x_k = x_{k-1} + \alpha_k p_k \\
& r_k = b - \mathcal{A} x_k \\
& end,
\end{aligned} \tag{1.19}$$

where we see

$$\begin{aligned} x_k \ &\in \ x_0 + span\{p_1, ..., p_k\} \\ &= \{x_0 + \gamma_1 p_1 + ... + \gamma_k p_k : \gamma_i \in \mathbb{R}\} \end{aligned} \tag{1.20}$$

Now the key point is to find the searching directions that could avoid the problem of the steepest descent.

Discussion about the Searching Direction of Conjugate Gradient Method

From the previous discussion, we know that in order to get an n-th step convergent about the problem $\mathcal{A}x = b$, it is sufficient to search for an x_n, which could minimize $\phi(x)$ over $\mathbb{R}^n$. therefore, we only need to suppose that x_k solved the problem

$$\min_{x \in x_0 + span\{p_1, p_2, ..., p_k\}} \phi(x), \tag{1.21}$$

where $p_i, i = 1, ..., k$ are linear independent. Then the scheme will be guaranteed to be convergent in at most n steps.

Since we suppose to have a method which should be easy enough to be implemented. And it is an iteration method, which means by given x_{k+1}, we could compute x_k without expensive computational cost. Then we set

$$x_k = x_0 + P_{k-1}y + \alpha_k, \tag{1.22}$$

with $y \in \mathbb{R}^{k-1}$, $P_{k-1} = [p_1, p_2, ..., p_k]$, and $\alpha \in \mathbb{R}$. Substitute x in $\phi(x)$ by x_k, we can have

$$\phi(x_k) = \phi(x_0 + P_{k-1}y) + \alpha y^T P_{k-1}^T \mathcal{A} p_k + \frac{\alpha^2}{2} p_k^T \mathcal{A} p_k - \alpha_k^T r_0. \tag{1.23}$$

In a simple case, we just let $p_k \in span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}^\perp$, and the second term of previous equation becomes to zero, then the key part of minimization problem of the k-yh step can be expressed as follows

$$\begin{aligned}
\min_{x_k \in x_0 + span\{p_1, p_2, ..., p_k\}} \phi(x_k) &= \min_{y,\alpha} \phi(x_0 + P_{k-1}y + \alpha_k) \\
&= \min_{y,\alpha} (\phi(x_0 + P_{k-1}y) + \frac{\alpha^2}{2} p_k^T \mathcal{A} p_k - \alpha_k^T r_0) \qquad (1.24) \\
&= \min_{y} \phi(x_0 + P_{k-1}y) + \min_{\alpha} (\frac{\alpha^2}{2} p_k^T \mathcal{A} p_k - \alpha_k^T r_0),
\end{aligned}$$

involving two separate minimization problems.

Now let us consider the first part of the problem, that $x_{k-1} = x_0 + P_{k-1}y_{k-1}$ minimizes ϕ over $x_0 + span\{p_1, p_2, ..., p_{k-1}\}$ if y_{k-1} is the solution of that part. For the second part, by the following A-conjugacy,

$$\begin{aligned} p_k^T r_{k-1} &= p_k^T(b - \mathcal{A}x_{k-1}) \\ &= p_k^T(b - \mathcal{A}(x_0 + P_{k-1}y_{k-1})) \\ &= p_k^T r_0 \end{aligned} \tag{1.25}$$

we shall have a result, $\alpha_k = p_k^T r_0 / p_k^T \mathcal{A} p_k$.

Then we can get the following scheme with $x_k = x_{k-q} + \alpha_k p_k$

$$\begin{aligned} &x_0 = initial\ guess \\ &k = 0 \\ &r_0 = b - \mathcal{A}x_0 \\ &while\ r_k \neq 0 \\ &k = k + 1 \\ &Choose p_k \in span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}^\perp\ so\ p_k^T r_{k-1} \neq 0 \\ &\alpha_k = p_k^T r_{k-1} / p_k^T \mathcal{A} p_k \\ &x_k = x_{k-1} + \alpha_k p_k \\ &r_k = b - \mathcal{A}x_k \\ &end. \end{aligned} \tag{1.26}$$

Here let us introduce the following lemma, which can support us to find the possible direction.

Lemma If $r_k \neq 0$, there exists a $p_k \in span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}^\perp$, such that $p_k^T r_{k-1} \neq 0$.

Simply proved by the following recurrence method. For $k = 1$, we set $p_1 = r_0$, then for $k > 1$, because of $r_{k-1} \neq 0$, we have

$$\mathcal{A}^{-1}b \notin x_0 + span\{p_1, p_2, ..., p_{k-1}\}.$$

Then we have

$$b \notin \mathcal{A}x_0 + span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}$$

which implies

$$r_0 \notin span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}$$

.

Hence we can say that there exists a $p \in span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}^{\perp}$, such that $p^T r_{k-1} \neq 0$. Moreover, since

$$x_{k-1} \in x_0 + span\{p_1, p_2, ..., p_{k-1}\}$$

,

it is true that

$$r_{k-1} \in r_0 + span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}$$

,

and

$$p^T r_{k-1} = p^T r_0 \neq 0$$

.

In the scheme (1.26), the searching direction is named A-conjugate, since $p_i^T \mathcal{A} p_j = 0$ for all $i \neq j$.

Also we can easily see that

$$P_k^T \mathcal{A} P_k = diag(p_1^T \mathcal{A} p_1, p_2^T \mathcal{A} p_2, ..., p_k^T \mathcal{A} p_k)$$

is non-singular because of the properties of $\mathcal{A}$ and $p_i \neq 0$, then it follows that P_k is full column ranked. The convergence of the scheme (1.26) can be guaranteed in at most n steps, since x_n shall minimize $\phi(x)$ over $\mathbb{R}^n$.

Based on the discussion above, we almost achieve our goal. Here easily comes to an idea that to combine the method of steepest descent and the A-conjugate scheme, we suppose to choose p_k with restricted condition in the scheme (1.26), and to be the closest vector to r_{k-1}. By these constrains, we shall define the following "conjugate gradient" scheme,

$$\begin{aligned}
&x_0 = initial\ guess \\
&k = 0 \\
&r_0 = b - \mathcal{A}x_0 \\
&while\ r_k \neq 0, \quad k = k+1 \\
&if\ k = 1 \\
&p_1 = r_0 \\
&else \quad let \\
&p_k \in span\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\}^{\perp} \\
&end \\
&\alpha_k = p_k^T r_{k-1} / p_k^T \mathcal{A} p_k \\
&x_k = x_{k-1} + \alpha_k p_k \\
&r_k = b - \mathcal{A}x_k \\
&end \\
&x = x_k
\end{aligned} \tag{1.27}$$

The remaining work is to find an efficient method to compute the searching direction p_k.

Lemmas and Theorems

In this subsection, we are going to introduce several important lemmas and theorems to derive the faithful scheme of conjugate gradient method. First of all, let us recall the Krylov subspace, which is defined by

$$\mathcal{K}(r_0, \mathcal{A}, k) = span\{r_0, \mathcal{A}r_0, \mathcal{A}^2 r_0, ..., \mathcal{A}^{k-1} r_0\}.$$

Lemma 1.1. *For $k \geq 2$, the vectors p_k are generated by* (1.27)*, which satisfy*

$$p_k = r_{k-1} - \mathcal{A}P_{k-1} z_{k-1},$$

where $P_k = [p_1, p_2, ..., p_{k-1}]$, and z_{k-1} solves the least square problem

$$\min_{z \in \mathbb{R}^{k-1}} ||r_{k-1} - \mathcal{A}P_{k-1} z||_2.$$

Proof. Here we set p be the associated minimum residual

$$p = r_{k-1} - \mathcal{A}P_{k-1} z_{k-1},$$

and by the assumption that z_{k-1} is the solution of the previous least square problem, then we have

$$p^T \mathcal{A}P_{k-1} = 0$$

.

Furthermore, the orthogonal projection of r_{k-1} into $ran(\mathcal{A}P_{k-1})^{\perp}$

$$p = [I - (\mathcal{A}P_{k-1})(\mathcal{A}P_{k-1})^{+}] r_{k-1}$$

is the closest vector in $ran(\mathcal{A}P_{k-1})^{\perp}$ to r_{k-1}, then

$$p = p_k$$

. ∎

By this lemma, we can derive many important relationships between the searching direction p_k, residual r_k, and the Krylov subspace.

Next, let us recall the most important theorem, which can help us to show that p_k could be a linear combination of the direction p_{k-1} of last step and the current residual r_{k-1}.

Theorem 1.2. *After k iterations of the schem* (1.27)*, we have*

$$r_k = r_{k-1} - \alpha_k \mathcal{A} p_k \tag{1.28}$$

$$p_k^T r_k = 0 \tag{1.29}$$

$$\begin{aligned} span\{p_1, p_2, ..., p_k\} &= span\{r_1, r_2, ..., r_{k-1}\} \\ &= \mathcal{K}(r_0, \mathcal{A}, k) \\ &= span\{r_0, \mathcal{A}r_0, \mathcal{A}^2 r_0, ..., \mathcal{A}^{k-1} r_0\} \end{aligned} \tag{1.30}$$

and the residual $r_0, r_1, ..., r_k$ are mutually orthogonal.

Proof. By the definition of residual $r_p = b - \mathcal{A}x_p$, we have

$$\begin{aligned} r_k &= b - \mathcal{A}x_k \\ &= b - \mathcal{A}(x_{k-1} + \alpha_k p_k) \\ &= b - \mathcal{A}x_{k-1} - \alpha_k \mathcal{A} p_k \\ &= r_{k-1} - \alpha_k \mathcal{A} p_k \end{aligned} \tag{1.31}$$

since $x_k = x_{k-1} + \alpha_k p_k$. Which is (1.28).

And for (1.29), we use that

$$x_k = x_0 + P_k y_k$$

, where y_k minimized the equation

$$\phi(x_0 + P_k y) = \phi(x_0) + \frac{1}{2} y^T (P_k^T \mathcal{A} P_k) y - y^T P_k (b - \mathcal{A} x_0),$$

however, it means that y_k is the solution of the linear system

$$(P_k^T \mathcal{A} P_k) y = P_k^T (b - \mathcal{A} x_0)$$

.

Therefore, we have

$$\begin{aligned} 0 &= P_k^T (b - \mathcal{A} x_0) - (P_k^T \mathcal{A} P_k) y_k \\ &= P_k^T (b - \mathcal{A}(x_0 + P_k y_k)) \\ &= P_k^T r_k. \end{aligned} \tag{1.32}$$

It is true that

$$\{\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}\} \subset span\{r_0, r_1, ..., r_{k-1}\}$$

, because of (1.28). By the lemma, we shall see

$$\begin{aligned} p_k &= r_{k-1} - [\mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-1}] z_{k-1} \\ &\in span\{r_0, r_1, ..., r_{k-1}\}, \end{aligned} \tag{1.33}$$

and then we have

$$[p_1, p_2, ..., p_k] = [r_0, r_1, ..., r_{k-1}] T,$$

where T is an upper triangular matrix. Because of the non-singularity of T, and the independence of the searching directions, we could conclude that

$$span\{p_1, p_2, ..., p_k\} = span\{r_0, r_1, ..., r_{k-1}\}$$

.Again by (1.28), we say that

$$\begin{aligned} r_k &\in span\{r_{k-1}, \mathcal{A}p_k\} \\ &\subset span\{r_{k-1}, \mathcal{A}r_0, ..., \mathcal{A}r_{k-1}\}. \end{aligned} \tag{1.34}$$

Then, by induction we can get the Krylov space connection. And by (1.29), (1.30) we can easily get the orthogonality of the residuals.

□

Theorem 1.3. *The residual and searching direction in* (1.27) *have the property that* $p_k \in span\{\partial_{k-1}, r_{k-1}\}$ *for* $k \geq 2$.

Proof. We shall prove this from $k = 2$, since $p_2 \in span\{r_0, r_1\}$ from (1.30), and $p_1 = r_0$, then $p_2 \in span\{p_, r_1\}$. And for $k > 2$, we separate the vector z_{k-1} from the lemma to

$$z_{k-1} = [\omega,\ \mu]^T$$

where ω is an $1 - by - (k-2)$ vector and μ is a single number. And because of

$$r_{k-1} = r_{k-2} - \alpha_{k-1}\mathcal{A}p_{k-1}$$

,

we can have

$$
\begin{aligned}
p_k &= r_{k-1} - \mathcal{A}P_{k-1}z_{k-1} \\
&= r_{k-1} - \mathcal{A}P_{k-2}\omega - \mu\mathcal{A}p_{k-1} \\
&= (1 + \frac{\mu}{\alpha_{k-1}})r_{k-1} + s_{k-1},
\end{aligned}
\tag{1.35}
$$

where

$$
\begin{aligned}
s_{k-1} &= -\frac{\mu}{\alpha_{k-1}}r_{k-2} - \mathcal{A}P_{k-2}\omega \\
&\in span\{r_{k-2}, \mathcal{A}P_{k-2}\omega\} \\
&\subset span\{r_{k-2}, \mathcal{A}p_1, \mathcal{A}p_2, ..., \mathcal{A}p_{k-2}\} \\
&\subset span\{r_1, r_2, ..., r_{k-2}\},
\end{aligned}
\tag{1.36}
$$

it is obvious that s_{k-1} and r_{k-1} are orthogonal to each other, since r_i are mutually orthogonal. Then we just need to find μ and ω, which minimized

$$||p_k||_2^2 = (1 + \frac{\mu}{\alpha_{k-1}})^2||r_{k-1}||_2^2 + ||s_{k-1}||_2^2.$$

We conclude that $p_k \in span\{r_{k-1}, p_{k-1}\}$. Because s_{k-1} is a multiple of p_{k-1}, which because z_{k-2} minimized the 2-norm of $r_{k-2} - \mathcal{A}P_{k-2}z$ by giving residual p_{k-1}.

□

The Conjugate Gradient Method

By the theorem before, we can choose that

$$p_k = r_{k-1} + \beta_k p_{k-1},$$

by the A-conjugacy $p_{k-1}^T\mathcal{A}p_k = 0$, we have

$$\beta_k = -\frac{p_{k-1}^T \mathcal{A} r_{k-1}}{p_{k-1}^T \mathcal{A} p_{k-1}},$$

then the conjugate gradient scheme can be given as the following

$$\begin{aligned}
&x_0 = initial\ guess \\
&k = 0 \\
&r_0 = b - \mathcal{A}x_0 \\
&while\ r_k \neq 0 \\
&k = k + 1 \\
&if\ \ k = 1 \\
&p_1 = r_0 \\
&else \\
&\beta_k = -p_{k-1}^T \mathcal{A} r_{k-1} / p_{k-1} T \mathcal{A} p_{k-1} \\
&p_k = r_{k-1} + \beta_k p_{k-1} \\
&end \\
&\alpha_k = p_k^T r_{k-1} / p_k^T \mathcal{A} p_k \\
&x_k = x_{k-1} + \alpha_k p_k \\
&r_k = b - \mathcal{A}x_k \\
&end \\
&x = x_k
\end{aligned} \tag{1.37}$$

Since it needs three separate matrix-vector multiplications of each iteration in the above scheme, we will use $r_k = r_{k-1} - \alpha \mathcal{A} p_k$ to derive out that

$$r_{k-1}^T r_{k-1} = -\alpha_{k-1} r_{k-1}^T \mathcal{A} p_{k-1}, \tag{1.38}$$

and

$$r_{k-2}^T r_{k-2} = \alpha_{k-1} p_{k-1}^T \mathcal{A} p_{k-1}. \tag{1.39}$$

Then the conjugate gradient method scheme will be written as

$$\begin{aligned}
&x_0 = initial\ guess\\
&k = 0\\
&r_0 = b - \mathcal{A}x_0\\
&while\ r_k \neq 0\\
&k = k + 1\\
&if\ k = 1\\
&p_1 = r_0\\
&else\\
&\beta_k = r_{k-1}^T r_{k-1} / r_{k-2}^T r_{k-2}\\
&p_k = r_{k-1} + \beta_k p_{k-1}\\
&end\\
&\alpha_k = r_{k-1}^T r_{k-1} / p_k^T \mathcal{A} p_k\\
&x_k = x_{k-1} + \alpha_k p_k\\
&r_k = r_{k-1} - \alpha_k \mathcal{A} p_k\\
&end\\
&x = x_k
\end{aligned} \tag{1.40}$$

And we shall use this version of the scheme of the conjugate gradient method in future computation.

Convergence Analysis

The convergence of the conjugate gradient method is shown by the following two famous theorems.

Theorem 1.4. *If $\mathcal{A} = \mathcal{I} + \mathcal{B}$ is an $n-by-n$ symmetric positive definite matrix and $rank(\mathcal{B}) = r$, then the scheme* (1.40) *converges in at most $r+1$ steps.*

After we defined the A-norm as

$$||\omega||_A = \sqrt{\omega^T A \omega}$$

, the error estimation can be given by the following theorem.

Theorem 1.5. *Suppose $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite and $b \in \mathbb{R}$. If the scheme* (1.40) *produces iterates $\{x_k\}$ and $\kappa = \kappa_2(A)$ then*

$$||x - x_k||_A \leq 2||x - x_0||_A (\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1})^k. \tag{1.41}$$

In conclusion, the conjugate gradient method converges rapidly in the A-norm if $\kappa_2(A) \approx 1$.

1.6 A brief discussion of Krylov subspace iteration method

By previous study, we observed that the conjugate gradient method is the ideal method in solving the constant-coefficient peridynamic problems. Since the constant-coefficient peridynamic problems always lead to a symmetric positive definite stiffness matrix. However, in later discussion we shall find that the stiffness matrix of a variable-coefficient peridynamic or non-local diffusion problem is even not symmetric, and always differs from entry to entry. We then hope to choose a similar Krylov subspace method to solve our nonsymmtric linear systems.

There are many Krylov subspace methods have been developed for nonsymmetric linear systems. CGNR, CGNE and GMRES are known for their easily implementations. The stability of these methods are also conspicuous. Nevertheless, we cannot choose these methods for our fast method, since the schemes of CGNR and CGNE involve a transpose-vector product in each iteration, and the coefficient matrix we used is $A^T A$ or AA^T, the computation has been increased and the condition numbers have been squared. For GMRES, the storage of requirement will be huge when we are going to solve an ill-conditioned problem.

We suppose to implement an ideal iteration method, we only need matrix-vector multiplications, and the storage requirement should not depend on the number of the iterations. Bi-CG has been considered at the beginning, and we have the following algorithm

$$\begin{aligned}
&x_0 = initial\ guess \\
&k = 0 \\
&r = b - \mathcal{A}x_0 \\
&\hat{r} = r \\
&\rho_0 = 1 \\
&\hat{p} = p = 0 \\
&while||r||_2 > \epsilon||b||_2 \quad and \quad k < kmax \\
&k = k + 1 \\
&\rho_k = \hat{r}^T r,\ \beta = \rho_k/\rho_{k-1} \\
&p = r + \beta p,\ \hat{p} = \hat{r} + \beta\hat{p} \\
&v = \mathcal{A}p \\
&\alpha = \rho_k/(\hat{p}^T v), \quad x = x + \alpha p \\
&r = r - \alpha v;\ \hat{r} = \hat{r} - \alpha\mathcal{A}^T\hat{p}.
\end{aligned} \tag{1.42}$$

Here the stiffness matrix $\mathcal{A}$ does not need to be symmetric and positive definite. However, a transpose-vector product of the stiffness matrix $\mathcal{A}$ is needed. This will cause more computational work. Furthermore, the transpose of $\mathcal{A}$ may not exist sometimes.

Finally, we come to a remedy of Bi-CG method, named CGS. In order to avoid the transpose of $\mathcal{A}$, we shall use the observation that there is $\bar{p_k} \in \mathcal{P}_k$ such that

$$r_k = \bar{p}_k(\mathcal{A})r_0 \quad and \quad \hat{r}_k = \bar{p}_k(\mathcal{A}^T)\hat{r}_0,$$

then the scalar product $\hat{r}^T r$ in (1.42) can be written as

$$r_k^T \hat{r}_k = (\bar{p}_k(\mathcal{A})r_0)^T(\bar{p}_k(\mathcal{A}^T)\hat{r}_0) = (\bar{p}_k(\mathcal{A}^2)\hat{r}_0)^T \hat{r}_0.$$

Here $\mathcal{A}$ can be eliminated and we get the algorithm of CGS method

$$\begin{aligned}
&x_0 = initial\ guess \\
&p_0 = u_0 = r_0 = b - \mathcal{A}x_0 \\
&v_0 = \mathcal{A}p_0; \quad \hat{r}_0 = r_0 \\
&\rho_0 = \hat{r}_0^T r_0; \quad k = 0 \\
&while ||r||_2 > \epsilon ||b||_2\ and k < kmax \\
&k = k+1, \quad \sigma_{k-1} = \hat{r}_0^T v_{k-1} \\
&\alpha_{k-1} = \rho_{k-1}/\sigma_{k-1}, \quad q_k = u_{k-1} - \alpha_{k-1}v_{k-1} \\
&x_k = x_{k-1} + \alpha_{k-1}(u_{k-1}q_k) \\
&r_k = r_{k-1} - \alpha_{k-1}\mathcal{A}(u_{k-1}q_k) \\
&\rho_k = \hat{r}_0^T r_k; \quad \beta_k = \rho_k/\rho_{k-1} \\
&u_k = r_k + \beta_k q_k \\
&p_k = u_k + \beta_k(q_k + \beta_k p_{k-1}), \quad v_k = \mathcal{A}p_k,
\end{aligned} \tag{1.43}$$

where $\bar{q}_k$ is defined by $\bar{q}_0 = 1$ and for $k \geq 1$ by

$$\bar{q}_k(z) = \bar{p}_k(z) + \beta_k \bar{q}_{k-1}(z).$$

Although this CGS method has some disadvantages, like it is not stable, and breakdowns occur when either ρ_{k-1} or $\delta_k - 1$ vanish, it meets most of our requirement for our fast method.

Chapter 2

A Fast Collocation Method for a Variable Coefficient Peridynamic Model

2.1 Introduction

Compared to the constant-coefficient peridynamic model[23], the superiority of a variable-coefficient peridynamic model is in accounting for the heterogeneity of the elastic material. However, it also requires much more computational cost and storage memory. Our fast method is based on the structure of stiffness matrix of the numerical simulations. In order to show the idea intuitively, we start our discussion with the collocation method of a one dimensional variable-coefficient peridynamic model.

In this chapter we develop a fast numerical method for a variable-coefficient peridynamic model for describing a heterogeneous finite elastic bar. In section two we go over the finite element method for the peridynamic model. In section three we develop a fast method with an efficient matrix assembly and storage. In section four we conduct numerical experiments to investigate the computational benefits of the fast method.

2.2 A variable-coefficient peridynamic model and its collocation discretization

A variable-coefficient peridynamic model for describing a finite heterogeneous microelastic bar is formulated as follows

$$
\begin{aligned}
&\int_{x-\delta}^{x+\delta} (\alpha(x)+\alpha(y)) \frac{(u(x)-u(y))}{|x-y|^{1+\gamma}} dy = f(x), \quad x \in (a,b) \\
&u(x) = g(x), \quad x \in (a-\delta, a] \cup [b, b+\delta).
\end{aligned}
\tag{2.1}
$$

Here $\delta > 0$ refers to the size of the material horizon, the index $\gamma < 1$ specifies the singularity of the kernel. The elasticity coefficient $\alpha(\cdot)$ has positive lower and upper bounds.

Let N be a positive integer. We define a uniform partition $x_i := a + ih$ for $i = -1, 0, 1, \ldots, N, N+1$ with $h := (b-a)/N$. Let $\psi(\xi) := 1 - |\xi|$ for $\xi \in [-1,1]$ and 0 elsewhere be the hat function on the reference element $[-1,1]$. Let $\phi_i(x) := \psi\Big((x-x_i)/h\Big)$. The trial function is

$$
u(x) = \sum_{j=0}^{N} u_j \phi_j(x). \tag{2.2}
$$

We enforce the governing equation in (2.1) at the collocation points $\{x_i\}_{i=1}^{N-1}$ to obtain the following collocation scheme

$$
\int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{u(x_i)-u(y)}{|x_i-y|^{1+\gamma}} dy = f(x_i), \quad 1 \le i \le N-1. \tag{2.3}
$$

We substitute the trial function (2.2) into (2.3) to rewrite (2.3) as follows

$$
\begin{aligned}
&u_i \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{1-\phi_i(y)}{|x_i-y|^{1+\gamma}} dy \\
&\quad - \sum_{j=1, j\neq i}^{N-1} u_j \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_j(y)}{|x_i-y|^{1+\gamma}} dy \\
&= f(x_i) + g(a) \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_0(y)}{|x_i-y|^{1+\gamma}} dy \\
&\quad + g(b) \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_N(y)}{|x_i-y|^{1+\gamma}} dy, \quad 1 \le i \le N-1.
\end{aligned}
\tag{2.4}
$$

The numerical scheme (2.4) can be formulated in the following matrix form

$$
Au = f. \tag{2.5}
$$

Here the unknown vector $u := [u_1, u_2, \dots, u_{N-1}]^T$ with $\{u_j\}_{j=1}^{N-1}$ being given in (2.2), and the stiffness matrix $A := [A_{i,j}]_{i,j=1}^{N-1}$ and the right-hand side $f := [f_1, f_2, \dots, f_{N-1}]^T$ are defined by

$$
\begin{aligned}
A_{i,j} &:= \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_j(y)}{|x_i-y|^{1+\gamma}} dy, \quad j \neq i, \\
A_{i,i} &:= \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{1-\phi_i(y)}{|x_i-y|^{1+\gamma}} dy, \\
f_i &:= f(x_i) + g(a) \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_0(y)}{|x_i-y|^{1+\gamma}} dy \\
&\quad + g(b) \int_{x_i-\delta}^{x_i+\delta} (\alpha(x_i)+\alpha(y)) \frac{\phi_N(y)}{|x_i-y|^{1+\gamma}} dy.
\end{aligned}
\tag{2.6}
$$

We observe from (2.6) that each row of the stiffness matrix A has $O(K)$ nonzero entries, where

$$
K := \lfloor \delta/h \rfloor \tag{2.7}
$$

is the floor of δ/h. We note that $K = O(N)$ as N increases. In other words, the stiffness matrix A is a dense matrix, for which direct solvers require $O(N^2)$ storage and have $O(N^3)$ computational complexity. To develop a fast solution method we shall explore the structure of the stiffness matrix.

The goal of this paper is to develop a fast numerical solution technique for the numerical scheme (2.5)–(2.6), whicn in turn relies on the Toeplitz-like structure of the stiffness matrix. However, in contrast to the case of the constant-coefficient peridynamic model and the non-local diffusion model in which the stiffness matrices of the corresponding numerical schemes automatically have a Toeplitz-like structure [50, 49], it is not clear whether the stiffness matrix of the numerical scheme (2.5)–(2.6) has such a structure. We shall show that with a carefully chosen numerical quadrature, the resulting stiffness matrix actually has a Toeplitz-like structure, and then develop a corresponding fast solution technique for (2.5)–(2.6).

In this section we shall study an efficient evaluation of the stiffness matrix. Recall that the stiffness matrices of numerical methods for integer-order partial differential equations are sparse. Consequently, the evaluation of these matrices naturally has linear complexity. However, in the current context of nonlocal diffusion or peridynamic models, the stiffness matrices are dense, which requires $O(N^2)$ computational work to assemble and $O(N^2)$ memory to store. In the context of a constant-coefficient peridynamic or nonlocal diffusion model, we proved a Toeplitz-like structure for the stiffness matrices of the corresponding numerical methods [50, 49]. However, in the current context, the presence of a variable coefficient destroys a Toeplitz-like structure of the stiffness matrix A of the numerical scheme (2.5)–(2.6). Hence, we have to carefully design an efficient evaluation mechanism to assemble the matrix A.

Evaluation of the diagonal entries of the stiffness matrix

We begin by evaluating the diagonal entries of the stiffness matrix A in (2.6) for $i = 1, 2, \ldots, N-1$

$$A_{i,i} = \alpha(x_i) \int_{x_i-\delta}^{x_i+\delta} \frac{1-\phi_i(y)}{|x_i-y|^{1+\gamma}} dy + \int_{x_i-\delta}^{x_i+\delta} \frac{\alpha(y)(1-\phi_i(y))}{|x_i-y|^{1+\gamma}} dy, \tag{2.8}$$

We utilize the symmetry of the integration to evaluate the first term on the right-hand as follows

$$\begin{aligned} &\int_{x_i-\delta}^{x_i+\delta} \frac{1-\phi_i(y)}{|x_i-y|^{1+\gamma}} dy \\ &= 2\int_{x_{i+1}}^{x_i+\delta} \frac{1}{(y-x_i)^{1+\gamma}} dy + 2\int_{x_i}^{x_{i+1}} \frac{1-\phi_i(y)}{(y-x_i)^{1+\gamma}} dy \\ &= \frac{2(h^{-\gamma}-\delta^{-\gamma})}{\gamma} + \frac{2h^{-\gamma}}{1-\gamma}, \end{aligned} \tag{2.9}$$

However, the evaluation of the second term on the right-hand side of (2.8) presents a sustantial numerical difficulty. As the variable-coefficient $\alpha(y)$ is inside the integration with respect to y, a numerical quadrature has to be applied to evaluate the integral in general. Furthermore, the domain of integration, i.e., the horizon of the material at the collocation point x_i, is $(x_i-\delta, x_i+\delta)$, which is asymptotically of order $O(N)$. A naive application of a numerical quadrature would require $O(N^2)$ computations to assemble the matrix A. To reduce the computational work to assemble the stiffness matrix A, we need to evaluate this term very carefully. We note that this term can be decomposed as

$$\begin{aligned}
&\int_{x_i-\delta}^{x_i+\delta} \frac{\alpha(y)(1-\phi_i(y))}{|x_i-y|^{1+\gamma}}dy \\
&\quad= \int_{x_i-\delta}^{x_{i-1}} \frac{\alpha(y)}{(x_i-y)^{1+\gamma}}dy + \int_{x_{i+1}}^{x_i+\delta} \frac{\alpha(y)}{(y-x_i)^{1+\gamma}}dy \\
&\qquad+ \int_{x_{i-1}}^{x_i} \frac{\alpha(y)(1-\phi_i(y))}{(x_i-y)^{1+\gamma}}dy + \int_{x_i}^{x_{i+1}} \frac{\alpha(y)(1-\phi_i(y))}{(y-x_i)^{1+\gamma}}dy.
\end{aligned} \tag{2.10}$$

We use a piecewise-constant approximation $\alpha^I(x)$ to approximate $\alpha(x)$

$$\alpha^I(x) := \sum_{i=1}^{N} \alpha(x_{-\frac{1}{2}}) 1_{[x_{i-1},x_i)}(x), \quad x \in [a,b]. \tag{2.11}$$

Here $1_{[x_{i-1},x_i)}(x)$ is the indicator function such that $1_{[x_{i-1},x_i)}(x) = 1$ for $x \in [x_{i-1}, x_i)$ or 0 elsewhere. $x_{i-\frac{1}{2}} := (x_{i-1}+x_i)/2$ are the cell centers of the cells $[x_{i-1}, x_i]$.

We substitute α^I for α in the last two terms on the right-hand side of (2.9) to evaluate these two terms in a similar fashion to (2.9) as follows

$$\begin{aligned}
&\alpha(x_{i-\frac{1}{2}}) \int_{x_{i-1}}^{x_i} \frac{(1-\phi_i(y))}{(x_i-y)^{1+\gamma}}dy + \alpha(x_{i+1/2}) \int_{x_i}^{x_{i+1}} \frac{(1-\phi_i(y))}{(y-x_i)^{1+\gamma}}dy \\
&\quad= \frac{h^{-\gamma}\Big(\alpha(x_{i-\frac{1}{2}}) + \alpha(x_{i+1/2})\Big)}{1-\gamma}.
\end{aligned} \tag{2.12}$$

We similarly evaluate the first two terms on the right-hand side of (2.9) as

$$
\begin{aligned}
&\alpha(x_{i-K-\frac{1}{2}})\int_{x_i-\delta}^{x_{i-K}} \frac{1}{(x_i-y)^{1+\gamma}}dy \\
&\quad + \sum_{k=i-K+1}^{i-1} \alpha(x_{k-\frac{1}{2}})\int_{x_{k-1}}^{x_k} \frac{1}{(x_i-y)^{1+\gamma}}dy \\
&\quad + \sum_{k=i+2}^{x_{i+K}} \alpha(x_{k-\frac{1}{2}})\int_{x_{k-1}}^{x_k} \frac{1}{(x_i-y)^{1+\gamma}}dy \\
&\quad +\alpha(x_{i+K+\frac{1}{2}})\int_{x_{i+K}}^{x_i+\delta} \frac{\alpha(y)}{(y-x_i)^{1+\gamma}}dy \\
&= \alpha(x_{i-K-\frac{1}{2}})\frac{(Kh)^{-\gamma}-\delta^{-\gamma}}{\gamma} \\
&\quad + \sum_{k=i-K+1}^{i-1} \alpha(x_{k-\frac{1}{2}})\frac{((i-K)^{-\gamma}-(i-K+1)^{-\gamma})h^{-\gamma}}{\gamma} \\
&\quad + \sum_{k=i+2}^{x_{i+K}} \alpha(x_{k-\frac{1}{2}})\frac{((K-1-i)^{-\gamma}-(K-i)^{-\gamma})h^{-\gamma}}{\gamma} \\
&\quad +\alpha(x_{i+K+\frac{1}{2}})\frac{(Kh)^{-\gamma}-\delta^{-\gamma}}{\gamma}
\end{aligned}
\tag{2.13}
$$

Evaluation of the off-diagonal entries of the stiffness matrix

We now turn to the off-diagonal entries of the matrix A given in (2.6)

$$A_{i,j} = \alpha(x_i)\int_{x_i-\delta}^{x_i+\delta} \frac{\phi_j(y)}{|x_i-y|^{1+\gamma}}dy + \int_{x_i-\delta}^{x_i+\delta} \frac{\alpha(y)\phi_j(y)}{|x_i-y|^{1+\gamma}}dy, \quad j \neq i. \tag{2.14}$$

Since the support of ϕ_j is the interval $[x_{j-1}, x_{j+1}]$, it is clear that

$$A_{i,j} = 0, \quad |i-j| \geq K+2. \tag{2.15}$$

In other words, the stiffness matrix A is a dense banded matrix with the band width of $O(N)$ asymptotically.

We need only evaluate the off-diagonal entries $A_{i,j}$ for $1 \leq |i-j| \leq K+1$ in different cases.

Case 1: We consider the case $i+1 \leq j \leq i+K-1$, when the support of ϕ_j lies inside $[x_i, x_{i+K}]$. In this case, we evaluate $A_{i,j}$ as in Section 3.1

$$
\begin{aligned}
A_{i,j} &= \alpha(x_i)\int_{x_{j-1}}^{x_{j+1}} \frac{\phi_j(y)}{(y-x_i)^{1+\gamma}}dy + \int_{x_{j-1}}^{x_{j+1}} \frac{\alpha(y)\phi_j(y)}{(y-x_i)^{1+\gamma}}dy \\
&\approx \alpha(x_{j-\frac{1}{2}})\int_{x_{j-1}}^{x_j} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}dy + \alpha(x_{j+\frac{1}{2}})\int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy\Big] \\
&\quad +\alpha(x_i)\Big[\int_{x_{j-1}}^{x_j} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}} + \int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy\Big] \\
&= \alpha(x_{j-\frac{1}{2}})l_{j-i}^{(1)} + \alpha(x_{j+\frac{1}{2}})r_{j-i}^{(1)} \\
&\quad +\alpha(x_i)(l_{j-i}^{(1)} + r_{j-i}^{(1)}),
\end{aligned}
\tag{2.16}
$$

where we have

$$
\begin{aligned}
l_{j-i}^{(1)} &= \int_{x_{j-1}}^{x_j} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}dy \\
&= \begin{cases}
((i-j+1)\frac{h^{-\gamma}}{\gamma}((j-i-1)^{-\gamma} - (j-i)^{-\gamma}) \\
+\frac{h^{-\gamma}}{1-\gamma}((j-i)^{1-\gamma} - (j-i-1)^{1-\gamma})) \\
for\ \gamma \neq 0 \\
1 + (i-j+1)ln^{\frac{j-i}{j-i-1}} \\
for\ \gamma = 0
\end{cases},
\end{aligned}
\tag{2.17}
$$

$$
\begin{aligned}
r_{j-i}^{(1)} &= \int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy \\
&= \begin{cases}
((j-i+1)\frac{h^{-\gamma}}{\gamma}((j-i)^{-\gamma} - (j-i+1)^{-\gamma}) \\
+\frac{h^{-\gamma}}{1-\gamma}((j-i)^{1-\gamma} - (j-i+1)^{1-\gamma})) \\
for\ \gamma \neq 0 \\
(j-i+1)ln^{\frac{j-i+1}{j-i}} - 1 \\
for\ \gamma = 0
\end{cases},
\end{aligned}
\tag{2.18}
$$

then we defined

$$
l_k^{(1)} = \begin{cases} ((1-k)\frac{h^{-\gamma}}{\gamma}((k-1)^{-\gamma}-(k)^{-\gamma}) \\ \\ +\frac{h^{-\gamma}}{1-\gamma}((k^{1-\gamma}-(k-1)^{1-\gamma})) \\ \\ for\ \gamma \neq 0 \\ 1+(1-k)ln^{\frac{k}{k-1}} \\ for\ \gamma = 0 \end{cases}, \tag{2.19}
$$

$$
r_k^{(1)} = \begin{cases} ((k+1)\frac{h^{-\gamma}}{\gamma}((k)^{-\gamma}-(k+1)^{-\gamma}) \\ \\ +\frac{h^{-\gamma}}{1-\gamma}((k)^{1-\gamma}-(k+1)^{1-\gamma})) \\ \\ for\ \gamma \neq 0 \\ (k+1)ln^{\frac{k+1}{k}}-1 \\ for\ \gamma = 0 \end{cases}, \tag{2.20}
$$

Case 2: We can similarly evaluate $A_{i,j}$ for $i-K+1 \leq j \leq i-1$ as follows

$$
\begin{aligned}
A_{i,j} &= \alpha(x_i)\int_{x_{j-1}}^{x_{j+1}} \frac{\phi_j(y)}{(x_i-y)^{1+\gamma}}dy + \int_{x_{j-1}}^{x_{j+1}} \frac{\alpha(y)\phi_j(y)}{(x_i-y)^{1+\gamma}}dy \\
&\approx \alpha(x_{j-\frac{1}{2}})\int_{x_{j-1}}^{x_j} \frac{\phi_j^L(y)}{(x_i-y)^{1+\gamma}}dy + \alpha(x_{j+\frac{1}{2}})\int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}}dy\Big] \\
&\quad +\alpha(x_i)\Big[\int_{x_{j-1}}^{x_j} \frac{\phi_j^L(y)}{(x_i-y)^{1+\gamma}} + \int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}}dy\Big] \\
&= \alpha(x_{j-\frac{1}{2}})r_{i-j}^{(1)} + \alpha(x_{j+\frac{1}{2}})l_{i-j}^{(1)} \\
&\quad +\alpha(x_i)(l_{i-j}^{(1)} + r_{i-j}^{(1)}),
\end{aligned} \tag{2.21}
$$

Case 3: We consider the case $j = i + K$ when part of the support of ϕ_j, the interval $[x_j, x_{j+1}]$, does not lie in the interval $[x_i, x_i + \delta]$ in general. In this case, we evaluate $A_{i,j}$ as follows

$$
\begin{aligned}
A_{i,j} &= \alpha(x_i)\int_{x_{j-1}}^{x_j+\delta-Kh}\frac{\phi_j(y)}{(y-x_i)^{1+\gamma}}dy+\int_{x_{j-1}}^{x_j+\delta-Kh}\frac{\alpha(y)\phi_j(y)}{(x_i-y)^{1+\gamma}}dy \\
&\approx \alpha(x_{j-\frac{1}{2}})\int_{x_{j-1}}^{x_j}\frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}dy+\alpha(x_{j+\frac{1}{2}})\int_{x_j}^{x_j+\delta-Kh}\frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy\Big] \\
&\quad +\alpha(x_i)\Big[\int_{x_{j-1}}^{x_j}\frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}+\int_{x_j}^{x_j+\delta-Kh}\frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}}dy\Big] \\
&= \alpha(x_{j-\frac{1}{2}})l_{j-i}^{(1)}+\alpha(x_{j+\frac{1}{2}})r_{j-i}^{(2)} \\
&\quad +\alpha(x_i)(l_{j-i}^{(1)}+r_{j-i}^{(2)}),
\end{aligned}
\tag{2.22}
$$

for which,

$$
\begin{aligned}
r_{j-i}^{(2)} &= \textstyle\int_{x_j}^{x_j+\delta-Kh}\frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy \\
&= \begin{cases}
((j-i+1)\frac{1}{\gamma}((j-i)^{-\gamma}h^{-\gamma}-((j-i-K)h+\delta)^{-\gamma}) \\
-\frac{1}{h(1-\gamma)}((((j-i-K)h+\delta)^{1-\gamma}-(j-i)^{1-\gamma})) \\
for\ \gamma\neq 0 \\
(j-i+1)ln^{\frac{(j-i-K)h+\delta}{(j-i)h}}+\frac{Kh-\delta}{h} \\
for\ \gamma=0
\end{cases},
\end{aligned}
\tag{2.23}
$$

here we defined

$$
r_k^{(2)} = \begin{cases}
((k+1)\frac{1}{\gamma}((k)^{-\gamma}h^{-\gamma}-((k-K)h+\delta)^{-\gamma}) \\
-\frac{1}{h(1-\gamma)}((((k-K)h+\delta)^{1-\gamma}-(k)^{1-\gamma})) \\
for\ \gamma\neq 0 \\
(k+1)ln^{\frac{(k-K)h+\delta}{kh}}+\frac{Kh-\delta}{h} \\
for\ \gamma=0
\end{cases},
\tag{2.24}
$$

Case 4: We can similarly treat the case $j=i-K$.

$$
\begin{aligned}
A_{i,j} &= \alpha(x_i)\int_{x_j-\delta+Kh}^{x_{j+1}} \frac{\phi_j(y)}{(x_i-y)^{1+\gamma}}dy + \int_{x_j-\delta+Kh}^{x_{j+1}} \frac{\alpha(y)\phi_j(y)}{(x_i-y)^{1+\gamma}}dy \\
&\approx \alpha(x_{j-\frac{1}{2}})\int_{x_j-\delta+Kh}^{x_j} \frac{\phi_j^L(y)}{(x_i-y)^{1+\gamma}}dy + \alpha(x_{j+\frac{1}{2}})\int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}}dy\Big] \\
&\quad +\alpha(x_i)\Big[\int_{x_j-\delta+Kh}^{x_j} \frac{\phi_j^L(y)}{(x_i-y)^{1+\gamma}} + \int_{x_j}^{x_{j+1}} \frac{\phi_j^R(y)}{(y-x_i)^{1+\gamma}}dy\Big] \\
&= \alpha(x_{j-\frac{1}{2}})r_{i-j}^{(2)} + \alpha(x_{j+\frac{1}{2}})l_{i-j}^{(1)} \\
&\quad +\alpha(x_i)(r_{i-j}^{(2)} + l_{i-j}^{(1)}),
\end{aligned}
\tag{2.25}
$$

Case 5: We consider the case $j = i + K + 1$ when part of the support of ϕ_j, $[x_{j-1}, x_j]$ does not necessarly lie in the interval $[x_i, x_i+\delta]$. We evaluate $A_{i,j}$ as follows .

$$
\begin{aligned}
A_{i,j} &= \alpha(x_i)\int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\phi_j(y)}{(y-x_i)^{1+\gamma}}dy + \int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\alpha(y)\phi_j(y)}{(y-x_i)^{1+\gamma}}dy \\
&\approx \alpha(x_i)\int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}dy + \alpha(x_{j-\frac{1}{2}})\int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}}dy \\
&= \alpha(x_{j-\frac{1}{2}})l_{j-i}^{(2)} + \alpha(x_i)l_{j-i}^{(2)},
\end{aligned}
\tag{2.26}
$$

and we simply write

$$l_{j-i}^{(2)} = \int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\phi_j^L(y)}{(y-x_i)^{1+\gamma}} dy$$

$$= \begin{cases} ((i-j+1)\frac{1}{\gamma}((j-i-1)^{-\gamma}h^{-\gamma} - ((j-i-1-K)h+\delta)^{-\gamma}) \\ \\ +\frac{1}{h(1-\gamma)}((((j-i-1-K)h+\delta)^{1-\gamma} - (j-i-1)^{1-\gamma})) \\ \\ for\ \gamma \neq 0 \\ \\ (i-j+1)ln^{\frac{(j-i-1-K)h+\delta}{(j-i-1)h}} + \frac{\delta-Kh}{h} \\ \\ for\ \gamma = 0 \end{cases}, \tag{2.27}$$

again we can define

$$l_k^{(2)} = \begin{cases} ((1-k)\frac{1}{\gamma}((k-1)^{-\gamma}h^{-\gamma} - ((k-1-K)h+\delta)^{-\gamma}) \\ \\ +\frac{1}{h(1-\gamma)}((((k-1-K)h+\delta)^{1-\gamma} - (k-1)^{1-\gamma})) \\ \\ for\ \gamma \neq 0 \\ (1-k)ln^{\frac{(k-1-K)h+\delta}{(k-1)h}} + \frac{\delta-Kh}{h} \\ \\ for\ \gamma = 0 \end{cases}, \tag{2.28}$$

Case 6: We can similarly treat the case $j = i - K - 1$.

$$\begin{aligned} A_{i,j} &= \alpha(x_i) \int_{x_{j+1}-\delta+Kh}^{x_{j+1}} \frac{\phi_j(y)}{(x_i-y)^{1+\gamma}} dy + \int_{x_{j-1}}^{x_{j-1}+\delta-Kh} \frac{\alpha(y)\phi_j(y)}{(x_i-y)^{1+\gamma}} dy \\ &\approx \alpha(x_i) \int_{x_{j+1}-\delta+Kh}^{x_{j+1}} \frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}} dy + \alpha(x_{j+\frac{1}{2}}) \int_{x_{j+1}-\delta+Kh}^{x_{j+1}} \frac{\phi_j^R(y)}{(x_i-y)^{1+\gamma}} dy \\ &= \alpha(x_{j+\frac{1}{2}}) l_{i-j}^{(2)} + \alpha(x_i) l_{i-j}^{(2)}. \end{aligned} \tag{2.29}$$

2.4 Structure of the stiffness matrix

In this subsection, we shall introduce two important theorems, which can be used to achieve our fast method.

Theorem 2.1. *The stiffness matrix A of the collocation method of previous variable coefficients non-local peridynamic model can be written as*

$$A = Ad + A^{(1)}T^{(1)} + T^{(2)}A^{(2)} + T^{(3)}A^{(3)}$$

, where Ad contains all the diagonal entries of A.

Proof. By previous discussion, we can first set Ad as a diagonal matrix and all the diagonal entries are the same as the diagonal entries of A.

For remaining parts, let us recall the expressions of the entries of A, (2.16), (2.21), (2.22), (2.25), (2.26), and (2.29), then we write A^1 in the following structure

$$T^{(1)} = \begin{pmatrix} 0 & T^{(1)}_{1,2} & \cdots & T^{(1)}_{1,m+1} & \cdots & 0 \\ T^{(1)}_{1,2} & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & T^{(1)}_{N-m-1,N-1} \\ T^{(1)}_{m+1,1} & \ddots & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & T^{(1)}_{N-2,N-1} \\ 0 & \cdots & T^{(1)}_{N-1,N-m-1} & \cdots & T^{(1)}_{N-1,N-2} & 0 \end{pmatrix}, \tag{2.30}$$

where for $j = i + m$

$$T^{(1)}_{i,i+m} = -\int_{x_{j-1}}^{x_{j-1}+\delta-nh} \frac{\phi^L_{i+m}(y)}{(y - x_i)^{1+\gamma}} dy, \tag{2.31}$$

and for $j = i - m$

$$T^{(1)}_{i,i-m} = -\int_{x_{j+1}+nh-\delta}^{x_{j+1}} \frac{\phi^R_{i-m}(y)}{(x_i - y)^{1+\gamma}} dy, \tag{2.32}$$

when $j=i+n$ and $j=i-n$, we can have

$$\begin{aligned} T^{(1)}_{i,i+n} &= -\int_{x_{i+n-1}}^{x_{i+n}} \frac{\phi^L_{i+n}(y)}{(y-x_i)^{1+\gamma}}dy - \int_{x_{i+n}}^{x_{i+n}+\delta-nh} \frac{\phi^R_{i+n}(y)}{(y-x_i)^{1+\gamma}}dy \\ T^{(1)}_{i,i-n} &= -\int_{x_{i-n}}^{x_{i-n+1}} \frac{\phi^L_{i+n}(y)}{(x_i-y)^{1+\gamma}}dy - \int_{x_{i-n}-\delta+nh}^{x_{i-n}} \frac{\phi^R_{i-n}(y)}{(x_i-y)^{1+\gamma}}dy, \end{aligned} \tag{2.33}$$

then for $j=i+1,...,i+n-1$

$$T^{(1)}_{i,j} = -\int_{x_{j-1}}^{x_j} \frac{\phi^L_j(y)}{(y-x_i)^{1+\gamma}}dy - \int_{x_j}^{x_{j+1}} \frac{\phi^R_j(y)}{(y-x_i)^{1+\gamma}}dy, \tag{2.34}$$

finally, for $j=i-1,...,i-n+1$

$$T^{(1)}_{i,j} = -\int_{x_{j-1}}^{x_j} \frac{\phi^L_j(y)}{(x_i-y)^{1+\gamma}}dy - \int_{x_j}^{x_{j+1}} \frac{\phi^R_j(y)}{(x_i-y)^{1+\gamma}}dy. \tag{2.35}$$

Next, we shall let the structure of T^2 to be

$$T^{(2)} = \begin{pmatrix} 0 & T^{(2)}_{1,2} & \cdots & T^{(2)}_{1,m+1} & \cdots & 0 \\ T^{(2)}_{1,2} & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & T^{(2)}_{N-m-1,N-1} \\ T^{(2)}_{n+1,1} & \ddots & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & T^{(2)}_{N-2,N-1} \\ 0 & \cdots & T^{(2)}_{N-1,N-n-1} & \cdots & T^{(2)}_{N-1,N-2} & 0 \end{pmatrix}. \tag{2.36}$$

We then consider the entries in several cases, for $j=i+m$ $T^{(2)}_{i,j} = T^{(1)}_{i,j}$ and for $j=i+n$

$$T^{(2)}_{i,i+n} = -\int_{x_{i+n-1}}^{x_{i+n}} \frac{\phi^L_{i+n}(y)}{(y-x_i)^{1+\gamma}}dy. \tag{2.37}$$

When $j=i+1,...,i+n-1$, we have

$$T^{(2)}_{i,j} = -\int_{x_{j-1}}^{x_j} \frac{\phi^L_j(y)}{(y-x_i)^{1+\gamma}}dy. \tag{2.38}$$

Similarly, $T^{(2)}_{i,i-n}$ can be expressed as

$$T^{(2)}_{i,i-n} = -\int_{x_{i-n}-\delta+nh}^{x_{i-n}} \frac{\phi^R_{i-n}(y)}{(x_i - y)^{1+\gamma}} dy. \tag{2.39}$$

The last case for $j = i-1, ..., i-n+1$

$$T^{(2)}_{i,j} = -\int_{x_{j-1}}^{x_j} \frac{\phi^L_j(y)}{(x_i - y)^{1+\gamma}} dy. \tag{2.40}$$

Finally we set T^3 as

$$T^{(2)} = \begin{pmatrix} 0 & T^{(2)}_{1,2} & \dots & T^{(2)}_{1,n+1} & \dots & 0 \\ T^{(2)}_{1,2} & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & T^{(2)}_{N-n-1,N-1} \\ T^{(2)}_{m+1,1} & \ddots & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & T^{(2)}_{N-2,N-1} \\ 0 & \dots & T^{(2)}_{N-1,N-m-1} & \dots & T^{(2)}_{N-1,N-2} & 0 \end{pmatrix}, \tag{2.41}$$

where $T^{(3)}_{i,j} = T^{(1)}_{i,j}$ when $j = i - m$. For $j = i + n$

$$T^{(3)}_{i,i+n} = -\int_{x_{i+n}}^{x_{i+n}+\delta-nh} \frac{\phi^R_{i+n}(y)}{(y - x_i)^{1+\gamma}} dy, \tag{2.42}$$

when $j = i+1, ..., i+n-1$

$$T^{(3)}_{i,j} = -\int_{x_j}^{x_{j+1}} \frac{\phi^R_j(y)}{(y - x_i)^{1+\gamma}} dy, \tag{2.43}$$

and for $j = i - n$

$$T^{(3)}_{i,i-n} = -\int_{x_{i-n}}^{x_{i-n+1}} \frac{\phi^L_{i+n}(y)}{(x_i - y)^{1+\gamma}} dy. \tag{2.44}$$

Again, for $j = i-1, ..., i-n+1$

$$T^{(3)}_{i,j} = -\int_{x_j}^{x_{j+1}} \frac{\phi^R_j(y)}{(x_i - y)^{1+\gamma}} dy. \tag{2.45}$$

Then we can write $\{A^{(i)}\}_{i=1}^3$ as three diagonal matrices, where $A_{i,i}^{(1)} = \alpha(x_i)$, $A_{i,i}^{(2)} = \alpha(x_i + \frac{h}{2})$, and $A_{i,i}^{(3)} = \alpha(x_{i-1} + \frac{h}{2})$.

Now we have

$$A = Ad + A^{(1)}T^{(1)} + T^{(2)}A^{(2)} + T^{(3)}A^{(3)}$$

.

□

Theorem 2.2. *The matrices $\{A^{(i)}\}_{i=1}^3$ in Theorem 1 are diagonal matrices, and $\{T^{(i)}\}_{i=1}^3$ are Toeplitz matrices.*

Proof. From Theorem 1, we already set $\{A^{(i)}\}_{i=1}^3$ as diagonal matrices, then we only need to prove that $\{T^{(i)}\}_{i=1}^3$ are Toeplitz matrices.

Since the proof of $T^{(1)}$, $T^{(2)}$, and $T^{(3)}$ are almost the same, without losing generality, we will only prove the up triangular entries of $T^{(1)}$.

Firstly, for $j = i + m$ we do the following substitution

$$y = x_j + s, dy = ds$$

,

then $s \in [-h, -h + \delta - nh]$, (2.31) can be written as

$$\begin{aligned} T_{i,i+m}^{(1)} &= -\int_{x_{j-1}}^{x_{j-1}+\delta-nh} \frac{\phi_{i+m}^L(y)}{(y-x_i)^{1+\gamma}} dy \\ &= -\int_{-h}^{-h+\delta-nh} \frac{\psi(\frac{x_j+s-x_j}{h})}{(x_j+s-x_i)^{1+\gamma}} ds \\ &= -\int_{-h}^{-h+\delta-nh} \frac{\psi(\frac{s}{h})}{((j-i)h+s)^{1+\gamma}} ds, \end{aligned} \tag{2.46}$$

so its value only depends on $j - i$.

Then for $j = i + n$, we do the substitution

$$y = x_j + s, dy = ds$$

,

where $s \in [-h, \delta - nh]$, (2.33) can be written as

$$\begin{aligned} T^{(1)}_{i,i+n} &= -\int_{x_{j-1}}^{x_j+\delta-nh} \frac{\phi_{i+n}(y)}{(y-x_i)^{1+\gamma}} dy \\ &= -\int_{-h}^{\delta-nh} \frac{\psi(\frac{x_j+s-x_j}{h})}{(x_j+s-x_i)^{1+\gamma}} ds \\ &= -\int_{-h}^{-h+\delta} \frac{\psi(\frac{s}{h})}{((j-i)h+s)^{1+\gamma}} ds, \end{aligned} \tag{2.47}$$

the value of $T^{(1)}_{i,i+n}$ also only depends on $j-i$.

Finally, for $j = i+1, ..., i+n-1$, the expression

$$T^{(1)}_{i,j} = -\int_{x_{j-1}}^{x_{j+1}} \frac{\phi_j(y)}{(y-x_i)^{1+\gamma}} dy, \tag{2.48}$$

after doing the substitution $y = x_j + s$, where $s \in [-h, h]$, we have

$$\begin{aligned} T^{(1)}_{i,j} &= -\int_{x_{j-1}}^{x_{j+1}} \frac{\phi_j(y)}{(y-x_i)^{1+\gamma}} dy \\ &= -\int_{-h}^{h} \frac{\psi(\frac{x_j+s-x_j}{h})}{(x_j+s-x_i)^{1+\gamma}} ds \\ &= -\int_{-h}^{-h} \frac{\psi(\frac{s}{h})}{((j-i)h+s)^{1+\gamma}} ds. \end{aligned} \tag{2.49}$$

We have the same conclusion, such that the value of $T^{(1)}$ only depends on $j-i$, the values of the entries are the same if $j-i$ is a constant, so $T^{(1)}$ is a Toeplitz matrix.

We do not need to repeat the procedure, and observed that $T^{(2)}$, $T^{(3)}$ are also Toeplitz matrices.

$\square$

Now we can implement our fast method. Since we are going to accelerate the matrix vector multiplication Ap_k, by Theorem 1, we rewrite

$$\begin{aligned} Ap_k &= (Ad + A^{(1)}T^{(1)} + T^{(2)}A^{(2)} + T^{(3)}A^{(3)})p_k \\ &= Adp_k + A^{(1)}T^{(1)}p_k + T^{(2)}A^{(2)}p_k + T^{(3)}A^{(3)}p_k \qquad , \\ &= Adp_k + A^{(1)}(T^{(1)}p_k) + T^{(2)}(A^{(2)}p_k) + T^{(3)}(A^{(3)}p_k) \end{aligned} \tag{2.50}$$

Because $A^{(2)}$ and $A^{(3)}$ are diagonal matrices, $A^{(2)}p_k$ and $A^{(2)}p_k$ can be found out with computational cost $O(N)$. We simply write $A^{(2)}p_k = p_k^{(2)}$ and $A^{(3)}p_k = p_k^{(3)}$, so the Toeplitz matrix vector multiplication $T^{(1)}p_k$, $T^{(2)}p_k^{(2)}$, and $T^{(3)}p_k^{(3)}$ could be accelerated.

By previous studies, Toeplitz matrices can be embedded into an circulant matrix C as follows, without losing generality, we shall use $T^{(1)}$ as our example

$$C := \begin{pmatrix} T^{(1)} & B \\ B & T^{(1)} \end{pmatrix} \qquad B := \begin{pmatrix} 0 & q_{-m} & \dots & q_{-1} & 0 \\ q_m & 0 & q_{-m} & \dots & q_{-1} \\ \vdots & q_m & 0 & \ddots & \vdots \\ q_1 & \vdots & \ddots & \ddots & q_{-m} \\ 0 & q_1 & \dots & q_m & 0 \end{pmatrix}. \tag{2.51}$$

In our problem, C should be an $2(N-1) \times 2(N-1)$ circulant matrix, where $q_u = T^{(1)}_{i,j}, u = j - i$.

By previous studies, we know that C has the following decomposition

$$C = F^{-1}\mathrm{diag}(Fc)F, \tag{2.52}$$

where c is the first column of C, F is the $2(N-1) \times 2(N-1)$ discrete Fourier transform matrix.

Then we we embed p_k into an $2(N-1)$ dimensional vector w, where

$$w := \begin{pmatrix} p_k \\ 0 \end{pmatrix}. \tag{2.53}$$

Here we can get $T^{(1)}p_k$ by keeping the first $N-1$ entries of Cw, since Fc and Fw only need $O(NlogN)$ computational cost, we can implement the conjugate gradient squared method by our fast matrix vector multiplication in only $O(NlogN)$ steps per iteration, which is a big improvement compared to the standard matrix vector multiplication.

2.5 Numerical experiments

In this section, we will implement our fast method to a series of numerical experiments, to investigate the performance of the fast method. To run the numerical examples, we set the spatial domain $(a, b) = (-1, 1)$, and the real solution $u(x) = (1-x)^2(1+x)^2$, with the variable coefficient $\alpha(x) = 1 + \epsilon(1-x^2)$, where we choose $\epsilon = 0.1$, and $\delta = 1/32$.

In our numerical examples, the analytic expression of the right-hand side term $b(x)$ could be find out in each collocation point x_i. Then we shall use Gaussian Elimination, standard Conjugate Gradient Squared, and our Fast Conjugate Gradient Squared method to seek the numerical solution of the linear equation system (2.5). We ran those Matlab programs in a 16GB-ROM laptop.

We inspected the performance of our fast method by switching the size of the grid from $h = 2^{-7}$ to $h = 2^{14}$, and observed that all three of the above numerical solvers have the same computational error and convergence rate. Then the order of convergence could be fitted by linear regression. If we set u^h as the numerical solution and u as the real solution of our problem, the error estimate could be expressed as follows

$$||u^h - u||_{L^p(a,b)} \leq C_\beta h^\beta, \quad p = 2, \infty$$

We concluded that the order of convergence of the numerical methods we used in our experiments is second order, and since when $h = 2^{-13}$ the error is small enough, the numerical scheme loses the second order accuracy as h gets smaller[16, 15].

There were no research findings have been announced directly on the collocation method for peridynamic or nonlocal diffusion model, but it has been shown that the singularity of the stiffness matrix depends on the power γ in the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ when performing the Galerkin finite element method in previous studies[1, 10, 16, 36, 56]. We tentatively chose $\gamma = 1/10, 1/2, 3/4, and - 1/2$, and try to observed the effect on our collocation method.

Example 1. We set $\gamma = 1/10$, where the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ is nonintegrable, and has little singularity. We showed the numerical solution in Table 1.

Example 2. We set $\gamma = 1/2$, where the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ is still nonintegrable, but has more singularity and we found that the convergence rate is almost second order. However, the number of iterations increased a lot. We showed the numerical solution in Table 2.

Example 3. We set $\gamma = 3/4$, where the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ is still nonintegrable and the singularity became even larger. We shall inspect that the convergence rate decreased to 1.5 order, and we need more iterations to get the ideal numerical solution. We showed the numerical solution in Table 3.

Example 4. We set $\gamma = -1/2$, where the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ is integrable and we found that the convergence rate was about second order. We will need fewer iterations. We showed the numerical solution in Table 4.

Example 5. Finally, consider $\gamma = 1/10$, where the kernel function $\frac{1}{|x-y|^{1+\gamma}}$ is still non-integrable, but we set $\delta = 1/16$. We will have better accuracy with a smaller grid size and need fewer iteration steps. We showed the numerical solution in Table

Table 2.1: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 1/10$, $\delta = 1/32$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	2^{-7}	$3.04166771e-02$	–	0.05s
	2^{-8}	$8.18334943e-03$	–	0.23s
	2^{-9}	$1.89265584e-03$	–	1.47s
	2^{-10}	$4.17492908e-04$	–	12.11s
	2^{-11}	$9.62105995e-05$	–	1m58s
	2^{-12}	$2.70050257e-05$	–	16m54s
	2^{-13}	$8.67911833e-06$	–	2h54m
	2^{-14}	$2.78957323e-06$	–	1d5h
CGS	2^{-7}	$3.04166771e-02$	462	0.2s
	2^{-8}	$8.18334943e-03$	496	0.35s
	2^{-9}	$1.89265584e-03$	500	1.53s
	2^{-10}	$4.17492908e-04$	549	8.72s
	2^{-11}	$9.62105995e-05$	527	22.9s
	2^{-12}	$2.70050257e-05$	495	1m02s
	2^{-13}	$8.67911833e-06$	527	5m27s
	2^{-14}	$2.78957323e-06$	570	21m21s
FCGS	2^{-7}	$3.04166771e-02$	462	1.38s
	2^{-8}	$8.18334943e-03$	496	1.38s
	2^{-9}	$1.89265584e-03$	500	1.39s
	2^{-10}	$4.17492908e-04$	549	3.11s
	2^{-11}	$9.62105995e-05$	527	6.02s
	2^{-12}	$2.70050257e-05$	495	11.4s
	2^{-13}	$8.67911833e-06$	527	57s
	2^{-14}	$2.78957323e-06$	570	1m01s

Table 2.2: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 1/2$, $\delta = 1/32$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	2^{-8}	$1.80518676e-02$	–	0.22s
	2^{-9}	$5.45753657e-03$	–	1.37s
	2^{-10}	$1.49600916e-03$	–	12.12s
	2^{-11}	$3.95745488e-04$	–	2m18s
	2^{-12}	$1.05727455e-04$	–	16m57s
CGS	2^{-8}	$1.80518676e-02$	758	0.68s
	2^{-9}	$5.45753657e-03$	896	2.57s
	2^{-10}	$1.49600916e-03$	1011	14.6s
	2^{-11}	$3.95745488e-04$	1139	51.98s
	2^{-12}	$1.05727455e-04$	1358	3m58s
FCGS	2^{-8}	$1.80518676e-02$	758	1.19s
	2^{-9}	$5.45753657e-03$	896	3.00s
	2^{-10}	$1.49600916e-03$	1011	5.35s
	2^{-11}	$3.95745488e-04$	1139	12.02s
	2^{-12}	$1.05727455e-04$	1358	30.07s

5.

In conclusion, a heuristic observation a obtained through the previous discussion and numerical experiments. Namely, that the convergence rate and number of iterations all depend on the kernel function $\frac{1}{|x-y|^{1+\gamma}}$. When $\gamma < 0$, we have a better convergence rate and fewer iteration steps; when γ is close to 1, the singularity causes more iteration steps and a lower convergence rate.

Table 2.3: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 3/4$, $\delta = 1/32$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	2^{-8}	$3.45784187e-02$	–	0.22s
	2^{-9}	$1.38426155e-02$	–	1.42s
	2^{-10}	$4.82842571e-03$	–	12.31s
	2^{-11}	$1.55326827e-03$	–	2m20s
	2^{-12}	$4.81220832e-04$	–	16m59s
CGS	2^{-8}	$3.45784187e-02$	1005	0.74s
	2^{-9}	$1.38426155e-02$	1464	4.07s
	2^{-10}	$4.82842571e-03$	1836	29.75s
	2^{-11}	$1.55326827e-03$	2362	1m43s
	2^{-12}	$4.81220832e-04$	2954	6m14s
FCGS	2^{-8}	$3.45784187e-02$	1005	1.82s
	2^{-9}	$1.38426155e-02$	1464	4.94s
	2^{-10}	$4.82842571e-03$	1836	9.57s
	2^{-11}	$1.55326827e-03$	2362	24.97s
	2^{-12}	$4.81220832e-04$	2954	1m06s

Table 2.4: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = -1/2$, $\delta = 1/32$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	2^{-8}	$3.76621601e-03$	–	0.28s
	2^{-9}	$7.35901274e-04$	–	1.61s
	2^{-10}	$1.52092692e-04$	–	13.18s
	2^{-11}	$4.13432509e-05$	–	2m7s
	2^{-12}	$1.81069791e-05$	–	18m43s
CGS	2^{-8}	$3.76621601e-03$	322	0.23s
	2^{-9}	$7.35901274e-04$	291	0.8s
	2^{-10}	$1.52092692e-04$	252	4.19s
	2^{-11}	$4.13432509e-05$	254	11.53s
	2^{-12}	$1.81069791e-05$	259	34.85s
FCGS	2^{-8}	$8.18334943e-03$	322	0.67s
	2^{-9}	$1.89265584e-03$	291	1.02s
	2^{-10}	$4.17492908e-04$	252	1.41s
	2^{-11}	$9.62105995e-05$	254	2.64s
	2^{-12}	$2.70050257e-05$	259	5.78s

Table 2.5: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 1/10$, $\delta = 1/16$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	2^{-8}	$2.76438847e-03$	–	0.25s
	2^{-9}	$6.65091032e-04$	–	1.36s
	2^{-10}	$1.96415968e-04$	–	12.32s
	2^{-11}	$8.51694885e-05$	–	1m58s
	2^{-12}	$4.94857046e-05$	–	17m40s
CGS	2^{-8}	$2.76438847e-03$	259	0.12s
	2^{-9}	$6.65091032e-04$	285	0.48s
	2^{-10}	$1.96415968e-04$	281	1.92s
	2^{-11}	$8.51694885e-05$	299	7.94s
	2^{-12}	$4.94857046e-05$	293	25.4s
FCGS	2^{-8}	$2.76438847e-03$	259	0.59s
	2^{-9}	$6.65091032e-04$	285	1.15s
	2^{-10}	$1.96415968e-04$	281	1.48s
	2^{-11}	$8.51694885e-05$	299	3.05s
	2^{-12}	$4.94857046e-05$	293	6.46s

Chapter 3

A Fast Galerkin Method for a variable Coefficient Peridynamic Model

3.1 Introduction

In the previous chapter, we worked out the numerical solution of the variable-coefficient peridynamic model by using collocation method. And in the studying of constant coefficient peridynamic problems, we observed that it will lead to a bilinear form when implement finite element method to search for numerical solution.

In this chapter, we consider a special case of the variable-coefficient peridynamic model[55, 54]. When the material horizon δ goes to infinity, our globel variable-coefficient peridynamic model can be seen as

$$\begin{aligned} &\int_a^b \mathbb{C}(x,y)(u(x)-u(y))dy = f(x), \qquad x \in (a,b) \\ &u(a) = u_a, \quad (b) = u_b, \end{aligned} \tag{3.1}$$

then we will use the finite element method to find the numerical solution. Since this model will lead to a full stiffness matrix, and the stiffness matrix does not have a Toeplitz structure directly, we shall analyze the entries of it and make it a sum of several special matrices. Moreover, we will show that the diagonal entries of the stiffness matrix of the variable-coefficient peridynamic model also require a great deal of computation, a numerical approximation of the variable coefficient technically helped us with accelerating the computation of the diagonal entries by using the Toeplitz structure again.

3.2 Numerical scheme of one dimensional variable-coefficient peridynamic model

In order to discuss the numerical scheme, we see the model which is derived out from (3.1),

$$\int_a^b (\alpha(x) + \alpha(y)) \frac{(u(x) - u(y))}{|x-y|^{1+\gamma}} dy = f(x),$$
$$u(a) = u_a, \qquad u(b) = u_b. \tag{3.2}$$

where we considered $\alpha(x, y) = \alpha(x) + \alpha(y)$ be the elasticity coefficient, and the kernel function $\kappa(x, y) = \frac{1}{|x-y|^{1+\gamma}}$. Then we are going to discuss the variational formulation of our model. For convenience, we simply introduce the notation

$$\mathbb{C}(x, y) = \alpha(x, y)\kappa(x, y), \tag{3.3}$$

To see the variational formulation, we multiply both sides of the non-local diffusion equation by a test function $v(x)$, $v \in L^2(\Omega; \mathbb{R})$, and integral both sides about x over the whole filed Ω, we get the weak formulation

$$a(u, v) = l(v), \tag{3.4}$$

where

$$a(u, v) = \int_a^b \int_a^b \mathbb{C}(x, y)(u(x) - u(y))v(x) dy dx, \tag{3.5}$$

and

$$l(v) = \int_a^b v(x) f(x) dy dx. \tag{3.6}$$

First, let us work on the right-hand side of the equation,

$$\begin{aligned} a(u,v) &= \int_a^b \int_a^b (\alpha(x)+\alpha(y))\mathbb{C}(x,y)(u(x)-u(y))v(x)dydx && (3.7)\\ &= \int_a^b \int_a^b (\alpha(x)+\alpha(y))\frac{u(x)-u(y)}{|x-y|^{1+\gamma}}v(x)dydx && (3.8)\\ &= \int_a^b \int_a^b (\alpha(y)+\alpha(x))\frac{u(y)-u(x)}{|y-x|^{1+\gamma}}v(y)dxdy && (3.9)\\ &= \int_a^b \int_a^b (\alpha(x)+\alpha(y))\frac{u(y)-u(x)}{|x-y|^{1+\gamma}}v(y)dydx, && (3.10) \end{aligned}$$

we first interchanged x and y from (3.7) and (3.8) to (3.9), and then exchanged the order of integration. Since we have (3.8) and (3.10) are equivalent, subtract (3.10) from (3.8), the following equation is true

$$\int_a^b \int_a^b \mathbb{C}(x,y)\left((u(x)-u(y))v(x)-(u(y)-u(x)v(y))\right)dydx = 0. \tag{3.11}$$

Let us consider the inside part of the integration above

$$\begin{aligned} &(u(x)-u(y))v(x)-(u(y)-u(x))v(y)\\ &= v(x)((u(x)-u(y))-(u(y)-u(x)))\\ &\quad + v(x)(u(y)-u(x))-v(y)(u(y)-u(x))\\ &= v(x)((u(x)-u(y))-(u(y)-u(x)))\\ &\quad - (v(y)-v(x))(u(y)-u(x))\\ &= v(x)(u(x)-u(y))-v(x)(u(y)-u(x))\\ &\quad - (v(y)-v(x))(u(y)-u(x))\\ &= 2v(x)(u(x)-u(y))-(v(y)-v(x))(u(y)-u(x)), \end{aligned} \tag{3.12}$$

then we plug in the last expression of (3.12) to (3.11), and get

$$\begin{aligned}
&\textstyle\int_a^b \int_a^b \mathbb{C}(x,y)(2v(x)(u(x)-u(y)) \\
&\qquad -(v(y)-v(x))(u(y)-u(x)))dydx = 0,
\end{aligned} \tag{3.13}$$

so that

$$\begin{aligned}
&\int_a^b \int_a^b \mathbb{C}(x,y)\left(2v(x)(u(x)-u(y))\right) dydx \\
&= \int_a^b \int_a^b \mathbb{C}(x,y)\left((v(y)-v(x))(u(y)-u(x))\right) dydx,
\end{aligned} \tag{3.14}$$

and here come to the second form of $a(u,v)$

$$a(u,v) = \frac{1}{2}\int_a^b \int_a^b \mathbb{C}(x,y)\left((v(y)-v(x))(u(y)-u(x))\right) dydx, \tag{3.15}$$

so we can conclude that $a(u,v)$ is symmetric.

Next we replaced the $\mathbb{C}(x,y)$ in (3.16) by

$$\mathbb{C}(x,y) = (\alpha(x)+\alpha(y))\frac{1}{|x-y|^{1+\gamma}}$$

,

then (3.16) can be expressed as

$$\begin{aligned}
&a(u,v) \\
&= \frac{1}{2}\int_a^b \int_a^b \mathbb{C}(x,y)\left((v(y)-v(x))(u(y)-u(x))\right) dydx \\
&= \frac{1}{2}\int_a^b \int_a^b (\alpha(x)+\alpha(y))\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}} dydx \\
&= \frac{1}{2}(\int_a^b \int_a^b \alpha(x)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}} dydx \\
&\quad + \int_a^b \int_a^b \alpha(y)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}} dydx),
\end{aligned} \tag{3.16}$$

in which,

$$\begin{aligned}&\int_a^b\int_a^b \alpha(y)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx\\&=\int_a^b\int_a^b \alpha(y)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dxdy,\end{aligned}\tag{3.17}$$

then if we interchanged x and y in the previous expression, we shall find

$$\begin{aligned}&\int_a^b\int_a^b \alpha(y)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dxdy\\&=\int_a^b\int_a^b \alpha(x)\frac{((v(x)-v(y))(u(x)-u(y)))}{|x-y|^{1+\gamma}}dydx,\end{aligned}\tag{3.18}$$

so

$$\begin{aligned}&a(u,v)\\&=\frac{1}{2}(\int_a^b\int_a^b \alpha(x)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx\\&\quad+\int_a^b\int_a^b \alpha(y)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx)\\&=\frac{1}{2}(\int_a^b\int_a^b \alpha(x)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx\\&\quad+\int_a^b\int_a^b \alpha(x)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx)\\&=\int_a^b\int_a^b \alpha(x)\frac{((v(y)-v(x))(u(y)-u(x)))}{|x-y|^{1+\gamma}}dydx.\end{aligned}\tag{3.19}$$

In order to discrete the integral equation and use the Galerkin finite element method to find the numerical solution of this model, we perform uniform partition on the whole field $\Omega \cup \Omega_\epsilon$, in which Ω_ϵ is an area around the boundary, and in this section, we considered $\Omega \cup \Omega_\epsilon$ as $(a-\epsilon, b+\epsilon)$. Then we defined the partition as following

$$x_i = a + ih,$$

$$-K \leq i \leq N + K, \tag{3.20}$$

$$h = \frac{b-a}{N}.$$

We shall introduce the standard hat function as we need, that is

$$\psi(\xi) = 1 - |\xi|, \qquad \xi \in [-1, 1],$$

then we can define $\phi_i(x) = \psi(\frac{x-x_i}{h})$, $x \in [x_{i-1}, x_{i+1}]$ in our one dimensional problem. In other words,

$$\phi_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{h} & x \in [x_{i-1}, x_i] \\ \\ \dfrac{x_{i+1} - x}{h} & x \in [x_i, x_{i+1}] \end{cases}, \tag{3.21}$$

with the hat functions $\{\phi_i(x)\}_{i=0}^N$ we introduced, the numerical solution of $u(x)$ can be approximated by the following equation

$$u(x) = \phi_0(x)u_a + \sum_{j=1}^{N-1} \phi_j(x)u_j + \phi_N(x)u_b, \tag{3.22}$$

for further discussion, we sometimes use the notations u_0 and u_N for u_a and u_b.

Then we choose the test function $v(x) = \phi_i(x)$, for $i = 1, ..., N-1$. According to (3.3), we proceed to discretization of $a(u, v)$,

$$\begin{aligned} &a(u,v) \\ &= \int_a^b \int_a^b \alpha(x) \frac{(\sum\limits_{j=0}^N \phi_j(y)u_j - \sum\limits_{j=0}^N \phi_j(x)u_j)(\phi_i(y) - \phi_i(x))}{|x-y|^{1+\gamma}} dydx, \qquad (3.23) \\ &= u_j \int_a^b \int_a^b \alpha(x) \frac{(\sum\limits_{j=0}^N \phi_j(y) - \sum\limits_{j=0}^N \phi_j(x))(\phi_i(y) - \phi_i(x))}{|x-y|^{1+\gamma}} dydx \end{aligned}$$

Before proceeding further, a simple manipulation is necessary

$$
\begin{aligned}
&(\sum_{j=0}^{N}\phi_j(y)-\sum_{j=0}^{N}\phi_j(x))(\phi_i(y)-\phi_i(x)) \\
&\quad=(\phi_0(y)-\phi_0(x)+\phi_1(y)-\phi_1(x) \\
&\qquad+...+\phi_N(y)-\phi_N(x))(\phi_i(y)-\phi_i(x)) \\
&\quad=(\phi_0(y)-\phi_0(x))(\phi_i(y)-\phi_i(x)) \qquad\qquad , \\
&\qquad+(\phi_1(y)-\phi_1(x))(\phi_i(y)-\phi_i(x)) \\
&\qquad+...+(\phi_N(y)-\phi_N(x))(\phi_i(y)-\phi_i(x)) \\
&\quad=\sum_{j=0}^{N}(\phi_j(y)-\phi_j(x))(\phi_i(y)-\phi_i(x))
\end{aligned}
\tag{3.24}
$$

finally, we remark that

$$
\begin{aligned}
&a(u,v) \\
&\quad=\sum_{j=0}^{N}\int_a^b\int_a^b\alpha(x)\frac{(\phi_j(y)-\phi_j(x))(\phi_i(y)-\phi_i(x))}{|x-y|^{1+\gamma}}dydxu_j.
\end{aligned}
\tag{3.25}
$$

After this sequence of works, we can derive out the following finite element formulation

$$\sum_{j=0}^{N}a(\phi_i,\phi_j)u_j=l(\phi_i), \tag{3.26}$$

since we have $u_0=u_a$ and $u_N=u_b$ on the boundary, the above formulation is equivalent to

$$\sum_{j=a}^{N-1}a(\phi_i,\phi_j)u_j=l(\phi_i)-a(\phi_i,\phi_0)u_a-a(\phi_i,\phi_N)u_b, \tag{3.27}$$

for $i=1,...,N-1$.

Now we get the matrix form of the equations system,

$$A\vec{u} = \vec{b} \tag{3.28}$$

where $\vec{u}$ is the numerical solution of $u(x)$, $\vec{u} = (u_1, u_2, ..., u_{N-1})^T$, $\vec{b} = (b_1, b_2, ..., b_{N-1})$,

$$b_i = l(\phi_i) - a(\phi_i, \phi_0)u_a - a(\phi_i, \phi_N)u_b, \tag{3.29}$$

and A is the stiffness matrix, which can be expressed as

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} & \dots & A_{1N-1} \\ A_{21} & A_{22} & A_{23} & & \\ A_{31} & A_{32} & A_{33} & & \vdots \\ \dots & & & \ddots & \\ A_{N-11} & A_{N-12} & A_{N-13} & \dots & A_{N-1N-1} \end{pmatrix}, \tag{3.30}$$

for each entry, we have

$$\begin{aligned} A_{ij} &= \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))(\phi_i(y) - \phi_i(x))}{|x-y|^{1+\gamma}} dydx \\ &= a(\phi_i, \phi_j). \end{aligned} \tag{3.31}$$

Up to now, we have gotten the general numerical scheme of the entries of stiffness matrix and right-hand terms, which can be used in standard numerical solver to find out the numerical solutions.

There are many previous studies that focus on the error estimates and computational difficulties. Since non-local peridynamic models always lead to a full stiffness matrix, when we are going to consider a problem with high partition density or high dimension, the cost of memory and computation will become very expensive. That is also our motivation of us to discover the structure of the stiffness matrix and use it to perform the fast method. We shall do some computation and then make some manipulation about the stiffness matrix, then try to introduce the fast method of this one dimensional variable coefficient non-local peridynamic model in the following sections.

3.3 A second look about the entries of the stiffness matrix

By using the expression of A_{ij} we derived before, we shall have

$$A_{ij} = \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))(\phi_i(y) - \phi_i(x))}{|x-y|^{1+\gamma}} dydx \tag{3.32}$$

$$= \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \tag{3.33}$$

$$- \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(x)}{|x-y|^{1+\gamma}} dydx, \tag{3.34}$$

in order to find the structure of A_{ij} with less computational works, we consider (3.34) first,

$$\begin{aligned} & - \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(x)}{|x-y|^{1+\gamma}} dydx \\ &= - \int_a^b \int_a^b \alpha(y) \frac{(\phi_j(x) - \phi_j(y))\phi_i(y)}{|y-x|^{1+\gamma}} dxdy \\ &= \int_a^b \int_a^b \alpha(y) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy, \\ &= \int_a^b \int_a^b \alpha(y) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \end{aligned} \tag{3.35}$$

here we interchanged x and y, and then switched the order of integration. Now it is obvious that (3.34) and (3.33) have a similar structure, so we shall only discuss (3.33) later.

First, we switch the order of integration of (3.33)

$$
\begin{aligned}
&\int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \\
&\quad = \int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy,
\end{aligned} \tag{3.36}
$$

then,

$$
\begin{aligned}
&\int_a^b \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy \\
&\quad = \int_a^b \phi_i(y) \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy,
\end{aligned} \tag{3.37}
$$

since when x is on the interval $[a, x_{i-1}] \cup [x_{i+1}, b]$, $\phi_i(x) = 0$,

$$
\begin{aligned}
&\int_a^b \phi_i(y) \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \\
&\quad = \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy,
\end{aligned} \tag{3.38}
$$

While on different subintervals of the whole field $[a-\epsilon, b+\epsilon]$, $\alpha(x)$ and $\{\phi_i(x)\}_{i=0}^N$ have different values and expressions, we write (3.33) as

$$
\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^b \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.39}
$$

$$
= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_{j-1}} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.40}
$$

$$
+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.41}
$$

$$
+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j+1}}^{b} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy. \tag{3.42}
$$

Because the stiffness matrix is symmetric, we only need to consider the up triangular matrix. And in order to explain the structure clearly, we shall discuss (3.39) in three cases. For each entry, we discuss $j \geq i+2$, $j = i+1$, and $j = i$ one by one.

By definition, $\phi_j(x) = 0$ and $\phi_j(y) = 0$ when $j \geq i+2$, which means

$$\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{a}^{x_{j-1}} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy$$

$$= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j+1}}^{b} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.43}$$

$$= 0.$$

Then we have, for $j \geq i+2$ and $x \in [x_{i-1}, x_{i+1}]$, $\phi_j(x) = 0$

$$\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{a}^{b} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy$$

$$= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x) \frac{(\phi_j(y) - \phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.44}$$

$$= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy,$$

next, we are going to show that the value of (3.41) only depends on the value of $\alpha(x)$ and the value of $|i-j|$.

For one more step, we write (3.44) as

$$\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.45}$$

$$= \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.46}$$

$$+ \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.47}$$

$$+ \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.48}$$

$$+ \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy. \tag{3.49}$$

Since (3.46), (3.47), (3.48) and (3.49) have similar structure, it suffices to prove that the value of (3.46) only depends on the value of $\alpha(x)$ and the value of $|i-j|$.

Let us assume initially $\alpha(x)$ can be approximated by a piece-wise defined function(2.11),

We proceed to prove this, by assumption before

$$\begin{aligned} &\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \alpha(x) \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \\ &= \alpha(x_{j-1} + \frac{h}{2}) \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \end{aligned}, \tag{3.50}$$

then we set $x = x_j + s$, and $y = x_i + t$ where $s \in [-h, 0]$ and $t \in [-h, 0]$, so we shall have $dx = ds$, $dy = dt$, furthermore, let us recall the definition of the hat function $\psi(\xi) = 1 - |\xi|$, where

$$\phi_i(x) = \psi(\frac{x - x_i}{h})$$

.

Now we have

$$\begin{aligned} \phi_j(x) &= \psi(\frac{x - x_j}{h}) \\ &= \psi(\frac{x_j - x_j + s}{h}) \\ &= \psi(\frac{s}{h}) \\ &= 1 - |\frac{s}{h}|, \end{aligned} \tag{3.51}$$

similarly,

$$
\begin{aligned}
\phi_i(y) &= \psi(\frac{y - x_i}{h}) \\
&= \psi(\frac{x_i - x_i + t}{h}) \\
&= \psi(\frac{t}{h}) \\
&= 1 - |\frac{t}{h}|,
\end{aligned} \tag{3.52}
$$

moreover,

$$
|x - y|^{1+\gamma} = |(j - i)h + t - s|^{1+\gamma},
$$

it follows that

$$
\begin{aligned}
&\alpha(x_{j-1} + \frac{h}{2}) \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x - y|^{1+\gamma}} dx dy \\
&= \alpha(x_{j-1} + \frac{h}{2}) \int_{-h}^{0} \int_{-h}^{0} \frac{-(1 - |\frac{s}{h}|)(1 - |\frac{t}{h}|)}{|(j - i)h + t - s|^{1+\gamma}} ds dt,
\end{aligned} \tag{3.53}
$$

then we can conclude that, the value of (3.46) only depends on the value of $\alpha(x)$ and the value of $|i - j|$, the remainder of (3.41) is analogous to that which is shown in before.

3.4 Analysis of the structure of the stiffness matrix

We first classified the entries of the stiffness matrix to three cases in the previous section, and discussed the case when $j \geq i + 2$. We derived out the expression of each entry, and separated it to two parts. Since two of them are similar, we focus on the first part, which is (3.33) only involved one term for the case $j \geq i + 1$. The other two cases are much more complicated, so we shall discuss them later.

According to this kind of classification, the stiffness matrix can be written as the sum of two matrices

$$A = A^{(tr)} + T, \tag{3.54}$$

where A^{tr} contains the two complicated cases, which is combined with diagonal and sub-diagonal entries. T is actually the case we discussed before.

Obviously T is also symmetric, the diagonal and sub-diagonal entries of T are all 0s. The other entries of the up-triangular part is defined as (3.41).

under the assumption that $\alpha(x)$ could be approximated by the piece-wise defined function $\sum_{k=0}^{N} \alpha(x_k + \frac{h}{2})\Lambda_{x\in[x_k,x_{k+1}]}(x)$, after manipulating them we can get

$$\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&= \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy,
\end{aligned} \tag{3.55}$$

then we factor out $\alpha(x)$ from the above expression,

$$\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&= \alpha(x_{j-1} + \frac{h}{2})(\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy) \\
&\quad + \alpha(x_j + \frac{h}{2})(\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy),
\end{aligned} \tag{3.56}$$

and because another term of (3.32), mean (3.34) could similarly be written as

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{x_{j+1}} \alpha(x)\frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&= \alpha(x_{i-1}+\frac{h}{2})(\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy) \\
&\quad + \alpha(x_i+\frac{h}{2})(\int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}}dxdy).
\end{aligned}
\tag{3.57}
$$

Finally, we can say that the matrix T could be written as a sum of four matrices

$$T = T^{(a)} + T^{(b)} + T^{(c)} + T^{(d)}, \tag{3.58}$$

furthermore, we have

$$
\begin{aligned}
T^{(a)} &= T^{(1)}A^{(1)} \\
T^{(b)} &= T^{(2)}A^{(2)} \\
T^{(c)} &= A^{(3)}T^{(3)} \\
T^{(d)} &= A^{(4)}T^{(4)},
\end{aligned}
\tag{3.59}
$$

where $A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}$ are diagonal matrices, and $T^{(1)}, T^{(2)}, T^{(3)}, T^{(4)}$ are Toeplitz matrices.

Finally, we conclude that the stiffness matrix A can be written as the following

$$A = A^{(tr)} + T^{(1)}A^{(1)} + T^{(2)}A^{(2)} + A^{(3)}T^{(3)} + A^{(4)}T^{(4)}, \tag{3.60}$$

A_{tr} is a tridiagonal matrix with all tridiagonal entries of A.

3.5 Computation and Storage about entries of the Stiffness Matrix

According to the results we got from the previous section, we can conclude that the matrices $\{A^{(p)}\}_{p=1}^{4}$ have the following structure

$$A^{(p)} = \begin{pmatrix} A_{11}^{(p)} & 0 & 0 & \dots & 0 \\ 0 & A_{22}^{(p)} & 0 & & \\ 0 & 0 & A_{33}^{(p)} & & \vdots \\ \dots & & & \ddots & \\ 0 & 0 & 0 & \dots & A_{N-1N-1}^{(p)} \end{pmatrix}, \tag{3.61}$$

where for A^1, A^2

$$A_{jj}^{(p)} = \begin{cases} \alpha(x_j + \frac{h}{2}) & p = 1 \\ \alpha(x_{j-1} + \frac{h}{2}) & p = 2 \end{cases}, \tag{3.62}$$

and for A^3, A^4

$$A_{ii}^{(p)} = \begin{cases} \alpha(x_i + \frac{h}{2}) & p = 3 \\ \alpha(x_{i-1} + \frac{h}{2}) & p = 4 \end{cases}. \tag{3.63}$$

The structure and entries of $\{T^{(p)}\}_{p=1}^{4}$ also can be observed,

$$T^{(p)} = \begin{pmatrix} 0 & 0 & T_{13}^{(p)} & \dots & T_{1N-1}^{(p)} \\ 0 & 0 & 0 & & \\ T_{31}^{(p)} & 0 & 0 & & \vdots \\ \dots & & & \ddots & \\ T_{N-11}^{(p)} & T_{N-12}^{(p)} & T_{N-13}^{(p)} & \dots & 0 \end{pmatrix}, \tag{3.64}$$

and for different p, $T_{ij}^{(p)}$ can be written as

$$T_{ij}^{(1)} = (\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.65}$$

$$+ \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy) \tag{3.66}$$

$$T_{ij}^{(2)} = (\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.67}$$

$$+ \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy) \tag{3.68}$$

$$T_{ij}^{(3)} = (\int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{j-1}}^{j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.69}$$

$$+ \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_j}^{j+1} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy) \tag{3.70}$$

$$T_{ij}^{(4)} = (\int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{j-1}}^{x_j} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.71}$$

$$+ \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_j}^{x_{j+1}} \frac{(-\phi_j(x))}{|x-y|^{1+\gamma}} dxdy), \tag{3.72}$$

since $\{T^{(p)}\}_{p=1}^{4}$ are $N-1$ by $N-1$ symmetric matrices, and by the proof from previous section, they are Toeplitz. Therefore, we only need to find out the values of $T_{13}^{(p)}$, $T_{14}^{(p)}$, ..., $T_{1N-1}^{(p)}$.

Next we are going to focus on the other two cases we have mentioned before, which are the tridiagonal entries of A and $A^{(tr)}$.

Since $A^{(tr)}$ is a symmetric tridiagonal matrix, we can write it as the following

$$A^{(tr)} = \begin{pmatrix} A_{11}^{(tr)} & A_{12}^{(tr)} & 0 & \dots & 0 \\ A_{21}^{(tr)} & A_{22}^{(tr)} & A_{23}^{(tr)} & & \\ 0 & A_{32}^{(tr)} & A_{33}^{(tr)} & & \vdots \\ \dots & & & \ddots & \\ 0 & 0 & A_{N-1N-2}^{(tr)} & \dots & A_{N-1N-1}^{(tr)} \end{pmatrix}, \tag{3.73}$$

and only consider the cases $j = i + 1$ and $j = i$.

Let us start from the case $j = i + 1$, recalling the first term of (3.32), we have

$$\int_a^b \int_a^b \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \tag{3.74}$$

$$= \int_a^b \int_a^b \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy \tag{3.75}$$

$$= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_i} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.76}$$

$$+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_i}^{x_{i+2}} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.77}$$

$$+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+2}}^{b} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy, \tag{3.78}$$

in our previous discussion, we know that when $j \geq i + 2$, the terms (3.76) and (3.78) are 0.

But in this case, for (3.76), we shall find that

$$\begin{aligned} &x \in [a, x_i], \quad \phi_{i+1}(x) = 0; \\ &y \in [x_{i-1}, x_{i+1}], \quad \phi_i(y) \neq 0; \\ &y \in [x_i, x_{i+1}], \quad \phi_{i+1}(y) \neq 0. \end{aligned} \tag{3.79}$$

Again, we assume the $\alpha(x)$ could be approximated by a piece-wise defined function

$$\alpha(x) = \sum_{k=0}^{N} \alpha(x_k + \frac{h}{2}) \Lambda_{x \in [x_k, x_{k+1}]}(x), \tag{3.80}$$

the term (3.76) could be expressed as

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_i} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \\
&= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y)\phi_{i+1}(y) \int_a^{x_i} \frac{(\alpha(x)}{(y-x)^{1+\gamma}} dxdy \\
&= \sum_{k=0}^{i-1} \int_{x_{i-1}}^{x_{i+1}} \phi_i(y)\phi_{i+1}(y) \int_{x_k}^{x_{k+1}} \frac{(\alpha(x_k + \frac{h}{2})}{(y-x)^{1+\gamma}} dxdy \\
&= \sum_{k=0}^{i-1} (\alpha(x_k + \frac{h}{2}) \int_{x_{i-1}}^{x_{i+1}} \phi_i(y)\phi_{i+1}(y) \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy \\
&= \sum_{k=0}^{i-1} (\alpha(x_k + \frac{h}{2}) \int_{x_{i-1}}^{x_{i+1}} (\frac{x_{i+1}-y}{h})(\frac{y-x_i}{h}) \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy.
\end{aligned}
\tag{3.81}
$$

We then discuss the second term (3.77)

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_i}^{x_{i+2}} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \\
&= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_i}^{x_{i+1}} \alpha(x_i + \frac{h}{2}) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \\
&\quad + \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+1}}^{x_{i+2}} \alpha(x_{i+1} + \frac{h}{2}) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \\
&= \alpha(x_i + \frac{h}{2}) (\int_{x_{i-1}}^{x_i} \frac{y-x_{i-1}}{h} \int_{x_i}^{x_{i+1}} \frac{-\frac{x-x_i}{h}}{(x-y)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \frac{x_{i+1}-y}{h} \int_{x_i}^{y} \frac{\frac{y-x_i}{h} - \frac{x-x_i}{h}}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \frac{x_{i+1}-y}{h} \int_{y}^{x_{i+1}} \frac{\frac{y-x_i}{h} - \frac{x-x_i}{h}}{(x-y)^{1+\gamma}} dxdy) \\
&\quad + \alpha(x_{i+1} + \frac{h}{2}) (\int_{x_i}^{x_{i+1}} \frac{x_{i+1}-y}{h} \int_{x_{i+1}}^{x_{i+2}} \frac{\frac{y-x_i}{h} - \frac{x_{i+2}-x}{h}}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} \frac{y-x_{i-1}}{h} \int_{x_{i+1}}^{x_{i+2}} \frac{-\frac{x_{i+2}-x}{h}}{(x-y)^{1+\gamma}} dxdy),
\end{aligned}
\tag{3.82}
$$

now the remainder works are computing each part of the above expression.

For the last term (3.78), the following should be true

$$
\begin{aligned}
&x \in [x_{i+2}, b], \quad \phi_{i+1}(x) = 0; \\
&y \in [x_{i-1,x_i}], \quad \phi_{i+1}(y) = 0.
\end{aligned} \tag{3.83}
$$

we consider the piece-wise defined approximation of $\alpha(x)$, which is defined by (2.11), the term (3.78) can be written as

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+2}}^{b} \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|x-y|^{1+\gamma}} dxdy \\
&= \int_{x_i}^{x_{i-1}} \phi_i(y)\phi_{i+1}(y) \int_{x_{i+2}}^{b} \frac{\alpha(x)}{(x-y)^{1+\gamma}} dxdy \\
&= \sum_{k=i+2}^{N-1} \alpha(x_k + \frac{h}{2}) \int_{x_i}^{x_{i-1}} (\frac{x_{i+1}-y}{h})(\frac{y-x_i}{h}) \int_{x_k}^{x_{k+1}} \frac{1}{(x-y)^{1+\gamma}} dxdy.
\end{aligned} \tag{3.84}
$$

Here we discussed all terms of the case when $j = i+1$. We are now going to consider the last case, where $j = i$.

We consider it in different intervals

$$
\int_a^b \int_a^b \alpha(x) \frac{(\phi_i(y) - \phi_i(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \tag{3.85}
$$

$$
= \int_a^b \int_a^b \alpha(x) \frac{(\phi_i(y) - \phi_i(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy \tag{3.86}
$$

$$
= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_{i-1}} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(y-x)^{1+\gamma}} dxdy \tag{3.87}
$$

$$
+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i-1}}^{x_{i+1}} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{|x-y|^{1+\gamma}} dxdy \tag{3.88}
$$

$$
+ \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+1}}^{b} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(x-y)^{1+\gamma}} dxdy. \tag{3.89}
$$

For (3.87), we see

$$\begin{aligned} &x \in [a, x_{i-1}], \quad \phi_i(x) = 0; \\ &y \in [x_{i-1}, x_{i+1}], \quad \phi_i(y) \neq 0. \end{aligned} \tag{3.90}$$

We first shall have the following derivation

$$\begin{aligned} &\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_{i-1}} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\ &= \int_{x_{i-1}}^{x_i} (\frac{y - x_{i-1}}{h})^2 \int_a^{x_{i-1}} \frac{\alpha(x)}{(y-x)^{1+\gamma}} dxdy \\ &+ \int_{x_i}^{x_{i+1}} (\frac{x_{i+1} - y}{h})^2 \int_a^{x_{i-1}} \frac{\alpha(x)}{(y-x)^{1+\gamma}} dxdy, \end{aligned} \tag{3.91}$$

then under the assumption that

$$\alpha(x) = \sum_{k=0}^{N} \alpha(x_k + \frac{h}{2}) \Lambda_{x \in [x_k, x_{k+1}]}(x),$$

the term (3.87) could be written as

$$\begin{aligned} &\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_a^{x_{i-1}} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\ &= \sum_{k=0}^{i-2} \alpha(x_k + \frac{h}{2}) (\int_{x_{i-1}}^{x_i} (\frac{y - x_{i-1}}{h})^2 \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy \\ &+ \int_{x_i}^{x_{i+1}} (\frac{x_{i+1} - y}{h})^2 \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy), \end{aligned} \tag{3.92}$$

the second term of (3.85) should be a little bit complicated, since x and y sit in the same interval. The following holds

$$\begin{aligned} &x \in [x_{i-1}, x_{i+1}], \quad \phi_i(x) \neq 0; \\ &y \in [x_{i-1}, x_{i+1}], \quad \phi_i(y) \neq 0. \end{aligned} \tag{3.93}$$

And because x and y shall meet at some point, we need to discuss it in six cases

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i-1}}^{x_{i+1}} \alpha(x) \frac{(\phi_i(y)-\phi_i(x))}{|x-y|^{1+\gamma}} dxdy \\
&\quad = \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_{i-1}}^{y} \alpha(x_{i-1}+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{y}^{x_i} \alpha(x_{i-1}+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(x-y)^{1+\gamma}} dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} \phi_i(y) \int_{x_i}^{x_{i+1}} \alpha(x_i+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(x-y)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_{i-1}}^{x_i} \alpha(x_{i-1}+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{x_i}^{y} \alpha(x_i+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} \phi_i(y) \int_{y}^{x_{i+1}} \alpha(x_i+\frac{h}{2}) \frac{(\phi_i(y)-\phi_i(x))}{(x-y)^{1+\gamma}} dxdy,
\end{aligned} \tag{3.94}
$$

after manipulation we can get

$$
\begin{aligned}
&\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i-1}}^{x_{i+1}} \alpha(x) \frac{(\phi_i(y)-\phi_i(x))}{|x-y|^{1+\gamma}} dxdy \\
&= \alpha(x_{i-1}+\frac{h}{2})(\int_{x_{i-1}}^{x_i} (\frac{y-x_{i-1}}{h}) \int_{x_{i-1}}^{y} \frac{((\frac{y-x_{i-1}}{h})-(\frac{x-x_{i-1}}{h}))}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_{i-1}}^{x_i} (\frac{y-x_{i-1}}{h}) \int_{y}^{x_i} \frac{((\frac{y-x_{i-1}}{h})-(\frac{x-x_{i-1}}{h}))}{(x-y)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} (\frac{x_{i+1}-y}{h}) \int_{x_{i-1}}^{x_i} \frac{((\frac{x_{i+1}-y}{h})-(\frac{x-x_{i-1}}{h}))}{(y-x)^{1+\gamma}} dxdy) \\
&\quad + \alpha(x_i+\frac{h}{2})(\int_{x_{i-1}}^{x_i} (\frac{y-x_{i-1}}{h}) \int_{x_i}^{x_{i+1}} \frac{((\frac{y-x_{i-1}}{h})-(\frac{x_{i+1}-x}{h}))}{(x-y)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} (\frac{x_{i+1}-y}{h}) \int_{x_i}^{y} \frac{((\frac{x_{i+1}-y}{h})-(\frac{x_{i+1}-x}{h}))}{(y-x)^{1+\gamma}} dxdy \\
&\quad + \int_{x_i}^{x_{i+1}} (\frac{x_{i+1}-y}{h}) \int_{y}^{x_{i+1}} \frac{((\frac{x_{i+1}-y}{h})-(\frac{x_{i+1}-x}{h}))}{(x-y)^{1+\gamma}} dxdy),
\end{aligned} \tag{3.95}
$$

Here we find that there are many terms in case two, since we are trying to figure out the integration involving absolute value, and later we shall use several subfunction to simplify our coding.

The last term that needs to be discussed is (3.89), where

$$\begin{aligned} &x \in [x_{i+1}, b], \quad \phi_i(x) = 0; \\ &y \in [x_{i-1}, x_{i+1}], \quad \phi_i(y) \neq 0. \end{aligned} \tag{3.96}$$

And write (3.89) as

$$\begin{aligned} &\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+1}}^{b} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(x-y)^{1+\gamma}} dxdy \\ &= \int_{x_{i-1}}^{x_i} (\frac{y - x_{i-1}}{h})^2 \int_{x_{i+1}}^{b} \frac{\alpha(x)}{(x-y)^{1+\gamma}} dxdy \\ &+ \int_{x_i}^{x_{i+1}} (\frac{x_{i+1} - y}{h})^2 \int_{x_{i+1}}^{b} \frac{\alpha(x)}{(x-y)^{1+\gamma}} dxdy, \end{aligned} \tag{3.97}$$

with the assumption of $\alpha(x)$, we now derive the following expression

$$\begin{aligned} &\int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_{i+1}}^{b} \alpha(x) \frac{(\phi_i(y) - \phi_i(x))}{(y-x)^{1+\gamma}} dxdy \\ &= \sum_{k=i+1}^{N-1} \alpha(x_k + \frac{h}{2}) (\int_{x_{i-1}}^{x_i} (\frac{y - x_{i-1}}{h})^2 \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy \\ &+ \int_{x_i}^{x_{i+1}} (\frac{x_{i+1} - y}{h})^2 \int_{x_k}^{x_{k+1}} \frac{1}{(y-x)^{1+\gamma}} dxdy), \end{aligned} \tag{3.98}$$

we finally got all the entries of the stiffness matrix. We shall use Gaussion-elimination and Conjugate Gradient Squared methods to find the numerical solution of the model, and derive the fast numerical method in the next section.

3.6 The fast method for one dimensional variable-coefficient peridynamic model

In previous sections, we discussed the variational formulation, and derived out the matrix equation of the variable coefficient non-local peridynamic model. We recall that

$$A\vec{u} = \vec{b}$$

,

where A could be written as the sum of the tridiagonal matrix $A^{(tr)}$ and a symmetric Toeplitz matrix T.

It is obvious that our variable coefficient non-local peridynamic model always leads to a full stiffness matrix, so the computational cost shall be $O(N^3)$ and $O(N^2)$ storage memory to be needed when processing a standard Gaussian elimination method. Even by using a standard Conjugate Gradient Squared iteration method, $O(N^2)$ memory space shall be needed to store the full stiffness matrix, and in each iteration, the computational cost would be $O(N^2)$. Now we are going to introduce our fast method with efficient storage.

Let us recall the Conjugate Gradient Squared iteration scheme (1.40). The most computational cost is for the matrix vector multiplication, which is $O(N^2)$. So first, we consider the stiffness matrix separately, say $A^{(tr)} + T$.

Now for any matrix vector multiplication Ad, we can write it as

$$Ad = A^{(tr)}d + Td$$

,

since $A^{(tr)}$ is just a tridiagonal matrix, and furthermore it is a symmetric matrix, we only need to store the entries of diagonal and subdiagonal. For an $N \times N$ matrix, it shall need $2N-1$ memory, which is $O(N)$. And for the matrix vector multiplication, we need $3N-2$ multiplications and $3N-5$ additions, which is $O(N)$.

Here we come to the second term Td, by previous discussion, we know that A is symmetric, so T is symmetric. Then we write T as a sum of an up-triangular matrix T^{up} and its transpose T^{low}

$$T = T^{(up)} + T^{(low)}$$

,

as we mentioned before, $T^{(up)}$ and $T^{(low)}$ could both be written as a sum of four matrices. Since $T^{(low)} = T^{(upT)}$, we consider the up-triangular matrix,

$$T^{(up)} = T^{(aup)} + T^{(bup)} + T^{(cup)} + T^{(dup)}$$

,

where

$$\begin{aligned} T^{(aup)} &= T^{(1up)}A^{(1)} \\ T^{(bup)} &= T^{(2up)}A^{(2)} \\ T^{(cup)} &= A^{(3)}T^{(3up)} \\ T^{(dup)} &= A^{(4)}T^{(4up)}, \end{aligned} \tag{3.99}$$

and here $T^{(1up)}$, $T^{(2up)}$, $T^{(3up)}$, $T^{(4up)}$ involving the upper-triangular entries of corresponding Toeplitz matrix $T^{(1)}$, $T^{(2)}$, $T^{(3)}$, $T^{(4)}$. And then $T^{(low)}$ can be written as

$$T^{(low)} = (T^{(aup)} + T^{(bup)} + T^{(cup)} + T^{(dup)})^{(T)}$$

,

in one more step, we can see

$$T^{(low)} = A^{(1)}T^{(1upT)} + A^{(2)}T^{(2upT)} + T^{(3upT)}A^{(3)} + T^{(4upT)}A^{(4)}$$

.

To consider with our model, since in our model the entries of A are values of the piecewise defined coefficient function $\alpha(x)$, $A^{(1)} = A^{(3)}$, and $A^{(2)} = A^{(4)}$. We can have

$$\begin{aligned}
T &= T^{(1up)}A^{(1)} + T^{(2up)}A^{(2)} + A^{(3)}T^{(3up)} + A^{(4)}T^{(4up)} \\
&\quad + A^{(1)}T^{(1upT)} + A^{(2)}T^{(2upT)} + T^{(3upT)}A^{(3)} + T^{(4upT)}A^{(4)} \\
&= T^{(1up)}A^{(1)} + T^{(2up)}A^{(2)} + A^{(1)}T^{(3up)} + A^{(2)}T^{(4up)} \\
&\quad + A^{(1)}T^{(1upT)} + A^{(2)}T^{(2upT)} + T^{(3upT)}A^{(1)} + T^{(4upT)}A^{(2)} \\
&= (T^{(1up)} + T^{(3upT)})A^{(1)} + (T^{(2up)} + T^{(4upT)})A^{(2)} \\
&\quad + A^{(1)}(T^{(3up)} + T^{(1upT)}) + A^{(2)}(T^{(4up)} + T^{(2upT)}),
\end{aligned} \tag{3.100}$$

here we see that the multiplications of any $\{A^{(i)}\}_i$ and any matrix, the computational cost is $O(N)$. And since $\{T^{(iup)}\}_i$ are upper-triangular Toeplitz matrices, $\{T^{(iupT)}\}_i$ are lower-triangular Toeplitz matrices. the sum of these two types of matrices should be a full Toeplitz matrix.

Then the matrix-vector multiplication Td can be written as

$$\begin{aligned}
Td &= ((T^{(1up)} + T^{(3upT)})A^{(1)} + (T^{(2up)} + T^{(4upT)})A^{(2)} \\
&\quad + A^{(1)}(T^{(3up)} + T^{(1upT)}) + A^{(2)}(T^{(4up)} + T^{(2upT)}))d \\
&= (T^{(1up)} + T^{(3upT)})A^{(1)}d + (T^{(2up)} + T^{(4upT)})A^{(2)}d \\
&\quad + A^{(1)}(T^{(3up)} + T^{(1upT)})d + A^{(2)}(T^{(4up)} + T^{(2upT)})d,
\end{aligned} \tag{3.101}$$

because $A^{(1)}$ and $A^{(2)}$ are diagonal matrices, $A^{(1)}(T^{(3up)}+T^{(1upT)})$ and $A^{(2)}(T^{(4up)}+T^{(2upT)})$ are also Toeplitz matrices. So, we can implement the fast matrix-vector multiplication to each part of the above expression.

Without loss of generality, we consider the first one of the four multiplications. Here we write the Toeplitz matrix $(T^{(1up)}+T^{(3upT)})A^1(1)$ as $T^{(0)}$. Where

$$T^{(0)} = \begin{pmatrix} 0 & 0 & T^{(0)}_{13} & \dots & T^{(0)}_{1N-1} \\ 0 & 0 & 0 & & \\ T^{(0)}_{31} & 0 & 0 & & \vdots \\ \dots & & & \ddots & \\ T^{(0)}_{N-11} & T^{(0)}_{N-12} & T^{(0)}_{N-13} & \dots & 0 \end{pmatrix}, \tag{3.102}$$

in order to show the fast matrix-vector multiplication clearly, we denote $q_{j-i} = T^{(0)}_{ij}$, and the sub-indices could be replaced be one group of numbers. Then we shall embed the $N-1 \times N-1$ Toeplitz matrix T^0 into an $2(N-1) \times 2(N-1)$ circulant matrix C as follows

$$C := \begin{pmatrix} T^o & B \\ B & T^o \end{pmatrix} \qquad B := \begin{pmatrix} 0 & q_{2-N} & \dots & 0 & 0 \\ q_{N-2} & 0 & q_{2-N} & \dots & 0 \\ \vdots & q_{N-2} & 0 & \ddots & \vdots \\ 0 & \vdots & \ddots & \ddots & q_{2-N} \\ 0 & 0 & \dots & q_{N-2} & 0 \end{pmatrix}. \tag{3.103}$$

For the N dimensional vector d, we embed it into a $2N-1$ dimensional vector w, where

$$w := \begin{pmatrix} d \\ 0 \end{pmatrix}. \tag{3.104}$$

And by previous studies, we know that the $2(N-1) \times 2(N-1)$ circulant matrix C has the following decomposition

$$C = F^{-1}\mathrm{diag}(Fc)F, \tag{3.105}$$

we observe that Td should be the first $2N-1$ entries of Cw.

Since F is the $2(N-1) \times 2(N-1)$ discrete Fourier transform matrix, and c is the first column of C, we can conclude that Fc and Fw can be figured out in $O(2Nlog2N)$ operations by applying the Fast Fourier Transform (FFT). Then for our numerical scheme, the matrix-vector multiplication Td can be evaluated in $O(NlogN)$ operations.

Then by considering the previous discussion about the tri-diagonal matrix $A^{(tr)}$, we shall conclude that by using our fast matrix-vector multiplication, Ad can be carried out by $O(NlogN)$ operations. So, if we implement our fast method to the Conjugate Gradient Squared method, the computational cost should be $O(NlogN)$ per iteration.

3.7 Efficient evaluation and storage of the stiffness matrix

By previous discussion, we can see that the numerical scheme for the variable coefficient peridynamic model shall generate a $2N-1 \times 2N-1$ full stiffness matrix, and since the standard situation of all the entries are different, then we need to store the $(2N-1)^2$ entries separately.

But after we write the stiffness matrix A as the sum of $A^{(tr)}$ and T, we can simply store the $3N-5$ entries of $A^{(tr)}$. And for each part of T we only need to store $2N-6$ entries, since they are Toeplitz matrices with zeros on tri-diagonal entries.

Here we accurately stored the stiffness matrix without a loss of compression.

The next job is to discuss the computational cost of evaluating the stiffness matrix A. We still consider it as two parts, $A^{(tr)}$ and T.

For the matrix T, as we did before, it has been considered as a sum of four parts, and the Toeplitz matrix of each part just consists of entries we already evaluated. The cost of the product of a diagonal matrix and a Toeplitz matrix is only $O(N)$, which is about

$$(T^{(1up)} + T^{(3upT)})A^{(1)},$$

and

$$(T^{(2up)} + T^{(4upT)})A^{(2)}.$$

Then we are going to consider the tri-diagonal matrix $A^{(tr)}$. Since A is symmetric, we only focus on the upper-diagonal and diagonal entries.

For upper-diagonal entries, let us recall the expression (3.74), which consists three terms (3.76), (3.77) and (3.78). After we rewrite (3.76) and (3.78) as sums in terms of $\alpha(x_k + \frac{h}{2})$, it can be found that each entry of the upper-diagonal need at least $O(N)$ operations to be calculated, and for all the $N-2$ entries, the computational cost should be $O(N^2)$.

And recalling the expression (3.85), which is about the diagonal entries, we can say the same is true for one type of expression of diagonal entries. The computational cost of evaluating diagonal entries would also be $O(N^2)$.

So we say it is necessary to accelerate the computational work of evaluating diagonal and sub-diagonal entries of the stiffness matrix.

In order to derive out the fast method of evaluating the entries of tri-diagonal, we first consider the upper-diagonal entries. Without loss of generality, we recall the expression (3.74), and simply write a_{ii+1} as following

$$\int_a^b \int_a^b \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \tag{3.106}$$

$$= \int_a^b \int_a^b \alpha(x) \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy \tag{3.107}$$

$$= \sum_{k=0}^{N-1} (\alpha(x_k + \frac{h}{2}) \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_k}^{x_{k+1}} \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|y-x|^{1+\gamma}} dxdy, \tag{3.108}$$

since $i = 1, ..., N-2$, we can write all the entries of upper-diagonal as a vector a^u, where $a^u \in \mathbb{R}^{N-2}$, and it is a product of an $N-2 \times N$ matrix A^u and a vector $\alpha \in \mathbb{R}^N$.

Furthermore, the entries $A_{ik}^{(u)}$ can be expressed as

$$A_{ik}^{(u)} = \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_k}^{x_{k+1}} \frac{(\phi_{i+1}(y) - \phi_{i+1}(x))}{|y-x|^{1+\gamma}} dxdy, \tag{3.109}$$

for $k = i$ and $k = i+1$, which is corresponding to the term (3.77), we consider them as the diagonal and upper-diagonal entries of $A^{(u)}$, and after some simple operations we can see $A^{(u)}$ has the Toeplitz structure.

In a general case, then we set $x = x_k + s$, and $y = x_i + t$ where $s \in [0, h]$ and $t \in [-h, h]$, so we shall have $dx = ds$, $dy = dt$, further more, let us recall the definition of the hat function $\psi(\xi) = 1 - |\xi|$, where

$$\phi_i(x) = \psi(\frac{x - x_i}{h}),$$

then (3.109) is equivalent to

$$\begin{aligned} A_{ik}^{(u)} &= \int_{-h}^{h} \psi(\frac{x_i + t - x_i}{h}) \int_0^h \frac{\psi(\frac{x_i+t-x_{i+1}}{h}) - \psi(\frac{x_k+t-x_{i+1}}{h})}{|x_i - x_k + t - s|^{1+\gamma}} \\ &= \int_{-h}^{h} \psi(\frac{t}{h}) \int_0^h \frac{\psi(\frac{t-h}{h}) - \psi(\frac{t-(k-i-1)h}{h})}{|(i-k)h + t - s|^{1+\gamma}}, \end{aligned} \tag{3.110}$$

from the above expression, we can easily conclude that $A_{ik}^{(u)}$ only depends on the value of $i - k$, which means the $A^{(u)}$ is a Toeplitz matrix.

Since we are going to implement the fast matrix-vector multiplication, we first expend the $N - 2 \times N$ matrix $A^{(u)}$ to an $N \times N$ matrix. Here we just need to attach two rows to the bottom of A^u, which with the entries do not change the Toeplitz structure of $A^{(u)}$.

Here we can embed the $N \times N$ Toeplitz matrix $A^{(u)}$ into an $2N \times 2N$ circulant matrix $C^{(u)}$ as follows

$$C^{(u)} := \begin{pmatrix} A^{(u)} & B^{(u)} \\ B^{(u)} & A^{(u)} \end{pmatrix} \qquad B^{(u)} := \begin{pmatrix} 0 & q_{2-N} & \dots & 0 & 0 \\ q_{N-2} & 0 & q_{2-N} & \dots & 0 \\ \vdots & q_{N-2} & 0 & \ddots & \vdots \\ 0 & \vdots & \ddots & \ddots & q_{2-N} \\ 0 & 0 & \dots & q_{N-2} & 0 \end{pmatrix}, \tag{3.111}$$

we use the amended notation of q_{j-i} in above matrix, which is $q_{j-i} = A^u_{ij}$.

Then use the decomposition of $2N \times 2N$ circulant matrix

$$C^{(u)} = F^{-1}\mathrm{diag}(Fc^{(u)})F, \tag{3.112}$$

where $c^{(u)}$ is the first column of $C^{(u)}$.

The N dimensional vector then need to be embedded into a $2N$ dimensional vector, which has zeros on the $N+1$ to $2N$ entries. And the first $N-2$ entries of the product of the $2N \times 2N$ matrix and the $2N$ dimensional vector is the upper-diagonal we are seeking.

As we discussed before, the computational cost of evaluating the sub-diagonal entries of A can be reduced to $O(NlogN)$.

For the diagonal entries, we recall its expression (3.85), and rewrite a_{ii} as

$$\int_a^b \int_a^b \alpha(x) \frac{(\phi_i(y) - \phi_i(x))\phi_i(y)}{|x-y|^{1+\gamma}} dydx \tag{3.113}$$

$$= \int_a^b \int_a^b \alpha(x) \frac{(\phi_i(y) - \phi_i(x))\phi_i(y)}{|x-y|^{1+\gamma}} dxdy \tag{3.114}$$

$$= \sum_{k=0}^{N-1} (\alpha(x_k + \frac{h}{2}) \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_k}^{x_{k+1}} \frac{(\phi_i(y) - \phi_i(x))}{|y-x|^{1+\gamma}} dxdy, \tag{3.115}$$

with a similar discussion, we can write all the entries of diagonal as a vector $a^{(d)}$, where $a^{(d)} \in \mathbb{R}^{N-1}$, and it is a product of an $N-1 \times N$ matrix $A^{(u)}$ and a vector $\alpha \in \mathbb{R}^N$.

Where $A_{ik}^{(d)}$ can be written as

$$
\begin{aligned}
A_{ik}^{(d)} &= \int_{x_{i-1}}^{x_{i+1}} \phi_i(y) \int_{x_k}^{x_{k+1}} \frac{(\phi_i(y) - \phi_i(x))}{|y-x|^{1+\gamma}} dxdy \\
&= \int_{-h}^{h} \psi(\frac{x_i + t - x_i}{h}) \int_0^h \frac{\psi(\frac{x_i+t-x_{i+1}}{h}) - \psi(\frac{x_k+t-x_i}{h})}{|x_i - x_k + t - s|^{1+\gamma}} \\
&= \int_{-h}^{h} \psi(\frac{t}{h}) \int_0^h \frac{\psi(\frac{t-h}{h}) - \psi(\frac{t-(i-k)h}{h})}{|(i-k)h + t - s|^{1+\gamma}},
\end{aligned}
\tag{3.116}
$$

then we come to the same conclusion, $A_{ik}^{(d)}$ only depends on the value of $i-k$, so $A^{(d)}$ has the Toeplitz structure.

Here we just repeat the works we did for upper-diagonal entries of A, and we can get all the diagonal entries of A with $O(NlogN)$ operations without any loss of compression.

3.8 Numerical experiments

In this section, we shall use sequence of numerical experiments to show the priorities of our fast method. Here we use Gaussian elimination, conjugate gradient squared (CGS) method, and fast conjugate gradient squared(FCGS) method to generate Matlab codes, and ran them in a 16GB-ROM laptop.

In our numerical experiments, we set the spatial domain $(a, b) = (0, 1)$, and the real solution $u(x) = (1-x)^2(1+x)^2$, with the variable coefficient $\alpha(x) = 1 - x^2$. And the analytic expression of right-hand side term $b(x)$ could be found out in each point x_i. We shall use different γ in the kernel function $\sigma(x, y) = \frac{1}{|x-y|^{1+\gamma}}$ to show that for our simplified model, the effect of γ on convergence rate and speed has been limited.

Example 1. We set $\gamma = 1/10$, and let N be from 2^6 to 2^{12}. We shown the numerical solution in **Table 1**.

Example 2. We set $\gamma = -1/2$, and let N be from 2^6 to 2^{12}. We shown the numerical solution in **Table 2**.

Table 3.1: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 1/10$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPu Time
Gauss	2^{-6}	$2.28624498e-03$	–	0.40s
	2^{-7}	$8.88216468e-04$	–	1.51s
	2^{-8}	$3.31472991e-04$	–	6.02s
	2^{-9}	$1.20887038e-04$	–	23.79s
	2^{-10}	$4.34995144e-05$	–	103.08s
	2^{-11}	$1.55277259e-05$	–	499.92s
	2^{-12}	$5.41799011e-06$	–	2495.76s
CGS	2^{-6}	$2.28624498e-03$	31	0.38s
	2^{-7}	$8.88216468e-04$	47	1.49s
	2^{-8}	$3.31472991e-04$	62	5.81s
	2^{-9}	$1.20887038e-04$	78	22.66s
	2^{-10}	$4.34995144e-05$	94	91.39s
	2^{-11}	$1.55277259e-05$	110	364.27s
	2^{-12}	$5.41799011e-06$	125	1457.31s
FCGS	2^{-6}	$2.28624498e-03$	31	0.07s
	2^{-7}	$8.88216468e-04$	47	0.12s
	2^{-8}	$3.31472991e-04$	62	0.25s
	2^{-9}	$31.20887038e-04$	78	0.61s
	2^{-10}	$4.34995154e-05$	94	1.60s
	2^{-11}	$1.55277269e-05$	110	7.17s
	2^{-12}	$5.41798742e-06$	125	42.65s

Table 3.2: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = -1/2$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPu Time
Gauss	2^{-6}	$2.99603704e-03$	–	0.40s
	2^{-7}	$1.13013076e-03$	–	1.51s
	2^{-8}	$4.12885038e-04$	–	5.94s
	2^{-9}	$1.48412795e-04$	–	23.69s
	2^{-10}	$5.29108864e-05$	–	101.36s
	2^{-11}	$1.87798603e-05$	–	476.01s
	2^{-12}	$6.67340814e-06$	–	2481.82s
CGS	2^{-6}	$2.99603704e-03$	30	0.35s
	2^{-7}	$1.13013076e-03$	43	1.47s
	2^{-8}	$4.12885038e-04$	51	5.75s
	2^{-9}	$1.48412795e-04$	55	22.93s
	2^{-10}	$5.29108864e-05$	59	90.11s
	2^{-11}	$1.87798603e-05$	61	357.93s
	2^{-12}	$6.67340814e-06$	63	1435.33s
FCGS	2^{-6}	$2.99603704e-03$	30	0.08s
	2^{-7}	$1.13013076e-03$	43	0.11s
	2^{-8}	$4.12885038e-04$	51	0.22s
	2^{-9}	$1.48412795e-04$	55	0.53s
	2^{-10}	$5.29108866e-05$	59	1.25s
	2^{-11}	$1.87798605e-05$	61	6.35s
	2^{-12}	$6.67340804e-06$	63	41.46s

Example 3. We set $\gamma = -3/4$, and let N be from 2^6 to 2^{12}. We shown the numerical solution in **Table 3**.

Table 3.3: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = -3/4$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPu Time
Gauss	2^{-6}	$3.13988161e-03$	$-$	0.38s
	2^{-7}	$1.17614357e-03$	$-$	1.51s
	2^{-8}	$4.28042187e-04$	$-$	5.83s
	2^{-9}	$1.53545358e-04$	$-$	23.82s
	2^{-10}	$5.46819955e-05$	$-$	105.67s
	2^{-11}	$1.94029476e-05$	$-$	473.42s
	2^{-12}	$6.87573759e-06$	$-$	2494.09s
CGS	2^{-6}	$3.13988161e-03$	30	0.35s
	2^{-7}	$1.17614357e-03$	40	1.50s
	2^{-8}	$4.28042187e-04$	45	5.69s
	2^{-9}	$1.53545358e-04$	49	22.71s
	2^{-10}	$5.46819955e-05$	51	91.25s
	2^{-11}	$1.94029476e-05$	52	361.35s
	2^{-12}	$6.87573761e-06$	53	1504.99s
FCGS	2^{-6}	$3.13988161e-03$	30	0.07s
	2^{-7}	$1.17614357e-03$	40	0.11s
	2^{-8}	$4.28042187e-04$	45	0.21s
	2^{-9}	$1.53545358e-04$	49	0.47s
	2^{-10}	$5.46819955e-05$	51	1.22s
	2^{-11}	$1.94029478e-05$	52	5.59s
	2^{-12}	$6.87573728e-06$	53	41.39s

Example 4. We set $\gamma = 1/2$, and let N be from 2^6 to 2^{12}. We show the numerical

solution in **Table 4**.

Table 3.4: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 1/2$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPu Time
Gauss	2^{-6}	$1.51193824e-03$	–	0.37s
	2^{-7}	$5.91400260e-04$	–	1.52s
	2^{-8}	$2.22326309e-04$	–	5.71s
	2^{-9}	$8.18392023e-05$	–	23.26s
	2^{-10}	$2.97512041e-05$	–	102.87s
	2^{-11}	$1.07314245e-05$	–	471.01s
	2^{-12}	$3.85395494e-06$	–	2476.10s
CGS	2^{-6}	$1.51193824e-03$	30	0.35s
	2^{-7}	$5.91400260e-04$	45	1.48s
	2^{-8}	$2.22326309e-04$	66	5.58s
	2^{-9}	$8.18392023e-05$	95	22.38s
	2^{-10}	$2.97512040e-05$	130	90.39s
	2^{-11}	$1.07314243e-05$	174	359.51s
	2^{-12}	$3.85395526e-06$	225	1465.47s
FCGS	2^{-6}	$1.51193824e-03$	30	0.06s
	2^{-7}	$5.91400260e-04$	45	0.11s
	2^{-8}	$2.22326309e-04$	66	0.24s
	2^{-9}	$8.18392020e-05$	95	0.65s
	2^{-10}	$2.97512051e-05$	130	1.67s
	2^{-11}	$1.07314248e-05$	174	8.16s
	2^{-12}	$3.85395541e-06$	225	43.56s

Example 5. We set $\gamma = 3/4$, and let N be from 2^6 to 2^{12}. We show the numerical solution in **Table 5**.

Table 3.5: Convergence of the Gaussian elimination, the conjugate gradient squared(CGS) method, and the fast conjugate gradient squared(FCGS) method. $\gamma = 3/4$

	h	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPu Time
Gauss	2^{-6}	$1.00901433e-03$	–	0.40s
	2^{-7}	$3.88438683e-04$	–	1.50s
	2^{-8}	$1.42639480e-04$	–	5.99s
	2^{-9}	$5.13222513e-05$	–	24.00s
	2^{-10}	$1.83050672e-05$	–	102.04s
	2^{-11}	$6.50495908e-06$	–	478.13s
	2^{-12}	$2.31089586e-06$	–	2583.39s
CGS	2^{-6}	$1.00901433e-03$	31	0.39s
	2^{-7}	$3.88438683e-04$	49	1.49s
	2^{-8}	$1.42639480e-04$	75	5.69s
	2^{-9}	$5.13222514e-05$	111	23.32s
	2^{-10}	$1.83050673e-05$	154	93.23s
	2^{-11}	$6.50495937e-06$	222	370.92s
	2^{-12}	$2.31089598e-06$	329	1491.85s
FCGS	2^{-6}	$1.00901433e-03$	29	0.08s
	2^{-7}	$3.88438683e-04$	49	0.14s
	2^{-8}	$1.42639480e-04$	71	0.27s
	2^{-9}	$5.13222518e-05$	111	0.86s
	2^{-10}	$1.83050696e-05$	154	1.91s
	2^{-11}	$6.50496581e-06$	221	8.38s
	2^{-12}	$2.31089500e-06$	328	47.38s

Here we conclude that, for variable-coefficient peridynamic models, we need to consider the stiffness matrix as a sum of several Toeplitz matrices and diagonal matrices. The fast Fourier transforms and matrix-vector multiplication will cost more time for computational works than the constant-coefficient problems. Especially for the problem we considered in this chapter, the diagonal entries need a long time to be evaluated. By implementing our fast method, when the mesh size goes to 2^{12} we only need 42s, the CGS needs 24m17s, and normal Gaussian elimination takes 1h60m36s to get the numerical solution, which has significantly improved computational efficiency without loss of accuracy.

Chapter 4

A Fast Collocation Method for a Two Dimensional Variable-Coefficient Non-local Diffusion Model

4.1 Introduction

In this chapter, we develop a fast numerical method for a two dimensional variable-coefficient non-local diffusion model for describing a heterogeneous finite elastic bar. We shall introduce the two dimensional variable-coefficient non-local diffusion model in section 2, and develop the fast method in section 3. Finally, we will use numerical experiments to show the computational superiority of our fast method[34, 39].

4.2 A variable-coefficient non-local diffusion model and its bi-linear collocation discretization

In this section, we first introduce a two dimensional variable-coefficient non-local diffusion model, and then derive the collocation scheme to search for a numerical solution.

A variable-coefficient non-local diffusion model

The two dimensional variable-coefficient non-local diffusion model can be expressed as

$$\int_{B_\delta(x,y)} (\alpha(x,y) + \alpha(x',y'))\sigma(x-x',y-y')$$

$$(u(x,y) - u(x',y'))dx'dy'$$

$$= f(x,y), \qquad for\ (x,y) \in \Omega \tag{4.1}$$

$$u(x,y) = g(x,y), \qquad for\ \ (x,y) \in \Omega_C.$$

Here we specify that

$$\sigma(x,y) = \frac{1}{|x^2+y^2|^{1+s}}, \tag{4.2}$$

where s shows the singularity of the kernel, and $\alpha(x,y)$ is the elasticity coefficient which has positive lower and upper bounds[47].

Furthermore, we define Ω as a rectangular area, and $\delta > 0$ as the horizon parameter of the material. We again see $f(x,y)$ as the prescribed source term and $u(x,y)$ as the density of the diffusing material.

The neighborhood of the material around the area (x,y) can always be defined as an open area by $|\cdot|_p$ as following

$$B_\delta(x,y) = \{(x',y') \in \Omega \cup \Omega_c : |(x-x',y-y')|_p < \delta\}, \tag{4.3}$$

with

$$\Big|(x,y)\Big|_p = \begin{cases} \Big(|x|^p + |y|^p\Big)^{1/p}, & 1 \le p < +\infty, \\ \max\Big\{|x|,|y|\Big\}, & p = +\infty. \end{cases}$$

In this paper, we consider $B_\delta(x,y)$ as an open disk with the radius δ and center (x,y), which means $p = 2$.

A collocation discretization of the two dimensional variable-coefficient non-local diffusion model

To derive the collocation method, we set the rectangular area to be an open domain, where $\Omega = (0, x_R) \times (0, y_R)$[2]. And marked that $\Delta x = x_R/I$ and $\Delta y = y_R/J$, where I and J are integers which tell the mesh density. Then we shall define a uniform spatial partition $x_i = i\Delta x$ for $i = 0, 1, ..., I$ and $y_j = j\Delta y$ for $j = 0, 1, ..., J$. In order to complete the partition in the whole field $\Omega \cup \Omega_c$, it is necessary to extend (x_i, y_j) for $i = -K, -K+1, ..., -1, 0, 1, ..., I+K$ and $j = -L, -L+1, ..., -1, 0, 1, ..., J+L$ where K and L are the ceilings of $\delta/\Delta x$ and $\delta/\Delta y$. We write them as

$$K = \lceil \delta/\Delta x \rceil, \qquad L = \lceil \delta/\Delta y \rceil.$$

Then we introduce $\psi(\xi) = 1 - |\xi|$, for $\xi \in [-1, 1]$ and 0 otherwise. Now we can define the two dimensional pyramid function $\phi_{i,j}(x, y)$ as

$$\phi_{i,j}(x, y) = \psi(\frac{x - x_i}{\Delta x})\psi(\frac{y - y_i}{\Delta y}), \qquad 0 \leq i \leq I, 0 \leq j \leq J, \tag{4.4}$$

and the trial function u should be

$$u(x, y) = \sum_{i=0}^{I}\sum_{j=0}^{J} u_{i,j}\phi_{i,j}(x, y), \qquad (x, y) \in \Omega. \tag{4.5}$$

Since $u(x_i, y_j) = g(x_i, y_j)$ when $i = 0, I$ and $j = 0, J$, we carry out the governing equation (4.1) at the collocation points (x_i, y_j) for $i = 1, 2, ..., I-1$ and $j = 1, 2, ..., J-1$. And then we write the collocation formulation as follow

$$
\begin{aligned}
&\int_{B_\delta(x_i,y_j)}(\alpha(x_i,y_j)+\alpha(x',y'))\sigma(x_i-x',y_j-y')\\
&(u(x_i,y_j)-u(x',y'))dx'dy'\\
&=f(x_i,y_j)\\
&1\le i\le I-1, 1\le j\le J-1.
\end{aligned}
\tag{4.6}
$$

Next, we substitute $u(x',y')$ by the trial function, so we replace (i,j) by (i',j') and (x,y) by (x',y') in (4.5). The collocation numerical scheme can be written as

$$
\begin{aligned}
&\int_{B_\delta(x_i,y_j)}(\alpha(x_i,y_j)+\alpha(x',y'))\sigma(x_i-x',y_j-y')\\
&(u(x_i,y_j)-\sum_{i'=0}^{I}\sum_{j'=0}^{J}u_{i',j'}\phi_{i',j'}(x',y'))dx'dy'\\
&=f(x_i,y_j)\\
&1\le i\le I-1, 1\le j\le J-1.
\end{aligned}
\tag{4.7}
$$

After manipulations, we can have

$$
\begin{aligned}
&u(x_i,y_j)\int_{B_\delta(x_i,y_j)}(\alpha(x_i,y_j)+\alpha(x',y'))\\
&\sigma(x_i-x',y_j-y')(1-\phi_{i,j}(x',y'))dx'dy'\\
&-\sum_{i'\neq i}\sum_{j'\neq j}u_{i',j'}\int_{B_\delta(x_i,y_j)}(\alpha(x_i,y_j)+\alpha(x',y'))\\
&\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&=f(x_i,y_j)\\
&1\le i\le I-1, 1\le j\le J-1.
\end{aligned}
\tag{4.8}
$$

For $N=(I-1)\times(J-1)$, we define N dimensional vectors

$$u = [u_{1,1}, u_{2,1}, ...u_{I-1,1}, u_{1,2}, u_{2,2}, ..., u_{I-1,2}, ..., u_{1,J-1}, u_{2,J-1}, ..., u_{I-1,J-1}]^T,$$
$$f = [f_{1,1}, f_{2,1}, ...f_{I-1,1}, f_{1,2}, f_{2,2}, ..., f_{I-1,2}, ..., f_{1,J-1}, f_{2,J-1}, ..., f_{I-1,J-1}]^T, \tag{4.9}$$

furthermore, we defined the global indices m and n as

$$m = (j-1)(I-1) + i, \qquad 1 \le i \le I-1, 1 \le j \le J-1,$$
$$n = (j'-1)(I-1) + i', \qquad 1 \le i' \le I-1, 1 \le j' \le J-1. \tag{4.10}$$

Obviously, m and n are related to (i,j) and (i',j'),

then the matrix form of our collocation method can be expressed as

$$Au = f,$$

and the entries of $N \times N$ stiffness matrix $\{A\}_{m,n=1}^N$ should be defined by

$$A_{m,n} = \int_{B_\delta(x_i,y_j)} (\alpha(x_i,y_j) + \alpha(x',y'))$$
$$\sigma(x_i - x', y_j - y')(\delta_{m,n} - \phi_{i',j'}(x',y'))dx'dy', \tag{4.11}$$

$\delta_{m,n}$ is the characteristic function, such that $\delta_{m,n} = 1$ for $m = n$ or 0 otherwise, and the entries of $\{f\}_{m=1}^N$ could be written as

$$f_m = f(x_i,y_j) + \sum_{i''=0,I} \sum_{j''=0,J} \int_{B_\delta(x_i,y_j)} (\alpha(x_i,y_j) + \alpha(x',y'))$$
$$\sigma(x_i - x', y_j - y')g(x_{i''}, y_{j''})\phi_{i'',j''}(x',y'))dx'dy', \tag{4.12}$$

here the indices (i",j") do not only refer to the boundary spatial nodes of Ω, but also the spatial nodes on Ω_c, the supports of whose basis function $\phi_{i'',j''}$ has an non-empty intersection with the neighborhood $B_\delta(x_i,y_j)$ of the node (x_i,y_j).

In consideration of physical relevance, our fast method does not rely on the assumption that the material property δ depends on the computational mesh size. Then the block structure stiffness matrix is almost a full matrix when the mesh size is small enough. Moreover, unlike the constant coefficient problem, the stiffness matrix does not have a clear Toeplitz structure. So we will discuss the structure of the stiffness matrix, and go on to develop our fast method of the variable-coefficient problem.

4.3 Development of the fast collocation method

In this section, we shall start with the discussion about the entries of the stiffness A from previous section, and then decompose the stiffness matrix based on its corresponding structure. Finally, we shall show the fast method[3, 6].

The entries of the stiffness matrix

By previous discussion, we already found the expression of the entries of the stiffness matrix A. We rewrite A by one step manipulation

$$
\begin{aligned}
A_{m,n} = \alpha(x_i, y_j) \int_{B_\delta(x_i,y_j)} & \sigma(x_i - x', y_j - y') \\
& (\delta_{m,n} - \phi_{i',j'}(x', y'))dx'dy' \\
& + \int_{B_\delta(x_i,y_j)} \alpha(x', y')\sigma(x_i - x', y_j - y') \\
& (\delta_{m,n} - \phi_{i',j'}(x', y'))dx'dy'.
\end{aligned}
\tag{4.13}
$$

Then we shall introduce several important theorem in order to get our fast method.

Decomposition of the stiffness matrix

The following theorems could be deduced from before (4.13).

Theorem 4.1. *The stiffness matrix A of the collocation method of previous variable* *coefficients non-local diffusion model can be written as*

$$A = DA^{(1)} + Ad^{(2)} + A^{(2-1)}D^{(1)} + A^{(2-2)}D^{(2)} + A^{(2-3)}D^{(3)} + A^{(2-4)}D^{(4)}$$

, where Ad is a diagonal matrix.

Proof. Let us recall the formula (4.13) we derived before, and simply discuss it as a sum of two terms. For the first term,

$$\alpha(x_i, y_j) \int_{B_\delta(x_i,y_j)} \sigma(x_i - x', y_j - y')(\delta_{m,n} - \phi_{i',j'}(x', y'))dx'dy'$$

let it correspond to $DA^{(1)}$, and then we can have the entries of D and the entries of $A^{(1)}$, where

$$D_{m,m} = \alpha(x_i, y_j), \tag{4.14}$$

which is a diagonal matrix with

$$A^{(1)}_{m,n} = \int_{B_\delta(x_i,y_j)} \sigma(x_i - x', y_j - y') \\ (\delta_{m,n} - \phi_{i',j'}(x', y'))dx'dy'. \tag{4.15}$$

Here m and n are the global indices defined by (4.10).

For the second term, we discuss it in two cases.

The first case, when $m = n$ we have

$$\int_{B_\delta(x_i,y_j)} \alpha(x', y')\sigma(x_i - x', y_j - y')(1 - \phi_{i,j}(x', y'))dx'dy'$$

,

then we let $Ad^{(2)}$ be a diagonal matrix, and all the diagonal entries are defined by the above expression.

For the second case, we consider that $m \neq n$. Before our discussion, we introduce a piecewise-constant approximation $\alpha^I(x, y)$ to approximate $\alpha(x, y)$

$$\begin{aligned}&\alpha^I(x,y) := \textstyle\sum_{i=1}^N \sum_{j=1}^N \alpha(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}) 1_{[x_i,x_{i+1})\times[y_j,y_{j+1})}(x,y),\\ &(x,y) \in [0,x_R]\times[0,y_R].\end{aligned} \tag{4.16}$$

And then we can have

$$\begin{aligned}
&\int_{B_\delta(x_i,y_j)} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&\quad= \int_{B_\delta(x_i,y_j)\cap supp(\phi_{i',j'})} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&\quad= \int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'},y_{j'+1})} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&\qquad+ \int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'},y_{j'+1})} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&\qquad+ \int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'-1},y_{j'})} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'\\
&\qquad+ \int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'-1},y_{j'})} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy',
\end{aligned} \tag{4.17}$$

by substituting α^I for α in the above expression, which is then written as

$$
\begin{aligned}
&\int_{B_\delta(x_i,y_j)} \alpha(x',y')\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy' \\
&= \alpha(x_{i'+\frac{1}{2}},y_{j'+\frac{1}{2}}) \int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'},y_{j'+1})} \\
&\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy' \\
&\quad + \alpha(x_{i'-\frac{1}{2}},y_{j'+\frac{1}{2}}) \int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'},y_{j'+1})} \\
&\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy' \\
&\quad + \alpha(x_{i'-\frac{1}{2}},y_{j'-\frac{1}{2}}) \int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'-1},y_{j'})} \\
&\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy' \\
&\quad + \alpha(x_{i'+\frac{1}{2}},y_{j'-\frac{1}{2}}) \int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'-1},y_{j'})} \\
&\sigma(x_i-x',y_j-y')\phi_{i',j'}(x',y')dx'dy'.
\end{aligned}
\tag{4.18}
$$

Now we shall observe that $D^{(1)}$, $D^{(2)}$, $D^{(3)}$, and $D^{(4)}$ are diagonal matrices, and their diagonal entries can be written as

$$
\begin{aligned}
D^{(1)}_{n,n} &= \alpha(x_{i'+\frac{1}{2}},y_{j'+\frac{1}{2}}) \\
D^{(2)}_{n,n} &= \alpha(x_{i'-\frac{1}{2}},y_{j'+\frac{1}{2}}) \\
D^{(3)}_{n,n} &= \alpha(x_{i'-\frac{1}{2}},y_{j'-\frac{1}{2}}) \\
D^{(4)}_{n,n} &= \alpha(x_{i'+\frac{1}{2}},y_{j'-\frac{1}{2}}),
\end{aligned}
\tag{4.19}
$$

and let the entries of $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, $A^{(2-4)}$ be expressed as following

$$
\begin{aligned}
A_{m,n}^{(2-1)} = &\int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'},y_{j'+1})} \\
&\qquad \sigma(x_i - x', y_j - y')\phi_{i',j'}(x',y')dx'dy' \\
A_{m,n}^{(2-2)} = &\int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'},y_{j'+1})} \\
&\qquad \sigma(x_i - x', y_j - y')\phi_{i',j'}(x',y')dx'dy' \\
A_{m,n}^{(2-3)} = &\int_{B_\delta(x_i,y_j)\cap[x_{i'-1},x_{i'})\times[y_{j'-1},y_{j'})} \\
&\qquad \sigma(x_i - x', y_j - y')\phi_{i',j'}(x',y')dx'dy' \\
A_{m,n}^{(2-4)} = &\int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'-1},y_{j'})} \\
&\qquad \sigma(x_i - x', y_j - y')\phi_{i',j'}(x',y')dx'dy'.
\end{aligned}
\tag{4.20}
$$

In conclusion, the stiffness matrix A can be written as

$$
A = DA^{(1)} + Ad^{(2)} + A^{(2-1)}D^{(1)} + A^{(2-2)}D^{(2)} + A^{(2-3)}D^{(3)} + A^{(2-4)}D^{(4)}
$$

. □

Theorem 4.2. *Let $B_\delta(x_i, y_j)$, K, and L are defined as before in section 1, while the matrices $A^{(1)}$, $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ have block-Toeplitz-Toeplitz-block structures with corresponding blocks, and matrices D, $D^{(1)}$, $D^{(2)}$, $D^{(3)}$, $D^{(4)}$ are all diagonal matrices.*

Proof. By definition in theorem 1, D, $D^{(1)}$, $D^{(2)}$, $D^{(3)}$, $D^{(4)}$ are all diagonal matrices. Moreover, we can easily find that

$$
A^{(1)} = Ad^{(1)} + A^{(2-1)} + A^{(2-2)} + A^{(2-3)} + A^{(2-4)}
$$

, where $Ad^{(1)}$ contains all the diagonal entries of $A^{(1)}$. If we proved that $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ have block-Toeplitz-Toeplitz-block structures and all the diagonal entries of Ad^1 are a constant, we can say $A^{(1)}$ is also a block-Toeplitz-Toeplitz-block matrix.

We are going to prove that $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ have block-Toeplitz-Toeplitz-block structures. Let us recall (4.20), where it was found that $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ have similar structures, then we can just prove that $A^{(2-1)}$ has a block-Toeplitz-Toeplitz-block structure.

For convenience, we denote that $P = A^{(2-1)}$, and

$$P_{m,n} = \int_{B_\delta(x_i,y_j)\cap[x_{i'},x_{i'+1})\times[y_{j'},y_{j'+1})} \sigma(x_i - x', y_j - y')\phi_{i',j'}(x', y')dx'dy', \tag{4.21}$$

it can be observed that when $B_\delta(x_i, y_j) \cap [x_{i'}, x_{i'+1}) \times [y_{j'}, y_{j'+1})$ is not empty set, $P_{m,n} \neq 0$. Then we can investigate that P has $2L$ bands of blocks,

$$P = \begin{pmatrix} Q^{1,1} & Q^{1,2} & \cdots & Q^{1,L} & \cdots & 0 \\ Q^{2,1} & Q^{2,2} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & Q^{J-L,J-1} \\ Q^{L+1,1} & \ddots & \ddots & Q^{L+1,L+1} & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & Q^{J-2,J-1} \\ 0 & \cdots & Q^{J-1,J-L-1} & \cdots & Q^{J-1,J-2} & Q^{J-1,J-1} \end{pmatrix}, \tag{4.22}$$

and each block $P^{j,j'}$ has $2K$ bands of non-zero entries

$$
Q^{j,j'} = \begin{pmatrix}
q^{j,j'}_{1,1} & q^{j,j'}_{1,2} & \cdots & q^{j,j'}_{1,K} & \cdots & 0 \\
q^{j,j'}_{2,1} & q^{j,j'}_{2,2} & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & q^{j,j'}_{I-K,I-1} \\
q^{j,j'}_{K+1,1} & \ddots & \ddots & q^{j,j'}_{K+1,K+1} & \ddots & \vdots \\
\vdots & \ddots & & \ddots & \ddots & q^{j,j'}_{I-2,I-1} \\
0 & \cdots & q^{j,j'}_{I-1,I-K-1} & \cdots & q^{j,j'}_{I-1,I-2} & q^{j,j'}_{I-1,I-1}
\end{pmatrix}. \tag{4.23}
$$

Here we will prove that P has a block-Toeplitz structure first, and the prove that each block of P is a Toeplitz matrix.

In order to prove that P has a block-Toeplitz structure, we need to show that for any j_1, j_1' and j_2, j_2', $Q^{j_1,j_1'} = Q^{j_2,j_2'}$ if $j_1 - j_1' = j_2 - j_2'$. Let P_{m_1,n_1} and P_{m_2,n_2} be the corresponding entries of $Q^{j_1,j_1'}$ and $Q^{j_2,j_2'}$, and then investigate the corresponding global indices

$$
\begin{aligned}
m_1 &= (j_1-1)(I-1)+i, \qquad 1 \le i \le I-1, 1 \le j \le J-1, \\
n_1 &= (j_1'-1)(I-1)+i', \qquad 1 \le i' \le I-1, 1 \le j' \le J-1 \\
m_2 &= (j_2-1)(I-1)+i, \qquad 1 \le i \le I-1, 1 \le j \le J-1, \\
n_2 &= (j_2'-1)(I-1)+i', \qquad 1 \le i' \le I-1, 1 \le j' \le J-1,
\end{aligned} \tag{4.24}
$$

then by the formula (4.21), we have

$$
\begin{aligned}
P_{m_1,n_1} &= q_{i,i'}^{j_1,j_1'} \\
&= \int_{B_\delta(x_i,y_{j_1})\cap[x_{i'},x_{i'+1})\times[y_{j_1'},y_{j_1'+1})} \\
&\qquad \sigma(x_i - x', y_{j_1} - y')\phi_{i',j_1'}(x',y')dx'dy'.
\end{aligned} \tag{4.25}
$$

Here we introduce the following substitution

$$
x' = x_i + x, \qquad y' = y_j + y,
$$

and replace x' and y' in the basis function $\phi_{i',j'}(x',y')$. Through simple transformations, we have

$$
\begin{aligned}
\phi_{i',j'}(x',y') &= \psi(\frac{x' - x_{i'}}{\Delta x})\psi(\frac{y' - y_{j'}}{\Delta y}) \\
&= \psi(\frac{x_i + x - x_{i'}}{\Delta x})\psi(\frac{y_j + y - y_{j'}}{\Delta y}) \\
&= \psi(\frac{x' - x_{i'-i}}{\Delta x})\psi(\frac{y' - y_{j'-j}}{\Delta y}) \\
&= \phi_{i'-i,j'-j}(x',y').
\end{aligned} \tag{4.26}
$$

The formula (4.25) has the following equivalent transformation

$$
\begin{aligned}
P_{m_1,n_1} &= q_{i,i'}^{j_1,j_1'} \\
&= \int_{B_\delta(0,0)\cap[0,\Delta x)\times[0,\Delta y)} \sigma(-x,-y)\phi_{i'-i,j_1'-j_1}(x,y)dxdy \\
&= \int_{B_\delta(0,0)\cap[0,\Delta x)\times[0,\Delta y)} \sigma(-x,-y)\phi_{i'-i,j_2'-j_2}(x,y)dxdy \\
&= P_{m_2,n_2} = q_{i,i'}^{j_2,j_2'},
\end{aligned} \tag{4.27}
$$

then we are going to prove that each block is a Toeplitz matrix. We do the same substitution, and show for any $Q^{j,j'}$, if $i_1 - i'_1 = i_2 - i'_2$, $q^{j,j'}_{i_1,i'_1} = q^{j,j'}_{i_2,i'_2}$ should always be true.

By introducing the global indices

$$
\begin{aligned}
m_3 &= (j-1)(I-1) + i_1, \qquad 1 \le i_1 \le I-1, 1 \le j \le J-1, \\
n_3 &= (j-1)(I-1) + i'_1, \qquad 1 \le i'_1 \le I-1, 1 \le j' \le J-1 \\
m_4 &= (j-1)(I-1) + i_2, \qquad 1 \le i_2 \le I-1, 1 \le j \le J-1, \\
n_4 &= (j'-1)(I-1) + i'_2, \qquad 1 \le i'_2 \le I-1, 1 \le j' \le J-1.
\end{aligned}
\tag{4.28}
$$

We investigate that

$$
\begin{aligned}
P_{m_3,n_3} &= q^{j,j'}_{i_1,i'_1} \\
&= \int_{B_\delta(0,0)\cap[0,\Delta x)\times[0,\Delta y)} \sigma(-x,-y)\phi_{i'_1-i_1,j'-j}(x,y)dxdy \\
&= \int_{B_\delta(0,0)\cap[0,\Delta x)\times[0,\Delta y)} \sigma(-x,-y)\phi_{i'_2-i_2,j'-j}(x,y)dxdy \\
&= P_{m_4,n_4} = q^{j,j'}_{i_2,i'_2}.
\end{aligned}
\tag{4.29}
$$

Now we conclude that $A^{(2-1)}$ has a block-Toeplitz-Toeplitz-block structure. Since the proofs of $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ are almost the same, we will not repeat the proof.

Next, we are going to show that all the diagonal entries of $Ad^{(1)}$ are a constant. By recalling (4.13) and doing the same substitution, the proof is obverse.

$$
\begin{aligned}
Ad^{(2-2)}_{m,m} &= \int_{B_\delta(x_i,y_j)} \sigma(x_i - x', y_j - y') \\
&\quad (1 - \phi_{i',j'}(x', y'))dx'dy' \\
&= \int_{B_\delta(0,0)} \sigma(-x, -y) \\
&\quad (1 - \phi_{0,0}(x, y))dxdy,
\end{aligned}
\tag{4.30}
$$

which is a constant and does not depend on m.

Finally, we conclude that matrices $A^{(1)}$, $A^{(2-1)}$, $A^{(2-2)}$, $A^{(2-3)}$, and $A^{(2-4)}$ have block-Toeplitz-Toeplitz-block structures.

$\square$

Theorem 4.3. *The matrix-vector multiplication Mv is obtained in $O(NlogN)$ operations, if M is an $N \times N$ block-Toeplitz-Toeplitz-block matrix with $(J-1) \times (J-1)$ blocks, each block is a $(I-1) \times (I-1)$ Toeplitz matrix, and v could be any N dimensional vector.*

Proof. Without losing generality, we consider a block-Toeplitz-Toeplitz-block matrix as the following

$$
M = \begin{pmatrix}
M_0 & M_1 & \dots & M_L & \dots & 0 \\
M_{-1} & M_0 & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & M_L \\
M_{-L} & \ddots & \ddots & M_0 & \ddots & \vdots \\
\vdots & \ddots & & \ddots & \ddots & M_1 \\
0 & \dots & M_{-L} & \dots & M_{-1} & M_0
\end{pmatrix},
\tag{4.31}
$$

which has $2L+1$ Toeplitz blocks, and each block has $2K+1$ bands of Toeplitz entries like the following

$$M_i = \begin{pmatrix} m_0^{(i)} & m_1^{(i)} & \dots & m_K^{(i)} & \dots & 0 \\ m_{-1}^{(i)} & m_0^{(i)} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & m_K^{(i)} \\ m_{-K}^{(i)} & \ddots & \ddots & m_0^i & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & m_1^{(i)} \\ 0 & \dots & m_{-K}^{(i)} & \dots & m_{-1}^{(i)} & m_0^{(i)} \end{pmatrix}. \tag{4.32}$$

We then embed each block into an $2(I-1) \times 2(I-1)$ circulant matrix, where

$$C_i = \begin{pmatrix} M_i & \widetilde{M_i} \\ \widetilde{M_i} & M_i \end{pmatrix}, \tag{4.33}$$

and let $\widetilde{M_i}$ be

$$\widetilde{M_i} = \begin{pmatrix} 0 & \dots & 0 & m_{-K}^{(i)} & \dots & m_{-1}^{(i)} \\ \vdots & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & m_{-K}^{(i)} \\ m_K^{(i)} & \ddots & \ddots & 0 & \ddots & 0 \\ \vdots & \ddots & & \ddots & \ddots & \vdots \\ m_1^{(i)} & \dots & m_K^{(i)} & 0 & \dots & 0 \end{pmatrix}. \tag{4.34}$$

Now we have a block-Toeplitz-circulant-block matrix C

$$C = \begin{pmatrix} C_0 & C_1 & \dots & C_L & \dots & 0 \\ C_{-1} & C_0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & C_L \\ C_{-L} & \ddots & \ddots & C_0 & \ddots & \vdots \\ \vdots & \ddots & & \ddots & \ddots & C_1 \\ 0 & \dots & C_{-L} & \dots & C_{-1} & C_0 \end{pmatrix}. \tag{4.35}$$

Finally, the block-Toeplitz-circulant-block matrix C should be embedded into a block-circulant-circulant-block matrix B, where

$$B = \begin{pmatrix} C & \widetilde{C} \\ \widetilde{C} & C \end{pmatrix}, \tag{4.36}$$

and $\widetilde{C}$ is defined as

$$\widetilde{C} = \begin{pmatrix} 0 & \dots & 0 & C_{-L} & \dots & C_{-1} \\ \vdots & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & C_{-L} \\ C_L & \ddots & \ddots & 0 & \ddots & 0 \\ \vdots & \ddots & & \ddots & \ddots & \vdots \\ C_1 & \dots & C_L & 0 & \dots & 0 \end{pmatrix}. \tag{4.37}$$

Here the first column b should have all information of the matrix B. Let $F_{2(J-1)} \otimes F_{2(I-1)}$ be the two dimensional discrete Fourier transform matrix, the the Fourier transform $\hat{b}$ of c could be written as $(F_{2(J-1)} \otimes F_{2(I-1)})b$. Since the block-circulant-circulant-block matrix B has the decomposition

$$(F_{2(J-1)} \otimes F_{2(I-1)})^{-1} diag(\hat{b})(F_{2(J-1)} \otimes F_{2(I-1)}),$$

we then work on the N dimensional vector v[8, 11, 21]. Because $N = (J-1)(I-1)$, we write $v = [v_{(1)}, v_{(2)}, ..., v_{(J-1)}]^T$, where $v_{(j)} = [v_{1,j}, v_{2,j}, ..., v_{I-1,j}]$, for $1 \leq j \leq J-1$. Our goal is to expand v to an $4N$ dimensional vector, so we first expand it to a $2N$ dimensional vector by setting $v^{(2N)} = [v_{(1)}, \vec{0}, v_{(2)}, \vec{0}, ..., v_{(J-1)}, \vec{0}]^T$, where $\vec{0}$ is an $I-1$ dimensional 0 vector. Then we have $v^{(4N)} = [v^{(2N)}, 0]^T$, where 0 is a $2N$ dimensional zero vector.

Since $(F_{2(J-1)} \otimes F_{2(I-1)})v^{(4N)}$ and the Hadamard product of $\hat{c}$ and any $4N$ dimensional vector can be carried out in $O(NlogN)$ operations. Mv can be carried out in $O(NlogN)$ operations, which is a part of $Bv^{(4N)}$.

□

Since the stiffness matrix A is no longer symmetric, we prefer a conjugate gradient squared method to search for numerical solutions. Based on the above theorem, we can consider the matrix-vector multiplication Ad_k as

$$(DA^{(1)} + Ad^{(2)} + A^{(2-1)}D^{(1)} + A^{(2-2)}D^{(2)} + A^{(2-3)}D^{(3)} + A^{(2-4)}D^{(4)})d_k,$$

which only involves Hadamard production and block-Toeplitz-circulant-block matrix-vector production, so the computational cost will be reduced to $O(NlogN)$.

4.4 Numerical experiment

Since for the two dimensional variable-coefficient non-local diffusion model, we need to use numerical integrals to generate five Toeplitz matrices, one diagonal matrix, and the right-hand side terms, the shall use one small mesh size with $N = 2^5$ to show the performance of our fast method. And based on previous study, we conclude that the advantages of our fast method are more obvious at larger mesh size.

Let the spatial domain be $(0,1) \times (0,1)$ and $\delta = 1/8$, and the kernel function $\sigma(x,y)$ be expressed as

$$\sigma(x,y) = \frac{1}{(x^2+y^2)^{1+s}}, \tag{4.38}$$

here we choose $u(x,y) = x(1-x)y(1-y)$ as the true solution of the problem (4.1), and use it to define the value of u on the boundary zone Ω_c, which means $u(x,y) = x(1-x)y(1-y)$ on the whole field $\Omega \cup \Omega_c$. The variable coefficient is defined as $\alpha(x,y) = 1 + 16\epsilon(x-\frac{1}{2})^2(y-\frac{1}{2})^2$, where ϵ is a small constant. Without loss of generality, we chose the same grid size in both the x and y directions i.e., $\Delta x = \Delta y = h$. And the right-hand side term $f(x,y)$ at each collocation point can be computed by numerical integration.

Then we implement the Matlab codes of standard Gaussian elimination (Gauss), conjugate gradient squared(CGS) method, and fast conjugate gradient squared (FCGS) method in a 16GB-ROM laptop.

Table 4.1: Gaussian elimination, conjugate gradient squared(CGS) method, and fast conjugate gradient squared(FCGS) method. $\delta = 1/8$

	s	$\|\|e^h\|\|_{L_2}$	# of Iter.	CPU Time
Gauss	0	$7.75185824e-03$	–	28.97s
	1/4	$1.26690474e-02$	–	29.18s
	3/8	$1.83823456e-02$	–	37.81s
CGS	0	$7.75185824e-03$	14	20.02s
	1/4	$1.26690474e-02$	28	21.22s
	3/8	$1.83823456e-02$	36	21.58s
FCGS	0	$7.75185824e-03$	26	0.46s
	1/4	$1.26690474e-02$	56	0.87s
	3/8	$1.83823456e-02$	72	1.08s

In conclusion, we found that the standard Gaussian elimination (Gauss), conjugate gradient squared(CGS) method, and fast conjugate gradient squared (FCGS) method have the same error, but the fast conjugate gradient squared method only needs 1/40 of times of conjugate gradients quared method in searching the numerical solution of the problem (4.1) when $N = 2^5$, which becomes much more significant for a larger grid size. Also, the number of iteration depends on the kernel function $\frac{1}{|x^2+y^2|^{1+s}}$[16, 15]. When $s < 0$, we have a better convergence rate and less iteration steps, and when s is close to 1, the singularity of the kernel function will cause more iteration steps.

Bibliography

[1] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*. San Diego: ELsevier, 2003.

[2] B. Aksoylu and M. L. Parks. “Variational theory and domain decomposition for nonlocal problems”. In: *Appl. Math. Comp.* 217 (2011), pp. 6498–6515.

[3] G. Ammar and P. Gader. “A variant of the Gohberg-Semencul formula involving circulant matrices”. In: *SIAM J Matrix Anal* A 12 (1991), pp. 534–540.

[4] G.S. Ammar and W.B. Gragg. “Superfast solution of real positive definite Toeplitz systems”. In: *SIAM Journal on Matrix Anal. Appl.* 9 (1988), pp. 61–67.

[5] F. Andreu et al. “A nonlocal p-Laplacian evolution equation with Neumann boundary conditions”. In: *J. Math. Pures Appl.* 90 (2008), pp. 201–227.

[6] R. Barrett et al. “Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods”. In: *SIAM, Philadelphia* PA (1994).

[7] F. Bobaru et al. “Convergence, adaptive refinement, and scaling in 1D peridynamics”. In: *Int J. Numer. Methods Engrg.* 77 (2009), pp. 852–857.

[8] A. Böttcher and B. Silbermann. *Introduction to Large Truncated Toeplitz Matrices*. New York: Springer, 1999.

[9] R.H. Chan and M.K. Ng. “Conjugate Gradient Methods for Toeplitz Systems”. In: *SIAM Review* 38 (1996), pp. 427–482.

[10] X. Chen and M. Gunzburger. “Continuous and discontinuous finite element methods Continuous and discontinuous finite element methods for a peridynamics model of mechanics”. In: *Comput. Methods Appl. Mech. Engrg.* 200 (2011), pp. 1237–1250.

[11] P. J. Davis. *Circulant Matrices*. New York: Wiley-Intersciences, 1979.

[12] D. del-Castillo-Negrete, B.A. Carreras, and V. E. Lynch. “Fractional diffusion in plasma turbulence”. In: *Phys. Plasmas* 11 (2004), p. 3854.

[13] Ning Du, Hong Wang, and Che Wang. “A fast method for a generalized nonlocal elastic model”. In: *J. Comput. Phys.* 297 (2015), pp. 72–83.

[14] Q Du and K Zhou. “Mathematical analysis for the peridyanmic nonlocal continuum theory”. In: *ESIAM: M2AN Math. Mod. Numer. Anal.* 45 (2011), pp. 217–234.

[15] Q. Du et al. “A posteriori error analysis of finite element method for linear nonlocal diffusion and peridynamic models”. In: *Mathematics of Computation* 82 (2013), pp. 1889–1922.

[16] Q. Du et al. “Analysis and approximation of nonlocal diffusion problems with volume constraints”. In: *SIAM Review* 54 (2012), pp. 667–696.

[17] E. Emmrich and O.Weckner. “Analysis and numerical approximation of an intergrodifferential equation modeling non-local effects in linear elasticity”. In: *Math. Meth. Solids* 12 (2007), pp. 363–384.

[18] E. Emmrich and O. Weckner. “The peridynamic equation and its spatial discretization”. In: *Math. Model. Anal.* 12 (2007), pp. 17–27.

[19] V.J. Ervin and J.P. Roop. “Variational Solution of Fractional Advection Dispersion Equations on Bounded Domains in R^d”. In: *Numer. Methods for Partial Differential Equations* 23 (2007), pp. 256–281.

[20] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins university, 1996.

[21] R. M. Gray. “Toeplitz and Circulant Matrices: A Review”. In: *Foundations and Trends in Comm 2* (2006), pp. 155–239.

[22] M. Gunzburger and R Lehoucq. “A nonlocal vector calculus with application to nonlocal boundary value problems”. In: *Multiscale Model. Simul.* 8 (2010), pp. 1581–1598.

[23] Y. D. Ha and F. Bobaru. “Studies of dynamic crack propagation and crack branching”. In: *Int. J. Fract.* 162 (2010), pp. 229–244.

[24] G.C. Hsiao, P. Kopp, and W. Wendland. “A Galerkin-collocation method for some integral equations of the first kind”. In: *Computing* 25 (1980), pp. 89–130.

[25] G.C. Hsiao and W. Wendland. “A finite element method for an integral equation of the first kind”. In: *J. Math. Anal. Appl.* 58 (1977), pp. 449–481.

[26] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. Philadelphia: SIAM, 1995.

[27] B. Kilic and E. Madenci. "An adaptive dynamic relaxation method for quasi-static simulations using the peridynamic theory". In: *Theoretical and Applied Fracture Mechanics* 53 (2010), pp. 194–204.

[28] C. Li and H. Ding. "Higher order finite difference method for the reaction and anomalous-diffusion equation". In: *Appl. Math. Modell.* (2014).

[29] R. Lin et al. "Stability and convergence of a new explicit finite-difference approximation for the variable-order nonlinear fractional diffusion equation". In: *Appl. Math. Comp.* 212 (2009), pp. 435–445.

[30] F. Liu, V. Anh, and I. Turner. "Numerical solution of the space fractional Fokker-Planck equation". In: *J. Comput. Appl. Math.* 166 (2004), pp. 209–219.

[31] R. Maceka and S. Silling. "Peridynamics via finite element analysis". In: *Finite Element in Analysis and Design* 183 (2006), pp. 365–372.

[32] M. M. Meerschaert and C. Tadjeran. "Finite difference approximations for fractional advection-dispersion flow equations". In: *J. Comput. Appl. Math.* 172 (2004), pp. 65–77.

[33] M. M. Meerschaert and C. Tadjeran. "Finite difference approximations for two-sided space-fractional Finite difference approximations for two-sided space-fractional partial differential equations". In: *Applied Numer. Math.* 56 (2006), pp. 80–90.

[34] T. Mengesha and Q. Du. "Nonlocal constrained value problems for a linear peridynamic Navier equation". In: *J. Elast.* 116 (2014), pp. 27–51.

[35] R. Metzler and J. Klafter. "The random walk's guide to anomalous diffusion: A fractional dynamics approach". In: *Phys. Rep.* 339 (2000), pp. 1–77.

[36] J.T. Oden and J.N. Reddy. *An Introduction to the Mathematical Theory of Finite Elements*. New York: John Wiley & Sons, 1976.

[37] H. K. Pang and H. W. Sun. "Multigrid method for fractional diffusion equations". In: *J. Comput. Phys.* 231 (2012), pp. 693–703.

[38] Y. Saad. *Iterative methods for sparse linear systems, 2nd ed.* Rhode Island: SIAM, 2003.

[39] P. Seleson, M. Gunzburger, and M. L. Parks. “Interface problems in nonlocal diffusion and sharp transitions between local and nonlocal domains”. In: *Comput. Methods Appl. Mech. Engrg.* 266 (2013), pp. 185–204.

[40] S. Silling. “Linearized theory of peridynamic states”. In: *J. Elast.* 99 (2010), pp. 85–111.

[41] S. Silling. “Reformulation of elasticity theory for discontinuities and long-range forces”. In: *N.Mech.Phys. Solids* 48 (2000), pp. 175–209.

[42] S. Silling and E. Askari. “A meshfree method based on the peridynamic model of solid mechanics”. In: *Computers and Structures* 83 (2005), pp. 1584–1611.

[43] S. Silling and R. Lehoucq. “Peridynamic theory of solid mechanics”. In: *Adv. Appl. Mech.* 44 (2010), pp. 73–168.

[44] S. Silling, M. Zimmermann, and R. Abeyaratne. “Deformation of a peridynamic bar”. In: *J. Elast.* 73 (2003), pp. 173–190.

[45] S. Silling et al. “Peridynamic states and constitutive modeling”. In: *J. Elast.* 88 (2007), pp. 151–184.

[46] H. Tian, H. Wang, and W. Wang. “An efficient collocation method for a nonlocal diffusion model”. In: *Int'l J. Numer. Anal. Modeling* 10 (2013), pp. 815–825.

[47] M.E.A. El Tom. “On the numerical stability of spline function approximations to solutions of Volterra integral equations of the second kind”. In: *BIT* 14 (1974), pp. 136–143.

[48] H. Wang and T.S. Basu. “A fast finite difference method for two-dimensional space-fractional diffusion equations”. In: *SIAM J. Sci. Comput.* 34 (2012), A2444–A2458.

[49] H. Wang and H. Tian. “A fast and faithful collocation method with efficient matrix assembly for a two dimensional nonlocal diffusion model”. In: *Comput. Methods Appl. Mech. Engrg.* 273C (2014), pp. 19–36.

[50] H. Wang and H. Tian. “A fast Galerkin method with efficient matrix assembly and storage for a peridynamic model”. In: *J. Comput. Phys.* 231 (2012), pp. 7730–7738.

[51] H. Wang and K. Wang. “An $O(N \log^2 N)$ alternating-direction finite difference method for two-dimensional fractional diffusion equations”. In: *J. Comput. Phys.* 230 (2011), pp. 7830–7839.

[52] H. Wang, K. Wang, and T. Sircar. “A direct $O(N \log^2 N)$ finite difference method for fractional diffusion equations”. In: *J. Comput. Phys.* 229 (2010), pp. 8095–8104.

[53] K. Wang and H. Wang. “A fast characteristic finite difference method for fractional advection-diffusion equations”. In: *Advances in Water Resources* 34 (2011), pp. 810–816.

[54] O. Weckner and E. Emmrich. “Numerical simulation of the dynamics of a nonlocal, inhomogeneous, infinite bar”. In: *J. Comput. Appl. Mech.* 6 (2005), pp. 311–319.

[55] O. Weckner and E. Emmrich. “The effect of long-range forces on the dynamics of a bar”. In: *J. Mech. Phys. Solids.* 53 (2005), pp. 705–728.

[56] K. Zhou and Q. Du. “Mathematical and numerical analysis of linear peridynamic models with nonlocal boundary condition”. In: *SIAM N.Numer Anal.* 48 (2010), pp. 1759–1780.