



UNIVERSITÄT HAMBURG



UNIVERSITÀ DEGLI STUDI  
DELL'AQUILA



UNIVERSITAT AUTÒNOMA DE  
BARCELONA

**Large Erasmus Mundus Consortium "MathMods"**

Joint Degree of Master of Science in  
Mathematical Modelling in Engineering: Theory, Numerics, Applications

In the framework of the  
Consortium Agreement and Award of a Joint/Multiple Degree 2013-2019

**Master's Thesis**

**A Non-local Particle Approximation of the Porous  
Medium Equation**

*Supervisor:*  
Prof. Marco DI FRANCESCO

*Candidate:*  
Aida ABDALLAH  
*Matricola:* 238809

2015/2016

Laurea Magistrale in Ingegneria Matematica  
Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica  
Università degli Studi dell'Aquila



## *Declaration of Authorship*

I, Aida ABDALLAH, declare that this thesis titled, A non-local approximation of the porous medium equation and the work presented in it are my own. I confirm that:

- This work was done wholly while in candidature for a masters' degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given.
- I have acknowledged all main sources of help.

This thesis was not previously presented to another examination board so has not been published.

Signature:

---

Date:

---



# *Acknowledgement*

*I give all praises and thanks to God for a successful completion of this work.*

*“ My utmost gratitude goes to my supervisor (Prof. Marco Di francesco), whom aside the valuable advisory role played in this thesis, has greatly improved my understanding and desire for studying mathematical analysis through his rigorousness and profound skill. A special thanks also goes to Simone Fagioli, whose contribution to this work is exceptionally profound. Thanks to him, I can now boast of some useful skills in Matlab. I would also like to thank the Mathmods consortium including the coordinator of the program, Prof. Bruno Rubino, the administrative staff and the lecturers at the University of L'Aquila and the University of Hamburg for giving me such a great learning opportunity and experience.*

*“A special thank you also goes out to my family for their supportive role. Especially to my dad, Hon. Abdallah Shuaib and my husband, Mr. Yahaya Halidu. Although miles and oceans away, they were always there to support me in diverse ways and urge me on even when my self motivation was not enough to see me through this program. Baaba, naa goodey; MD, you are simply the best! Allah ya barr mu taarey. I would also like to thank my colleagues and friends, particularly, Plamen Dimitrov and Fonyuy Theophile Fondzefe for their great help and contributions to my studies throughout this program.”*



# Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
1.1	Motivation . . . . .	7
1.2	Summary . . . . .	7
1.3	Organization . . . . .	9
<b>2</b>	<b>Mathematical modelling of biological dispersal and animal swarming.</b>	<b>10</b>
2.1	An overview of biological dispersal. . . . .	10
2.1.1	Quantifying Dispersion . . . . .	12
2.2	Mathematical Models For Animal Dispersal. . . . .	12
2.2.1	The Continuity Equation . . . . .	13
2.2.2	The Diffusion Equations . . . . .	16
2.3	Porous Medium Equation (PME) . . . . .	17
2.4	Swarming Models: Non-local Interaction . . . . .	19
2.4.1	General Approach to Interacting Individuals . . . . .	20
2.4.2	Non-Local Porous Medium Equation (NLPME) . . . . .	22
<b>3</b>	<b>Numerical Analysis 1: Simulating the PME and NLPME</b>	<b>26</b>
3.1	Simulating the PME with the Finite-Difference Method . . . . .	26
3.1.1	Procedure with the Barenblatt Initial Density . . . . .	27
3.1.2	Using the Constant Initial density . . . . .	34
3.1.3	Using a Triangular Initial Density . . . . .	38
3.2	Simulating the NLPME with the Particle Method . . . . .	42

3.2.1	Procedure Using a Constant Distribution of Initial Density.	42
3.2.2	Simulating with the Gaussian Potential . . . . .	46
3.2.3	Simulating with the Morse Potential . . . . .	52
3.2.4	Procedure Using a Triangular Distribution of the Initial Density . . . . .	58
3.2.5	Simulating with the Gaussian Potential . . . . .	59
3.2.6	Simulating with the Morse Potential . . . . .	66
3.2.7	A comparison of the PME and the NLPME. . . . .	71
<b>4</b>	<b>Numerical Analysis 2: <math>L_1</math> Error Analysis</b>	<b>78</b>
4.1	A variation of the $L_1$ error with $\beta$ . . . . .	78
4.1.1	$L_1$ Error Analysis with the both potentials and the con- stant initial density . . . . .	79
4.2	A variation of the $L_1$ error with $N$ . . . . .	83
<b>5</b>	<b>Conclusion</b>	<b>87</b>
	Bibliography . . . . .	90



# Chapter 1

## Introduction

### 1.1 Motivation

The motivation for this thesis comes from the desire to sharpen my computational and analytical skills while studying how mathematics has been used to model a real world phenomenon like biological dispersion. The choice of a biological subject is solely due to my educational background and interest in life science.

### 1.2 Summary

Indeed, there are mathematical models that reasonably describe the dispersion of organisms such as animals. Non-linear Partial Differential Equations (PDEs) such as the Porous Medium Equation (PME) and Non-local Porous Medium Equation (NLPME) are examples of such mathematical models. Among the benefits of studying these models is that, they provide an important framework for modelling more complicated biological phenomena like chemotaxis and organisational behaviours like the aggregation of biological organisms.

Unlike the PME where the forces that lead to dispersion are assumed to be local, for the NLPME, the forces are rather non-local. Our reference to the term "non-local forces" is in the sense that, interactions exist between each organism

(in the total number under consideration) and other ones of the same kind that are within its neighbourhood.

In this thesis, the objective is to verify the hypothesis that, for infinitely many organisms, the non-local forces become local and consequently, the NLPME coincides with the PME. Intuitively, one may see that the non-local model describes the phenomenon more closely than the local one since it regards the influence of other organisms on the resultant force that leads to the dispersal of a single organism.

Notwithstanding that, as indicated in a number of literature on the subject, one advantage of the local model over the non-local one is that of ease in analysis [10]. Holding on to that advantage, an immediate benefit of validating this hypothesis is that, if a very large number of organisms is involved, then it is reasonable to analyse dispersal models with the local model instead of the more complicated non-local model .

To substantiate this hypothesis, we implore the following methodology. First, we use two finite difference approaches, namely, the Explicit Euler and Runge Kutta to obtain a numerical scheme for the PME. Afterwards, we solve the resulting linear system and obtain the approximated solution to the PME.

Secondly, we use the so called particle method to reconstruct the solution to the NLPME. By this method, a fundamental step is to consider the initial locations of all the organisms such that, with their masses given, the distribution of their initial density can be constructed. In both simulations, we consider a constant as well as a triangular distribution of the organisms' initial density.

Finally, we evaluate the  $L_1$  error between the two approximated solutions. Ultimately, we analyse its behaviour as the range of the dispersion forces varies and particularly, as the number of organisms increases. As a matter of choice, we do all simulations and numerical computations in a Matlab environment.

## 1.3 Organization

This thesis is divided into five chapters. In chapter 2, we present an outline for the biological description of animal dispersion including its causes and benefits. We also take a look at how animal dispersion is quantified and present some mathematical models that reasonably describe the evolution of a dispersing population.

In chapter 3, after outlining the preliminaries for a numerical analysis, we begin an elaborate computational exercise of simulating both the local and non-local models. In addition, we illustrate the simulated solutions with graphs for certain values of the "scaling coefficient", a parameter which regulates the range of the forces that lead to animal dispersal.

The main result of this thesis is presented in chapter 4, where we compute the  $L_1$  error between the two solutions: the solution to the local and the non-local models. Furthermore, we study and compare the behaviour of the error with respect to two types of kernels that model the dispersive forces; and also with respect to the "scaling coefficient" and the total number of animals under consideration.

Finally, in chapter 5, we conclude with an evaluation of the computational procedure and the dispersal kernels used, in order to determine which of them is most suitable for our analysis. Most importantly, we evaluate the hypothesis for this thesis.

# Chapter 2

## Mathematical modelling of biological dispersal and animal swarming.

### 2.1 An overview of biological dispersal.

1

In Biology, animal dispersal is defined as the scattering of animals over periods within a given area. It has also been defined as the movement of individuals of animals from their birth site to their breeding site and from one breeding site to another. In general, dispersal occurs when the fitness benefits of moving outweigh the costs.

Its benefits include: locating new resources, escaping unfavorable conditions, avoiding competing with siblings, and avoiding breeding with closely related individuals which could lead to inbreeding depression. [1]

There are also some notable costs associated with dispersal. These costs can be thought of in terms of four main currencies: energy, risk, time and

<sup>1</sup><http://www.britannica.com/science/dispersion-biology>

opportunity.[3] Energetic costs include the extra energy required to move as well as energetic investment in movement machinery (e.g. wings). Risks include increased injury and mortality during dispersal and the possibility of settling in an unfavorable environment.

Also, time spent dispersing is time that often cannot be spent on other activities such as growth and reproduction. Finally, dispersal can also lead to outbreeding depression if an individual is better adapted to its natal (birth) environment than the one it ends up in. In social animals (such as many birds and mammals) a dispersing individual must find and join a new group, which can lead to loss of social rank. [3]

In general, there are two basic types of biological dispersal: Density independent (passive) dispersal, where organisms have evolved adaptations for a dispersal that is caused by various forms of kinetic energy occurring naturally in the environment; [8, 9] and Density dependent (active) dispersal where dispersal is largely as a result of factors such as local population size, resource competition, habitat size and quality. [2, 1, 4]

The forces that govern animal dispersal are classified under vectorial (deterministic or directed motion), which is caused by wind, water, food sources, overcrowding; or stochastic (random) as in case of changes in weather, which give no indication of where the organisms may ultimately settle.

Moreover, animals can establish one of three possible patterns of dispersion: a random pattern; and aggregated pattern, in which organisms gather in clumps; or a uniform pattern in which organisms are arranged in such a way that the spaces between them are roughly equal. The type of pattern formed by animals is often determined by the nature of the relationships within the population. For example, while territorial animals like birds tend to assume a uniform pattern, social animals like chimpanzees tend to gather in groups, an aggregate pattern.

In order to obtain an accurate reading of the pattern formed, it is important to be mindful of the scale of study. For instance, if we imagine a group of monkeys occupying three widely separated trees, after observation, our inference would obviously be that they have formed an aggregated pattern while if the focus is on only one tree, then their arrangement may appear to be rather uniform.

### 2.1.1 Quantifying Dispersion

Arguably, the most intuitive quantitative definition of dispersion is that, it is the degree of the variation of a variable about a central value. Most commonly, it is quantified either in terms of rate or distance.

Dispersal rate describes the probability that any individual leaves an area in a given time. On the other hand, dispersal distance is usually described by a dispersal kernel which gives the probability distribution of the distance travelled by any individual. Some of the distribution functions used as dispersal kernels or potentials in theoretical models for dispersion include: the negative exponential distribution, [13] the extended negative exponential distribution [7] and the normal distribution (Gaussian curve) [13].

## 2.2 Mathematical Models For Animal Dispersal.

There is a significant number of mathematical models that reasonably describe the behaviour of a dispersing population. In a book edited by Tilman and Kareiva (1998) [6], it has been demonstrated that, the spatial structure of a habitat can fundamentally alter both the qualitative and quantitative dynamics and outcomes of ecological processes such as dispersion. This implies that, when developing ecological models, although it is reasonable to assume in some cases that population is a homogeneous quantity which only depends on time, in many other situations, the dependence on spatial variables is relevant. Therefore, it is typical that in dispersal models, population is considered as a function of space and time.

In an assemblage of individuals (or particles) of animals (or cells, bacteria,

chemicals), each individual usually moves (actively or passively) in a random way. They spread out and settle in undeterminable locations as a result of some random forces. When this individual (or microscopic) irregular movements result in some gross (or macroscopic) regular motion of the group, we may think of it as a diffusion process.

### 2.2.1 The Continuity Equation

Indeed, it is a laborious exercise to obtain a macroscopic behaviour from a knowledge of an individual microscopic behaviour. Authors like Masaaki Inoue and Morale et. al. have rigorously obtained it in various forms. [5, 10] Here, we follow the derivation of a continuum model for the global behaviour in terms of a quantity called **population density function** or **concentration**. Unless otherwise stated, most of the material in the remaining parts of this chapter is based on the lecture notes of Mathematical Biology by Prof. Marco Di Francesco.

The diffusion mechanism models the movement of many individuals of the same kind in an environment. The sizes of the individuals may be very small such as: basic particles in physics and chemistry (like atoms and molecules) and microscopic particles in biology (like bacteria and cells); or very large objects such as: animals (like ants) and plants; or certain events like epidemics and tumours.

We assume an ensemble of particles residing in a region (habitat), which we denote as  $\Omega$ , with  $\Omega$  being an open subset of  $\mathbb{R}^n$  for  $n \geq 1$ . In particular, we are interested in the case of  $n = 1$  although the formulation here holds regardless of the spatial dimension. We also introduce the main mathematical variable for our analysis: the population density function of the particles or simply the population density, denoted by  $\rho(t, x)$ , where  $t$  is the time and  $x \in \Omega$  is the location of the particles. Usually, the dimension of population density is expressed as: the number of individuals per unit length (if  $n = 1$ ), per unit area (if  $n = 2$ ) or unit volume (if  $n = 3$ ). For example, the human population density is often expressed in number of people per square kilometer.

As in many other mathematical models, we also assume that  $\rho(t, x)$  has nice properties like continuity and differentiability, which is in fact reasonable when a population with a large number of organisms (or particles) is considered. More precisely, we assume that,  $\rho \in C^1([0, +\infty); L^1(\Omega))$ .

Technically, we define the population density function,  $\rho(t, x)$  as follows. Let  $x$  be a point in the habitat  $\Omega$ , and let  $\{O_n\}_{n=1}^{\infty}$  be a sequence of spatial regions (which have the same dimension as  $\Omega$ ) surrounding  $x$ ; Here,  $O_n$  is chosen in a way such that, the spatial measurement  $|O_n|$  of  $O_n$  (length, area, volume, or in a more general sense, the Lebesgue measure) tends to zero as  $n \rightarrow +\infty$ , and  $O_n \subset O_{n+1} \quad \forall n$ ; then

$$\rho(t, x) = \lim_{n \rightarrow +\infty} \frac{\text{Number of organisms in } O_n \text{ at time } t}{|O_n|},$$

if the limit exists. Realistically, as long as the scale of data collecting is small enough, the population density function is usually very well defined. The total population in any sub-region,  $O$  of  $\Omega$  at time  $t$  is

$$\int_O \rho(t, x) dx. \quad (2.1)$$

Our interest is in knowing how  $\rho(t, x)$  changes as time  $t$  evolves and as the location  $x$  changes. Basically, the events that cause changes in populations can be grouped into two: movement related causes (migration and emmigration); and non-movement related causes (birth and death).

For the movement related causes, we focus on diffusion, which is the most classical natural phenomenon that contributes to the movement of particles with a heterogeneous distribution. During diffusion, particles move from a region of high population density to a region of low population density. Perhaps, a more intuitive way of thinking about the movement of the particles in the above context is, as a directed flow of a fluid (the population density). The flow of  $\rho(t, x)$  is known as the **flux** of the population density, which is a vector quantity that we denote by  $\vec{J}(t, x)$ . On the other hand, we denote the rate of change of the population density with respect to death and birth by  $f(t, x, \rho)$ ,



which is known as the reaction rate.

From the definition of total population in the region  $O$  given in 2.1, the rate of change of the total population is  $\frac{d}{dt} \int_O \rho(t, x) dx$ . Additionally, the net growth of the population inside the region  $O$  is  $\int_O f(t, x, \rho(t, x)) dx$  and the total outward flux is  $-\int_{\partial O} \vec{J}(t, x) \cdot \vec{n}(x) dS$ , where  $\partial O$  is the boundary of  $O$ , and  $n(x)$  is the outward normal vector at a location  $x$ . Now, using the mass balance law, we derive a differential equation in terms of  $\rho(t, x)$  as

$$\frac{d}{dt} \int_O \rho(t, x) dx = - \int_{\partial O} \vec{J}(t, x) \cdot \vec{n}(x) dS + \int_O f(t, x, \rho(t, x)) dx. \quad (2.2)$$

Since diffusion is pure transport phenomena, it has no reaction. If we further assume that there is no reaction rate, that is  $f = 0$ , then the balance equation becomes

$$\frac{d}{dt} \int_O \rho(t, x) dx = - \int_{\partial O} \vec{J}(t, x) \cdot \vec{n}(x) dS \quad (2.3)$$

From the Fundamental Theorem of Calculus in the one dimensional case or the Divergence Theorem in multi-variable calculus, assuming that the flux  $\vec{J}(t, x)$  is  $C^1$  with respect to  $x$ , we have

$$\int_{\partial O} \vec{J}(t, x) \cdot \vec{n}(x) dS = \int_O \operatorname{div} \vec{J}(t, x) dx. \quad (2.4)$$

Combining (2.3) and (2.4) and interchanging the order of differentiation and integration (which is possible since  $\rho(t, x) \in C^1([0, +\infty); L^1(O))$ ), we obtain

$$\int_O \frac{\partial}{\partial t} \rho(t, x) dx + \operatorname{div} \vec{J}(t, x) dx = 0.$$

Since the choice of the region  $O$  is arbitrary, we can apply the variational Lemma to finally obtain the partial differential equation (PDE)

$$\frac{\partial}{\partial t} \rho(t, x) + \operatorname{div} \vec{J}(t, x) = 0, \quad (2.5)$$

which holds for any  $(t, x) \in \Omega$ . This equation is called the **Continuity Equation**.

### 2.2.2 The Diffusion Equations

With the continuity equation in hand, we are now ready to obtain the so called diffusion equations. We recall that in section 2.1, it was mentioned that dispersal (in this context, the flow of the population density) can be grouped into two forms: density dependent and density independent dispersal. This implies that, what characterises the transport phenomena (diffusion) is the flux  $\vec{J}(t, x)$ .

One model for the flux is given by a principle called the **Fick's law**, which states that:

$$\vec{J}(t, x) = -d(x)\nabla_x\rho(t, x), \quad (2.6)$$

where  $d(x)$  is called diffusion coefficient at  $x$ . The function  $d : O \rightarrow \mathbb{R}$  such that  $d(x) > 0 \quad \forall x \in O$ , and  $\nabla_x$  is the gradient operator.

With this choice of  $\vec{J}$ , the Continuity Equation becomes

$$\frac{\partial}{\partial t}\rho(t, x) = \operatorname{div}(d(x)\nabla_x\rho(t, x)). \quad (2.7)$$

If we assume a constant diffusion coefficient, i.e  $d(x) = d > 0$ , then (2.7) becomes

$$\frac{\partial}{\partial t}\rho(t, x) = d\Delta_x\rho(t, x), \quad (2.8)$$

where  $\Delta_x$  is the Laplacian operator. This equation is called Linear Diffusion Equation (or Heat equation).

Another model of the flux is built on the principle that, the diffusion coefficient depends on the density, i.e  $d = d(\rho)$ . So if we think of the flux as a repulsive force, then as the individuals move closer, the repulsive force (or flux) becomes stronger. This provides a more realistic way of modelling diffusion and it is called **Darcy's Law**, which is given by:

$$\vec{J}(t, x) = -d(\rho(t, x))\nabla_x\rho(t, x). \quad (2.9)$$

Typically we assume that  $d = d(\rho)$  is increasing with respect to  $\rho$  and consider  $d(\rho) = C\rho^{m-1}$ , where  $C > 0$  is constant and  $m > 1$  is an integer. Substituting this into (2.5), with  $D = \frac{C}{m}$ , we recover the non-linear PDE

$$\frac{\partial}{\partial t}\rho = D\Delta_x\rho^m \quad m > 1. \quad (2.10)$$

### 2.3 Porous Medium Equation (PME)

Considering the PDE in (2.10), after a suitable scaling of the spatial and time variables, one can recover the dimensionless form

$$\frac{\partial}{\partial t}\rho = \Delta_x\rho^m. \quad m > 1 \quad (2.11)$$

This equation is called the **non-linear diffusion equation**, also known as the **Porous Medium Equation**.

The existence of a classical solution to the PME is not known. Our goal now is to derive a self-similar solution for it. For this, we look for a solution of the form  $\rho(t, x) = R^d(t)u(xR(t), \tau(t))$  where  $x \in \mathbb{R}^d$  and  $t \geq 0$ . Let  $y := xR(t)$  and  $\tau := \tau(t)$ , then putting  $\rho(t, x)$  in equation (2.11), with  $\lambda_d := d(m-1) + 2$ , it can be shown that

$$\begin{cases} R(t) = (1 + \lambda_d t)^{-\frac{1}{\lambda_d}} \\ \tau(t) = -\log R(t) \end{cases} \quad (2.12)$$

and that if  $\rho(t, x) = R^d(t)u(xR(t), \tau(t))$  solves

$$\begin{cases} \frac{\partial}{\partial t}\rho = \Delta_x\rho^m, \quad m > 1 \\ \rho(0, x) = \rho_0(x) \end{cases} \quad (2.13)$$

with  $R(t)$  and  $\tau(t)$  as in equation (2.12), then  $u = u(y, \tau)$  solves the Fokker-Plank Equation,

$$\begin{cases} u_\tau + \operatorname{div}_y(yu) = \nabla u^m \\ u(y, 0) = \rho_0(y) \end{cases} \quad (2.14)$$

Furthermore, it can be shown that equation (2.14) has a stationary solution which is given by

$$u_\infty(y) = \begin{cases} \left[ \frac{m-1}{m} \left( C - \frac{|y|^2}{2} \right) \right]^{\frac{1}{1-m}} & \text{if } |y| \leq \sqrt{2C} \\ 0 & \text{otherwise} \end{cases} \quad (2.15)$$

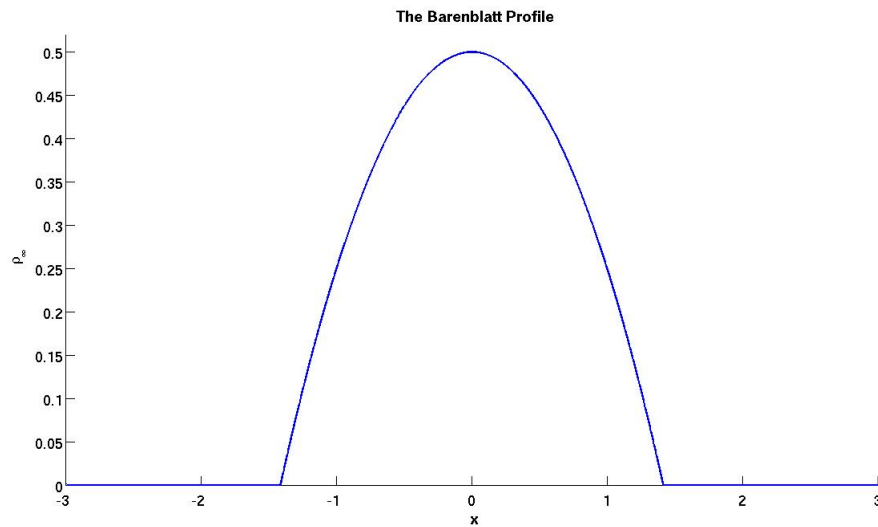
where  $C > 0$  constant. Hence, the self-similar solution to equation (2.13) is given by

$$\rho_\infty(t, x) = (1 + \lambda_d t)^{-\frac{d}{\lambda_d}} u_\infty(x(1 + \lambda_d t)^{-\frac{1}{\lambda_d}}). \quad (2.16)$$

This self-similar solution is called the **Barenblatt Solution**, and it has the following properties.

1. It diffuses in the sense that, it decays in  $L^\infty$  and the measure of its support goes to  $+\infty$  as  $t \rightarrow +\infty$ .
2. Since  $\rho_\infty$  remains compactly supported for  $t \in [0, +\infty)$ , we have a finite speed of propagation of the support, which is more realistic than in the linear diffusion case where we have an infinite speed of propagation.
3. It has  $C^\infty$  regularity on  $\{x \in \mathbb{R}^d : \rho_\infty(t, x) > 0\}$  and  $C^\alpha$  with  $\alpha = \frac{1}{m-1}$  (Hölder Regularity) on  $\mathbb{R}^d$ .

In the particular case of  $m = 2$ , the PME is called the **Quadratic Diffusion Equation** and the third property implies that,  $\rho_\infty \in C^1$



**Figure 2.1:** A Matlab representation of the Barenblatt profile defined on the interval  $[-3, 3]$ , with  $m=2$  and  $C=1$ .

## 2.4 Swarming Models: Non-local Interaction

More realistically, there often exist some kind of interaction between the individuals of a population. Consequently, their global movement may not be accurately described by a simple diffusion process. Biological aggregations such as insect swarms, ungulate herds, fish schools, and bacterial colonies are widespread examples of self-organization in nature. These groups often arise as a social phenomena, without directions from a leader or influence of external stimuli such as food and light sources. Social forces among organisms include attraction, for group cohesion, and repulsion, for collision avoidance. The resulting aggregations can confer benefits such as protection and mate choice to their members. These interactions are called non-local because each organisms interacts with other ones that are some distances away from its local position.

### 2.4.1 General Approach to Interacting Individuals

Several phenomena in particle physics, cell and population biology, and social sciences, can be modelled by a discrete set of  $N$  interacting agents, or particles, with positions  $X_1(t), \dots, X_N(t) \in \mathbb{R}^d$  depending on time, and with given masses  $m_1, \dots, m_N > 0$ . Each individual is treated as a discrete particle subject to the forces that govern animal dispersal. This approach of modelling the movement is called the Lagrangian approach: individuals are followed in their motion.

As mentioned earlier, the forces that govern the dispersal of animals are either random or vectorial. As such, possible randomness may be included in the motion of each particle. With the random location of the  $k^{\text{th}}$  particle being  $X^k(t) \in \mathbb{R}^d : k = 1, \dots, N$ , the variation in time of  $X^k(t)$  for each individual in the group at any  $t \geq 0$  gives a system of  $N$  stochastic differential equations (SDEs). [10]

More technically, from a Lagrangian point of view, the state of a system of  $N$  particles may be described as a stochastic process  $\{X_N^k(t)\}_{t \geq 0}$ , defined on a suitable probability space  $(\Omega, \mathcal{F}, \mathbb{P})$  and valued in  $(\mathbb{R}^d, \mathcal{B}_{\mathbb{R}^d})$  where  $\mathcal{B}_{\mathbb{R}^d}$  is the Borel  $\sigma$ -algebra generated by intervals on  $\mathbb{R}^d$ . [10]

According to Morale et. al. [10], one way is to model the state of the  $k^{\text{th}}$  individual with a random Dirac-Measure,  $\epsilon_{X_N^k(t)} \in \mathcal{M}(\mathbb{R}^d)$ , which is a random measure defined by:

$$\epsilon_{X_N^k(t)} = \begin{cases} 1 & \text{if } X_N^k(t) \in B, \quad \text{where } B \subset \mathcal{B} \\ 0 & \text{otherwise} \end{cases} . \quad (2.17)$$

Equation (2.17) means that, the generalized function of the  $k^{\text{th}}$  particle is a Dirac- $\delta$  function,  $\delta_{X_N^k(t)}$ . By an Eulerian approach, the collective behavior of the discrete (in the number of particles) system, may be given in terms of the spatial distribution of particles at time  $t$ , expressed in term of an empirical measure

$$X_N(t) = \sum_{k=1}^N \epsilon_{X_N^k(t)} \in \mathcal{M}(\mathbb{R}^d). \quad (2.18)$$

In case the of a vectorial force, the position of each individual at any  $t \geq 0$  is obtained in a deterministic way, and the variation in time of the position of each individual in the group gives a system of  $N$  deterministic differential equations. We then obtain the empirical measure by replacing the term " $\epsilon_{X_N^k(t)}$ " in equation (2.18) with the deterministic location of the  $k^{\text{th}}$  individual.

To describe the dynamics of the particles, one way is by a second order system given by Newton's law. For that, we set

$$\begin{cases} \dot{X}_k(t) = V_k(t) \\ m_k \dot{V}_k(t) = F_k(t) \end{cases}$$

where  $V_k$  and  $F_k(t) = F_i(X_1(t), \dots, X_N(t), V_1(t), \dots, V_N(t))$  are respectively the velocity and the interaction force of the  $K^{\text{th}}$  individual. Another way is by a first order description given by:

$$\dot{X}_k = F_k(X_1, \dots, X_N), \quad (2.19)$$

The first order approach comes from the observation that, external factors bias the motion of each single agent (particle) in terms of sudden changes in the velocity, rather than the acceleration. The chemotaxis (Keller Segel) model is an example of a first order model. Other examples are

1. Deterministic diffusion: consider particles on a real line, with all of them having the same mass,  $m_k = \frac{1}{N}$  for simplicity. The dynamics of the particles is given by

$$\begin{cases} \dot{X}_k = F_k(X_1, \dots, X_N) \\ F_k(X_1, \dots, X_N) = \frac{1}{N^2} \left( \frac{1}{X_k - X_{k-1}} - \frac{1}{X_{k+1} - X_k} \right) \\ \quad = \frac{1}{N} \left( \frac{m_k}{X_k - X_{k-1}} - \frac{m_k}{X_{k+1} - X_k} \right) \end{cases} \quad (2.20)$$

Equation (2.20) indicates that, the interaction force  $F_k$  is given by a "discretized gradient" for the "discrete density". It also can be observed that,

it is a repulsive force since each particle computes the distance between it and two neighbouring particles, and then moves away from the closest one.

2. Non-local Forces: The model is

$$\begin{cases} \dot{X}_k = F_k \\ F_k = -\sum_{\substack{j=1 \\ j \neq k}}^N m_j \nabla G(X_k - X_j) \end{cases} \quad (2.21)$$

where  $G$  is called interaction potential which is typically assumed to be (at least) continuous on  $\mathbb{R}^d \setminus \{0\}$ . A more precise choice of  $G$  depends on the phenomenon under study. Moreover,  $G$  is radial i.e.  $G(x) = g(|x|)$ . It is called an attractive potential if  $g' \geq 0$  on  $[0, +\infty)$ , or a repulsive potential if  $g' \leq 0$  on  $[0, +\infty)$ . It is an attractive repulsive potential if it is repulsive inside a ball  $|x| \leq R$  and attractive outside. Some examples of interaction potentials are:

- (a) Repulsive potentials: The Morse Potential,  $G(x) = \frac{1}{2} \exp(-|x|)$ ; and the Gaussian Potential,  $G(x) = \frac{1}{\sqrt{\pi}} \exp(-x^2)$ .
- (b) An attractive potential:  $G(x) = -\frac{1}{2} \exp(-|x|)$
- (c) An attractive repulsive potential:  $G(x) = M_R \exp(-\frac{|x|}{R}) - M_A \exp(-\frac{|x|}{A})$  where  $M_R, M_A, R, A > 0$ .

### 2.4.2 Non-Local Porous Medium Equation (NLPME)

Although the individual modelling approach does not compromise on the identity of the individuals, it may be problematic when we have a lot of particles. As a generalized approach, a continuum description has been provided in the discrete based setting. The continuum description of the density leads to the a model that is based on PDE's. This is a classical widespread approach which according to Morale et. al., is so due above all, to the wider spread knowledge on Non-linear PDE's. The advantages of the continuum approach are those of ease of analysis and elimination of arbitrary spatial discretization.



In other words, we want to replace the ODE (or SDE) system with a PDE. Here, we consider purely "deterministic" interactions. To formulate the PDE, we consider the ODE system  $\dot{X}_i = F_i(X_1, \dots, X_N)$  and then define the empirical measure of the particles as  $\mu(t) = \sum_{i=1}^N m_i \delta_{X_i(t)}$ , which is also an element of the space of distributions,  $\mathcal{D}'(\mathbb{R}^d)$ .

In a classical dynamic framework, and by neglecting both inertial effects and the interaction of a particle with itself, the movement of the particles can be described through the Cauchy problem on  $\mathbb{R}^{dN}$ :

$$\begin{cases} \dot{X}_i(t) = -\sum_{\substack{j=1 \\ j \neq i}}^N m_j \nabla G(X_i(t) - X_j(t)) \\ X_i(0) = X_i^0 \end{cases} \quad i = 1, \dots, N \quad (2.22)$$

where  $X_1, \dots, X_N \in \mathbb{R}^d$ ,  $m_1, \dots, m_N > 0$  and  $\sum_{i=1}^N m_i = 1$ . Now, we derive the PDE for (2.22) as follows. Let  $\varphi = \varphi(x, t) \in C_c^\infty(\mathbb{R}^d \times [0, +\infty))$  ( a test function). Then

$$\langle \partial_t \mu, \varphi \rangle = -\langle \mu, \partial_t \varphi \rangle = -\sum_{k=1}^N m_k \langle \delta_{X_k(t)}, \partial_t \varphi \rangle = -\sum_{k=1}^N m_k \partial_t \varphi(X_k(t), t) \quad (2.23)$$

where as the first equality is by the definition of derivatives in distributions, the second one is by the definition of the empirical measure and then in the final one, by the definition of the Dirac- $\delta$  function:  $\langle \delta_x, \varphi \rangle = \varphi(x)$ . Letting  $T > 0$ , we integrate (2.23) on  $[0, T)$  taking into account that  $\text{supp}(\varphi) \subset \mathbb{R}^d \times [0, T)$  and assuming that  $\nabla G(0) = 0$ .

$$\begin{aligned}
\int_0^T \langle \partial_t \mu, \varphi \rangle dt &= - \sum_{k=1}^N m_k \int_0^T \partial_t \varphi(X_k(t), t) dt \\
&= - \sum_{k=1}^N m_k \int_0^T \left[ \frac{d}{dt} \varphi(X_k(t), t) - (\nabla_{X_k} \varphi(X_k(t), t)) \dot{X}_k(t) \right] dt \\
&= \sum_{k=1}^N m_k \varphi(X_k(0), 0) \\
&\quad - \sum_{k=1}^N m_k \int_0^T \left[ (\nabla_{X_k} \varphi(X_k(t), t)) \sum_{\substack{j=1 \\ j \neq k}}^N m_j \nabla G(X_k(t) - X_j(t)) \right] dt \\
&= \langle \mu(0), \varphi(\cdot, 0) \rangle \\
&\quad - \int_0^T \sum_{k=1}^N m_k \nabla_{X_k} \varphi(X_k(t), t) (\mu(t) * \nabla G(X_k(t))) dt \\
&= \langle \mu(0), \varphi(\cdot, 0) \rangle - \int_0^T \langle \mu(t), (\mu(t) * \nabla G(X_k(t))) \nabla \varphi(\cdot, t) \rangle dt \\
&= \langle \mu(0), \varphi(\cdot, 0) \rangle - \int_0^T \langle (\mu(t) * \nabla G(X_k(t))) \mu(t), \nabla \varphi(\cdot, t) \rangle dt \\
&= \int_0^T \langle \operatorname{div} [(\mu(t) * \nabla G(X_k(t))) \mu(t)], \varphi(\cdot, t) \rangle dt.
\end{aligned}$$

Hence, we have

$$\int_0^T \langle \partial_t \mu - \operatorname{div} [\mu(\nabla G * \mu)], \varphi \rangle dt = 0,$$

which is the distributional formulation of the PDE,

$$\frac{\partial \mu}{\partial t} = \operatorname{div}(\mu \nabla G * \mu), \quad (2.24)$$

which is the continuum counterpart of equation (2.22). Putting  $\mu = \rho$  (the population density, which could be a function or a distribution), we get:

$$\frac{\partial \rho}{\partial t} = \operatorname{div}(\rho \nabla G * \rho), \quad (2.25)$$

which is the **Non-Local Continuum Interaction Equation**. We refer to it as the **Non-Local Porous Medium Equation (NLPME)**. Let's denote  $v = \nabla G * \rho$ . It has been shown that  $v$  the continuous analogue of equation 2.21, which gives the velocity of the  $k^{\text{th}}$  organism

In addition to equation (2.25), given the initial condition  $\rho(x, t) = \rho_0(x)$ , the following properties hold.

1. If  $\rho_0 = \mu(0) = \sum_{i=1}^N m_i \delta_{X_i(0)}$  (i.e the initial condition is a singular distribution), then the solution to the problem is (a singular distribution):  
 $\rho = \mu(t) = \sum_{i=1}^N m_i \delta_{X_i(t)}$ , with  $X_k(t)$  being solutions to the ODE with initial conditions  $X_k(0)$
2. If  $G \in C^2(\mathbb{R}^d)$  and  $\rho(0) \in L^1(\mathbb{R}^d)$ , then the solution to the problem is always a function (not singular distributions)  $\in L^1(\mathbb{R}^d \times [0, +\infty))$

Wrapping up on this chapter, it can be shown that, for:

- For a deterministic diffusion with  $\dot{X}_i = \frac{1}{N^2} \left( \frac{1}{X_i - X_{i-1}} - \frac{1}{X_{i+1} - X_i} \right)$ , if we consider the empirical measure  $\mu(t) = \sum_{i=1}^N m_i \delta_{X_i(t)}$  and let  $N \rightarrow +\infty$ , we recover the heat equation:  $\partial_t \mu = \partial_{xx} \mu$

In the remaining parts of this thesis, our interest is to use computational methods to show that, if one considers the NLPME and lets  $N \rightarrow +\infty$ , then the PME can be recovered within a reasonable margin of error.

# Chapter 3

## Numerical Analysis 1: Simulating the PME and NLPME

Moving on to the main purpose of this thesis, in this chapter, we present elaborate explanations of the procedure used to implement an approximation scheme for the PME and NLPME for various initial conditions.

### 3.1 Simulating the PME with the Finite-Difference Method

The problem under study is the PME imposed with a Dirichlet boundary and various initial conditions. First, we use a Barenblatt distribution for the organisms' initial density. This enables us to check that PME is well simulated since we expect the solution to remain a Barenblatt. In order to compare later with the NLPME, we also simulate the PME with a constant initial condition and a triangular one as well. We use a finite-difference approach as the approximation procedure for the problem.

More formally, for  $\Omega = [-a, a] \times [0, T]$  for any choice of  $a, T > 0$ , the problem

under consideration is:

$$\begin{cases} \frac{\partial u}{\partial t} &= \frac{\partial^2 u^2}{\partial x^2} & \text{for } (x, t) \in \Omega \\ u(t, -a) &= 0 & \text{for } t \in [0, T] \\ u(t, a) &= 0 & \text{for } x \in [-a, a] \\ u(0, x) &= u_i(0) & \text{for } x \in [-a, a] \end{cases} \quad (3.1)$$

with  $u_i$  defined in the following. For  $i = 1$ , we use a Barenblatt initial condition (BIC):

$$u_1 = \begin{cases} \frac{1}{2} \left( c - \frac{|x|^2}{2} \right) & \text{for } -\sqrt{2c} < x < \sqrt{2c} : c > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (3.2)$$

for  $i = 2$ , we use the constant initial condition (CIC):

$$u_2 = \begin{cases} 1 & \text{for } -\frac{1}{2} < x < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}, \quad (3.3)$$

and for  $i = 3$ , we use the triangular initial condition (TIC):

$$u_3 = \begin{cases} 1 - |x| & \text{for } -1 < x < 1 \\ 0 & \text{otherwise} \end{cases}. \quad (3.4)$$

Its important to note from the above initial conditions that,  $u_2$  and  $u_3$  have a total mass (or  $L_1$  norm) of 1.

### 3.1.1 Procedure with the Barenblatt Initial Density

The finite difference scheme formulation we use here is based on a lecture provided by Prof. Jens Struckmeier. Prior to defining a scheme for the PME, we discretize the domain by introducing a rectilinear grid on  $\Omega$  with sides parallel to the  $x$  and  $t$  axes. That is, on the time interval  $[0, 1]$ , we define the time grid with  $k$  as grid size and  $T$  as number of time steps.

$$\mathbb{G}^k = \{t_n = nk : k > 0, n = 0, 1, \dots, T\}$$

and on the spatial interval  $(-a, a)$ , we define the spatial with grid size  $h$  as

$$\mathbb{G}_h = \{x_m = mh : h > 0, m \in \mathbb{Z}\}$$

together with the two boundary points

$$\partial\mathbb{G} = \{-a, a\}$$

In particular, the functions satisfying the differential equations and the corresponding difference equation defined in the following will be denoted by  $u_m^n$  and  $U_m^n$  respectively. Where the indices  $m$  and  $n$  denote the grid points  $x_m = mh$  and  $t_n = nk$ . So,  $u_m^n = u(t_n, x_m)$  and  $U_m^n = U(t_n, x_m)$ .

Using the Explicit Euler Method defined on  $\mathbb{G}_k \times \mathbb{G}_h \cup \partial\mathbb{G}_h$ , the second order term of the PME in problem 3.1 gives

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{(U_{m+1}^n)^2 - 2(U_m^n)^2 + (U_{m-1}^n)^2}{h^2}$$

while the first order term gives

$$\frac{\partial u}{\partial t} \approx \frac{U_m^{n+1} - U_m^n}{k}$$

Hence, the corresponding difference equation for the PME is given by

$$\frac{U_m^{n+1}}{k} = \frac{U_m^n}{k} + \frac{U_{m+1}^n - 2U_m^n + U_{m-1}^n}{h^2}$$

Simplifying with  $\lambda = \frac{k}{h^2}$ , we obtain

$$U_m^{n+1} = U_m^n + \lambda(U_{m+1}^n - 2U_m^n + U_{m-1}^n)$$

Again from problem 3.1, we have the boundary conditions:  $U(t_n, -a) = 0$  and  $U(t_n, a) = 0 \quad \forall n$ ; and the initial condition  $U_i(0, x_n)$  for any of the  $i$ 's under consideration. One can observe without much difficulty that, the problem has now been transformed to a system of linear equations. Moreover, if we

### 3.1. SIMULATING THE PME WITH THE FINITE-DIFFERENCE METHOD 29

denote the number of spatial grid point by  $N$ , which is particularly the number of organisms (or particles) under study in the given spatial domain, then for  $(t_n, x_m) \in \mathbb{G}_k \times \mathbb{G}_h \cup \partial\mathbb{G}_h$ , the approximated solution to the problem gives a matrix of dimension  $T \times N$ , which we denote by  $U_P$ . Implementing the finite-difference scheme and solving the linear system, we obtain  $U_P$  via the following lines of pseudocode.

---

**Algorithm 1** : Using the Explicit Euler Method to simulate the PME with the BIC -

**Part 1**

---

1: **procedure** SET PARAMETERS FOR THE SIMULATION

2:  $c = \frac{1}{16}$

3:  $N = 100$

4:  $T = 500$

5:  $a = 1$

6:  $x_{max} = a$

7:  $x_{min} = -a$

8:  $h = \frac{x_{max} - x_{min}}{N - 1}$

▷ Assume equal spatial grid size

9:  $k = \frac{1}{T-1}$

▷ Assume equal time step size

**Require:**  $x = x_{min} : h : x_{max}$

▷ Descretize spatial domain

10:  $lx = length(x)$

**Require:**  $t = 0 : k : 1$

▷ Descretize time domain

11: **end procedure**

12: **procedure** CONSTRUCT THE BARENBLATT INITIAL CONDITION

13: **for**  $i \leftarrow 1 : N$  **do**

14:     **if**  $-\sqrt{2c} \leq x(i) \leq \sqrt{2c}$  **then**

15:          $U_1(i) = \frac{1}{2} \left( c - \frac{x(i)^2}{2} \right)$

16:     **else**

17:          $U_1(i) = 0$

18:     **end if**

19: **end for**

▷ This algorithm continues with algorithm 2 on the next page.

---

---

**Algorithm 2** : Using the Explicit Euler Method to simulate the PME with the BIC -

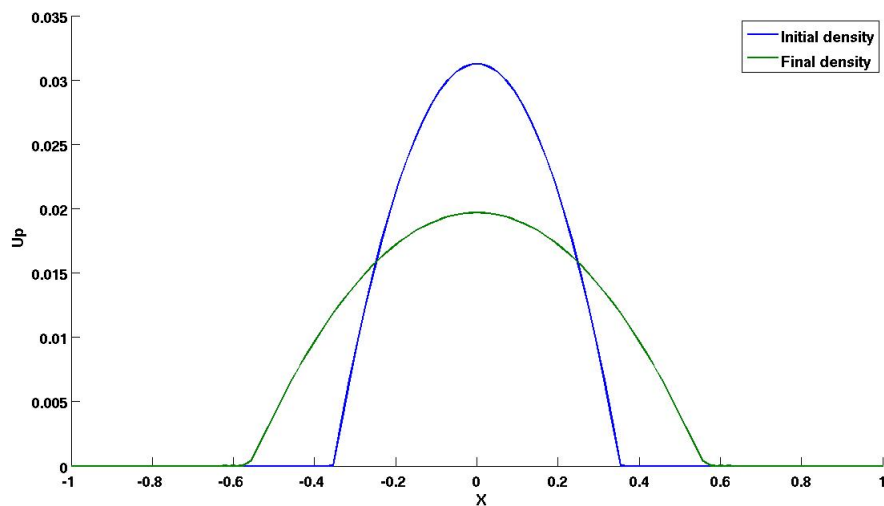
**Part 2**


---

20: **end procedure**21: **procedure** SET THE INITIAL AND BOUNDARY DENSITIES22:  $U_P(1, 1 : N) = U_1$  ▷ Set  $U_1$  to initial values of  $U_P$ 23:  $U_P(:, 1) = 0$  ▷ Set boundary values of  $U_P$ 24:  $U_P(:, lx) = 0$ 25: **end procedure**26: **procedure** SOLVE THE LINEAR SYSTEM27: **for**  $n \leftarrow 1 : T$  **do**28:  $U_P(n+1, 2 : N-1) = U_P(n, 2 : N-1) + \lambda(U_P^2(n, 1 : N-2) - 2U_P^2(n, 2 : N-1) + U_P^2(n, 3 : N))$ 29: **end for**30: **end procedure**


---

Figure 3.1 illustrates the result of the simulation.



**Figure 3.1:** Showing an approximated solution for the PME with a Barenblatt Initial data: solving with the Explicit Euler method.



### 3.1. SIMULATING THE PME WITH THE FINITE-DIFFERENCE METHOD 31

From figure 3.1, we observe that the algorithm simulates the behaviour of the PME well. This is because, as in the actual case of the PME, the initial distribution of the organisms' density reduces while spreading to other parts of the domain. However, we also observe that, the peak of the solution is still quite high at the end of the simulation. This may be quite problematic. Hence, to take care of this set back which arises from using equal time steps, we consider an alternative way of solving the problem.

Another useful approach in formulating a linear system for the problem is by using the Explicit Euler scheme together with the Runge Kutta Method. Henceforth, we refer to this approach as "Approach Two". By this approach, we reduce the PDE first to an ODE in the time variable. Next, we solve the resulting ODE with an in-built ODE solver in Matlab which uses the Runge Kutta method and comes in different varieties. Here, we use the one called **ODE45** and allow for adaptive time steps. The advantage is that, unlike in the previous approach where we use equal time steps, here, each time step is chosen such that there is a maximum change in  $U_P$  as a result of the change in time.

To explain further, applying the Explicit Euler Scheme defined on the set:  $G_h \cup \partial G_h$  to the second order term of the PME gives:

$$\frac{du}{dt} \approx \frac{U_{m+1}^2 - 2U_m^2 + U_{m-1}^2}{h^2}. \quad (3.5)$$

Next, we implement and store equation (3.5) as function in Matlab and then solve (or numerically integrate) it using the solver ODE45.

---

**Algorithm 3** : Using "Approach Two" to simulate the PME with the BIC
 

---

```

1: procedure SET VARIABLES FOR THE SIMULATION
2:   global  $N \ x \ h \ l$                                 ▷ Make global variables
3:    $c = \frac{1}{16}$                                           ▷ Set parameters
4:    $N = 100$ 
5:    $T = 500$ 
6:    $x_{max} = 1$ 
7:    $x_{min} = -1$ 
8:    $h = \frac{x_{max} - x_{min}}{N - 1}$                         ▷ Assume equal spatial grid size
Require:  $x = x_{min} : h : x_{max}$                         ▷ Descretize spatial domain
9:   Let  $length(x) = l$ 
10:  for  $i \leftarrow 1 : N$  do                                ▷ Construct the initial data
11:    if  $-\sqrt{2c} \leq y(i) \leq \sqrt{2c}$  then
12:       $U_1(i) = \frac{1}{2} \left( c - \frac{y(i)^2}{2} \right)$ 
13:    else
14:       $U_1(i) = 0$ 
15:    end if
16:  end for
17: end procedure

18: procedure IMPLEMENT A FUNCTION FOR EQUATION 3.5
19:  function  $PME_{Bar}(t, U_P)$ 
20:    global  $N \ x \ h \ l$                                 ▷ Make global variables
21:     $dU_P = zeros(l, 1)$                                 ▷ column vector, Set boundary data
22:    for  $i \leftarrow 2 : l - 1$  do                        ▷ Build the ODE's
23:       $dU_P = \frac{U_P^2(i-1) - 2U_P^2(i) + U_P^2(i+1)}{h^2}$     ▷ A vector
24:    end for
25:  end function
26: end procedure

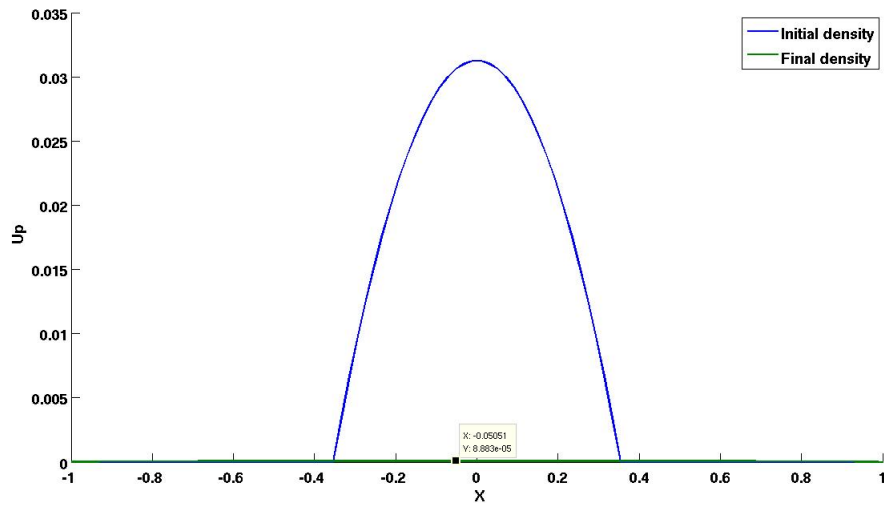
27: procedure SOLVE THE SYSTEM OF ODE'S
28:   $[NT, U_P] = ode45(@PME_{Bar}, [0, T], U_1)$   ▷ Use the ODE solver, NT is a
  vector containing all time grid points
29: end procedure

```

---

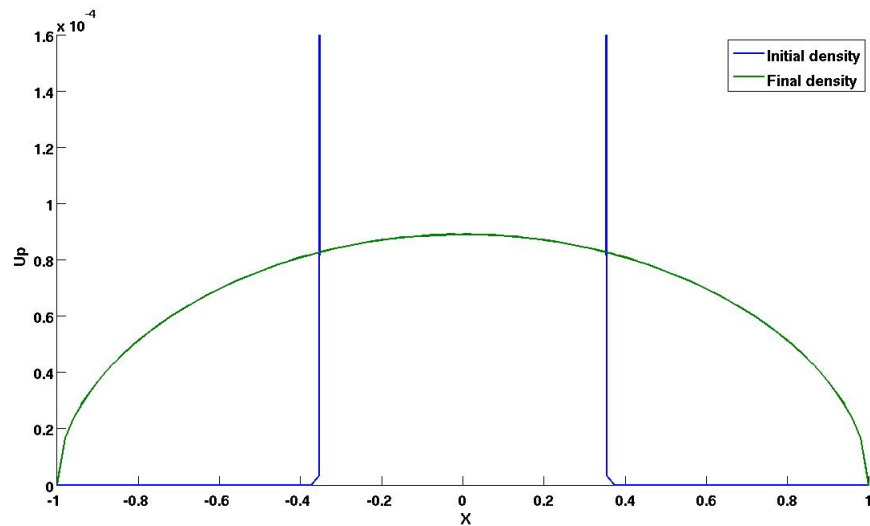
### 3.1. SIMULATING THE PME WITH THE FINITE-DIFFERENCE METHOD 33

Figures 3.2 and 3.3 illustrate the results of the simulation.



**Figure 3.2:** *Approximated solution for the PME with a Barenblatt Initial data: solving with both Explicit Euler and Runge Kutta methods*

Rescaling the axes in order to visualize the final solution better, we obtain the following figure.



**Figure 3.3:** *A rescaled version of figure 3.2, showing the final distribution of the organisms' densities.*

Comparing the two approaches used: Explicit Euler only and Explicit Euler with Runge Kutta, figures 3.1 and 3.3 demonstrate that, within the same simulation time, the latter approach gives a solution whose maximum value is much more smaller than in the former. Based on this observation, we may conclude that, the latter approach integrates the PDE faster.

Therefore, we subsequently use "Approach Two" with adaptive time steps for our simulations. In addition, since the code for "Approach Two is more versatile, even in cases where a time step-by-time step analysis is required, we use "Approach Two" with equal time steps and increase the simulation time to take care of the set back that arises from using the equal time steps.

### 3.1.2 Using the Constant Initial density

1

Now, we consider the initial condition  $u_2$ , as defined in equation 3.3. After simulating with  $u_2$ , we also consider another it's other variation which describes the organisms' population density as a mass that is concentrated in a small volume, like a Dirac- $\delta$  function. In application, one may see this as the distribution of organisms at the moment when they are produced (or prior to dispersion). The benefit of considering this other initial density is that, we are also able to check whether or not the results are affected as a result of using concentrated initial densities. To model the "Dirac- $\delta$ -like" function, we introduce the parameter  $\alpha = \frac{1}{N}$  and re-write the initial density as:

$$u_2 = \begin{cases} \frac{1}{\alpha} & \text{for } -\frac{\alpha}{2} < y < \frac{\alpha}{2} \\ 0 & \text{otherwise} \end{cases} \quad (3.6)$$

The following algorithms illustrate the set of pseudocode that we use to obtain  $U_p$ . Due to the non-smoothness of the initial density, we use the ODE solver **ODE23s** instead.<sup>1</sup>

<sup>1</sup><https://it.mathworks.com/help/matlab/math/choose-an-ode-solver.html>

---

**Algorithm 4** : Using "Approach Two" to simulate the PME with the constant initial density- **Part 1**

---

```

1: procedure SET VARIABLES FOR THE SIMULATION
2:   global  $h$   $lx$                                      ▷ Make global variables
3:    $N = 100$ 
4:    $T = 500$ 
5:    $a = 1$ 
6:    $\alpha = \frac{1}{N}$ 
7:    $x_{max} = a$ 
8:    $x_{min} = -a$ 
9:    $h = \frac{x_{max} - x_{min}}{N - 1}$                        ▷ Assume equal spatial grid size
Require:  $x = x_{min} : h : x_{max}$                        ▷ Descretize spatial domain
10:   $lx = \text{length}(x)$ 
11: end procedure

12: procedure IMPLEMENT A FUNCTION FOR EQUATION 3.5
13:  function  $PME_{Const}(t, U_P)$ 
14:    global  $N$   $x$   $h$   $l$ 
15:     $dU_P = \text{zeros}(lx, 1)$                                ▷ column vector, Set boundary data
16:    for  $i \leftarrow 2 : lx - 1$  do                       ▷ Build the ODE's
17:       $dU_P = \frac{U_P^2(i-1) - 2U_P^2(i) + U_P^2(i+1)}{h^2}$ 
18:    end for
19:  end function
20: end procedure

21: procedure CONSTRUCT THE CONSTANT INITIAL DENSITY
22:  for  $i \leftarrow 1 : N$  do
23:    if  $-\frac{\alpha}{2} \leq x(i) \leq \frac{\alpha}{2}$  then
24:       $U_2(i) = \frac{1}{\alpha}$ 
25:    else
26:       $U_2(i) = 0$ 
27:    end if

```

▷ This algorithm continues on the next page

---

---

**Algorithm 5** : Using "Approach Two" to simulate the PME with the constant initial density- Part 2

---

28:    **end for**

29: **end procedure**

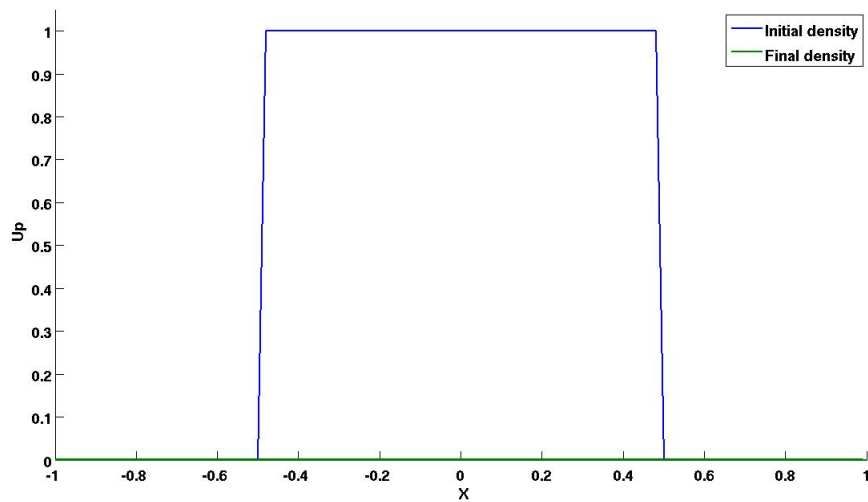
30: **procedure** SOLVE THE SYSTEM OF ODE'S

31:     $[NT, U_P] = \text{ode23s}(@\text{PME}_{Const}, [0, T], U_2)$  Use the ODE solver, NT is a vector containing all the time grid points used

32: **end procedure**

---

Figures 3.4 and 3.5 depict the results for simulating with  $\alpha = 1$  (non-concentrated constant initial density).



**Figure 3.4:** An approximated solution for the PME with the constant Initial density: solving with both Explicit Euler and Runge Kutta methods

3.1. SIMULATING THE PME WITH THE FINITE-DIFFERENCE METHOD 37

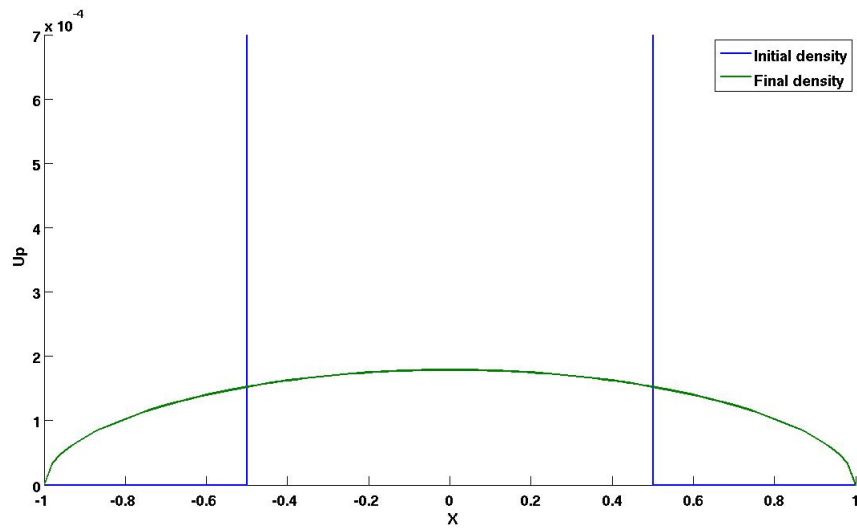


Figure 3.5: A rescaled version of figure 3.4, showing clearly the final density.

In the case of the concentrated constant initial density ( $\alpha = \frac{1}{N}$ ), we obtain the following results.

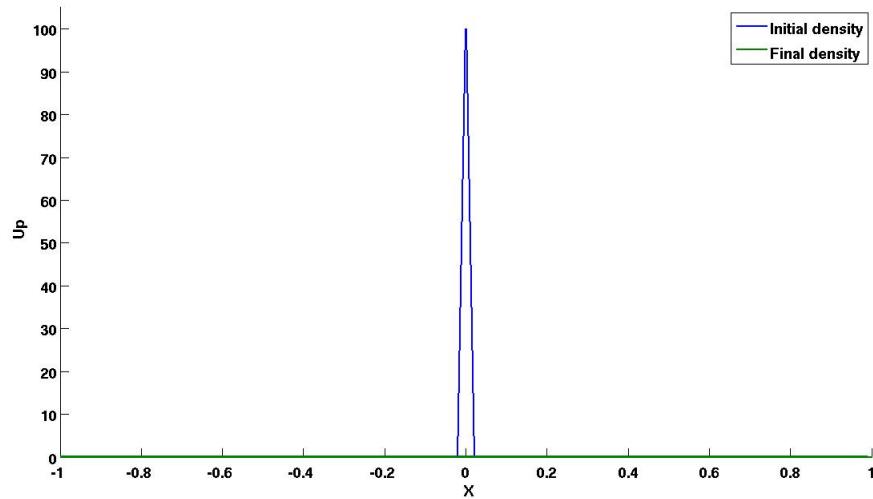


Figure 3.6: An approximated solution for the PME with the Dirac "δ-like" initial density: solving with both Explicit Euler and Runge Kutta methods

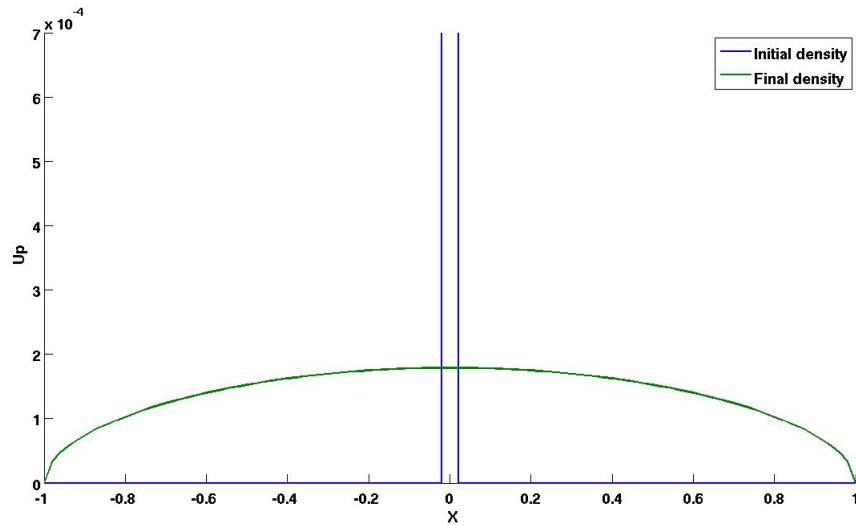


Figure 3.7: A rescaled version of figure 3.6, showing clearly the final density.

The above figures indicate that using a concentrated constant initial density here does not alter the results of the simulation significantly.

### 3.1.3 Using a Triangular Initial Density

Finally, we consider the triangular initial density  $u_3$  (3.4), which is by far more regular than the previous one used. In particular, it is Lipschitz continuous while the constant one is not even continuous. In this case also, we simulate for a concentrated version of initial density, given by:

$$u_3 = \begin{cases} \frac{1}{\alpha} \left(1 - \frac{1}{\alpha}|x|\right) & \text{for } -\alpha < x < \alpha \\ 0 & \text{otherwise} \end{cases} . \quad (3.7)$$



---

**Algorithm 6 :** Using "Approach 2" to solve the PME with a Triangular initial density- Part 1

---

```

1: procedure SET VARIABLES FOR THE SIMULATION
2:   global  $h$   $lx$                                 ▷ Make global variables
3:    $N = 101$ 
4:    $T = 200$ 
5:    $\alpha = \frac{1}{N}$ 
6:    $a = 5$ 
7:    $x_{max} = a$ 
8:    $x_{min} = -a$ 
9:    $h = \frac{x_{max} - x_{min}}{N - 1}$                     ▷ Assume equal spatial grid size
Require:  $x = x_{min} : h : x_{max}$                 ▷ Descretize spatial domain
10:   $lx = length(x)$ 
11: end procedure

12: procedure CONSTRUCT THE CONSTANT INITIAL DENSITY
13:  for  $i \leftarrow 1 : N$  do
14:    if  $-\alpha < x(i) < \alpha$  then
15:       $U_3(i) = \frac{1}{a} - \frac{1}{a^2} |x(i)|$ 
16:    else
17:       $U_3(i) = 0$ 
18:    end if
19:  end for
20: end procedure

21: procedure IMPLEMENT A FUNCTION FOR EQUATION 3.5
22:  function  $PME_{Tri}(t, U_P)$ 
23:    global  $h$   $lx$ 
24:     $dU_P = zeros(lx, 1)$                         ▷ Column vector, set boundary data

```

▷ This algorithm continues on the next page

---

---

**Algorithm 7 :** Using "Approach 2" to solve the PME with a Triangular initial density- Part 2

---

```

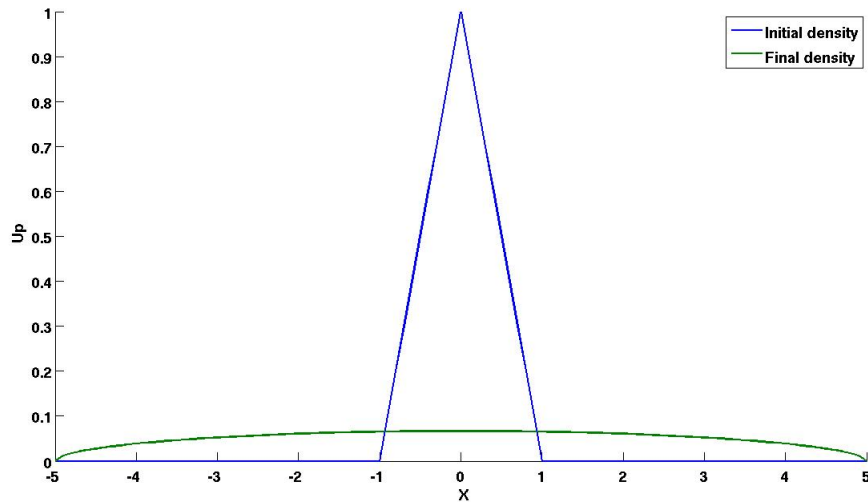
25:     for  $i \leftarrow 2 : lx - 1$  do                                     ▷ Build the ODE's
26:          $dU_P = \frac{U_P^2(i-1) - 2U_P^2(i) + U_P^2(i+1)}{h^2}$ 
27:     end for
28: end function
29: end procedure

30: procedure SOLVE THE SYSTEM OF ODE'S
31:      $[NT, U_P] = ode23s(@PME_{Tri}, [0, T], U_3)$  ▷ Use the ODE solver, NT is a
        vector containing all the time grid points used
32: end procedure

```

---

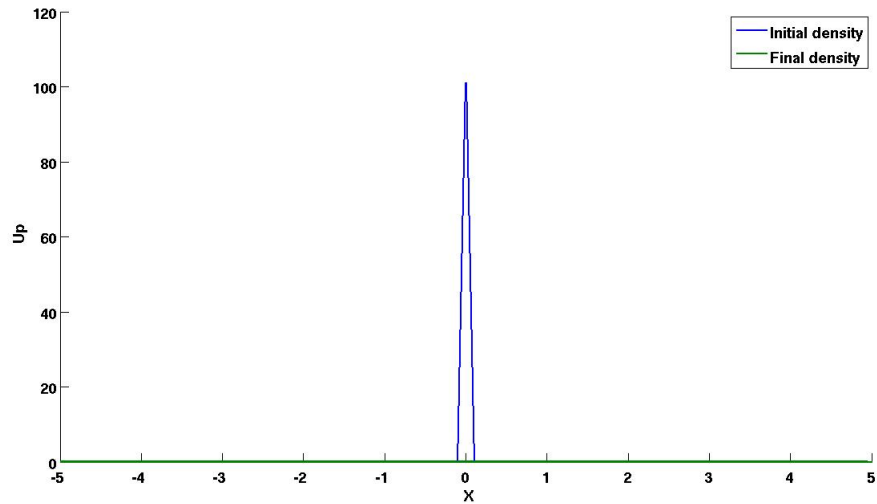
Simulating with  $\alpha = 1$  (non-concentrated initial density), we obtain the following result.



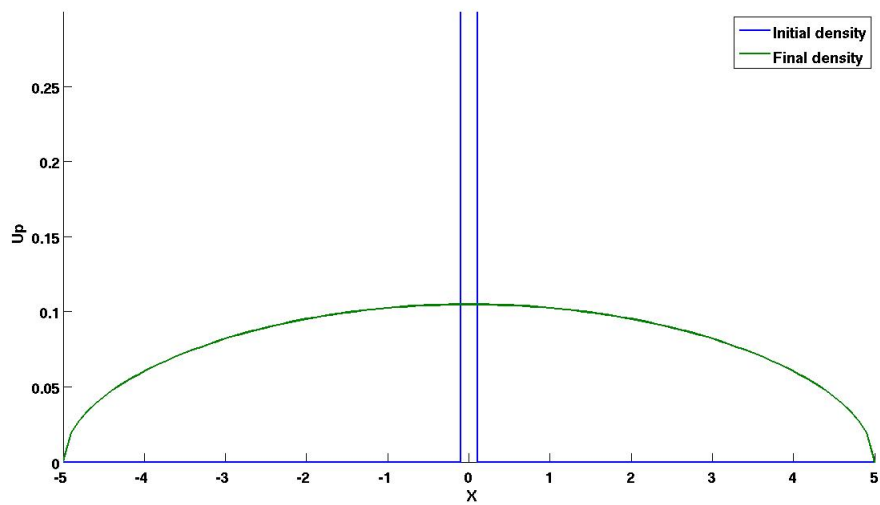
**Figure 3.8:** An approximated solution for the PME with the triangular initial density: solving with both Explicit Euler and Runge Kutta methods.

Also, we obtain the following results for simulating with  $\alpha = \frac{1}{N}$ .

3.1. SIMULATING THE PME WITH THE FINITE-DIFFERENCE METHOD 41



**Figure 3.9:** An approximated solution for the PME with the triangular initial density: solving with both Explicit Euler and Runge Kutta methods.



**Figure 3.10:** A rescaled version of figure 3.9 showing clearly the final solution.

Comparing the results, we observe that simulating on the domain  $[-5, 5]$  (i.e.  $a = 5$ ) gives the maximum value for the final density to be: 0.1049 when we use the concentrated triangular initial density and 0.0668 for the non-concentrated one. However, as seen earlier, simulating with  $a = 1$  rather shows

an insignificant change in the maximum value: 0.0045 for the concentrated initial density and 0.0044 for the non-concentrated one. Indeed, this observation suggests that in generally for PME, using a concentrated initial density has an effect on the results (the final density) especially in a bigger domain.

## 3.2 Similating the NLPME with the Particle Method

Our aim now is to obtain an approximated solution for the NLPME. Unlike in the previous section where we simulated the PME with a classical method, here, we simulate the NLPME via the so called particle method. As in the case of the PME, we simulate with the constant initial distribution of the organisms' density; followed by the triangular one. Finally, we also simulate for concentrated versions of the initial densities and then compare the results.

### 3.2.1 Procedure Using a Constant Distribution of Initial Density.

Let's consider a total of  $N + 1$  organisms with the initial density:

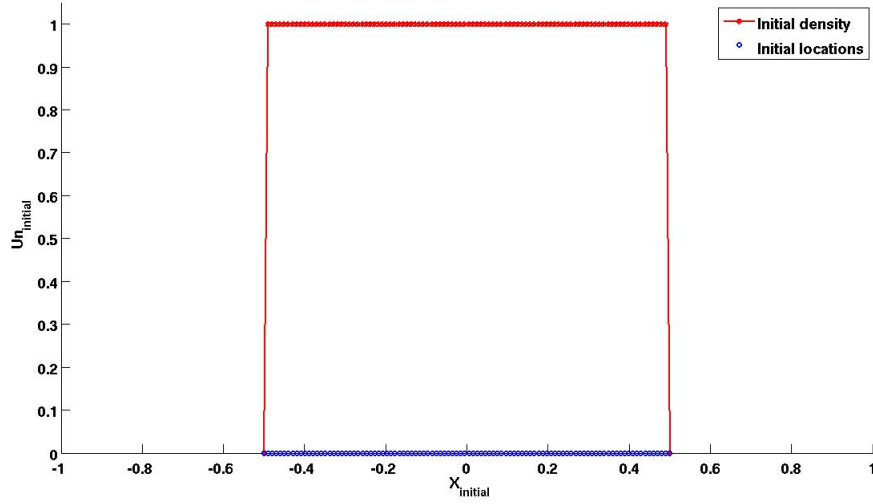
$$U_0 = \begin{cases} 1 & \text{for } -\frac{1}{2} < x < \frac{1}{2} \\ 0 & \text{for } x \in \{-\frac{1}{2}, \frac{1}{2}\} \end{cases} . \quad (3.8)$$

One can easily show that  $U_0 \in L^1([-\frac{1}{2}, \frac{1}{2}]) \cap L^\infty([-\frac{1}{2}, \frac{1}{2}]) \cap C_c([-\frac{1}{2}, \frac{1}{2}])$ . To initiate the particle method, we need to first find the locations of the organisms such that, their initial density is given by  $U_0$ . This is known as the **atomization procedure**. In this case of a constant density, one can easily obtain this by splitting the support of the initial density into  $N$  intervals with grid points:  $\{x_0, \dots, x_N\}$  representing the initial locations of the organisms. Assuming an equal distribution of mass in each interval, then the intervals must be equal with a density of 1 in each one of them. We illustrate this initial step with figure (3.11).

Formally, the total mass of the organisms is given by:

$$\begin{aligned} \int_{-\frac{1}{2}}^{\frac{1}{2}} U_0 &= U_0 \\ &= 1. \end{aligned}$$

Therefore, the mass within each interval is:  $\int_{x_i}^{x_{i+1}} U_0 = x_{i+1} - x_i = \frac{1}{N}$ , since we are assuming an equal distribution. In addition to the above, we assume that the law of conservation of mass,  $M$  holds. That is,  $M(0) = M(t) \forall t \in [0, T]$ , for a large enough  $T$  as defined previously.



**Figure 3.11:** Showing the initial location of organisms with their constant density in  $[-\frac{1}{2}, \frac{1}{2}]$ .

Now, for the dynamics of the organisms, we recall from equation (2.21) that, for a non-local interaction, the velocity of the  $i^{th}$  organism is given by:

$$\dot{x}_i(t) = - \sum_{\substack{j=0 \\ j \neq i}}^N m_j G'(x_i(t) - x_j(t)) \quad \forall i = 0, \dots, N, \quad (3.9)$$

Where  $m_i$  is the mass of the  $i^{th}$  organism and  $G$  is the repulsion potential which describes dispersal among the organisms. As mentioned earlier,  $v = G' * \rho$  is the continuous analogue of  $\dot{x}_i(t)$ . Particularly, while  $v$  is obtained via the Eulerian approach of considering the population density,  $\dot{x}_i(t)$  is obtained via the Lagrangian approach of considering each individual embedded in a finite

population of organisms of the same kind. [10]

In effect, the two different approaches (Lagrangian and Eulerian) describe the velocity at different scales: the finer scale description based on the (stochastic or deterministic) velocities of individuals (microscale) and the larger scale description based on the (continuum) velocities of population densities (macroscale). [10] As demonstrated by Morale et. al., we consider a multiscale approach here since the aim of our analysis is to provide a numerical framework for bridging the gap between the two different scales: the microscale which represents a non-local interaction (i.e the NLPME); and the macroscale which represent a local interaction (i.e the PME).

With the multiscale approach, to consider a rigorous limit, we also include the concept of **mesoscale** as introduced in [10, 11, 12]. That is, we view the velocity in a scale which is much larger than the microscale and much smaller than the macroscale, so that as the number of individuals increases, at the mesoscale we have a sufficient number of particles so that a law of large numbers may still be applied. The mesoscale can be obtained by a suitable rescaling of the potential function modelling the interaction (dispersion) among organisms.

As mentioned in subsection 2.1.1, the interaction among particles is mathematically modelled by an interaction potential which depends on the distance between any two organisms. As such, the range of the potential depends on the scale of the interaction. Since we are considering a dispersion potential,  $G$  here, we need to model the dependance of  $G$  on the three scales: microscale, mesoscale and macroscale. Mathematically, one way to distinguish between different scales, in the above sense, is based on the choice of a **scaling parameter** in  $G$ . [10]

We denote the scaled version of potential by  $G_\epsilon$  with  $\epsilon$  being its scaling parameter which also represents its range. This means that, each organism repels other organisms that are within a radial distance of  $\epsilon$  away from it. In particular, considering organisms in  $\mathbb{R}^d$ ,  $G_\epsilon(x) = \epsilon^{-1}G(\epsilon^{-\frac{1}{d}}x)$  is the appropriate scaling

for the potential. [10]

Moreover, a suitable choice for  $\epsilon$  is:  $\epsilon = \frac{1}{N^\beta}$  with  $\beta \in [0, 1]$  called the scaling coefficient. Moreover, if  $\beta = 0$ , then we have a macroscale with the McKean-Vlasov interaction (weak), a microscale with the hydrodynamic interaction (strong) if  $\beta = 1$  and finally, a mesoscale with a moderate interaction if  $\beta \in (0, 1)$ . [10]

Hence, by the multiscale approach, a more suitable equation for the velocity of the  $i^{\text{th}}$  particle is:

$$\dot{x}_i(t) = - \sum_{\substack{j=0 \\ j \neq i}}^N m_j G'_\epsilon(x_i(t) - x_j(t)) \quad \forall i = 0, \dots, N. \quad (3.10)$$

To proceed with the simulations, we need to first substitute an explicit formula for  $G$  into the above equation. In this study, we focus on two types of dispersion potentials: the Gaussian and Morse potentials.

The Gaussian Potential is given by:

$$G(x) = \frac{1}{\sqrt{\pi}} \exp(-x^2).$$

Therefore, its scaled form is:  $G_\epsilon(x) = \frac{N^\beta}{\sqrt{\pi}} \exp(-N^{2\beta} x^2)$  with  $G'_\epsilon = -\frac{2N^{3\beta} x}{\sqrt{\pi}} \exp(-N^{2\beta} x^2)$ . Equation (3.10) implies that, using the Gaussian potential, the velocity of the  $i^{\text{th}}$  organism is given by:

$$\dot{x}_i(t) = \frac{2N^{3\beta-1}}{\sqrt{\pi}} \sum_{\substack{j=0 \\ j \neq i}}^N (x_i - x_j) \exp(-N^{2\beta} (x_i - x_j)^2) \quad \forall i = 0, \dots, N. \quad (3.11)$$

The Morse potential is also given by:

$$G(x) = \frac{1}{2} \exp(-|x|).$$

Its scaled form is:  $G_\epsilon(x) = N^\beta \exp(-N^\beta |x|)$  with  $G'_\epsilon(x) = -\text{sign}(x) \frac{N^{2\beta}}{2} \exp(-N^\beta |x|)$ . Hence, equation (3.10) gives:

$$\dot{x}_i(t) = \frac{N^{2\beta-1}}{2} \sum_{\substack{j=0 \\ j \neq i}}^N \text{sign}(x_i - x_j) \exp(-N^\beta |x_i - x_j|) \quad \forall i = 0, \dots, N. \quad (3.12)$$

With the above formulations, we are now ready to begin the simulations. In the following lines of pseudocode, based on the type of potential under consideration, we implement the velocity functions in equation (3.11) or (3.12). After solving the system of ODE's we obtain the final locations,  $X_i(T)$  for each organism after dispersion. Ultimately, we make use of the final locations to reconstruct the final density of the  $i^{\text{th}}$  organism using the formular:

$$U_N(i) = \sum_{\substack{j=0 \\ j \neq i}}^N \frac{2m_i}{|X_{i-1} - X_{i+1}|} \mathbb{1}_{[x_{i-1}(t), x_{i+1}(t)]}(x(t)) \quad \text{for } i = 1, \dots, N-1.$$

As clearly seen in the above formular, we use a central difference approach or better still, the Voronoi tessellation (in one dimension) for the density reconstruction. This approach gives the symmetric final density that we desire. However, the price we pay for it is that, we only obtain the densities of  $N-1$  (out of the  $N+1$ ) organisms. To compensate for the loss, we compute the densities of the first and last organisms using a first order approximation given by:  $U_N(1) = \frac{U_N(2)}{2}$  and  $U_N(N+1) = \frac{U_N(N)}{2}$ .

### 3.2.2 Simulating with the Gaussian Potential

---

**Algorithm 8** : *NLPME with the Gaussian potential and the constant initial density*

**- Part 1**

---

- 1: **procedure** SET THE VARIABLES FOR THE SIMULATION
- 2:   **global**  $N$   $\beta$  ▷ global variables
- 3:    $N = 100$
- 4:    $T = 200$
- 5:    $\alpha = 1$
- 6:    $\beta = 0.1$  ▷ A scaling parameter,  $\beta \in (0, 1)$
- 7:    $m = \frac{1}{N}$  ▷ Assume equal mass for each organism

▷ We continue this algorithm in algorithm 9 on the next page.

---



---

**Algorithm 9** : NLPME with the Gaussian potential and the constant initial density

---

**- Part 2**

---

8:  $x_{max} = \frac{\alpha}{2}$

9:  $x_{min} = -\frac{\alpha}{2}$

10:  $h = \frac{2x_{max}}{N}$

▷ Assume equal spatial grid size

**Require:**  $x_0 = x_{min} : h : x_{max}$

▷ Initial position of organisms

11:  $length(x_0) = lx$

**Require:**  $U_0$  such that

▷ Initial density

12: **for**  $i \leftarrow lx$  **do**

13:     **if**  $-\frac{\alpha}{2} < x_0(i) < \frac{\alpha}{2}$  **then**

14:          $U_0(i) = \frac{1}{\alpha}$

15:     **else**

16:          $U_0(i) = 0$

17:     **end if**

18: **end for**

19: **end procedure**

20: **procedure** : USING THE GAUSSIAN TO IMPLEMENT THE ORGANISMS' VELOCITY  
FUNCTION

21: **function**  $Gauss_1(t, x)$

22:     **global**  $N \beta$

23:      $dx = zeros(N, 1)$

▷ Structure of the velocity: a column vector

24:     **for**  $i \leftarrow 1 : N$  **do**

25:          $Hx = x(i) - x$

26:          $V(i) = \frac{2N^{3\beta-1}}{\sqrt{\pi}} \sum_i Hx \cdot \exp\left(-\left(N^{2\beta}(Hx)\right)^2\right)$

27:     **end for**

28:      $dx = V^t$

▷ Transpose  $V$

29: **end function**

30: **end procedure**

31: **procedure** SOLVE THE SYSTEM OF ODE'S

32:  $[N_T, X_t] = ode23s(@Gauss_1, [0, T], U_0)$

▷ Use the ODE solver,  $N_T$  is a

vector containing all time grid points

▷ This algorithm continues on the next page

---

---

**Algorithm 10** : NLPME with the Gaussian potential and the constant initial density

---

- Part 3

---

```

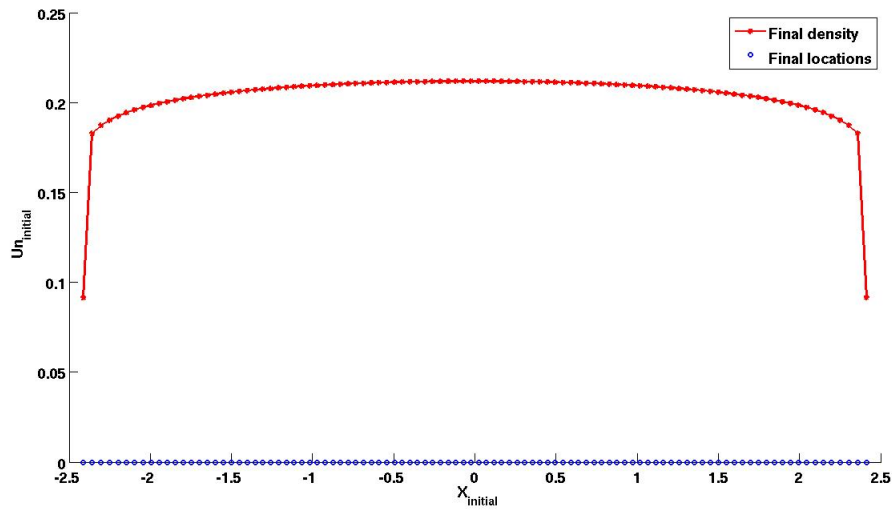
33:    $lt = \text{length}(NT)$ 
34: end procedure

35: procedure RECONSTRUCT THE DENSITY
36:    $U_N = \text{zeros}(lt, lx)$       ▷ Building the structure of the density matrix
37:   for  $i \leftarrow 2 : lx - 1$  do
38:     for  $j \leftarrow 1 : lt$  do
39:        $U_N(j, i) = \frac{2m}{X_t(k, i+1) - X_t(k, i-1)}$       ▷ Using a central difference
40:     end for
41:   end for
42: end procedure

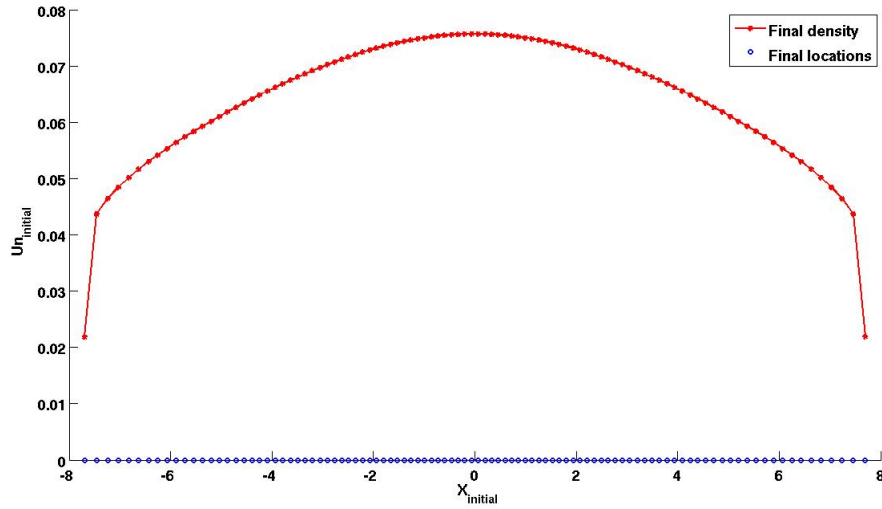
```

---

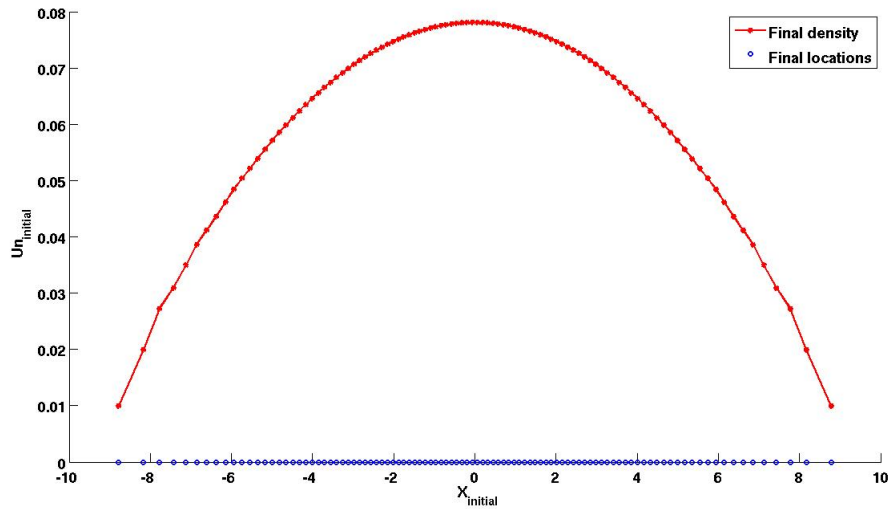
As seen in line 32 of the algorithm, we use adaptive time steps here in order to obtain our results faster. The following figures illustrate the simulation result for some values of the scaling coefficient,  $\beta$ .



**Figure 3.12:** Showing an approximated solution for the NLPME with: the Gaussian potential, the constant initial condition in  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.9$ .



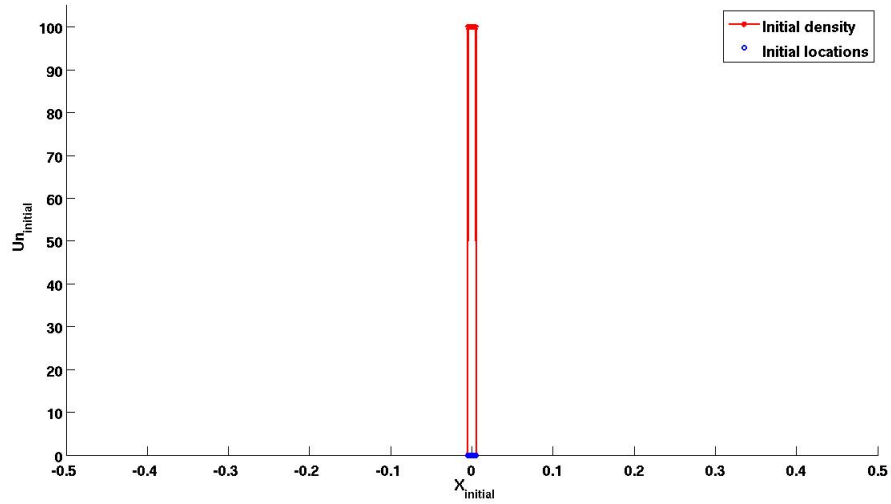
**Figure 3.13:** Showing an approximated solution for the NLPME with: the Gaussian potential, the constant initial condition in  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.5$ .



**Figure 3.14:** Showing an approximated solution for the NLPME with: the Gaussian potential, the constant initial condition in  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.1$ .

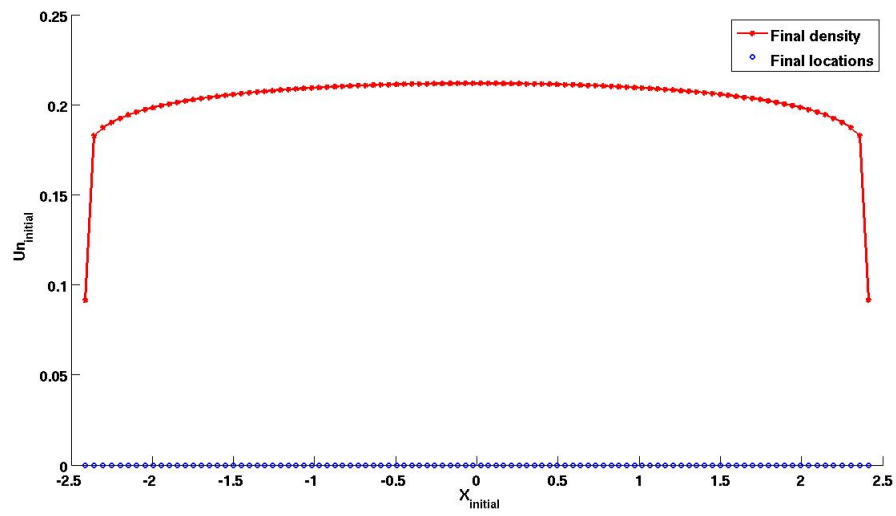
To check if a concentrated version of the initial condition would give a different result or not, we also perform the above simulation with a “Dirac  $\delta$ -like” initial density. That is, for  $\alpha = \frac{1}{N}$ , we use as initial density:

$$U_0 = \begin{cases} \frac{1}{\alpha} & \text{for } -\frac{\alpha}{2} < x < \frac{\alpha}{2} \\ 0 & \text{for } x \in \{-\frac{\alpha}{2}, \frac{\alpha}{2}\} \end{cases}, \quad (3.13)$$

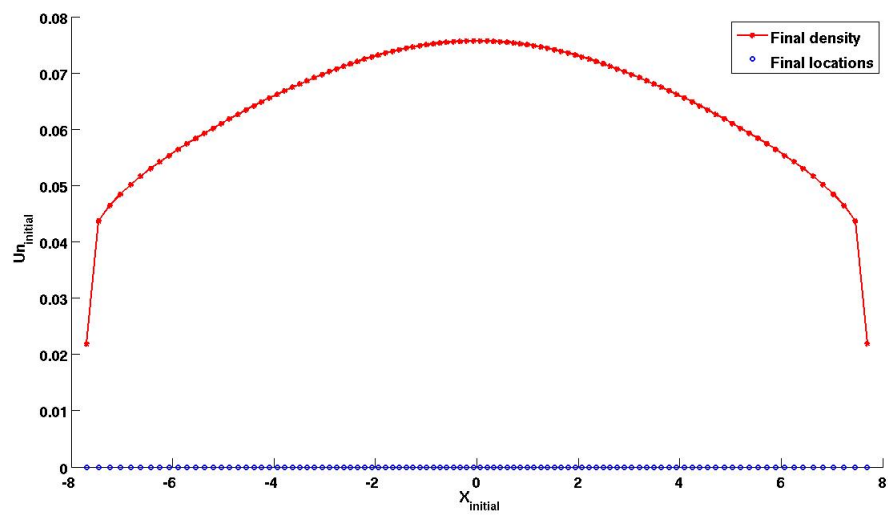


**Figure 3.15:** Showing the "Dirac  $\delta$ -like" initial condition in the domain  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  with  $\beta = 0.9$ .

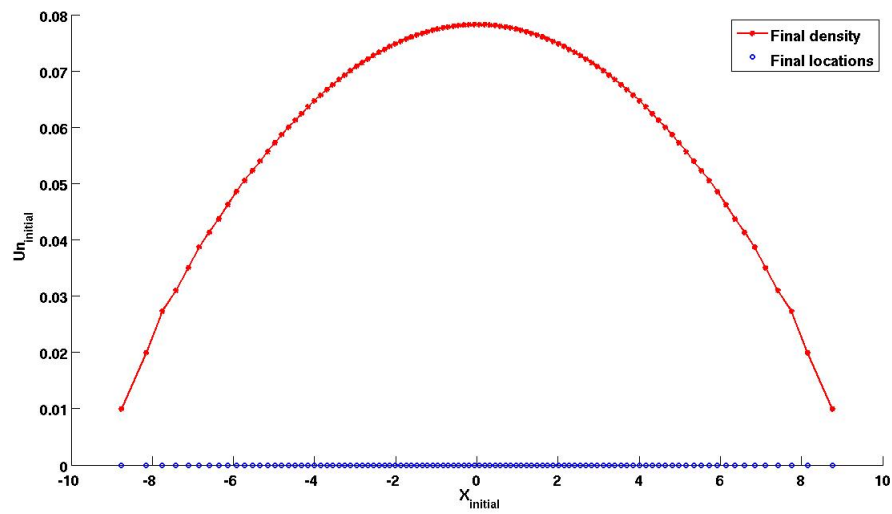
One can easily check that, all the computations shown for the non-concentrated version still holds here. In particular, the total mass of the organisms is still equal to 1. The simulation results are as follows.



**Figure 3.16:** Showing an approximated solution for the NLPME with: the Gaussian potential, a "Dirac  $\delta$ -like" initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.9$ .



**Figure 3.17:** Showing an approximated solution for the NLPME with: the Gaussian potential, a "Dirac  $\delta$ -like" initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.5$ .



**Figure 3.18:** Showing an approximated solution for the NLPME with: the Gaussian potential, a "Dirac  $\delta$ -like" initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.1$ .

Comparing the above figures with the previous, we get the indication that starting with a concentrated constant density does not alter the results of the simulation with a non-concentrated initial density.

### 3.2.3 Simulating with the Morse Potential

Next, considering the Morse potential and still using the constant distribution of the initial density, we simulate the NLPME in a similar way as before.

---

**Algorithm 11** : NLPME with the Morse potential and the constant initial density -

**Part 1**

---

```

1: procedure SET THE VARIABLES FOR THE SIMULATION
2:   global  $N \beta$                                      ▷ global variables
3:    $N = 100$ 
4:    $T = 200$ 
5:    $\alpha = 1$ 
6:    $\beta = 0.1$                                          ▷ A scaling parameter,  $\beta \in (0, 1)$ 
7:    $m = \frac{1}{N}$                                        ▷ Assume equal mass for each organism
8:    $x_{max} = \frac{\alpha}{2}$ 
9:    $x_{min} = -\frac{\alpha}{2}$ 
10:   $h = \frac{x_{max}}{N}$                                      ▷ Assume equal spatial grid size
Require:  $x_0 = x_{min} : h : x_{max}$                    ▷ Initial position of organisms
11:   $length(x_0) = lx$ 
Require:  $U_0$  such that                               ▷ Initial density
12:  for  $i \leftarrow lx$  do
13:    if  $-\frac{\alpha}{2} < x_0(i) < \frac{\alpha}{2}$  then
14:       $U_0(i) = \frac{1}{\alpha}$ 
15:    else
16:       $U_0(i) = 0$ 
17:    end if
18:  end for
19: end procedure
20: procedure : USING THE MORSE POTENTIAL TO IMPLEMENT THE ORGANISMS'
    VELOCITY FUNCTION
21:  function  $Morse_1(t, U_N)$ 
22:    global  $N \beta$ 
23:     $dx = zeros(N, 1)$ 

```

▷ We break this algorithm here in order to continue on the next page.

---

---

**Algorithm 12** : NLPME with the Morse potential and the constant initial density -

**Part 2**

---

```

24:   for  $i \leftarrow 1 : N$  do
25:        $Hx = |x(i) - x|$  ▷ A vector
26:        $sgnx = (x(i) - x) ./ Hx$  ▷ Vector containing normalized
            $x(i) - x(k)$ 
27:        $sgnx(i) = 0$  ▷ Get rid of the  $\frac{0}{0}$  terms in the sgnx vector
28:        $V(i) = \frac{N^{2\beta-1}}{2} \sum_i sgnx \cdot \exp -(N^\beta Hx)$ 
29:   end for
30:    $dx = V^t$  ▷  $V^t$  is the transpose of  $V$ 
31: end function
32: end procedure

33: procedure SOLVE THE SYSTEM OF ODE'S
34:    $[N_T, X_t] = ode23s(@Morse_1, [0, T], U_0)$  ▷ Use the ODE solver with
           adaptive time steps
35:    $length(N_T) = lt$ 
36: end procedure

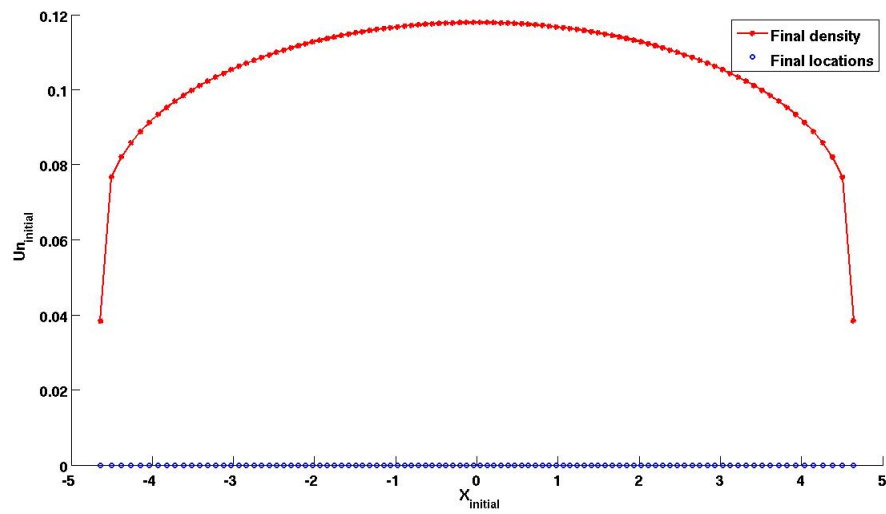
37: procedure RECONSTRUCT THE DENSITY
38:    $U_N = zeros(lt, lx)$  ▷ Building the structure of the density matrix
39:   for  $i \leftarrow 2 : lx - 1$  do
40:       for  $j \leftarrow 1 : lt$  do
41:            $U_N(j, i) = \frac{2m}{X_t(k, i+1) - X_t(k, i-1)}$  ▷ Using a central difference
42:       end for
43:   end for
44:    $U_N(:, 1) = \frac{U_N(:, 2)}{2}$ 
45:    $U_N(:, lx) = \frac{(U_N(:, lx-1))}{2}$  ▷ Set a boundary densities
46: end procedure

```

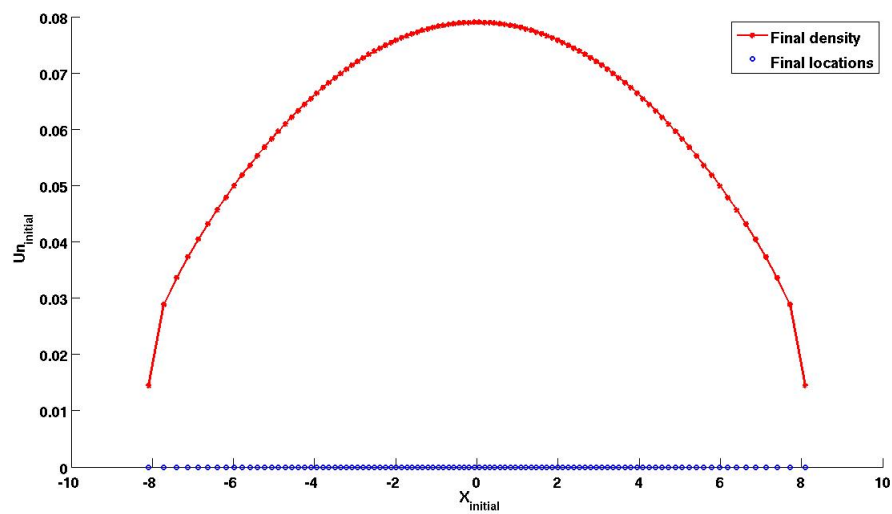
---

The following figures illustrate the simulation results for some values of  $\beta$ .

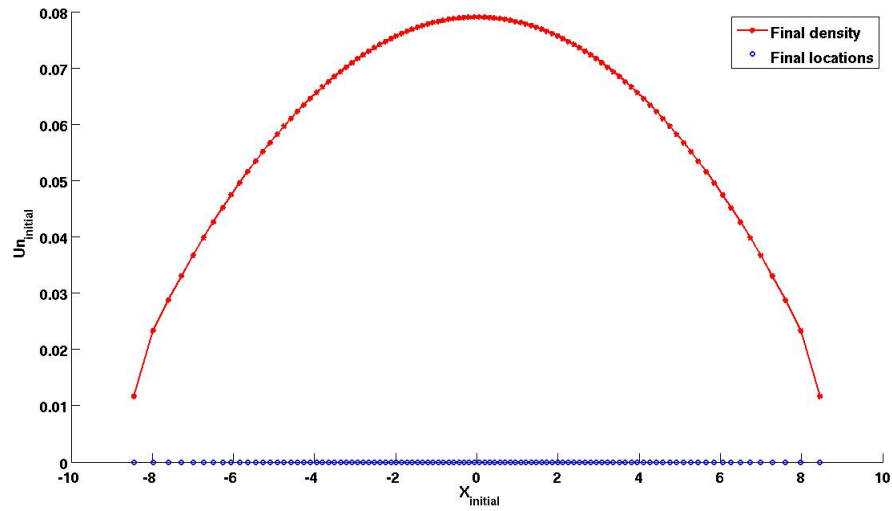




**Figure 3.19:** Showing an approximated solution for the NLPME with: the Morse potential, the constant initial density in  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.9$ .

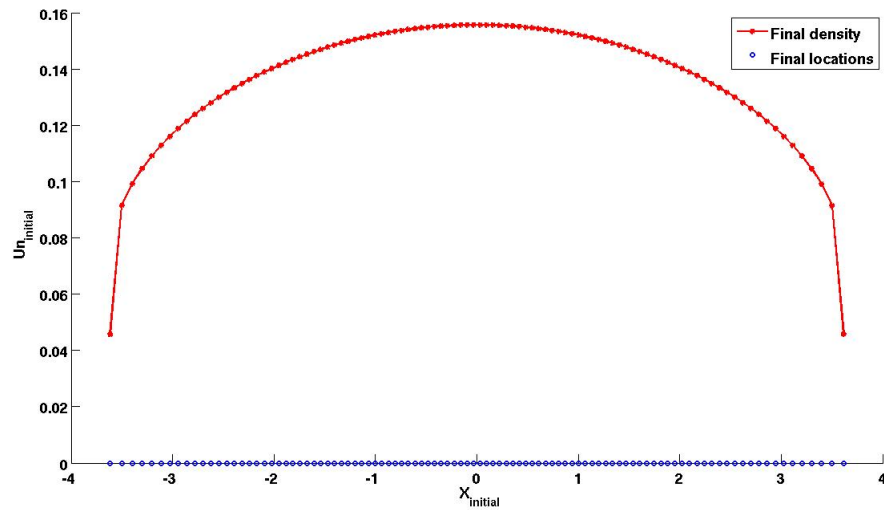


**Figure 3.20:** Showing an approximated solution for the NLPME with: the Morse potential, the constant initial density  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.5$ .

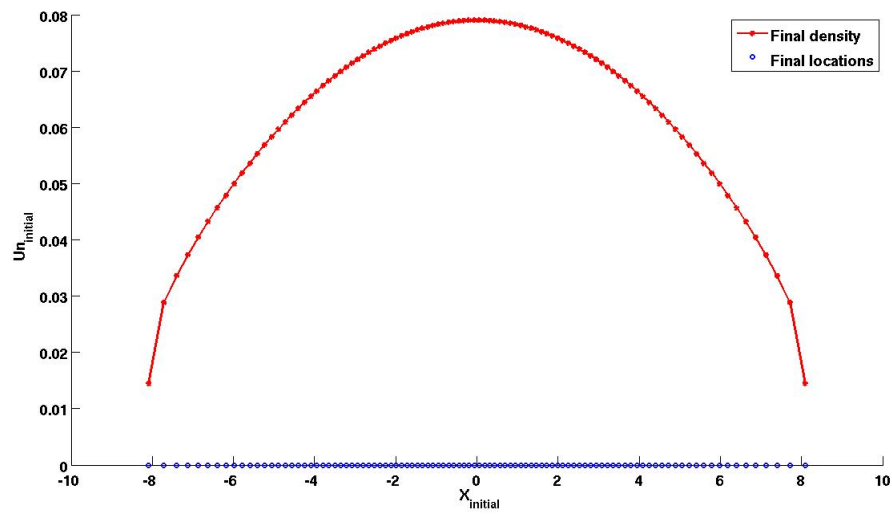


**Figure 3.21:** Showing an approximated solution for the NLPME with: the Morse potential, the constant initial density in  $[-\frac{1}{2}, \frac{1}{2}]$  and  $\beta = 0.1$ .

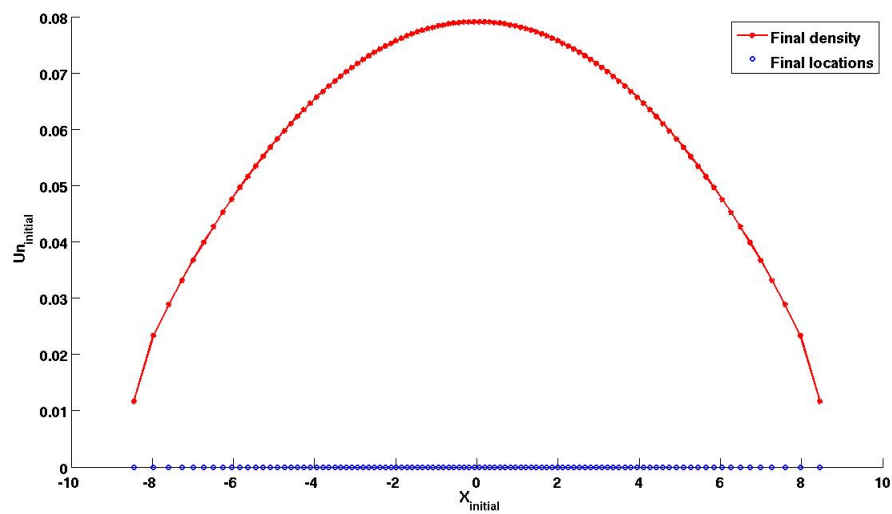
Additionally, repeating the above simulation for a concentrated initial density, i.e Dirac  $\delta$ -like density, we obtain the following results for some values of  $\beta$ .



**Figure 3.22:** Showing an approximated solution for the NLPME with: the Morse potential, the Dirac  $\delta$ -like initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.9$ .



**Figure 3.23:** Showing an approximated solution for the NLPME with: the Morse potential, the Dirac  $\delta$ -like initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.5$ .



**Figure 3.24:** Showing an approximated solution for the NLPME with: the Morse potential, the Dirac  $\delta$ -like initial density in  $[-\frac{\alpha}{2}, \frac{\alpha}{2}]$  and  $\beta = 0.1$ .

The above figures illustrate that, as in the case of the Gaussian potential, in this case also, simulating with a concentrated constant initial density does not give a different result from the simulation with the non-concentrated one.

### 3.2.4 Procedure Using a Triangular Distribution of the Initial Density

Now, we consider a triangular initial density which is compactly supported in the domain  $[-1, 1]$  with the  $L_1$  norm (or mass) of 1, just as the constant initial density.

$$U_0 = \begin{cases} 1 - |x| & \text{for } -1 < x < 1 \\ 0 & \text{otherwise} \end{cases} . \quad (3.14)$$

By the particle method, the organisms are initially located at points such that their density (initial) is given by  $U_0$ . Hence, using the simplicity assumption of the mass of each organism being  $\frac{1}{N}$ , one may compute their initial locations as follows.

Assuming that there exists only one organisms in each interval, then the mass contained between any  $k^{\text{th}}$  position and its subsequent one is given by:

$$\begin{aligned} \frac{1}{N} &= \int_k^{k+1} 1 - |x| dx. \quad \text{For } x < 0, \text{ we get} \\ \frac{1}{N} &= \int_k^{k+1} 1 + x dx = x_{k+1} + \frac{x_{k+1}}{2} - x_k - \frac{x_k}{2} \\ \implies \frac{2}{N} &= (x_{k+1} + 1)^2 - (x_k + 1)^2 \\ \implies x_{k+1} &= \pm \sqrt{\frac{2}{N} + (x_k + 1)^2} - 1 \end{aligned}$$

Particularly for  $-1 \leq x_k < 0$ , the admissible solution is:

$$x_{k+1} = \sqrt{\frac{2}{N} + (x_k + 1)^2} - 1.$$

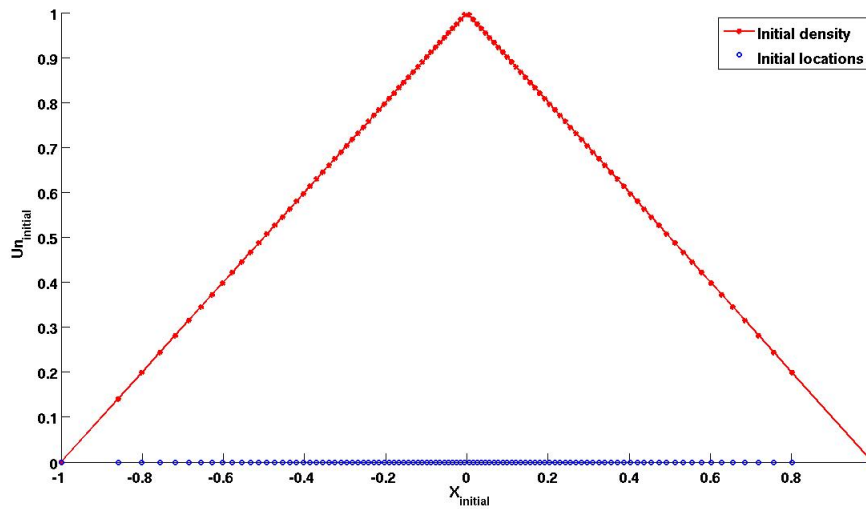
Also, for  $x > 0$ , we have:

$$\begin{aligned} \frac{1}{N} &= \int_k^{k+1} 1 - x dx = x_{k+1} - \frac{x_{k+1}}{2} - x_k + \frac{x_k}{2} \\ \implies \frac{2}{N} &= -(x_{k+1} - 1)^2 + (x_k - 1)^2 \\ \implies x_{k+1} &= \pm \sqrt{(x_k - 1)^2 - \frac{2}{N}} + 1 \end{aligned}$$

Particularly for  $0 < x_k \leq 1$ , the admissible solution is:

$$x_{k+1} = -\sqrt{(x_k - 1)^2 - \frac{2}{N}} + 1.$$

Therefore, initializing with  $x_k = -1$ , the location of the 1<sup>st</sup> organism, we can compute the locations of the remaining  $N - 2$  organisms through an interpolation procedure.



**Figure 3.25:** Showing the initial location of the organisms and their triangular density in the domain  $[-1, 1]$ .

Using the above initial density, we simulate the NLPME again with the Gaussian and Morse potentials.

### 3.2.5 Simulating with the Gaussian Potential

---

**Algorithm 13 :** NLPME with the Gaussian potential and a triangular initial density-

**Part 1**

- 
- 1: **procedure** SET THE PARAMETERS FOR THE SIMULATION  $(N, T, \beta, m, x_0, U_0)$
  - 2:     **global**  $N, \beta$  ▷ Make global variables
  - 3:      $N = 101$
  - 4:      $T = 200$
- ▷ We continue this algorithm with algorithm 14 on the next page.
-

---

**Algorithm 14** : NLPME with the Gaussian potential and a triangular initial density-

---

**Part 2**

---

- 5:  $\beta = 0.1$  ▷ Scaling parameter,  $\beta \in (0, 1)$   
6:  $m = \frac{1}{N}$  ▷ Assume uniform mass of organisms for simplicity  
7:  $x_0(1) = -1$  ▷ Initial location of 1<sup>st</sup> and N<sup>th</sup> organisms  
8:  $x_0(N) = 1$

**Require:**  $x_0$  such that

- 9: **for**  $k \leftarrow 1 : N - 2$  **do**  
10:     **if**  $x_0(k) < 0$  **then**  
11:          $x_0(k+1) = \sqrt{(x_0(k)+1)^2 + \frac{2}{N}} - 1$   
12:     **else**  
13:          $x_0(k+1) = -\sqrt{(x_0(k)-1)^2 - \frac{2}{N}} + 1$   
14:     **end if**  
15: **end for**  
16:  $U_0 = 1 - |x_0|$   
17:  $h = x_0(2 : N) - x_0(1 : N - 1)$   
18: **end procedure**

19: **procedure** : USING THE GAUSSIAN TO IMPLEMENT THE ORGANISMS' VELOCITY  
FUNCTION

- 20: **function**  $Gauss_1(t, x)$   
21:     **global**  $N \beta$   
22:      $dx = \text{zeros}(N, 1)$  ▷ Structure of the velocity: a column vector  
23:     **for**  $i \leftarrow 1 : N$  **do**  
24:          $Hx = x(i) - x$   
25:          $V(i) = \frac{2N^{3\beta-1}}{\sqrt{\pi}} \sum_i Hx \cdot \exp\left(-\left(N^{2\beta}(Hx.^2)\right)\right)$  ▷ Use derivative of  
scaled G. kernel  
26:     **end for**  
27:      $dx = V^t$  ▷ Transpose  $V$   
28: **end function**  
29: **end procedure**

▷ We continue this algorithm with algorithm 15 on the next page.

---

---

**Algorithm 15** : NLPME with the Gaussian potential and a triangular initial density-

**Part 3**

---

```

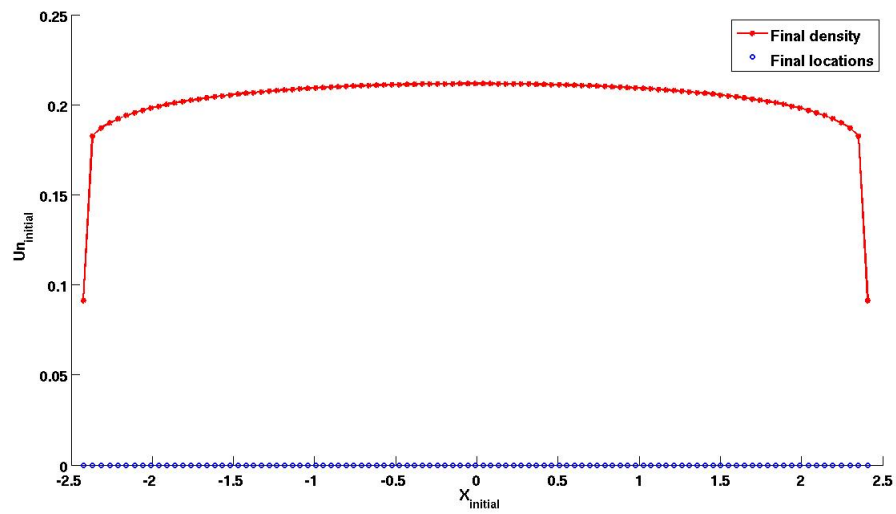
30: procedure SOLVE THE SYSTEM OF ODE'S
31:    $[N_T, X_t] = ode23s(@Gauss_1, [0, T], U_0)$    ▷ Use the ODE solver with
        adaptive time steps
32:    $length(N_T) = lt$ 
33: end procedure

34: procedure RECONSTRUCT THE DENSITIES( $X_t, U_N$ )
35:    $U_N = zeros(lt, lx)$    ▷ Building the structure of the density matrix
36:   for  $i \leftarrow 2 : lx - 1$  do
37:     for  $j \leftarrow 1 : lt$  do
38:        $U_N(j, i) = \frac{2m}{X_t(k, i+1) - X_t(k, i-1)}$    ▷ Using a central difference
39:     end for
40:   end for
41: end procedure
42:  $U(:, 1) = 0.5.U(:, 2)$    ▷ Setting a first order boundary condition
43:  $U(:, lx) = 0.5.U(:, lx - 1)$ 

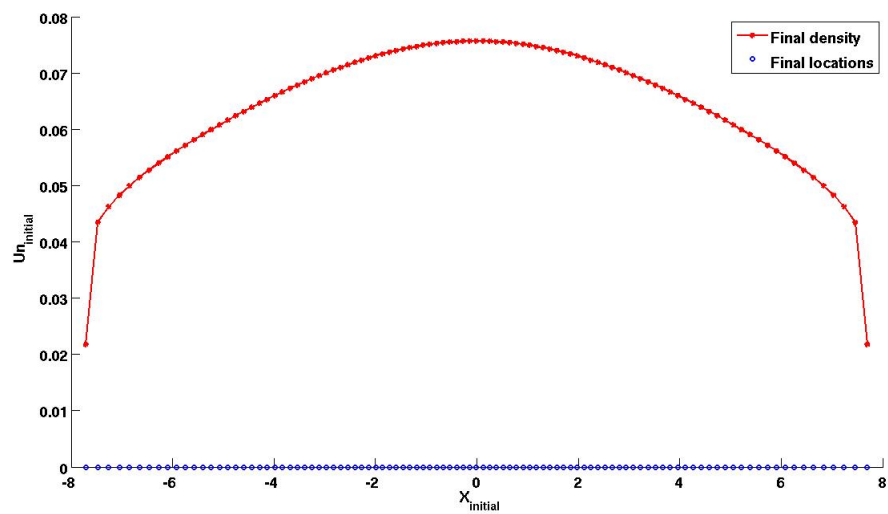
```

---

The following are results for some values of  $\beta$ .

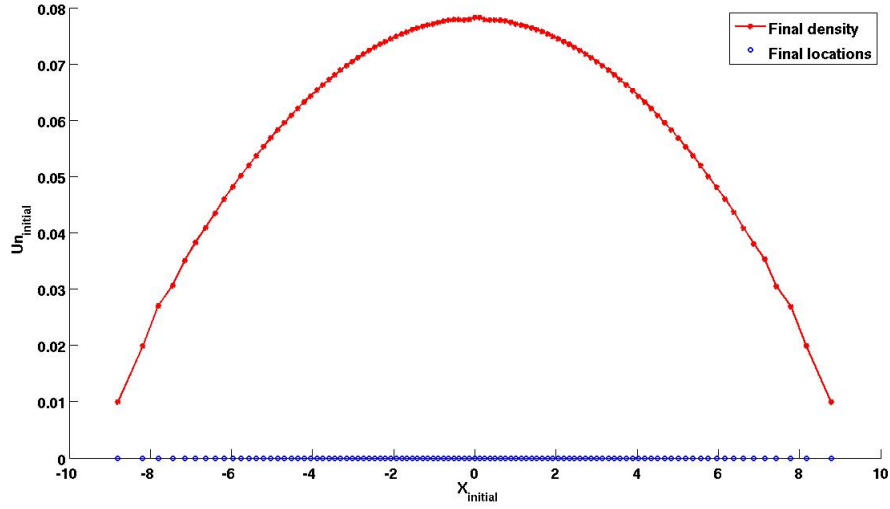


**Figure 3.26:** Showing an approximated solution for the NLPME with: the Gaussian potential, a triangular initial density in the domain  $[-1, 1]$  and  $\beta = 0.9$ .



**Figure 3.27:** Showing an approximated solution for the NLPME with: the Gaussian potential, a triangular initial density in the domain  $[-1, 1]$  and  $\beta = 0.5$ .





**Figure 3.28:** Showing an approximated solution for the NLPME with: the Gaussian potential, a triangular initial density in the domain  $[-1, 1]$  and  $\beta = 0.1$ .

As before, we also repeat the above simulation for the concentrated triangular initial condition. In order to maintain the total mass of 1, one can show that the rescaled triangular initial density is given by:

$$U_0 = \begin{cases} \frac{1}{\alpha} \left(1 - \frac{1}{\alpha}|x|\right) & \text{for } -\alpha < x < \alpha \\ 0 & \text{for } x \in \{-\alpha, \alpha\} \end{cases} . \quad (3.15)$$

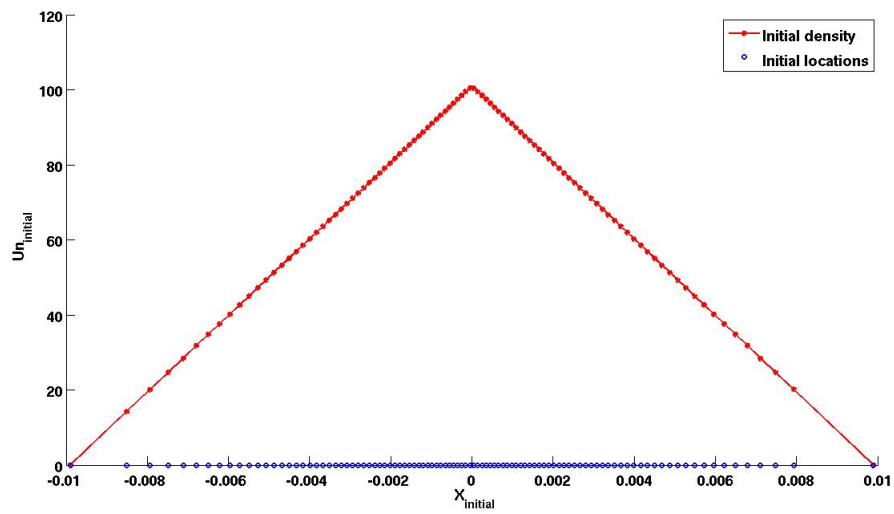
Moreover, given that the initial location of the 1<sup>st</sup> particle is  $-\alpha$ , we can obtain the initial locations of the successive  $N - 2$  organisms from the interpolation equations:

$$x_{k+1} = -\sqrt{\left(x_k - \alpha\right)^2 - \frac{2\alpha^2}{N}} + \alpha \quad \text{for } 0 < x_k \leq \alpha$$

and

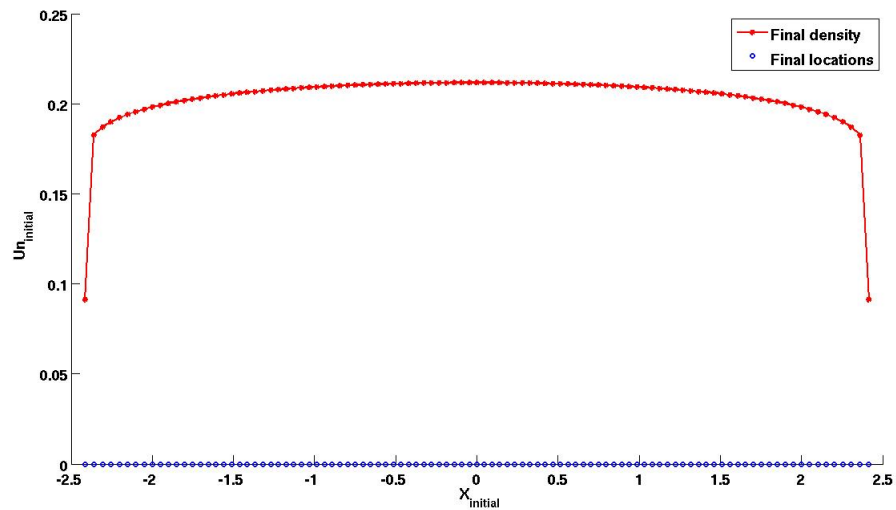
$$x_{k+1} = \sqrt{\left(x_k + \alpha\right)^2 + \frac{2\alpha^2}{N}} - \alpha \quad \text{for } -\alpha \leq x_k < 0.$$

Again, the initial locations of the organisms are computed as before.

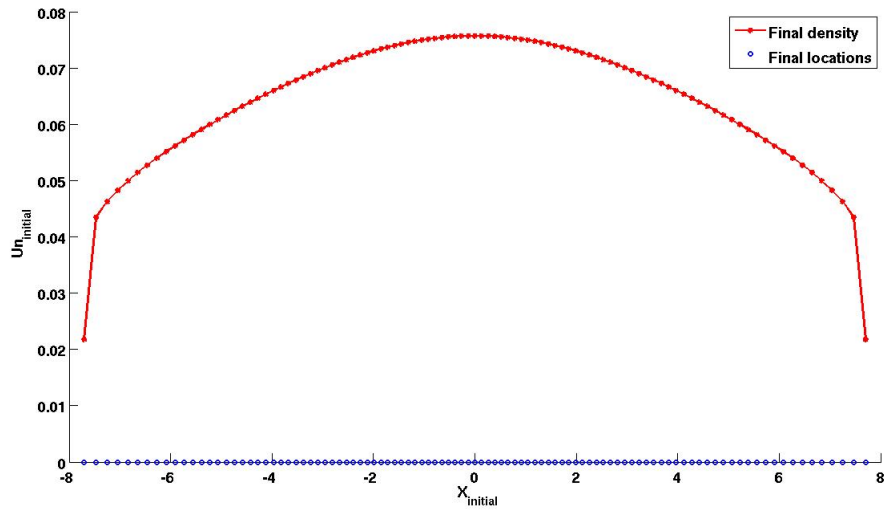


**Figure 3.29:** Showing the initial location of the organisms and their triangular density in the domain  $[-\epsilon, \epsilon]$ .

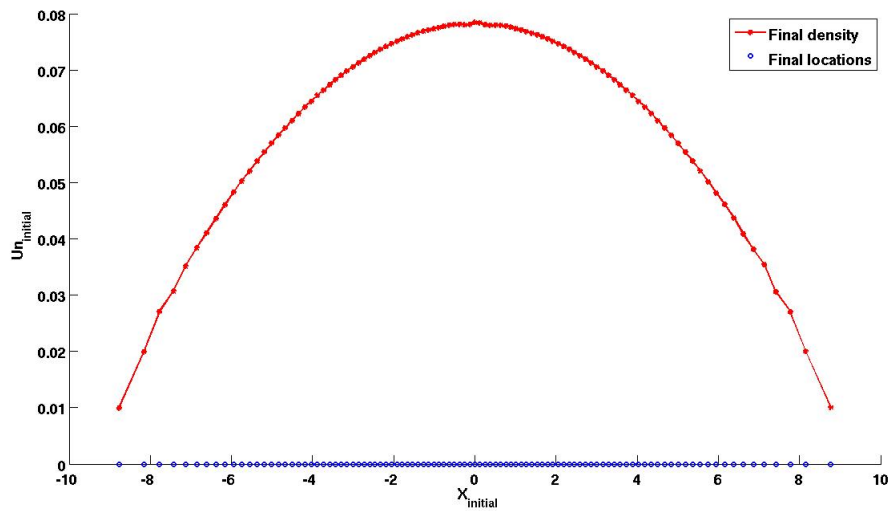
After simulating, we observe from the following figures that, the results here do not vary with the results obtained from using the non-concentrated triangular initial density.



**Figure 3.30:** Showing an approximated solution for the NLPME with: the Gaussian potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.9$ .



**Figure 3.31:** Showing an approximated solution for the NLPME with: the Gaussian potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.5$ .



**Figure 3.32:** Showing an approximated solution for the NLPME with: the Gaussian potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.1$ .

### 3.2.6 Simulating with the Morse Potential

---

**Algorithm 16** : NLPME with the Morse potential and a triangular initial density-

**Part 1**

---

- 1: **procedure** SET THE PARAMETERS FOR THE SIMULATION ( $N, T, \beta, m, x_0, U_0$ )
- 2:   **global**  $N \beta$  ▷ Make global variables
- 3:    $N = 100$
- 4:    $T = 200$
- 5:    $\beta = 0.1$  ▷ Scaling parameter,  $\beta \in (0, 1)$
- 6:    $m = \frac{1}{N}$  ▷ Assume uniform mass of organisms for simplicity
- 7:    $x_0(1) = -1$  ▷ Initial location of 1<sup>st</sup> and  $N^{th}$  organisms
- 8:    $x_0(N) = 1$

**Require:**  $x_0$  such that

- 9:   **for**  $k \leftarrow 1 : N - 2$  **do**
- 10:     **if**  $x_0(k) < 0$  **then**
- 11:        $x_0(k+1) = \sqrt{(x_0(k)+1)^2 + \frac{2}{N}} - 1$
- 12:     **else**
- 13:        $x_0(k+1) = -\sqrt{(x_0(k)-1)^2 - \frac{2}{N}} + 1$
- 14:     **end if**
- 15:   **end for**
- 16:    $U_0 = 1 - |x_0|$
- 17:    $h = x_0(2 : N) - x_0(1 : N - 1)$
- 18: **end procedure**

19: **procedure** : USING THE MORSE POTENTIAL TO IMPLEMENT THE ORGANISMS'

VELOCITY FUNCTION

- 20:   **function**  $Morse_1(t, U_N)$
- 21:     **global**  $N \beta m l x$
- 22:      $dx = \text{zeros}(N, 1)$  ▷ Building the stucture of the velocity vector

▷ We continue this algorithm with algorithm 17.

---

---

**Algorithm 17** : NLPME with the Morse potential and a triangular initial density-

**Part 2**

---

```

23:   for  $i \leftarrow 1 : N$  do
24:        $Hx = |x(i) - x|$ 
25:        $sgn = (x(i) - x) ./ Hx$  ▷ Normalize  $x(i) - x(k)$ 
26:        $sgn(i) = 0$  ▷ Get rid of the  $\frac{0}{0}$  terms in the sgn vector
27:        $V(i) = \frac{N^{2\beta-1}}{4} \sum_i sgn \cdot \exp\left(-\frac{N^\beta Hx}{2}\right)$ 
28:   end for
29:    $dx = V^t$  ▷  $V^t$  is the transpose of  $V$ 
30:   end function
31: end procedure

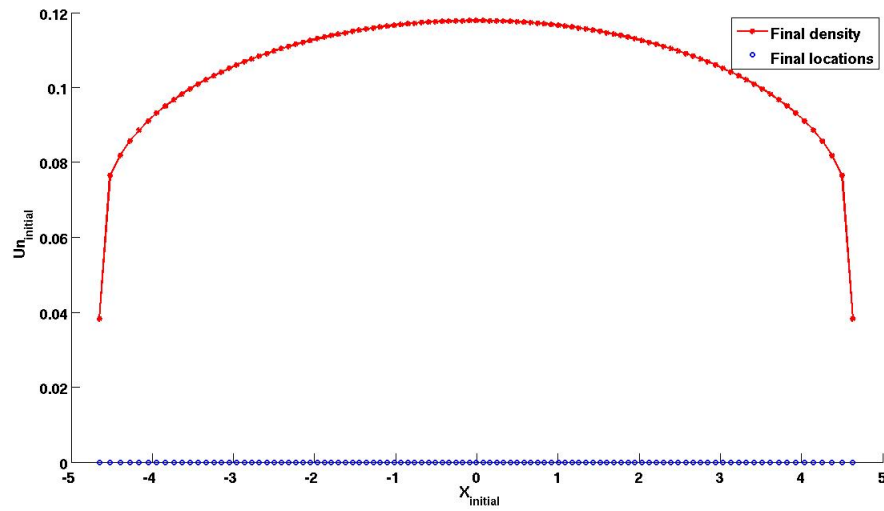
32: procedure SOLVE THE SYSTEM OF ODE'S :  $NLPME_{Gauss2}(dx, X_t, U_N)$ 
33:    $[N_T, X_t] = ode23s(@NLPME_{Morse1}, [0, T], U_0)$  ▷ Use the ODE solver
   with adaptive time steps
34:    $length(N_T) = lt$ 
35: end procedure

36: procedure RECONSTRUCT THE DENSITIES( $X_t, U_N$ )
37:    $U_N = zeros(lt, lx - 1)$  ▷ Building the structure of the density matrix
38:   for  $i \leftarrow 2 : lx - 2$  do
39:       for  $j \leftarrow 1 : lt$  do
40:            $U_N(j, i) = \frac{2m}{X_t(k, i+1) - X_t(k, i-1)}$  ▷ Using a central difference
41:       end for
42:   end for
43:    $U(:, 1) = 0.5 \cdot U(:, 2)$  ▷ Setting the boundary densities
44:    $U(:, lx) = 0.5 \cdot U(:, lx - 1)$ 
45: end procedure

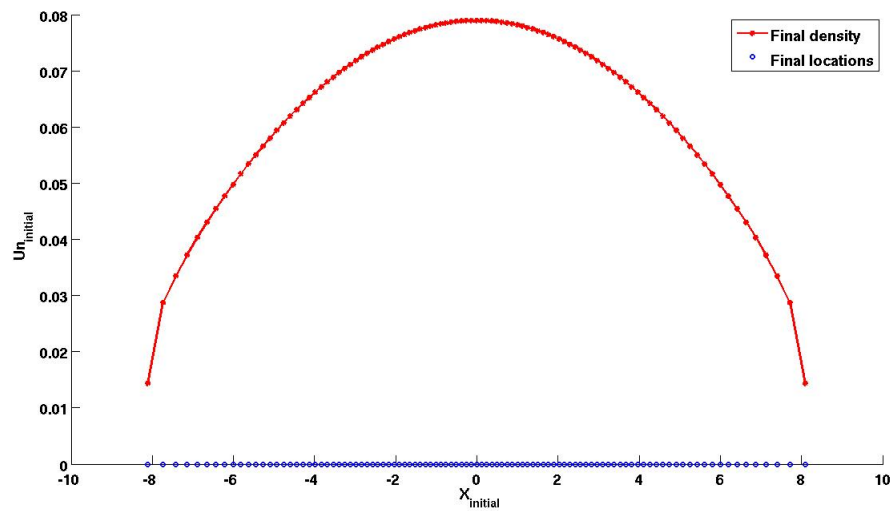
```

---

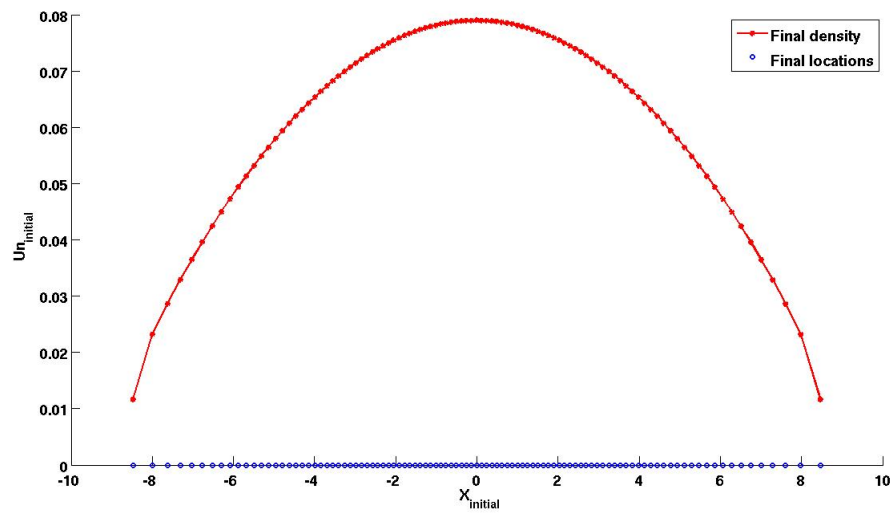
The simulation results are as follows.



**Figure 3.33:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-1, 1]$  and  $\beta = 0.9$ .

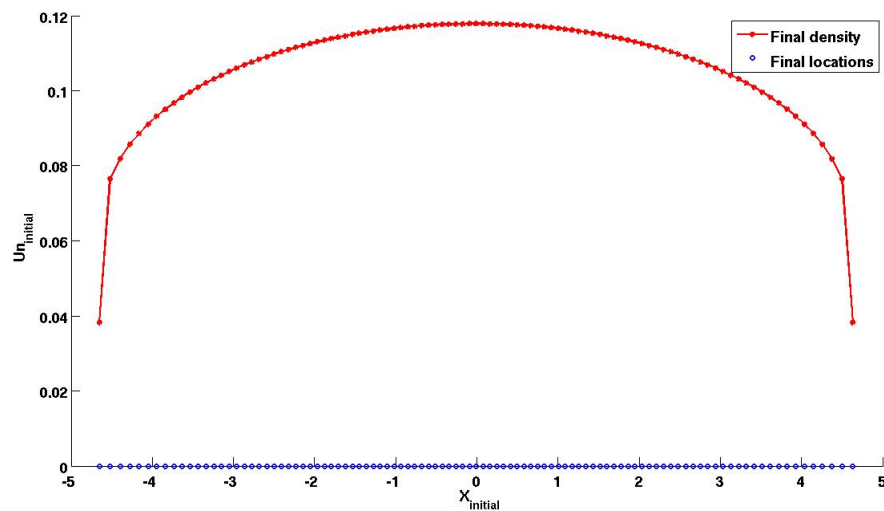


**Figure 3.34:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-1, 1]$  and  $\beta = 0.5$ .

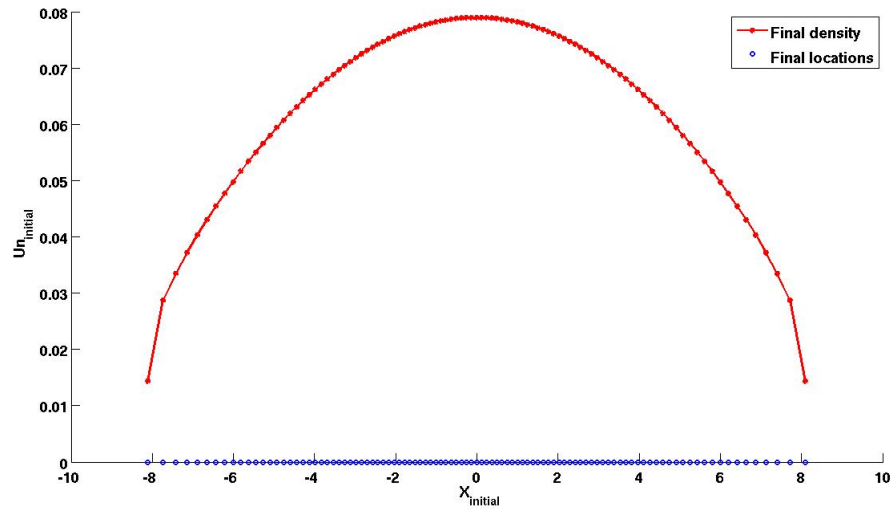


**Figure 3.35:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-1, 1]$  and  $\beta = 0.1$ .

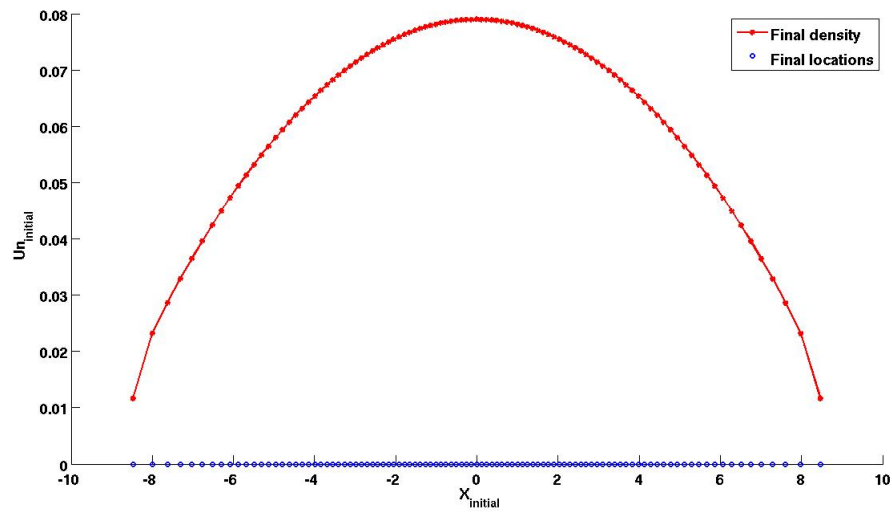
Again, repeating the algorithm for a concentrated triangular density, we obtained the following results.



**Figure 3.36:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.9$ .



**Figure 3.37:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.5$ .



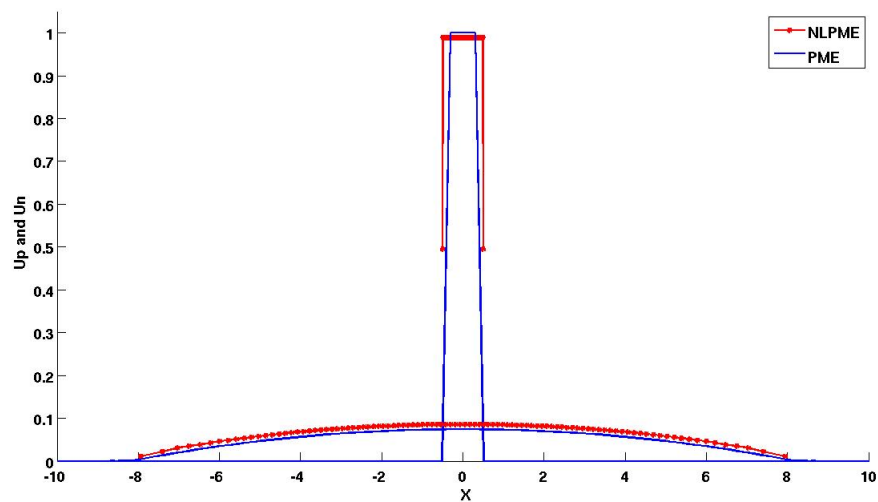
**Figure 3.38:** Showing an approximated solution for the NLPME with: the Morse potential, the triangular initial density in  $[-\alpha, \alpha]$  and  $\beta = 0.1$ .



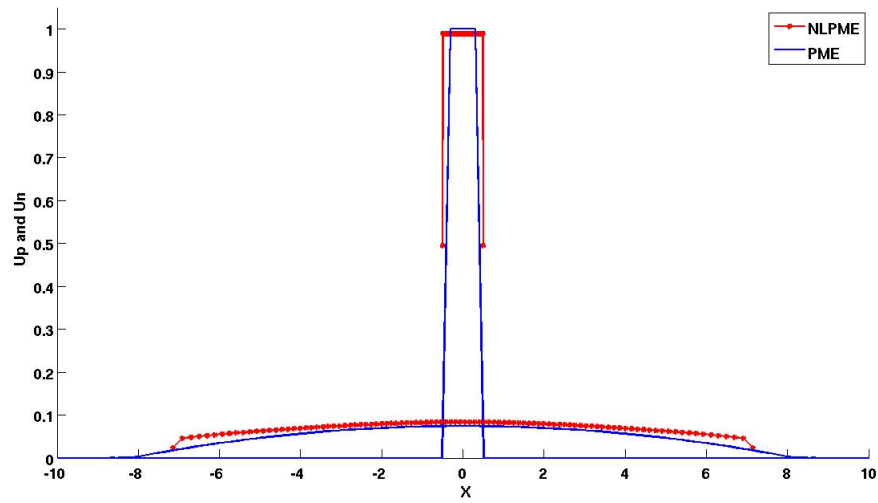
### 3.2.7 A comparison of the PME and the NLPME.

We conclude this chapter by comparing the simulation results of the PME and the NLPME. For this, we make sure to set equal parameters for the two simulations and plot their solutions together as shown in the figures below. In particular, setting the parameter  $a = 10$  (in the algorithm), we simulate the PME in the interval  $[-10, 10]$  and set all initial densities set in the same way as before. In addition, for a time step-by-time steps comparison, we use 200 equal time steps within a period of 150 (units of time in Matlab).

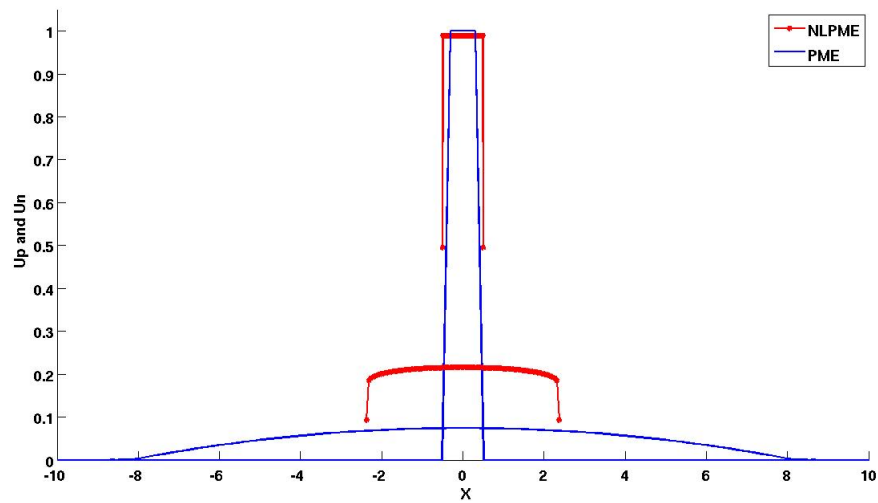
Simulating with the Gaussian potential, for the constant initial density, we obtain:



**Figure 3.39:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.1$ .



**Figure 3.40:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.5$ .



**Figure 3.41:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.9$ .

For the triangular initial density, we obtain:

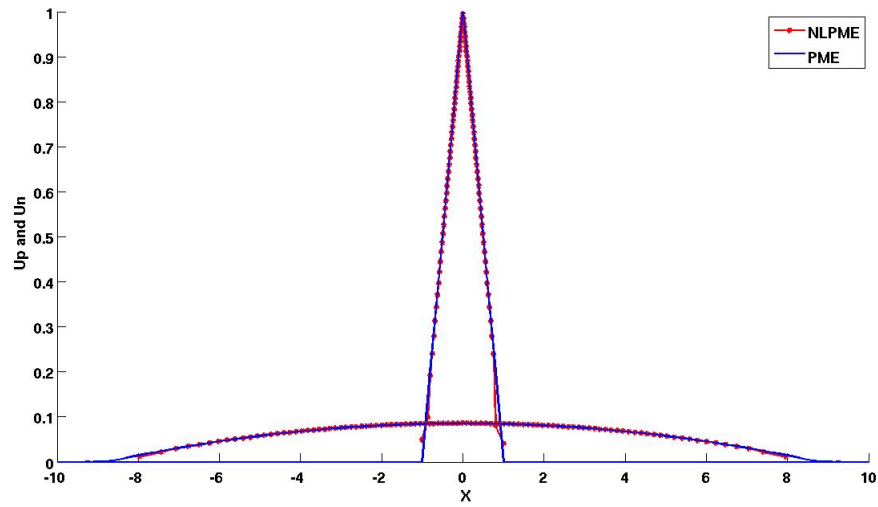


Figure 3.42: Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.1$ .

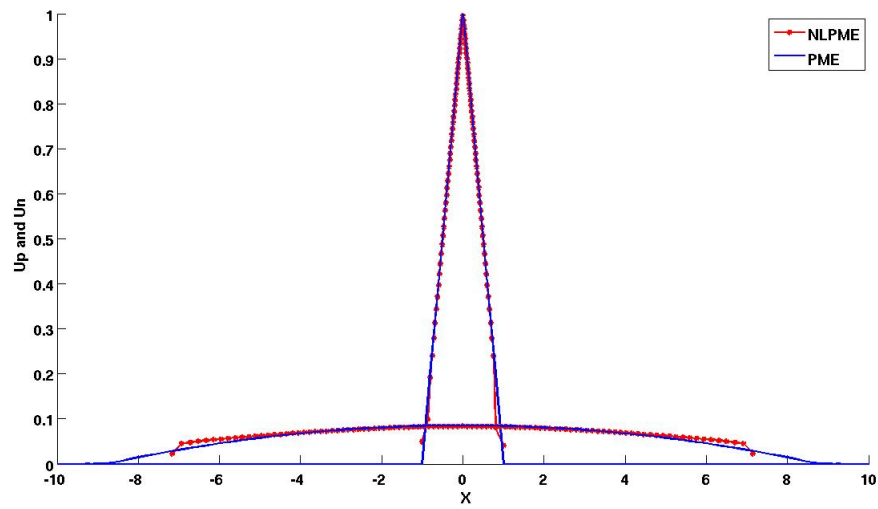
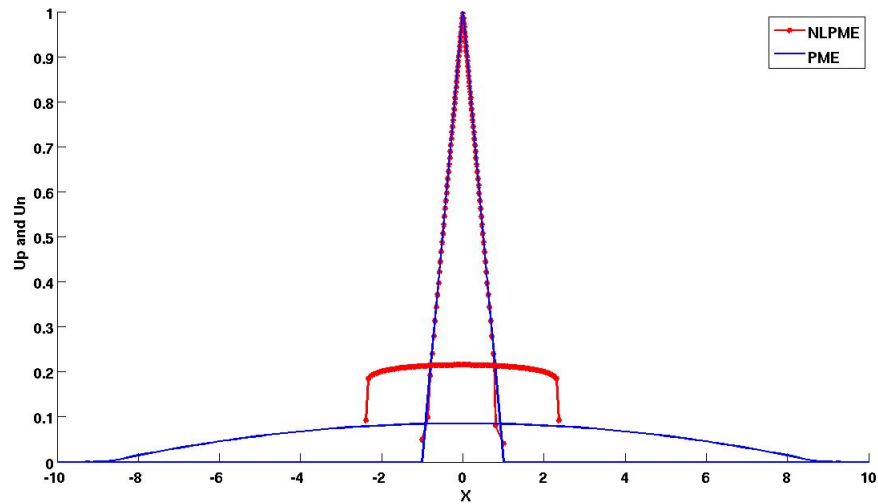
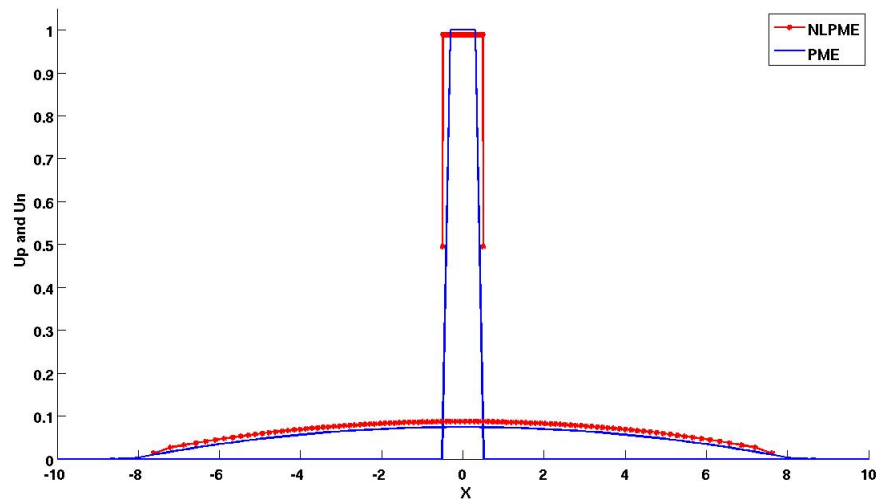


Figure 3.43: Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.5$ .

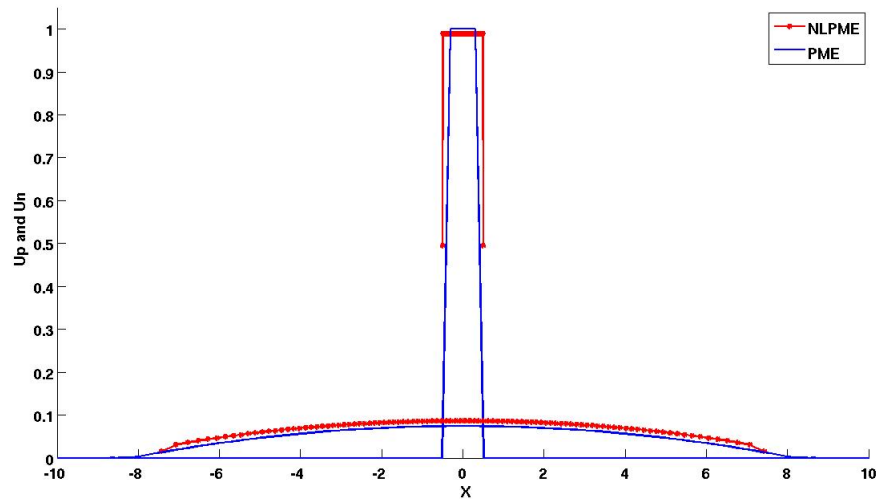


**Figure 3.44:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Gaussian with  $\beta = 0.9$ .

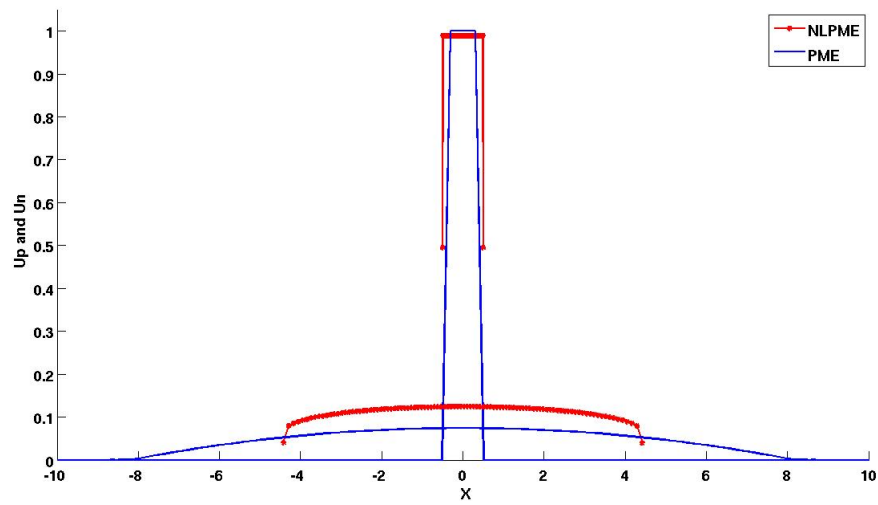
Also, simulating for the Morse potential with the constant initial density, we get:



**Figure 3.45:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.1$ .

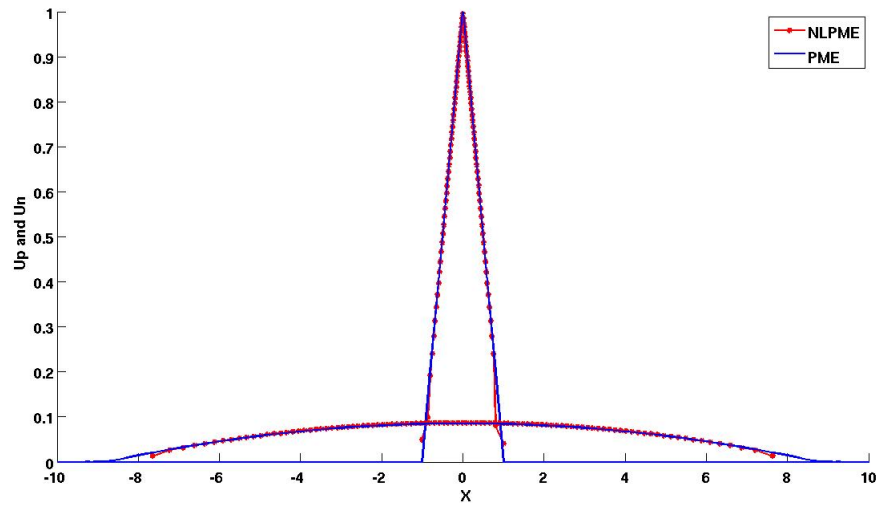


**Figure 3.46:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.5$ .

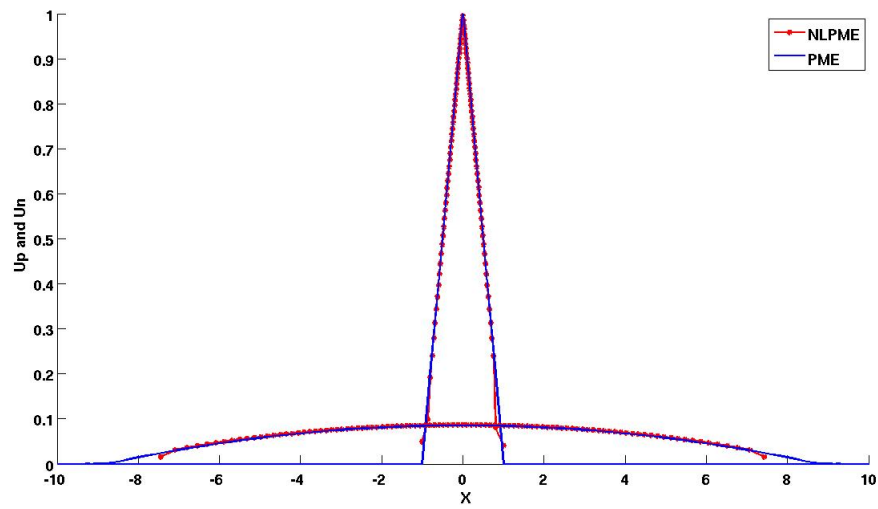


**Figure 3.47:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.9$ .

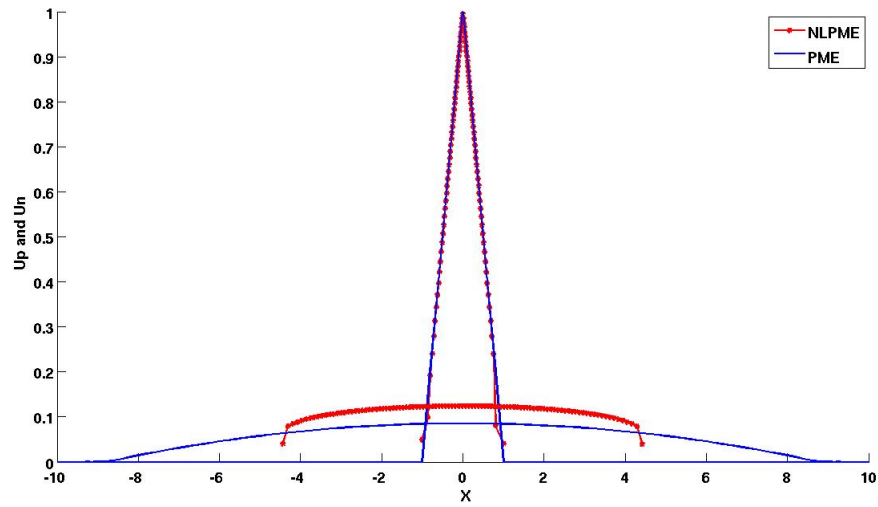
And for the triangular initial density, we get:



**Figure 3.48:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.1$ .



**Figure 3.49:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.5$ .



**Figure 3.50:** Showing both  $U_P$  and  $U_N$  together at their first and final time step: for the Morse with  $\beta = 0.9$ .

From the figures above, it is quite clear that the Gaussian potential with scaling coefficient,  $\beta = 0.1$ , justifies our hypothesis better. We ascertain this observation with an error analysis of the two solutions in the next chapter.

# Chapter 4

## Numerical Analysis 2: $L_1$ Error Analysis

In this chapter, we present the results for the numerical analysis of the  $L_1$  error between the solutions:  $U_P$  and  $U_N$ . We analyse the error for various values of the scaling parameter,  $\beta$  and the number of organisms,  $N$ .

### 4.1 A variation of the $L_1$ error with $\beta$ .

We begin by computing the error between the approximated solutions:  $U_P$  and  $U_N$ . Reasonably, we compute it with respect to the  $L_1$  norm since both solutions belong to that functional space. We obtain this by using the formular:  $\mathbf{Error} = \sum_{i=1}^N |U_P(end, i) - U_N(end, i)|h$ ; where  $h$  refers to the equal spatial step size. As the formular indicates, the focus here is only on the final error simply because the error reduces as the simulation time increases. Furthermore, we simulate with both potentials at the same time and compare their results.



### 4.1.1 $L_1$ Error Analysis with the both potentials and the constant initial density

---

**Algorithm 18** : A variation of  $L_1$  error with  $\beta$ : using the constant initial density-

**Part 1**

---

```

1: procedure SIMULATE FOR VARIOUS VALUES OF  $\beta$ .
2:   global  $N$   $\beta$   $ly$   $h$                                 ▷ Make global variables
3:    $count = 0$ 
4:    $\beta = 0$                                              ▷ Scaling coefficient
5:   while  $\beta < 0.8$  do                                  ▷ For varying  $\beta$ 
6:      $\beta = \beta + 0.1$ 
7:      $count = count + 1$ 
8:      $\beta_{span}(count) = \beta$                             ▷ Vector containing all values of  $\beta$ 
9:      $N = 100$ 
10:     $T = 200$ 
11:     $\alpha = 1$ 
12:     $m = \frac{1}{N}$ 

13:    procedure CONSTRUCTING THE INITIAL DENSITY FOR THE PME
14:      for  $i = 1 : ly$  do
15:        if  $-\frac{\alpha}{2} < y_0(i) < \frac{\alpha}{2}$  then
16:           $U_0(i) = \frac{1}{\alpha}$ 
17:        else
18:           $U_0(i) = 0$ 
19:        end if
20:      end for
21:    end procedure

```

▷ This algorithm continues on the next page.

---

---

**Algorithm 19** : A variation of  $L_1$  error with  $\beta$ : using the constant initial density-  
**Part 2**

---

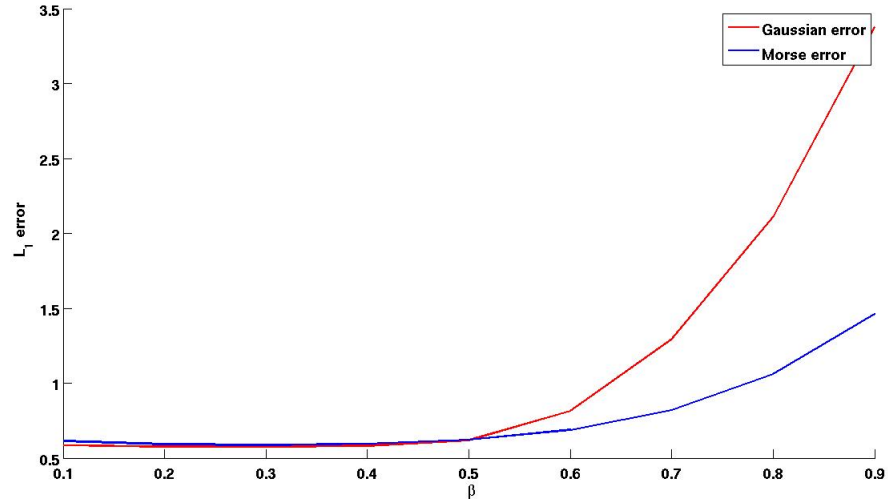
```

22:   procedure SOLVE THE SYSTEM OF ODE'S- RUNGE KUTTA
23:       options = odeset('RelTol', 1e - 4, 'AbsTol', 1e - 4)
24:       [Tp, Up] = ode23s(@PMEConst, tspan, U0, options)
25:       [Tng, Xtg] = ode23s('@NLPME'Gauss, tspan, x0, options)
26:       [Tnm, Xtm] = ode23s('@NPPLE'Morse, tspan, x0, options)
27:   end procedure
28:   procedure RECONSTRUCT  $U_N$  WITH BOTH POTENTIALS
29:        $U_{Ng} = \text{zeros}(lt, lx)$  ▷ For the Gaussian
30:       for  $i = 2 : lx - 1$  do
31:           for  $j = 1 : lt$  do
32:                $U_{Ng}(j, i) = \frac{2m}{X_{tg}(j, i+1) - X_{tg}(j, i-1)}$ 
33:           end for
34:       end for
35:        $U_{Ng}(:, 1) = 0.5.U_{Ng}(:, 2); U_{Ng}(:, lx) = 0.5.U_{Ng}(:, lx - 1)$ 
36:        $U_{Nm} = \text{zeros}(lt, lx)$  ▷ For the Morse
37:       for  $i = 2 : lx - 1$  do
38:           for  $j = 1 : lt$  do
39:                $U_{Nm}(j, i) = \frac{2m}{X_{tm}(j, i+1) - X_{tm}(j, i-1)}$ 
40:           end for
41:       end for
42:        $U_{Nm}(:, 1) = 0.5.U_{Nm}(:, 2)$ 
43:        $U_{Nm}(:, lx) = 0.5.U_{Nm}(:, lx - 1)$ 
44:   end procedure
45:   procedure COMPUTE THE  $L_1$  error WITH BOTH POTENTIALS
46:        $Eg = |U_{Ng}(end, :) - U_P(end, :)|$  ▷ Error with Gaussian
47:        $L_{1g} \text{ error}(count) = h \sum(Eg)$ 
48:        $Em = |U_{Nm}(end, :) - U_P(end, :)|$  ▷ Error with Morse
49:        $L_{1m} \text{ error}(count) = h \sum(Em)$ 
50:   end procedure
51: end while
52: end procedure

```

---

For various values of  $\beta \in (0, 1)$ , the  $L_1$  error varies as follows..



**Figure 4.1:** Showing the variation of the  $L_1$  error with the constant initial density  $\beta \in (0, 1)$  for both Gaussian and Morse Potentials.

Similarly, for the triangular initial density, after replacing the following lines of pseudocode in the above algorithm ?? and plotting the result, we obtain the following figure.

---

**Algorithm 20 :** A variation of  $L_1$  error with  $\beta$ : using the triangular initial density-

**Part 1**

1: **procedure** SPATIAL DESCRETISATION

2:  $h = \frac{2x_{max}}{N-1}$

3:  $y_0 = x_{min} : h : x_{max}$

▷ For the PME

4:  $ly = \text{length}(y_0)$

**Require:**  $x_0$  such that

▷ For the NLPME

5: **for**  $k \leftarrow 1 : N - 2$  **do**

6:     **if**  $x_0(k) < 0$  **then**

7:          $x_0(k+1) = \sqrt{\left(x_0(k) + \alpha\right)^2 + \frac{2}{N}} - \alpha$

▷ This algorithm continues on the next page.

---

---

**Algorithm 21** : A variation of  $L_1$  error with  $\beta$ : using the triangular initial density-

---

**Part 2**

---

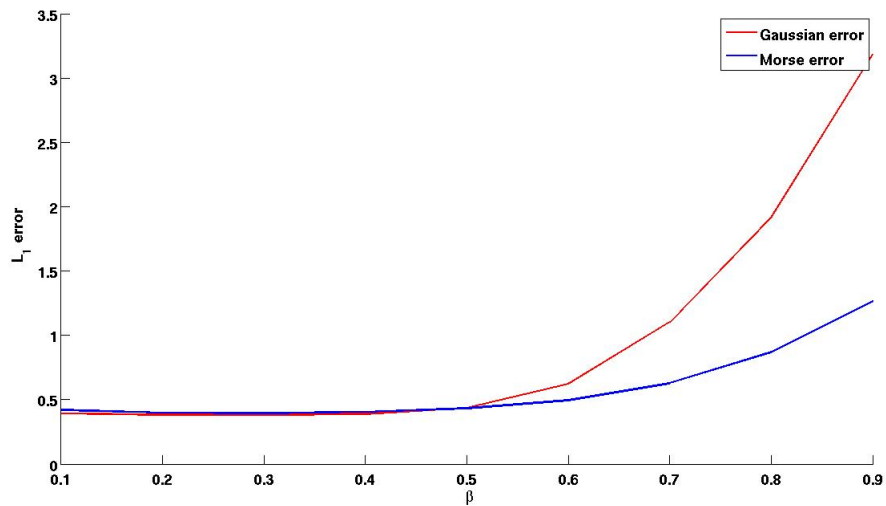
```

8:     else
9:          $x0(k+1) = -\sqrt{(x0(k) - \alpha)^2 - \frac{2}{N}} + \alpha$ 
10:    end if
11: end for
12:  $lx = \text{length}(x0)$ 
13: end procedure

14: procedure CONSTRUCTING THE INITIAL DENSITY FOR THE PME
15:   for  $i = 1 : ly$  do
16:     if  $-\alpha < y0(i) < \alpha$  then
17:        $U_0(i) = \frac{1}{\alpha} - \frac{1}{\alpha^2} |x(i)|$ 
18:     else
19:        $U_0(i) = 0$ 
20:     end if
21:   end for
22: end procedure

```

---



**Figure 4.2:** Showing the variation of the  $L_1$  error with the triangular initial density  $\beta \in (0, 1)$  for both Gaussian and Morse Potentials.

## 4.2 A variation of the $L_1$ error with $N$

Now, setting the value of  $\beta$  as 0.1, we compute the error for some values of  $N$ , the number of organisms.

---

**Algorithm 22** : A variation of  $L_1$  error with  $N$ : using the constant initial density-

---

### Part 1

---

```

1: procedure SIMULATE FOR VARIOUS VALUES OF  $N$ .
2:   global  $N$   $\beta$   $ly$   $h$                                 ▷ Make global variables
3:    $count = 0$ 
4:    $N = 0$ 
5:   while  $N < 500$  do                                    ▷ For varying  $N$ 
6:      $N = N + 50$ 
7:      $count = count + 1$ 
8:      $N_{span}(count) = N$                                 ▷ Vector containing all values of  $\beta$ 
9:      $\beta = 0.1$ 
10:     $T = 200$ 
11:     $\alpha = 1$ 
12:     $m = \frac{1}{N}$ 

13:    procedure TIME DESCRETISATION
14:       $t_0 = 0$ 
15:       $t_f = 150$ 
16:       $dt = \frac{t_f}{T-1}$ 
17:       $tspan = t_0 : dt : t_f$ 
18:       $lt = length(tspan)$ 
19:    end procedure

20:    procedure SPATIAL DESCRETISATION
21:       $h = \frac{2x_{max}}{N-1}$ 
22:       $y_0 = xmin : h : xmax$                                 ▷ For the PME
23:       $ly = length(y_0)$ 

```

▷ This algorithm continues on the next page.

---

---

**Algorithm 23** : A variation of  $L_1$  error with  $\beta$ : using the constant initial density-  
**Part 2**

---

**Require:**  $x_0 = -\frac{\alpha}{2} : \frac{\alpha}{N-1} : \frac{\alpha}{2}$  ▷ For the NLPME

24:  $lx = \text{length}(x_0)$

25: **end procedure**

26: **procedure** CONSTRUCTING THE INITIAL DENSITY FOR THE PME

27: **for**  $i = 1 : ly$  **do**

28: **if**  $-\frac{\alpha}{2} < y_0(i) < \frac{\alpha}{2}$  **then**

29:  $U_0(i) = \frac{1}{\alpha}$

30: **else**

31:  $U_0(i) = 0$

32: **end if**

33: **end for**

34: **end procedure**

35: **procedure** SOLVE THE SYSTEM OF ODE'S- RUNGE KUTTA

36:  $options = \text{odeset}('RelTol', 1e-4, 'AbsTol', 1e-4)$

37:  $[Tp, Up] = \text{ode23s}(@PME_{Const}, tspan, U_0, options)$

38:  $[Tng, Xtg] = \text{ode23s}(@NLPME'_{Gauss}, tspan, x_0, options)$

39:  $[Tnm, Xtm] = \text{ode23s}(@NPPLE'_{Morse}, tspan, x_0, options)$

40: **end procedure**

41: **procedure** RECONSTRUCT  $U_N$  WITH BOTH POTENTIALS

42:  $U_{Ng} = \text{zeros}(lt, lx)$  ▷ For the Gaussian

43:  $U_{Ng}(:, 1) = 0.5.U_{Ng}(:, 2); U_{Ng}(:, lx) = 0.5.U_{Ng}(:, lx - 1)$

▷ This algorithm continues on the next page.

---

---

**Algorithm 24** : A variation of  $L_1$  error with  $\beta$ : using the constant initial density-  
**Part 3**

---

```

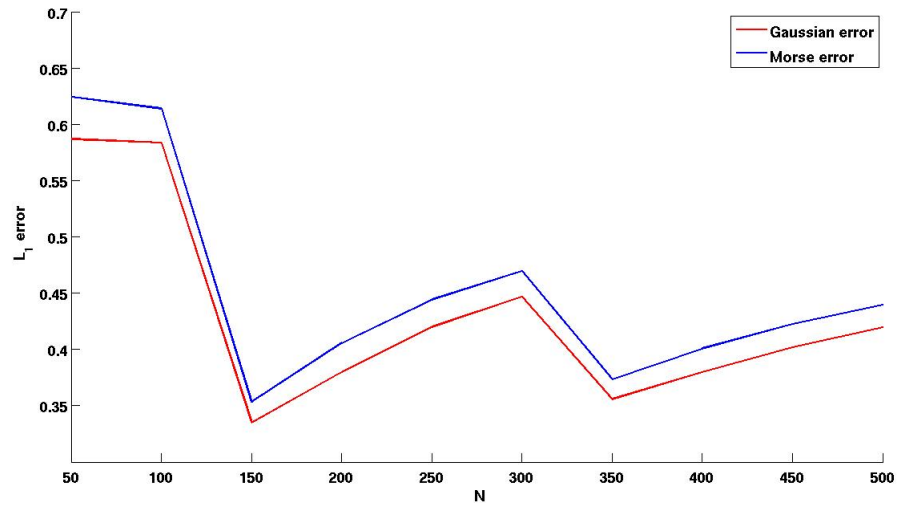
44:         for  $i = 2 : lx - 1$  do
45:             for  $j = 1 : lt$  do
46:                  $U_{Ng}(j, i) = \frac{2m}{X_{lg}(j, i+1) - X_{lg}(j, i-1)}$ 
47:             end for
48:         end for
49:          $U_{Nm} = \text{zeros}(lt, lx)$  ▷ For the Morse
50:         for  $i = 2 : lx - 1$  do
51:             for  $j = 1 : lt$  do
52:                  $U_{Nm}(j, i) = \frac{2m}{X_{tm}(j, i+1) - X_{tm}(j, i-1)}$ 
53:             end for
54:         end for
55:          $U_{Nm}(:, 1) = 0.5 \cdot U_{Nm}(:, 2)$ 
56:          $U_{Nm}(:, lx) = 0.5 \cdot U_{Nm}(:, lx - 1)$ 
57:     end procedure

58:     procedure COMPUTE THE  $L_1$  error WITH BOTH POTENTIALS
59:          $Eg = |U_{Ng}(\text{end}, :) - U_P(\text{end}, :)|$  ▷ Error with Gaussian
60:          $L_{1g} \text{ error}(\text{count}) = h \sum(Eg)$ 
61:          $Em = |U_{Nm}(\text{end}, :) - U_P(\text{end}, :)|$  ▷ Error with Morse
62:          $L_{1m} \text{ error}(\text{count}) = h \sum(Em)$ 
63:     end procedure
64: end while
65: end procedure

```

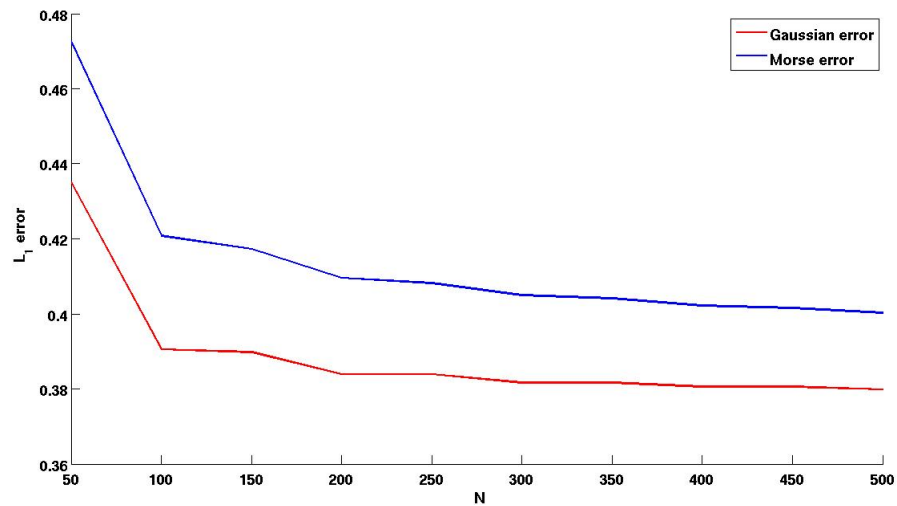
---

Plotting the  $L_1$  error of both potentials against the vector  $N_{span}$  defined algorithm 22, we obtain the following figure.



**Figure 4.3:** Showing the variation of the  $L_1$  error with the constant initial density and  $N$  for both Gaussian and Morse Potentials.

And finally, for the triangular initial density, we again replace in the algorithm above, the lines of pseudocode in algorithm 20. After plotting the results, we obtain the figure below.



**Figure 4.4:** Showing the variation of the  $L_1$  error with the triangular initial density and  $N$  for both Gaussian and Morse Potentials.



# Chapter 5

## Conclusion

One can draw the following conclusions from our simulations.

1. In simulating the porous medium equation, using a concentrated initial density has a significant effect on the results especially for large domains. The results differ more as the domain gets increases.
2. In simulating the non-local porous medium equation, using a concentrated initial density does not show a significant effect on the final density. This could be due to the fact that we did not define a domain for the simulation but only specified the initial locations.
3. An analysis of results for the  $L_1$  error computations indicates that, we record a minimal error for small values of the scaling parameter,  $\beta$ . In addition, the Gaussian potential performs slightly better for values of  $\beta$  in the range  $(0, 0.5)$  while the Morse potential rather performs much better for values in  $(0.5, 1)$ .  $\beta = 0.5$  seems to be a threshold value since it appears that both potentials perform equally with it.
4. Using a more regular initial density (the triangular initial density) records a lesser error as compared to using the less regular one (the constant initial density).
5. While increasing the total number of organisms for the triangular initial density shows a decrease in error, we record an oscillatory behavior of

the error for the irregular constant initial density. However, it appears that the amplitude of the oscillations decreases as the we simulate with more particles.

Undoubtedly,our result is in line with the hypothesis of this thesis. As a future endeavour to continue this work, should consider a more regular initial density like a parabola. Also, gathering the appropriate machinery (computers) to simulate for a very large number of organisms would definately be a useful attempt.

# Bibliography

- [1] Diana E. Bowler and Tim G. Benton, *Causes and consequences of animal dispersion strategies: relating individual behavior to spatial dynamics*, Biol. Rev. **80** (2005), no. 2, 205–225.
- [2] K. Johst and R. Brandl, *The effect of dispersal on local population dynamics*, Ecological Modelling **104** (1997), 87.
- [3] D. Bonte , H. Van Dyck , J. M. Bullock , A. Coulon , M. D. M. Delgado , M. Gibbs, et al., *Costs of dispersal*, Biological Reviews of the Cambridge Philosophical Society. **87** (2012), no. 2, 290–312.
- [4] L. Fernández, *Source-sink dynamics shapes the spatial distribution of soil protists in an arid shrubland of northern Chile*, Journal of Arid Environment. **113** (2015), 121–125.
- [5] M. Inoue, *Derivation of a porous medium equation from many Markovian particles and the propagation of chaos*, Hiroshima Math. Journal. **21** (1991), 85–110.
- [6] D. Tilman and P. Kareiva, *Spatial ecology: the role of space in population dynamics and interspecific interactions*, Princeton University Press, 1998.
- [7] Y. Zhou and M. Kot, *Discrete-time growth-dispersal models with shifting species ranges*, Theoretical Ecology. **4** (2011), no. 1, 549–561.
- [8] B. Maguire, *The passive dispersal of small aquatic organisms and their colonization of isolated bodies of water.*, Ecological Monographs **33** (1963), no. 2, 161–185.

- [9] R. Nathan, *The challenges of studying dispersal*, Trends in ecology and evolution **16** (2001), no. 9, 481.
- [10] D. Morale , V. Capasso and K. Oelschläger, *An interacting particle system modelling aggregation behavior: from individuals to populations*, J. Math. Biol. **50** (2005), 49–66.
- [11] K. Oelschläger, *On the derivation of reaction-diffusion equations as limit dynamics of systems of moderately interacting stochastic processes.*, Probability Theory Rel. **82** (1989), 565–586.
- [12] ———, *Large systems of interacting particles and porous medium equation.*, Journal for Diff. Eqs. **8** (1990), 294–346.
- [13] D. S. Chapman , C. Dytham and G. S. Oxford, *Modelling population redistribution in a leaf beetle: an evaluation of alternative dispersal functions*, Journal of Animal Ecology. **76** (2007), no. 1, 36–44.